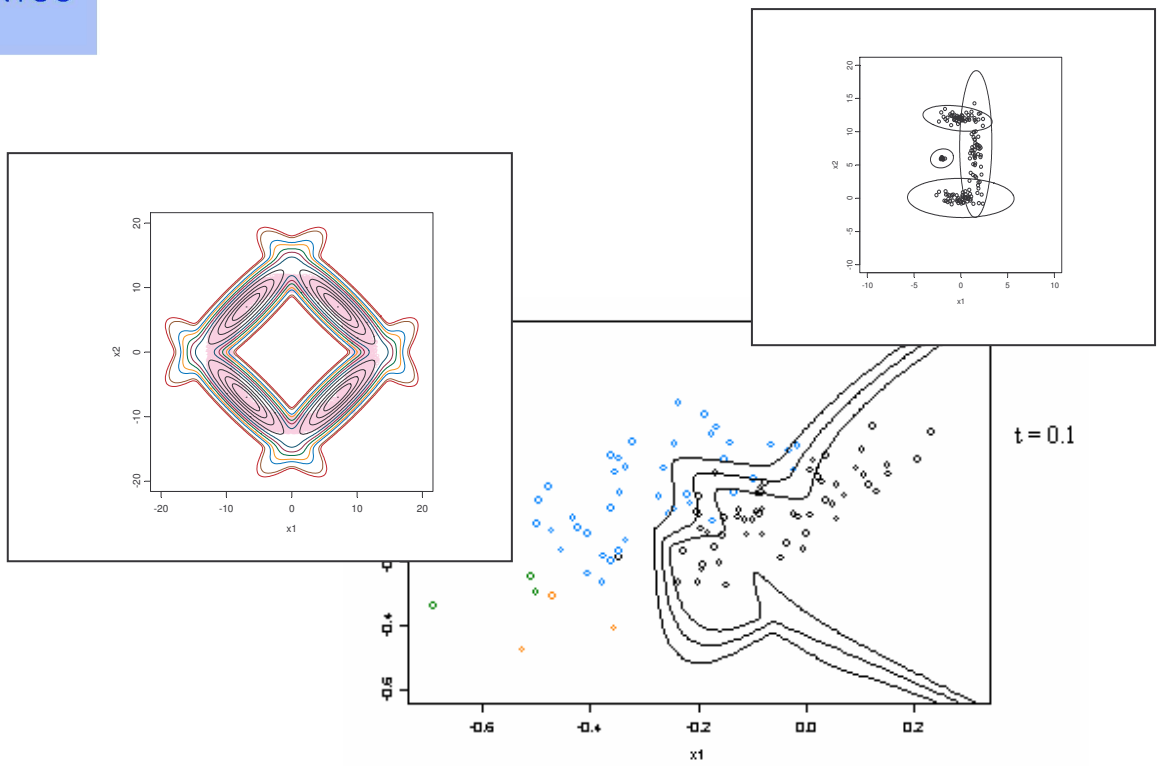
UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO



RECONHECIMENTO DE PADRÕES:
CLASSIFICAÇÃO SUPERVISIONADA COM REJEIÇÃO DE OBSERVAÇÕES

Carla Mónica Santos Dias Pereira
(Mestre)

Dissertação para obtenção do Grau de Doutor em Matemática

DOCUMENTO PROVISÓRIO

Julho de 2004

Resumo

A elaboração desta tese teve em atenção dois objectivos principais: em primeiro lugar dar uma visão global sobre o tema do reconhecimento de padrões e em particular sobre a classificação supervisionada; em segundo lugar, desenvolver uma metodologia de classificação de um objecto, em c+2 classes, alternativa à clássica, com introdução de duas classes de rejeição, uma por indecisão, outra para *outliers*. Os dois primeiros capítulos estão relacionados com o primeiro e os restantes com o segundo.

O terceiro e quarto capítulo abordam, respectivamente, a questão da rejeição por indecisão e por existência de *outliers*. No Capítulo 4 propõe-se um novo método de detecção de *outliers*, RRO, baseado na divisão dos dados em nuvens e na utilização de diferentes tipos de estimadores (robustos e clássicos).

No Capítulo 5 descrevem-se e discutem-se os resultados de um estudo de simulação, para comparação da RRO com os métodos tradicionais existentes. Mostra-se que além de funcionar bem nas situações tradicionais, também revela um bom desempenho em contextos mais gerais.

O Capítulo 6 é dedicado à apresentação do método genérico de classificação em c+2 classes. Finalmente no Capítulo 7 registam-se as principais conclusões e referem-se linhas de orientação para continuação do trabalho aqui iniciado.

Abstract

This thesis was developed bearing in mind two main purposes: first, to give an overview of the pattern recognition area, and supervised classification in particular; second, the development of a methodology to classify an object into *c*+2 classes, in alternative to the classical one, with the introduction of two rejection classes, one for indecision (doubt) and one for outliers. The first two chapters are related to the first purpose and the remaining to the second.

The third and fourth chapters address the question of rejection by indecision and rejection for outliers, respectively. In Chapter 4 a new method to detect outliers, RRO, based on clustering analysis and different estimators (classical and robust), is proposed.

Chapter 5 describes and discusses the results of a simulation study designed to compare the RRO with the traditional methods. It is showed that, while performing well in traditional situations, the method reveals a good performance in general contexts.

Chapter 6 is devoted to the presentation of a generic classification method into c+2 classes. Finally, Chapter 7 states the main conclusions and some guidelines to proceed with the work started here.

Palavras chave:

Análise discriminante
Classificação supervisionada
Compromisso erro/rejeição
Detecção de *outliers*
Estimação robusta
Análise de *clusters*

Keywords:

Discriminant analysis
Supervised classification
Error-reject tradeoff
Outlier detection
Robust estimation
Cluster analysis

À minha Mãe,
a quem permaneço eternamente ligada pelo cordão umbilical,
que em silêncio sopra continuamente nas minhas asas,
que nunca deixou de acreditar em mim,
que me ensinou a não me (pre)ocupar,
que me fez trazer ao de cima o que há de melhor em mim,
que me fez ver a diferença entre sabedoria e conhecimento
e me mostrou que se quisermos,
nada é impossível.

Ao meu Marido,
companheiro, amigo e confidente,
sempre presente,
que me deu a mão nos caminhos mais íngremes,
que riu e chorou sempre a meu lado,
mantendo permanentemente a calma,
acabando até por me contagiar,
e sem nunca desanimar.

À minha Irmã,
que eu muito admiro,
e que sem grandes palavras,
me encheu sempre o coração de alegria na hora certa,
dando-me a força necessária
para continuar.

Aos meus filhos Ana e David,
a quem dedico de uma forma especial esta tese,
que espero, mais tarde,
seja para eles motivo de orgulho e de incentivo,
para que nunca desistam dos seus sonhos,
percebendo que sem trabalho nada se consegue.

Agradecimentos

Fazer referência a todos os que me ajudaram a tornar possível a realização deste sonho seria praticamente impossível. Uns pelos incentivos, outros pela disponibilidade em me ouvirem, outros pela energia positiva que emanam, outros pelos conselhos, outros pelo seu exemplo, outros simplesmente por partilharem comigo a sua beleza interior ou por em algum momento sorrirem. Não posso deixar, no entanto, de agradecer de forma especial

à minha orientadora e amiga, Professora Ana Pires, por reunir todos os aspectos acabados de referir, pela grande mulher e profissional que é, pelo apoio e paciência incondicional mesmo em momentos que lhe eram difíceis, pelo constante estímulo, por tudo que com ela aprendi que se traduz não só em conhecimentos mas também em sabedoria e por não me deixar desistir;

ao Reitor da Universidade Portucalense Professor Doutor Francisco da Costa Durão pelo incentivo e apoios sempre presentes;

ao Professor Doutor Paulo Gomes pelo encorajamento e por me fazer enveredar por este caminho;

a todos os colegas da Universidade Portucalense que me incentivaram e "aturaram" e em especial à Graça, Raquel e Ricardo;

a todos os colegas do Instituto Superior Técnico que sempre me receberam de braços abertos nas minha idas a Lisboa;

a todos os meus amigos pela compreensão da minha ausência em momentos tão importantes das suas vidas.

"Se não houver frutos, valeu o perfume das flores.
Se não houver flores, valeu a sombra das folhas.
Se não houver folhas, valeu a intenção das sementes."
Henfil

Índice

Índice de tabelas

Índice de figuras

Nota introdutória

O reconhecimento de padrões é uma área científica multidisciplinar de inegável utilidade prática e grande actualidade. As suas aplicações são as mais diversas, podendo citar-se o reconhecimento de imagens, caracteres manuscritos ou fala, a classificação automática de objectos ou o diagnóstico médico. As ligações multidisciplinares vão desde a matemática e a estatística à inteligência artificial, passando por processamento de sinais, optimização, modelação estrutural, linguagens formais, percepção biológica ou ciências cognitivas.

Há muitas definições para o reconhecimento de padrões e que estão associadas às várias escolas de pensamento e às várias aplicações que caracterizam as diferentes abordagens frequentemente utilizadas para o tratamento deste tema. O termo "reconhecimento de padrões" é algo abrangente na medida em que designa todo um processo que passa pela selecção e extracção das características, pela aprendizagem, pelo treino e culmina na classificação ou descrição.

Não é de estranhar o grande interesse que o tema tem suscitado e que aliás se reflecte na vasta literatura que tem surgido nos últimos anos. O incremento verificado na quantidade de trabalho de investigação produzido tem sido tão grande, que se torna impossível acompanhar o seu desenvolvimento em tempo útil.

Como consequência da natureza interdisciplinar existem várias abordagens, necessariamente interrelacionadas e que são geralmente divididas em: estatística, estrutural (ou sintáctica) e neuronal. Apesar da existência de uma abordagem chamada "estatística", as contribuições e o interesse da comunidade matemática, e em particular da estatística, na área do reconhecimento de padrões, não têm sido tão significativas como se poderia à partida pensar. Sendo assim o objectivo primário desta tese começou por ser o de contribuir para o aumento dessa interdiscipliniaridade. Procurou-se em primeiro lugar relacionar e quando possível unificar ou criar pontes entre as diversas abordagens, numa perspectiva matemática. Em segundo lugar tentou-se fazer um cruzamento de conhecimentos isto é, ir ao reconhecimento de padrões "importar" ideias que possam ser úteis em áreas tradicionais e próximas da estatística (como é o caso da análise discriminante) e "exportar" para o reconhecimento de padrões desenvolvimentos recentes da estatística, como por exemplo os conceitos de robustez.

Podem agora enunciar-se os objectivos gerais desta tese e que são:

- em primeiro lugar dar uma visão global sobre o tema do reconhecimento de padrões e aspectos gerais associados à classificação e em particular à classificação supervisionada;
- desenvolver uma metodologia de classificação de um objecto (padrão) alternativa à clássica com introdução de uma classe de rejeição por indecisão e de uma classe de rejeição por existência de *outliers*, e que seja competitiva em termos de *performance* com as regras de decisão tradicionais.

Tendo em atenção estes objectivos estruturou-se o trabalho ao longo de sete capítulos da forma que a seguir se passa a descrever.

O primeiro capítulo pretende ser uma introdução ao tema do reconhecimento de padrões começando pela apresentação de várias definições, pela exposição dos aspectos gerais associados à classificação e à descrição dos termos/designações a este associados- com especial destaque no contexto de um problema de classificação- e terminando com a apresentação das principais características, diferenças e semelhanças entre as várias abordagens. Considera-se essencial a clarificação das designações adoptadas para que possam ser compreendidos mais tarde aquando a sua utilização nos restantes capítulos. Relativamente às abordagens ao reconhecimento de padrões, embora por razões óbvias se realce a abordagem estatística e alguns detalhes da abordagem neuronal que também irá ser utilizada na prática, optou-se por referir a estrutural por duas razões. Por um lado, porque se faz uma introdução ao tema r.p. e esta aparece na maior parte da literatura e, por outro, porque serve de introdução às árvores que têm um papel muito importante na abordagem estatística.

No segundo capítulo particulariza-se um dos objectivos gerais considerando-se o caso particular da classificação supervisionada e fazendo-se uma breve reflexão sobre os factores relevantes para a classificação. É feita a formulação de um problema típico de classificação e da notação adoptada, e são introduzidos os principais critérios de decisão para obtenção da regra de decisão. É também aqui que se trata a questão da avaliação da regra de decisão, enumerando-se os vários critérios de avaliação da *performance* de um classificador. É neste capítulo que também se aborda pela primeira vez a questão da rejeição, alertando-se não só para a necessidade e vantagens da sua introdução como também para o facto da rejeição não significar exclusão, mas apenas o

suster da decisão para tratamento posterior. O capítulo termina com a catalogação dos métodos de classificação existentes de acordo com os vários critérios de divisão tradicionalmente efectuados e que são dependentes das propriedades que se pretendem realçar. De salientar as dicotomias que compõem as duas grandes divisões geralmente consideradas: métodos paramétricos versus métodos não paramétricos e métodos de estimação de densidades versus métodos de estimação directa das probabilidades *a posteriori*. Algumas notas são também dedicadas à combinação de classificadores uma vez que começa a ser uma alternativa bastante popular nos dias de hoje. Este inventário permite ter uma ideia da grande diversidade de métodos existentes e das suas condições de aplicabilidade.

Os dois primeiros capítulos são bastante gerais e consideram-se essenciais para um melhor entendimento dos capítulos seguintes. A partir do terceiro capítulo apresentam-se as várias componentes que compõem o método genérico de classificação. Dado um objecto na forma de um vector com características $\mathbf{X}=(X_1,X_2,...,X_p)^T$, um conjunto de c classes distintas e uma amostra de treino (conjunto de objectos rotulados), o objectivo final é o da classificação desse novo objecto numa de c+2 classes distintas G_1, G_2, ..., G_c, I e O onde I representa uma classe de indecisão e O uma classe de *outliers*.

No terceiro capítulo aborda-se com mais detalhe a questão da rejeição no contexto da classificação supervisionada. Começa-se por enunciar os vários critérios de decisão com inclusão de uma opção de rejeição por indecisão, generalizando-se o que foi apresentado no capítulo anterior. Apresenta-se uma regra de decisão óptima, baseada na regra óptima de Chow (1970), que pode escrita quer em termos das probabilidades *a posteriori* quer em termos das densidades condicionais às classes. A regra proposta, embora baseada na teoria da decisão de Bayes, constitui uma generalização da regra de Chow, na medida em que se utiliza uma função de custos genérica e, que se saiba, nunca foi explorada na prática. Seguem-se alguns exemplos de aplicação, com os classificadores tradicionais da análise discriminante (linear, quadrática e logística), onde são deduzidas previamente as expressões associadas à regra e à avaliação da classificação para o caso particular de duas classes. Para finalizar o capítulo, estabelece-se a ligação entre a abordagem proposta, baseada na regra óptima de Chow, e uma abordagem baseada em curvas ROC proposta por Tortorella (2000), apresentando-se ainda um estudo experimental com três classificadores (discriminante linear, discriminante logística e redes neuronais) efectuado sobre um conjunto de dados reais, para comparação das duas abordagens.

O Capítulo 4 é dedicado em exclusivo à detecção de *outliers*. Discute-se o significado do termo, os procedimentos clássicos para a sua identificação e respectivas vantagens. A par de uma breve resenha histórica apresentam-se alguns procedimentos robustos alternativos entre os quais se destacam os estimadores MCD de Rousseeuw (1985) e o recente estimador OGK de Maronna e Zamar (2002). Depois da formalização do significado do termo *outlier*, propõe-se um novo método de detecção de *outliers*- designado por RRO e baseado na divisão dos dados em *clusters* (nuvens)- que além de funcionar bem nas situações tradicionais, também revela um bom desempenho em contextos não elipticamente simétricos. Procede-se em seguida à apresentação da RRO, dos motivos da necessidade da sua implementação, bem como dos aspectos que serviram de base à construção deste método. A proposta de um novo método de detecção de *outliers* é aqui apresentada com o objectivo de integrar uma das componentes que irá culminar no método genérico de classificação em c+2 classes. Para terminar discutem-se alguns procedimentos alternativos ao método proposto.

Antes de prosseguir deixa-se aqui uma advertência relativa à designação "*outlier*". Uma possível tradução seria "observação atípica" mas optou-se por não o fazer por a mesma não ser consensual.

No Capítulo 5 descrevem-se e discutem-se os resultados de um estudo de simulação extensivo, destinado a comparar o desempenho do método proposto com o método tradicional baseado nas distâncias de Mahalanobis, incidindo sobre várias configurações de dados normais e não normais com diferentes dimensões e diferentes percentagens de contaminação, com estimadores clássicos e robustos (MCD e OGK) e diferentes métodos de *clustering*. Este estudo tem como objectivo mostrar o bom desempenho da RRO em situações não *standard* sem detrimento do desempenho do método nas situações *standard*, tornando-o assim competitivo em relação aos métodos tradicionais existentes. O capítulo finaliza com uma breve descrição da estrutura e funcionamento do programa que implementa a RRO e permitiu levar a bom termo o estudo de simulação apresentado.

A inclusão da opção de rejeição por indecisão e por existência de *outliers* é efectuada no Capítulo 6. É neste capítulo que se apresenta a aplicação do método genérico de classificação em c+2 classes a dados simulados e a diversos conjuntos de dados reais, que constituem exemplos clássicos do reconhecimento de padrões. É também apresentada uma proposta de estimação de densidades, alternativa às

tradicionais referidas no Capítulo 2, demonstrando-se a sua utilização nos exemplos tratados.

Finalmente no Capítulo 7 registam-se as principais conclusões gerais e avança-se com algumas linhas de orientação para futuros trabalhos de investigação que permitam a continuação do trabalho aqui iniciado. Embora fosse um grande desafio explorar outros aspectos relevantes neste contexto, tal não foi possível neste trabalho limitado no espaço e no tempo.

"Farei rigorosamente perante o Sol e a Lua
o que me alegrar o espírito e que o meu coração me indicar"
Ralph Waldo Emerson

Capítulo 1

Sobre o reconhecimento de padrões

1.1 Introdução

O reconhecimento de padrões (r.p.) é uma actividade do nosso dia a dia e na qual somos particularmente bons: recebemos dados dos nossos sentidos e somos capazes de imediatamente e sem efeito consciente, identificar a respectiva fonte. Por exemplo, muitos de nós conseguem:

- Reconhecer uma cara que não vêem há muito tempo;
- Reconhecer uma voz através de uma linha telefónica;
- Reconhecer letras num texto.

Qual é o bebé que não reconhece a sua mãe pelo cheiro?

Citando Leondes (1998),

> *"Pattern recognition is the science and the art of giving names to natural objects in the real world."*

A palavra **padrão** deriva de *patron* que segundo Pavlidis (1977, Cap. 1, pág. 1) significa, na sua utilização original, algo que representa um exemplo perfeito a ser imitado; não menos rica a sua tradução de padroeiro, o padroado de um benefício. De certa forma, conforme este autor também refere, lembra-nos a ideia de Platão:

> *"The perfect forms, which the objects of the real world are imperfect replicas of."*

O termo padrão é uma palavra corrente do nosso vocabulário e representa algo que exibe uma certa regularidade, algo que pode ser visto de certa forma como um modelo[1] e que pode ser descrito por um conjunto de medidas ou observações.

[1] O termo modelo é aqui utilizado de uma forma genérica. Enquanto que este pode ser visto como um resumo de dados de grandes dimensões, um padrão é caracterizado por pequenas características de um conjunto de dados ou seja, uma estrutura exibida por um pequeno conjunto de pontos. Aliás, um padrão pode ser utilizado para descrever um modelo. Para mais detalhes sobre as diferenças entre estes dois termos ver Hand (2000).

Um padrão representa portanto um todo, uma estrutura associada a determinado conceito observado. Os elementos que constituem um padrão podem ser, por exemplo, acústicos, ópticos, eléctricos, números numa página, sinais de um radar, etc. Com base nestas ideias a metodologia ligada à classificação de padrões torna-a formal; faz a ligação entre dois mundos- o mundo das observações (medidas físicas) e o mundo dos conceitos, das ideias (Shürmann, 1996, Cap. 1, pág. 2).

Áreas como as da ciência e da tecnologia apresentam-nos com frequência incumbências tais como o diagnóstico de doenças, o reconhecimento de condições de condução perigosas, a leitura de códigos postais em envelopes, a inspecção de controlo de qualidade, a leitura de mapas, a identificação de chegada de mísseis através de radar ou de sinais sonoros, etc; uma lista interminável de problemas para resolver no seio deste tema. Os humanos conseguem realizar estas tarefas mas o que se pretende é criar tecnologia que permita construir "máquinas" capazes de a efectuarem de uma forma precisa, rápida e menos dispendiosa, até porque nem todas as tarefas são tão bem realizadas pelos humanos (por exemplo a leitura de códigos de barras). Claro que há outras situações em que o próprio reconhecimento não tem explicação lógica (por exemplo a forma como se processa a informação de caras) ou ainda casos, como o ainda agora referido (identificação de códigos de barras) onde não há necessidade de nenhuma explicação mas apenas de velocidade e precisão.

Segundo Ripley (1996, Cap. 1, pág. 2) o r.p. é precisamente a disciplina da construção de tais máquinas. Fukunaga (1972, Cap. 1, pág. 1) clarifica mais esta ideia ao afirmar que o objectivo desta área é o de ilustrar estes mecanismos complexos do processo de tomada de decisão (que apelida, como aliás outros autores, de *decision making*) e automatizar estas funções através da utilização da tecnologia informática.

Batchelor (1974) alerta-nos para um facto importante no sentido de poder orientar que caminho seguir para a resolução deste tipo de problemas: o facto do reconhecimento exigir *decisão*. Ao reconhecermos determinada característica, por exemplo uma letra num texto, estamos de facto a decidir que a característica é essa porque a nossa mente nos diz que ela é mais parecida com a letra A, do que por exemplo com a letra B ou C. Por outro lado quando a analisamos e lhe atribuímos um "rótulo" caímos efectivamente num problema de *classificação*. Tal como Bathelor

(1974, Cap. 1, pág. 1) também refere, os termos *reconhecimento*, *detecção*, *tomada de decisão* e *classificação* são aparentemente sinónimos. Contudo podem e devem, consoante o contexto em que são utilizados, ter significados diferentes. De seguida analisa-se um pouco a diferença entre dois dos "principais" termos neste contexto, **reconhecimento** e **classificação**.

Um problema de classificação consiste em encontrar uma forma de decidir se um dado objecto é idêntico a um determinado protótipo, a um padrão ou mesmo a uma dada estrutura. O reconhecimento, por outro lado, obriga a identificação e esta precede a classificação. Reconhecer é mais do que classificar já que "re-conhecer" obriga a especificar o padrão (no sentido genérico) a que se deve associar determinado objecto (Pavel, 1993). Ao classificar não necessitamos de ser específicos acerca da associação de um objecto com uma determinada estrutura enquanto que no reconhecimento esta tem de ser estipulada.

Nesta breve discussão tenta-se clarificar o facto do r.p. estar obviamente relacionado com a classificação; embora não sejam dois termos equivalentes, um não pode existir sem o outro. Sendo assim, e pelo facto do conceito de reconhecimento ser mais abrangente do que o conceito de classificação, opta-se nesta discussão inicial que compõe o Capítulo 1, por intitulá-la de r.p. no sentido de se referir a todo o processo que culmina na classificação.

Há muitas definições para o r.p. e que estão associadas às várias escolas de pensamento e às várias aplicações que caracterizam as diferentes abordagens frequentemente utilizadas para o tratamento deste tema. Apresentam-se de seguida algumas destas definições:

> *"Field concerned with machine recognition of meaningful regularities in noisy or complex environments".*
> Duda e Hart (1973)

> *"... is the search of structure in data… is, by its very nature, an inexact science and thus admits many approaches".*
> Bezdek (1992)

"… is about guessing or predicting the unknown nature of an observation…".
Devroye *et al.* (1996)

"... is the research area that studies the operation and the design of systems that recognize patterns in data".
Duin (2000)

"… is an information-reduction process: the assignment of visual or logical patterns to classes based on the features of these patterns and their relationships".
Leondes (1998)

Quais as principais motivações que levaram à necessidade de modelação de um sistema de r.p.? Bathelor (1974, Cap.1, pág.2) é um dos que responde a esta questão ao frisar que existem muitas aplicações onde o desenvolvimento de técnicas adequadas poderá trazer grandes benefícios. São estes, por exemplo, o dispensar dos humanos de determinadas tarefas onde se verifica tédio ou se lida com situações perigosas, o que conduz a economia de mão de obra e a um acréscimo na qualidade de produtos, etc.

Em resposta a esta questão julga-se que Bezdek (1992, Cap. 1, pág. 6) consegue sintetizar de forma concisa ao focar:

- a necessidade de processamento de dados e informação obtida em interacções entre cientistas, técnicos e a própria comunidade em geral,
- a necessidade que o indivíduo tem em comunicar através dos computadores numa linguagem natural entendida por todos e
- talvez a mais importante motivação, o facto dos cientistas se preocuparem (cada vez mais) com a construção de máquinas "inteligentes" que possam realizar certas tarefas com desempenho comparável ao dos seres humanos.

Mas será que é possível encontrar, ou melhor construir, uma teoria universal para o sistema de r.p.? Será que cada problema não é um caso particular devendo ser tratado como tal? Será que o que é melhor para uma aplicação não poderá ter um mau desempenho noutra situação? A resposta é bastante subjectiva como se pode perceber pela variedade de opiniões proferidas:

" Pattern recognition in general covers a wide range of problems, and it is hard to find a unified view or approach".
Fukunaga (1972)

"No single theory of pattern recognition embraces all of the important topics because each domain has unique characteristics that mold and shape the appropriate approach".
Duda e Hart (1973)

"The goal of pattern classification cannot be in finding a universal solution for a universal task. Instead pattern classification is the universal theory approach to solving a diversity of different tasks, all having their own solutions".
Shürmann (1996)

"Pattern recognition is not comprised of one approach, but rather is a broad body of often loosely related knowledge and techniques … as with most technical disciplines, the concept of 'pattern recognition' means different things to different people…".
Shalkoff (1992)

"The variety of areas of applications may cause one to wonder whether a general approach to pattern recognition is possible. This book is based in the assumption that an affirmative answer exists. The problem can be expressed in general terms and a general methodology can be developed. Of course, the engineer or scientist must exercise his judgment or lack of imagination. But they can help one to avoid reinventing the wheel".
Pavlidis (1977)

"What is essentially missing today in pattern recognition is a well grounded-based or, more precisely, its mathematical formalization… for us, the open problem, whose solutions is in progress, consists in constituting 'the theory of automatic pattern recognition".
Pavel (1993)

Deparamo-nos no último parágrafo com a questão da possibilidade de construção de uma teoria universal para o r.p. Neste momento, só pela breve diversidade de opiniões e como resposta a esta questão, nada melhor que deixar a resposta em aberto remetendo-a para o comentário de Batchelor (1974):

"Pattern recognition is steal a dream, but what a dream!".

Já aqui foi salientado que a área do r.p. está relacionada com muitas outras onde são desenvolvidos sistemas, técnicas e metodologias semelhantes. São exemplos disso:

- Inteligência artificial (*expert systems* e *machine learning*),
- Percepção visual,
- Ciências cognitivas, percepção biológica e taxonomia,
- Optimização,
- Estatística,
- Redes neuronais,
- Análise numérica,
- Sistemas de engenharia,
- Teoria dos conjuntos imprecisos (*fuzzy sets*),
- Teoria dos autómatos,
- Linguagens formais.

Como consequência de tão forte e vasta ligação interdisciplinar, o r.p. tem aplicações numerosas entre as quais se destacam[2]:

- Processamento, segmentação e análise de imagem (como por exemplo a identificação de indivíduos e impressões digitais),
- Reconhecimento de caracteres,
- Análise de discursos (*speech analysis*),
- Inspecção industrial,
- *Credit scoring*,
- Sismologia,
- Diagnóstico médico,
- *Data mining*.[3]

[2] Todos as aplicações aqui referidas são exemplos típicos frequentemente apresentados em qualquer livro na área do r.p.
[3] A relação desta aplicação com o r.p voltará a ser abordada mais adiante na secção 1.3.

Leondes (1998) destaca um grande conjunto de exemplos, no contexto da redes neuronais, dedicando também um capítulo à análise de imagens. No que diz respeito à teoria dos conjuntos imprecisos Bezdek (1992) constitui uma boa referência, não só pelo conjunto de técnicas apresentadas, como pela riqueza dos exemplos que disponibiliza.

1.2 Aspectos gerais associados à classificação e descrição

1.2.1 Padrões, medidas e *features*

Um padrão pode ser descrito como um vector de medidas (variáveis) referentes às características observadas. As classes de padrões podem ser representadas por um ou vários protótipos padrão conforme se descreve na Figura 1.1.

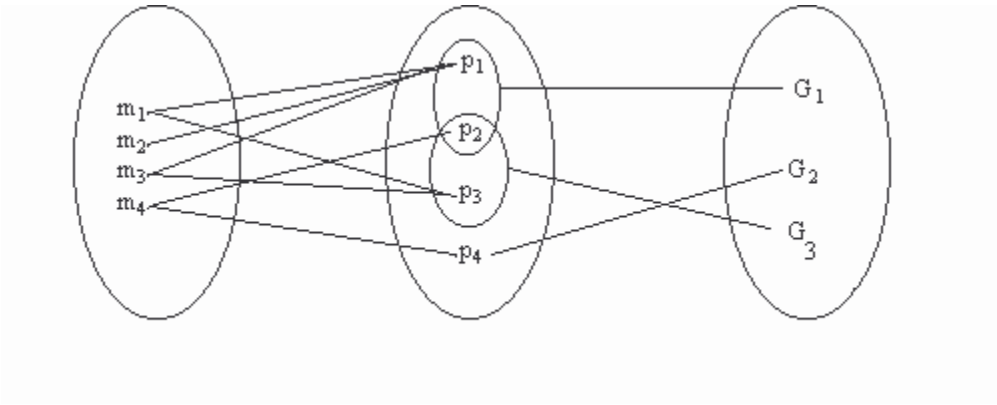


Figura 1.1- Representação básica da aplicação que constitui o r. p..

O termo *feature* é geralmente utilizado como sinónimo do termo "medida". No entanto, enquanto que as medidas são simplesmente observações que se espera que sejam úteis para a tarefa do r. p., as *features* são construídas por forma a terem poder discriminante. Esta alteração pode levar a uma grande melhoria desde que se proceda à escolha de transformações apropriadas e é de importância extrema na maior parte das aplicações, como, por exemplo, demonstra Shürmann (1996, Cap. 9) em problemas de reconstrução de imagens ou caracteres. Mais uma vez se reforça a ideia de que os padrões podem ser de vários tipos e são representados por variáveis (também de vários

tipos). Shalkoff (1992) constitui uma boa referência do ponto de vista de apresentação de exemplos de vários tipos de padrão.

Quanto à extracção e selecção de *features*, as quais se passará a designar por **características**, muito poderia ser dito mas sai fora âmbito desta dissertação. No contexto das redes neuronais, por exemplo, Bishop (1995, 1998) constitui uma boa referência. Neste momento o que se pretende é que fique claro a grande importância que está associada à transformação das medidas num conjunto menor de valores que irão ser as características necessárias para se avançar com um sistema de r.p. A escolha das características é um passo fundamental para um bom desempenho do classificador. Marques (1999, Cap. 1, pág. 10-12) refere que enquanto que esta escolha depende do problema específico em estudo, a concepção, propriamente dita, do classificador, pode ser formulada em termos mais abstractos. No entanto quando se passa à prática há que ter cuidado, se quisermos ser rigorosos e obter bons resultados, com a escolha do classificador que melhor se adapta ao problema em causa. Pretende-se com isto frisar, que esta escolha não poderá ser linear no sentido de optar por aquele classificador que satisfaz os requisitos impostos na teoria mas sim, deverá ser levado a cabo todo um processo de experimentação que inclusive poderá até culminar na obtenção de resultados surpreendentes relativamente ao que se esperava. Neste aspecto constata-se uma grande diferença entre a comunidade do r.p. e a comunidade estatística.

De uma forma geral pode ser dito que num sistema de r.p. o *input* é um vector p-dimensional (um ponto no espaço $\mathbb{R}^p$) cujas componentes são as características. A tarefa principal deste sistema consiste em atribuir o *input*, o qual passará a ser designado genericamente por objecto, a uma de c possíveis classes padrão (ou ainda classificá-lo numa classe de indecisão ou numa classe de observações *outliers*)[4]. Atribuições à mesma classe são baseadas em similaridades. Claro que a quantificação de similaridade é geralmente difícil e por vezes até subjectiva. Shalkoff (1992) e Webb (1999) são dois dos autores que dedicam apêndices apenas ao tratamento deste aspecto. Devijver e Kittler (1982, Cap. 1, pág. 8-10) defendem que a questão da similaridade pode ser abordada de uma forma informal por imposição de requisitos específicos no

[4] Deixa-se para mais tarde explicações detalhadas acerca destas duas "novas" opções, até porque este será o objectivo principal desta tese. Nesta fase, por uma questão de simplificação, falar-se-á apenas na classificação associada a um conjunto de c classes.

processo de obtenção dos dados traduzindo-se no seguinte: observações repetidas do mesmo padrão devem produzir a mesma representação no espaço dos padrões, dois pontos diferentes devem dar origem a duas representações também diferentes e uma ligeira distorção num dado padrão produz uma pequena deslocação na sua representação (de acordo com alguma métrica definida no espaço de representação). Estes requisitos sugerem então que uma pequena distância entre duas representações é sinónimo de similaridade. Contudo, como se julga óbvio, há que ter o cuidado de não cometer erros de extrapolação do tipo "quanto maior é a distância, menor é o grau de similaridade".

1.2.2 Aprendizagem e treino (*learning and training*)

Neste momento já fará sentido afirmar que um sistema de r.p. é composto por duas componentes essenciais, uma unidade de definição e uma unidade de classificação, conforme se descreve na Figura 1.2.

Figura 1.2 – Representação genérica de um sistema de r.p..

Os classificadores têm de lidar com diferentes fontes de informação pelo que à sua construção deve preceder uma clarificação da aplicação em causa; há que identificar as propriedades importantes antes que este seja treinado para a sua função específica.

A analogia das tarefas de r.p. com a aptidão humana constitui uma fonte de inspiração e proporciona, pelo menos, uma primeira ideia para o caminho que poderá levar à solução. Claro que não se pretende dizer que a facilidade com que os humanos reconhecem, ou mesmo classificam, um determinado padrão é a mesma com que se consegue automatizá-lo. Por outro lado existem outras situações onde nós próprios não conseguimos reconhecer. Se não conseguimos, por exemplo, diferenciar um "1" de um "7", como podemos esperar que uma máquina os reconheça?

O procedimento utilizado para se conseguir atingir o reconhecimento ou a classificação deverá começar pela recolha de um conjunto de dados observados- a que

se chamará **exemplos**, dentro de cada classe, e utilizá-los para a construção do sistema de r.p. A criação deste sistema obedece a um procedimento de aprendizagem dedutivo/indutivo que, curiosamente, tem um grande paralelismo com a forma como a criança aprende. Não podemos dizer a uma criança como falar ou como proceder, mas esta irá aprender através do nosso exemplo. A maior parte das nossas regras de aprendizagem são precisamente baseadas na "intuição": quase todas as situações com que nos deparamos diariamente são similares a situações pelas quais já passamos anteriormente.

O paradigma da aprendizagem científica é conhecido desde o tempo de Aristóteles (384-322 A.C.). O homem pode chegar a novos conhecimentos só pelo raciocínio, a partir de conhecimentos disponibilizados chega a conclusões e adquire novos conhecimentos; serve-se do que conhece para encontrar o que ignora partilhando uma verdade universal (dedução) ou a informação de um ou vários casos (indução). Julga-se que uma discussão mais detalhada acerca da forma como se processam os vários tipos de raciocínio utilizados habitualmente pelo homem, analogia, indução e dedução, bem como das suas vantagens e desvantagens, poderá ajudar na compreensão das técnicas de aprendizagem existentes, bem como na construção de novas.

Raciocinar por analogia é partir de certas semelhanças ou relações conhecidas entre objectos e supor que existem outras semelhanças ou relações. Embora este tipo de raciocínio tenha algumas limitações, tal como a ambiguidade ou a falta de rigor, poder-nos-á encaminhar no sentido das "conclusões prováveis".

O raciocínio do tipo indutivo é bastante relevante pelo seu sentido de generalização: a partir de uma certa amostra obtemos conclusões acerca de casos não incluídos, antecipando o que poderá ocorrer numa ocasião futura. Desta forma o papel da indução torna-se duplo: ao nível prático permite compreender as regularidades da natureza criando convicções e crenças, ao nível teórico serve para criar representações do real, para inventar hipóteses, modelos e teorias (Pissarra e Reis, 1996). O problema da validação da indução pode ser sempre minimizado recorrendo-se à indução probabilística.

Analogia e indução são generalizantes, a dedução por sua vez é discriminante; caracteriza-se pelo estabelecimento de uma relação entre dois ou mais juízos. O

raciocínio por dedução vai do geral para o particular, do abstracto para o concreto e do antecedente para o consequente (Pissarra e Reis, 1996).

Estudos ao longo de décadas do cérebro humano confirmam que este está dividido em duas partes construídas por forma a efectuar conjuntamente o processo iterativo de indução-dedução. Na generalidade dos humanos o lado esquerdo do cérebro concentra-se em particular na dedução, análise e pensamento racional, enquanto que o lado direito está mais direccionado para a indução, reconhecimento e criatividade. Consequentemente, e para justificar toda esta exposição, a obtenção de conhecimento através da investigação, com vista à construção de sistemas automáticos de r.p., deve ser realizada via processo iterativo de aprendizagem do tipo indutivo-dedutivo.

Box (1996) partilha a opinião de que a revolução verificada ao nível das potencialidades computacionais irá ajudar a solucionar os problemas relacionados com os aspectos indutivos-dedutivos, permitindo que se ultrapassem as barreiras ainda existentes no âmbito da "aprendizagem".

E voltando novamente à questão crucial da aprendizagem, há que definir a terminologia a esta associada. À colecção de exemplos rotulados, i.e., aqueles cujas classes a que pertencem são conhecidas, é habitual designar por conjunto ou **amostra de treino**. Por outras palavras, a amostra de treino pode ser vista como uma amostra de uma população de exemplos possíveis onde as similaridades de cada classe são extraídas, ou são encontradas as diferenças mais significativas. Se, por outro lado, a proveniência dos exemplos for desconhecida ou seja, se não se conhecem as verdadeiras classes de onde os exemplos são provenientes, então cai-se num problema de aprendizagem não supervisionada (*unsupervised learning*) a qual não irá ser abordada nesta dissertação. Todo o trabalho que irá aqui ser desenvolvido está integrado no contexto da **classificação supervisionada**.

Sempre que possível é habitual e de extrema importância para a avaliação da *performance*, considerar-se um outro conjunto de exemplos o qual é designado por **amostra de teste** para que se possa proceder à validação do classificador. Este novo conjunto tem a vantagem de eliminar o aparecimento de resultados demasiado optimistas que surgem naturalmente ao utilizar a mesma amostra para a construção e

validação do classificador. Ao processo de avaliação da *performanc*e baseado na amostra de treino é habitual chamar reclassificação. Quanto aos processos que utilizam uma outra amostra independente (amostra de teste) são geralmente referidos como generalização (*generalization*), termo que deriva da psicologia e que se refere à facilidade de se inferir correctamente sobre uma estrutura, através de exemplos (ver por exemplo Ripley, 1996).

Voltando à constituição de um sistema de r.p., a unidade de classificação representada na Figura 1.2 deverá ser processada em duas fases, uma que diz respeito à aprendizagem e treino, por vezes também designada por *store* (ver por exemplo Batchelor, 1974, Cap. 1) e uma subsequente que trata da própria decisão e a qual se passará a designar por fase operativa. É esta fase que nos permite dizer:

" este exemplo pertence à classe *c*" ou,

"este exemplo não pertence a nenhuma das classes" ou ainda,

"este exemplo é demasiado complexo".

A Figura 1.3 apresentada a seguir, já contempla estas duas novas fases no contexto de um sistema genérico de r.p.

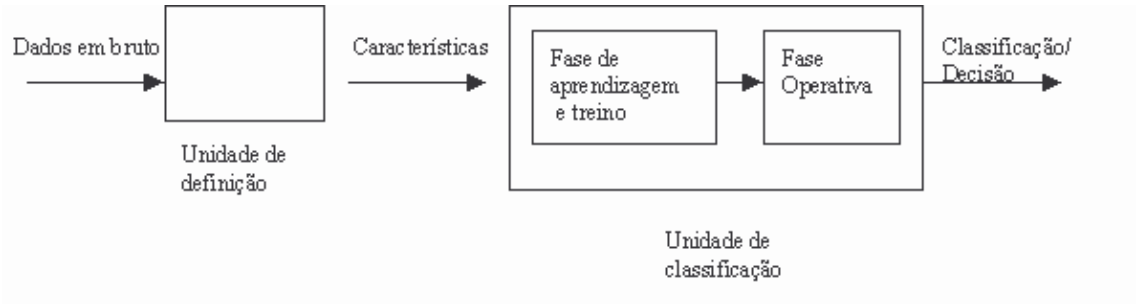


Figura 1.3 – Representação mais detalhada de um sistema de r.p..

De salientar ainda que treino e aprendizagem são dois conceitos completamente interligados. A amostra de treino é utilizada para permitir que o sistema "aprenda" a distinguir a informação relevante.

Na terminologia do r.p. a fase de aprendizagem/treino não é mais que o processo de estimação dos vários parâmetros (seja por métodos paramétricos, semi-paramétricos ou

não paramétricos) que permitirão construir a regra de classificação (ou classificador) necessária à tomada de decisão, na fase operativa.

1.2.3 Classificação versus descrição

Por muito bom que seja determinado método de classificação, dele só surgirão bons resultados se forem tomadas as medidas certas. Existem muitas formas de encontrar as características mais apropriadas de entre um vasto conjunto de candidatas (processo que já foi aqui referido como selecção de características). Mas isto se for possível encontrar primeiro o tal conjunto. A opção pela descrição ou classificação tem frequentemente origem nesta demanda. Há muitos problemas no contexto do r.p. para os quais o modelo de classificação é claramente inapropriado. Noutros casos o que se pretende é mesmo descrição e não classificação, quando analisamos uma figura, por exemplo. Tal descrição deve conter informação acerca de todas as partes individuais da mesma bem como das relações existentes entre as diversas partes. Idealmente deve reflectir directamente a estrutura presente na figura original.

Esta questão levou ao aparecimento de uma nova abordagem baseada na ideia de que uma estrutura complexa pode ser dividida em várias partes e descrita em termos das componentes dessas partes (por exemplo o diagrama de um circuito eléctrico). Leondes (1998, pág. 161-164) é de opinião que descrições adequadas permitem que sejam utilizadas técnicas de classificação simples pelo que considera o problema de classificação subordinado ao problema da descrição: para que uma descrição seja fiável, a decomposição do todo em partes tem de ser estável com respeito às variações entre amostras pertencentes à mesma classe, por forma a não destruir a informação necessária para a discriminação. Embora não seja fácil, esta abordagem torna-se interessante pelo simples facto das características, ao fazerem parte do padrão, permitirem uma espécie de avaliação *a priori* da fiabilidade das descrições. Há ainda uma agravante comum a problemas que surgem neste contexto: muitos padrões contêm informação estrutural ou relacional, difícil de quantificar em termos de características. A Figura 1.4 já contempla a opção de descrição no contexto de um sistema de r.p.

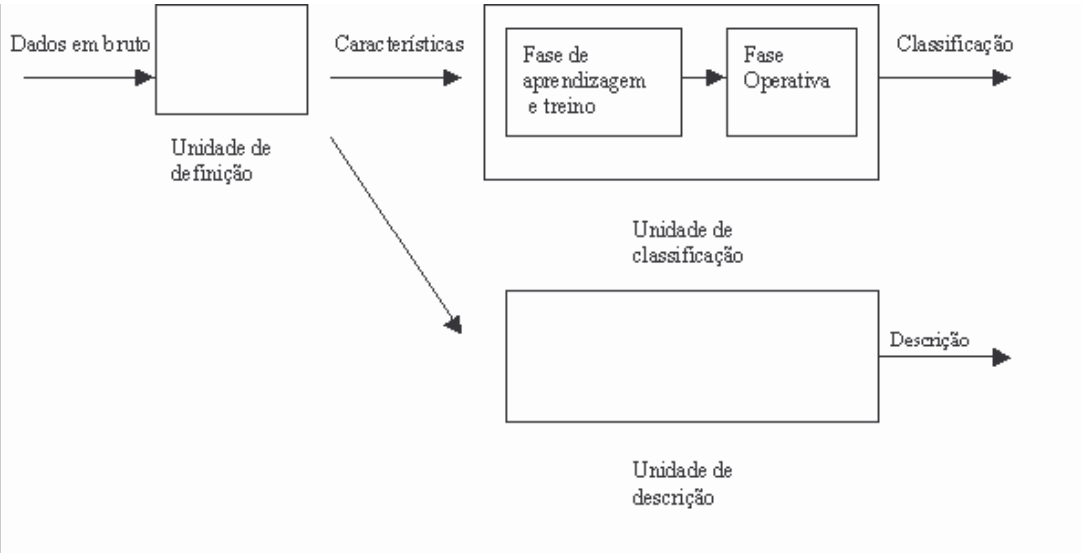


Figura 1.4 – Sistema genérico de r.p..

Estas ideias indiciam que a divisão do todo em partes tem uma analogia directa com a teoria formal da linguagem. Um texto pode ser descrito como um conjunto de frases, uma frase como um conjunto de palavras, uma palavra como um conjunto de letras,... Consequentemente, os termos **reconhecimento de padrões linguístico** e **reconhecimento sintáctico de padrões** (ou *parsing*) têm sido utilizados para descrever as metodologias baseadas nestas ideias. Mas, tal como refere Pavlidis (1977, Cap. 1, pág. 3-4) a teoria formal das linguagens, na qual a estrutura dos objectos é utilizada para a descrição, constitui apenas um aspecto desta metodologia. Por esta razão, no fim dos anos sessenta, passou a utilizar-se o termo **reconhecimento de padrões estrutural**. O objectivo consiste na descrição dos objectos com base em modelos matemáticos adequados. Este processo torna-se assim relativamente simples: se se utilizarem ferramentas sintácticas, as correspondentes gramáticas devem ter poucas regras e serem facilmente analisáveis. Se for utilizada uma abordagem geométrica, cai-se no que já aqui foi referido anteriormente, pontos que correspondam a objectos da mesma classe devem estar próximos de acordo com alguma métrica.

Duda e Hart (1973) dedicam alguns capítulos à apresentação dos fundamentos da *scene analysis* começando por referir alguns exemplos onde este tipo de abordagem se torna particularmente útil. O que se pretende neste caso é responder a questões do tipo. “esta figura mostra, por exemplo, uma cadeira?”, “indique todos os objectos que

aparecem na figura", "estas duas figuras mostram o mesmo objecto?", ou ainda de uma forma mais geral, "descreva esta figura".

1.3 As várias abordagens

Devido à sua importância de natureza prática, o r.p. tem sido um campo de pesquisa muito activo. Muitas das primeiras tentativas nesta área resultaram da utilização de uma abordagem baseada na teoria das configurações apelidada de *Gestalt*. O r.p. tornou-se nesta fase como sendo a procura de regiões no espaço onde caem os pontos que permitem a identificação de determinado padrão. Esta forma de abordar o problema surgiu de ideias muito simples. Os padrões podem ser representados por vectores de números resultantes das medições efectuadas a partir de uma amostra; estes números podem ser interpretados como coordenadas de pontos num determinado espaço com tantas dimensões quanto o número de entradas do correspondente vector. Escolhidas as medidas apropriadas, os pontos que correspondam a objectos com o mesmo padrão, estarão "próximos" em termos de distância geométrica. Parece então plausível que se assuma que os pontos que representam padrões idênticos tendam a agrupar-se de acordo com alguma métrica. Foi esta interpretação que levou à construção da estrutura que permitiu o recente desenvolvimento nesta área.

Um dos primeiros projectos associados ao r.p., apresentado por esta comunidade nos anos 50 (do século XX) foi o *perceptron*. As suas raízes derivam do trabalho de Fisher para a análise discriminante linear com duas classes que aliás constitui a primeira referência histórica de análise discriminante (Fisher, 1936). O *perceptron*, resulta também num modelo simples com uma diferença, a sua construção é baseada não em termos de um critério de separabilidade mas sim num algoritmo de optimização construído com o objectivo da minimização do número de observações mal classificadas, em suma um simples classificador em duas classes baseado numa combinação linear dos vectores medida.[5]

[5] Mais à frente irá ser abordado novamente este método, devido, nomeadamente à importância que assumiu para o desenvolvimento de outros trabalhos no contexto das redes neuronais.

Nos anos 60 e 70 (do século XX) as atenções viraram-se essencialmente para a extracção de medidas e para a forma de as combinar e transformar por forma a constituírem as características necessárias para a obtenção de "bons" classificadores (Hand, 1997). E foi, de certa forma, com o surgimento do p*erceptron*, que se começaram também a implementar métodos de escolha automática das mesmas.

O *perceptron* sobreviveu a anos de críticas severas nomeadamente pela investigação desenvolvida pela comunidade das ciências dos computadores. No entanto, o trabalho foi continuado e deu grandes frutos.

Uma classe de métodos, que surgiu em meados dos anos 80 (do século XX) e que veio pôr fim a um vasto conjunto de problemas, designadamente a implementação de estruturas complexas (atraindo ao mesmo tempo grandes massas), foi a classe dos chamados modelos de redes neuronais. Esta classe de modelos traduz-se numa combinação linear de transformações não lineares de combinações lineares de transformações não lineares... ; em destaque portanto, o facto das variáveis em bruto serem transformadas antes de combinadas (Hand, 1996).

Através dos exemplos que têm sido referidos, não restam dúvidas que as aplicações do r.p. surgem nas mais variadas formas. Em alguns casos existe uma base estatística subjacente para a geração de padrões. Noutros, a própria estrutura subjacente permite que se retire a informação fundamental para o reconhecimento. Noutros ainda, não há enquadramento em nenhuma das situações anteriores pelo que será necessário desenvolver sistemas que permitam associar o *input* a respostas que sejam "desejáveis". Daqui se infere que um problema particular pode dar origem à utilização de uma ou várias abordagens.

São sugeridas pelos investigadores diferentes divisões dos diferentes tipos de abordagens. De um modo geral podem destacar-se três principais: a abordagem estatística, a neuronal e a estrutural. Claro que além destes três grandes grupos aparecem na literatura algumas mais, embora sejam casos particulares (ou mesmo "generalizações") das anteriores. É o caso da abordagem *black box*, da abordagem gráfica (Shalkoff, 1992, Cap.8) ou, por exemplo, da abordagem pelo método dos conjuntos imprecisos (teoria dos *fuzzy sets*), descrita com pormenor em Bezdek (1992).

A abordagem *black box* é bastante adequada quando é necessária a utilização de um grande número de parâmetros como é o caso da análise de imagens (ver exemplos desta aplicação em Leondes, Cap. 2, 1998). Um sistema deste tipo é caracterizado por dois ingredientes, um computador e uma amostra de treino *input/output* tendo no entanto uma particularidade que a irá distinguir da abordagem neuronal (na forma como irá ser aqui apresentada). Esta particularidade, que aliás tem a ver com a própria designação, baseia-se no facto de não interessar o que se passa ao nível interno do sistema, embora seja implementado computacionalmente. Uma outra referência para sistemas deste tipo é o livro de Shalkoff (1992), onde aliás se estabelece a analogia com o cérebro humano para a concretização de objectivos ligados à implementação de problemas de inteligência artificial.

Uma outra área, actualmente com grande visibilidade, e com grande interacção com o r.p. na medida em que são utilizadas muitas técnicas comuns, é o *data mining*. Este termo está relacionado com exploração e procura de estruturas em grandes conjuntos de dados. Estes conjuntos de dados podem ter várias proveniências, não tendo sido geralmente recolhidos por amostragem nem com qualquer objectivo específico, apresentando por esta razão grandes deficiências (tais como valores omissos ou valores mal introduzidos). Hand (2000) define o *data mining* como sendo "as descobertas de interesse, inesperadas, ou estruturas valiosas em grandes conjuntos de dados". Neste artigo de Hand podem ser encontradas várias referências bibliográficas acerca de toda a problemática que envolve este tema.

Friedman (1997) relaciona a estatística e o *data mining* apresentando definições propostas por vários investigadores, aplicações, sugestões e "críticas" ao *data mining*, lançando inclusive algumas ideias que deixam em aberto o facto do *data mining* poder fazer parte integrante da estatística. Rocke (1998) apresenta a sua perspectiva de como as ferramentas estatísticas podem e devem ser utilizadas no *data mining* alertando para as vantagens e desvantagens da utilização de alguns métodos. A estatística trata da análise de dados e visto desta forma o *data mining* deve ser tido como um ramo da estatística. No entanto este campo tem sido desenvolvido de uma forma praticamente independente. Mas quais as razões desta separação? Uma das razões baseia-se no facto dos métodos serem desenvolvidos por pessoas que necessitam de resolver os problemas

com que se deparam e estes raramente incluem investigadores cuja área primária de interesse é a teoria e a metodologia estatística. Rocke (1998) contribui para esta discussão apontando algumas referências que indicam caminhos através dos quais as ideias vindas da estatística podem ter grande utilidade. No entanto existem outros problemas mais críticos. Tradicionalmente os métodos estatísticos foram desenvolvidos para fazer o maior uso da dispersão dos dados. O conceito de eficiência, por exemplo, é crucial quando os dados são "caros" e menos importante quando o "preço" da obtenção de novos casos é apenas o de ir carregar mais informação ao disco. Por outro lado a eficiência computacional deixou de ter um grande papel aquando dos avanços computacionais. O que acontece é que quando se analisam vastos conjuntos de dados as prioridades alteram-se, chegando mesmo a inverter-se. Talvez seja esta a razão que mais tem pesado para que a utilização dos métodos estatísticos seja pouco comum no campo do *data mining*. Mas algumas dificuldades que surgem não têm só a ver com a grande quantidade de dados; há muitos métodos exploratórios em que as respostas são claras e fáceis de obter independentemente da dimensão dos dados, é o caso das medidas de localização ou dispersão clássicas ou mesmo da estimação dos coeficientes duma regressão. No entanto há muitos problemas no *data mining* onde ocorre aquilo que é apelidado de "estrutura delicada" (*subtle structure*) no sentido de serem estruturas difíceis de encontrar mesmo em grandes conjuntos de dados. A detecção de *outliers* em dados multivariados e a análise de *clusters* caiem nesta categoria. No caso dos *clusters*, por exemplo, a grande quantidade de dados dificulta a procura da forma como os dados se dividem em nuvens. O melhor óptimo global pode ser fácil de encontrar em pequenos conjuntos de dados mas procedimentos computacionais bons com pequenos conjuntos de dados podem tornar-se impraticáveis com conjuntos grandes. Um exemplo bastante importante é o caso do estimador MCD proposto em Rousseeuw (1985) (ver também Rousseeuw e Leroy, 1987) onde o esforço computacional aumenta exponencialmente com a dimensão da amostra. Rocke (1998) explica com detalhe o que acontece para determinadas dimensões amostrais particulares. Em suma, uma perspectiva do ponto de vista estatístico no *data mining* poderá trazer grandes benefícios e deverá ser tomada em consideração em todo o processo devendo os métodos estatísticos ser utilizados independentemente do conhecimento *a priori* da estrutura dos dados. Rocke (1998) tece ainda algumas recomendações específicas entre as quais se destaca a seguinte:

> *" Although data mining has developed mostly independently of statistics as a discipline, a fusion of ideas from these two fields will lead to better methods for the analysis of massive data sets"*

Rocke e Woodruff (1998) acrescentam mais alguns comentários ao artigo de Rocke (1998) no que diz respeito a técnicas estatísticas de aplicação no *data mining*, nomeadamente com considerações sobre a detecção de *outliers* em dados multivariados e à análise de *clusters*. Hastie *et al.* (2001) apresentam uma série de conceitos e técnicas numa tentativa de juntar ideias novas que têm surgido no seio da aprendizagem[6] e com o objectivo de as inserir no âmbito da estatística. Embora sejam necessários alguns detalhes matemáticos, é dada maior ênfase aos métodos e às suas bases conceptuais do que propriamente às suas propriedades teóricas. Visto desta forma aquilo que estes autores esperam é que este livro sirva não só a comunidade estatística mas também investigadores e profissionais de uma vasta variedade de áreas. Sendo assim, e como o próprio nome indica (*The Elements of Statistical Learning: Data Mining, Inference and Prediction*) este livro também constitui uma boa referência para o *data mining*. Uma outra referência também interessante, mas do ponto de vista de uma aplicação prática concreta, é o artigo de Hébrail (2000).

Voltando às três grandes abordagens referidas anteriormente, e pelas considerações acerca de outras que aqui foram referidas, é fácil concluir que os respectivos limites são imprecisos e que elas partilham objectivos e características comuns. O que é mais natural é que, para um problema específico, se escolha a abordagem ou abordagens mais adequada(s). Schalkoff (1992, Cap. 1, pág. 21) apresenta uma tabela que resume as principais diferenças entre as três abordagens tradicionais.

A mais profunda dicotomia surge entre as abordagens estatística e sintáctica, embora também seja feita, por muitos autores, uma separação entre a abordagem estatística e a neuronal. Mas esta separação só poderá ser útil se quisermos olhar as duas como totalmente desligadas constituindo até, alternativas competitivas. Leondes (1998, pág. 21) refere que de uma forma extremista, as redes neuronais podem ser vistas como caminhos iterativos para atingir resultados clássicos dos métodos estatísticos

[6] No sentido referido na Secção 1.2.2.

tradicionais. Talvez o ideal fosse descrever todas estas abordagens através da utilização de termos comuns. Afinal o r.p. não é simplesmente o automatizar dos procedimentos realizados pelos humanos?

1.3.1. A abordagem estatística

Devijver e Kitler (1982) afirmam que a estatística e a teoria das probabilidades têm muito para oferecer na área do r.p.. E porque é que esta afirmação é válida? Por aquilo que já foi aqui referido julga-se óbvio que estes exemplificam todos os objectivos principais da estatística. Desde a formulação de problemas, passando pela recolha e análise de dados, estimação, validação, interpretação, culminando na tomada de decisão, ou ainda, na previsão.

Em algumas situações, por exemplo, as distribuições estatísticas multivariadas poderão ser utilizadas tanto como um modelo adequado para descrever a variabilidade dos padrões como também a própria geração dos mesmos. Todas as áreas da análise estatística multivariada têm ligações muito próximas com o r.p, em particular a análise discriminante (a qual, aliás, irá ser alvo de especial destaque nesta dissertação). A teoria da decisão, ou mesmo os testes de hipóteses, poderão contribuir para a questão de um dado padrão pertencer, ou não, a determinada classe.

Já que os problemas nesta área têm uma componente estatística bastante forte, deverão ser abordados como um procedimento de classificação onde se pretende minimizar a probabilidade de erro. Afinal não restam dúvidas que de uma forma sucinta o problema está associado à escolha de variáveis, recolha de amostras e divisão de espaços em regiões. O objectivo será então o de encontrar determinada função de classificação, o que implica que de uma forma ou de outra, terá de se proceder à estimação de parâmetros.

Por outro lado, será que a informação sugerida pela amostra se comporta do mesmo modo quando se obtêm novas amostras? O mais prudente seria recolher mais amostras e verificar quantas estariam correctamente classificadas. Infelizmente nem sempre isto é possível.

Todas as questões aqui levantadas encaminham-nos para situações em que existe um fundamento estatístico, quanto mais não seja porque o homem não pensa de uma forma linear; joga em simultâneo com vários dados que incrementam a complexidade do raciocínio e fazem com que este acabe por se mover no terreno do provável, levando ao aparecimento de um conjunto de hipóteses. O próprio raciocínio nunca é uma certeza, como na matemática. A estatística impõe rigor e exactidão necessários na análise de situações contempladas na área do r.p. Associada à estatística outra área fundamental serve-lhe de apoio (assim como a todas as outras), a computação (ver Hand, 1996).

Claro que, como é óbvio, nem tudo são vantagens. E a principal desvantagem apontada à abordagem estatística é a sua simplicidade em expressar relações estruturais; a estrutura do padrão é considerada de pouca relevância (ou mesmo insignificante, na opinião de alguns). Toda a estrutura é descoberta a partir dos dados, embora, pela escolha de variáveis adequadas, possa reflectir a ausência ou presença de determinadas relações. Claro que se esta informação não está disponível ou é irrelevante, se deve optar pela abordagem estatística.

Dependendo da informação disponibilizada, das suposições a ser efectuadas e das propriedades que se pretendem salientar, ainda poderá ser feita uma subdivisão em três sub-abordagens: a paramétrica, semi-paramétrica e não paramétrica. E uma vez que esta abordagem é aquela que irá dominar e prevalecer nos capítulos que se seguem nesta dissertação, deixar-se-ão para mais tarde considerações mais detalhadas.

1.3.2. A abordagem neuronal

As redes neuronais ou neurais, também conhecidas por ANN (*artificial neural networks*) ou outras designações mais específicas como *parallel distributed processing model* (PDF) são, de uma forma genérica, uma colecção de modelos matemáticos que se baseiam nas propriedades observadas do sistema nervoso humano. Ou ainda, entrando no campo da informática, um sistema de computação constituído por um conjunto de unidades de processamento interligadas que processam a informação, com capacidade de aprender, memorizar e criar relações a partir dos dados. Daí que, por vezes as redes

neuronais sejam descritas como sistemas de ligação devido às próprias ligações existentes entre unidades de processamento individuais. E, apesar da simplicidade de cada unidade de processamento, a utilização de um número elevado de unidades interligadas permite a execução de tarefas complexas.

Como é sabido, os computadores tradicionais (digitais) tiveram uma evolução notável e possuem um elevado poder de cálculo. No entanto, não conseguem realizar facilmente tarefas que o cérebro humano efectua continuamente, como o reconhecimento de objectos ou da fala. Este tipo de computadores centra-se na construção de programas que resolvem determinado problema através de séries pré determinadas de passos, os algoritmos. Os programas são controlados por uma única unidade de processamento central e armazenam informação em localizações específicas de memória: trata-se de uma estrutura de processamento em série em que tudo acontece como uma sequência de operações em contraste com o cérebro humano que realiza as suas operações de forma paralela e distribuída.

Uma rede neuronal não é sequencial nem necessariamente determinística; não tem uma memória para armazenamento de dados. É composta por um vasto conjunto de elementos de simples processamento que formam uma soma "pesada" do *input*. Não executa uma série de instruções, responde ao *input* que lhe é apresentado. O resultado não é armazenado numa posição de memória específica, consiste sim, no estado geral da rede depois de ter alcançado alguma condição. Não se pode aceder a um dado endereço de memória, por exemplo o 1532, e recuperar determinado valor. O conhecimento é armazenado nos chamados "sinapses" (por analogia aos pontos de contacto entre neurónios do cérebro humano) de uma forma diferente, tanto pelo processo como os elementos processadores estão interligados (a forma como um determinado *output* está ligado ao *input* de outro elemento processador) como pela sua importância (medida pelo correspondente "peso"). Torna-se assim uma função da arquitectura da rede e não do conteúdo de uma localização particular; o controlo fica assim distribuído pelas ligações entre os vários neurónios.

Não pretendendo replicar o cérebro humano, as redes neuronais consistem em simulações da forma como actuam as células nervosas proporcionando assim soluções

para determinados problemas que requeriam a intervenção directa dos humanos e o processo do pensamento. Por outro lado, o próprio cérebro humano tem limitações: os neurónios são propensos a falhas, morrem com frequência e são irregulares; os computadores têm de operar com "perfeição".

Embora, na maior parte da literatura se refira o aparecimento da teoria das redes neuronais com início na década de 40 (do século XX), ela realmente começou em 1890 com a publicação da teoria do funcionamento do cérebro pelo psicólogo americano William James. De entre muitas outras ideias relevantes, a teoria proposta por James (1890) refere que a eficiência da transmissão entre os estímulos o correspondente *output* aumenta significativamente com a aprendizagem.

Os trabalhos normalmente designadas como pioneiros apareceram em 1943 com McCulloch e Pitts na forma de um modelo simples, mas poderoso, de um neurónio real e que não é mais do que uma combinação linear de *inputs*, seguida de uma decisão binária (McCulloch e Pitts, 1948). Apesar da sua simplicidade, este modelo constitui a estrutura de processamento básica pelo que ainda se encontra em uso na maioria dos modelos de redes neuronais artificiais. Este modelo é baseado na plausibilidade neurofisiológica de um neurónio onde os pesos nas conexões (capazes de serem adaptados por um processo de aprendizagem de forma a que a rede desempenhe a função desejada) correspondem às sinapses inibidoras e excitadoras de um neurónio real.

A primeira teoria neurofisiológica para modificação de sinapses em neurónio reais foi proposta por Hebb (1949). Segundo ele, o fortalecimento da conexão entre neurónios deve ser aumentada se estes forem activados em conjunto, o que deu origem ao aparecimento da conhecida regra de aprendizagem de Hebb (ver por exemplo Friedman e Kandel, Cap. 8, 1999 ou Olmsted, 1998). Esta ideia foi adaptada e deu bastantes frutos, nomeadamente nos trabalhos de Widrow e Hoff (1960), autores da regra delta (ou *Least Mean Square*, LMS), uma das mais conhecidas regras de aprendizagem e da utilização dos sistemas *adalines* (*adaptive linear element*) bem como no trabalho de Rosenblatt o qual iniciou uma nova fase com a apresentação do *perceptron* em 1958 (Rosenblatt, 1958). Isto causou grande impacto na comunidade científica que viu nele a possibilidade de realizar tarefas de classificação com treino

automático a partir dos dados, de uma forma inspirada no cérebro humano. O *perceptron* (ou perceptrão, tradução adaptada, por exemplo, por Marques, 1999) é uma rede constituída por várias unidades interligadas estando estas unidades, frequentemente organizadas em camadas (*layers*). As entradas ou *inputs* de uma camada são saídas ou *outputs* da camada anterior. Cada camada é descrita através de um modelo de McCulloch e Pitts seguida de uma decisão binária ou contínua (sigmóide). Quando a rede é constituída por uma única camada, é designada por *classic perceptron* e quando tem várias camadas por *multi-layer perceptron,* MLP (perceptrão de multicamada). A última camada é geralmente designada por camada de saída, e as anteriores, cujas saídas são internas, são as camadas escondidas (*hidden layers*).

No entanto o *perceptron* teve uma vida curta nomeadamente devido à publicação do livro de Minsky e Papert (1969) onde são demonstradas formalmente as suas limitações. Uma das dificuldades surge quando os dados não são linearmente separáveis- o algoritmo de aprendizagem nunca termina. Mesmo que se pare não há garantias que o vector de pesos encontrado tenha um bom comportamento em novos dados, i.e. na sua generalização (Bishop, 1995). O que acontece é que a superfície de decisão de um *perceptron* não é mais do que um hiperplano. É aí que surge o MLP e com este o aparecimento do algoritmo de aprendizagem *backpropagation* (retropropagação de erro, segundo designação adoptada por Marques, 1999) através da investigação independente de vários autores em 1986 (ver Olmested, 1998) e que provocou novamente um forte interesse pela área. A par deste modelo surgiram também outros com destaque, tais como o modelo SOM (*self-organizing map*) de Kohonen (1982) motivado por Anderson (1972), o modelo de Hopfield em 1982 e as máquinas de Boltzman em 1985, estes últimos ligados à classificação não supervisionada (ver, por exemplo, Leondes, 1998).

Um outro tipo de rede que concorre com o MLP, aparece com Powell (1987) e mais tarde em 1995 com Lowe (1995) no contexto da literatura de técnicas para interpolação exacta de funções em espaços multi-dimensionais e é designada por *radial basis functions*, RBF (funções de base radiais). Matematicamente podem ser descritas como combinações lineares de funções base não lineares radiais simétricas; um modelo aliás bastante semelhante ao de *Kernel* (qual irá ser referido mais à frente já que se inclui no contexto da abordagem dita estatística). Esta rede é constituída por uma camada escondida, com funções de activação locais, ligada a uma camada de saída linear. A

saída é definida por uma função que depende apenas da distância do vector de características a um vector de "centros", por isso apelidada de função de base radial, sendo a escolha mais comum a função Gaussiana. As saídas de rede são somas ponderadas dessas mesmas funções.

As RBF têm um treino mais rápido do que o MLP e permitem interpretar a contribuição de cada unidade para o comportamento global da rede, pois cada unidade só é activada numa zona limitada do espaço de *input* (Marques, 1999, Cap. 5, pág. 181). Vantagens, desvantagens, motivações, propriedades e recomendações de utilização tanto do MLP como do RBF, podem ser encontradas numa vasta literatura de forma bastante explicita e pormenorizada de que são exemplo Bishop (1995, 1998), Webb (1999) e Leondes (1998), ou através de uma abordagem uma pouco mais superficial para quem inicia este tema, em Marques (1999). Ainda de um ponto de vista mais prático e crítico, em Alder (1997). Uma boa descrição de cada tipo de rede no contexto histórico é realizada por Olmsted (1998).

Em suma, uma rede neuronal é composta por dois elementos primários chave: os elementos processadores ou neurónios que recebem o *input* e geram o *output* e as interligações entre estes que determinam a arquitectura da rede. Numa rede neuronal:

- O processamento da informação é feito através de um vasto número de elementos processadores ou **neurónios**.
- A informação, por vezes também designada por sinal, é transmitida entre neurónios através de **elos de ligação**.
- A cada elo é atribuído um **peso** que propaga o sinal transmitido.
- Cada neurónio utiliza uma **função de activação,** não linear que é a soma pesada dos sinais de *input*, para obter o sinal de *output*.

Com base nesta descrição poder-se-á dizer que uma rede neuronal é caracterizada pelos seguintes elementos:

- Elos de ligação entre os neurónios que definem a arquitectura de rede (informação sobre o número de neurónios em cada camada e as suas ligações);
- Método para a obtenção dos pesos, ou seja, o algoritmo de aprendizagem;
- Função de activação (geralmente não linear);
- Controlo distribuído em forma paralela.

Poderá parecer, neste momento, por tudo aquilo que foi referido que os computadores tradicionais são um tanto antagónicos às redes neuronais. Tal não é verdade, enquanto que o computador tradicional tem de ser programado utilizando-se algum tipo de linguagem simbólica, por forma a prepará-lo para o seu trabalho, as redes neuronais são programadas por ajustamento dos seus pesos com base na optimização matemática. Através de uma escolha conveniente da função de activação pode-se fazer com que a rede sirva uma série de objectivos diferentes. Mais ainda, a maior parte do trabalho carece de computadores, embora estes sejam utilizados de uma forma diferente da tradicional. O *software* que realiza esta tarefa é geralmente conhecido por "*netware*" ou "*neural network simulation software"*. Hoje em dia, até alguns dos *packages* estatísticos mais utilizados já incluem módulos de redes neuronais: é o caso do STATISTICA. Neste *package* as redes neuronais estão incluídas no módulo das técnicas de *data mining* onde se poderá ler[7]:

> "*StatSoft's STATISTICA Neural Networks software contains the most comprehensive selection of neural network methods with intelligent problem solvers and automatic wizards; a C-code generator add-on is available. It also features:*
> *Intelligent Problem Solver Wizard,*
> *Automatic Search for Best Architecture,*
> *Multilayer Perceptrons,*
> *Radial Basis Function Networks, and Many Others,*
> *Self-Organizing Feature Map,*
> *Back Propogation,*

[7] Para mais detalhes ver http://www.statsoft.com/textbook/stathome.html)

Conjugate Gradient Descent,
Numerous Analytical Graphs,
Resampling (Cross Validation, Bootstrap),
Sensitivity Analysis, ROC Curves,
Network Ensembles,
API Interface,
and much, much more...”

Neste contexto e de uma forma simplista, o processo de aprendizagem consiste na procura de um óptimo local com base na modificação de um conjunto de pesos (o simular da intensidade de transferência de informação) ou ainda na procura de hiperplanos que separem o espaço das características em regiões de decisão. No entanto, de uma forma mais rigorosa, a questão da aprendizagem envolve duas fases: a escolha da arquitectura e só depois o cálculo dos pesos. Só a primeira requer antes de mais que se proceda a uma escolha apropriada da complexidade ou da ordem do modelo em dois sentidos, tanto no que envolve o número de funções e a complexidade das mesmas, como na definição do número de camadas, do número de neurónios por camada e das ligações. A resposta para esta fase está obviamente dependente dos dados na medida em que há um “*interplay*” entre a ordem do modelo, a dimensão da amostra de teste e a dimensão da amostra de treino, questão que aliás está muito em voga (Webb, 1999, Cap. 5, pág. 134) pela inexistência de uma equação que as relacione (pelo menos para dados de dimensão não elevada). Por consequência esta operação é realizada de forma não automática, com base em critérios heurísticos e dependentes da experiência do utilizador. Quanto à segunda fase, embora de mais fácil resolução, envolve procedimentos associados a critérios (como por exemplo, a minimização do risco empírico) de optimização não linear que poderão na prática adquirir uma certa complexidade.

A aprendizagem ocorre através do treino, já que utiliza exemplos (“exposição” a um conjunto do qual se conhece o verdadeiro *input* e *output*) através de um algoritmo iterativo que ajusta os pesos das ligações (sinapses) por forma a que o sistema possa aprender as relações existentes e armazenar o conhecimento necessário para a resolução de problemas específicos. Uma vez estabelecidas tais relações podem tomar-se novos

inputs por forma a serem feitas previsões. Mais tarde retomar-se-á a questão da aprendizagem.

A Figura 1.5 seguinte mostra uma rede neuronal simples com X_1, X_2, ..., X_n como neurónios de *input*, Y como neurónio escondido que activa o *input* através da função de activação e que permite gerar o *output* e onde w_1, w_2, ..., w_n são os pesos associados às ligações entre os neurónios de *input* e o neurónio escondido.

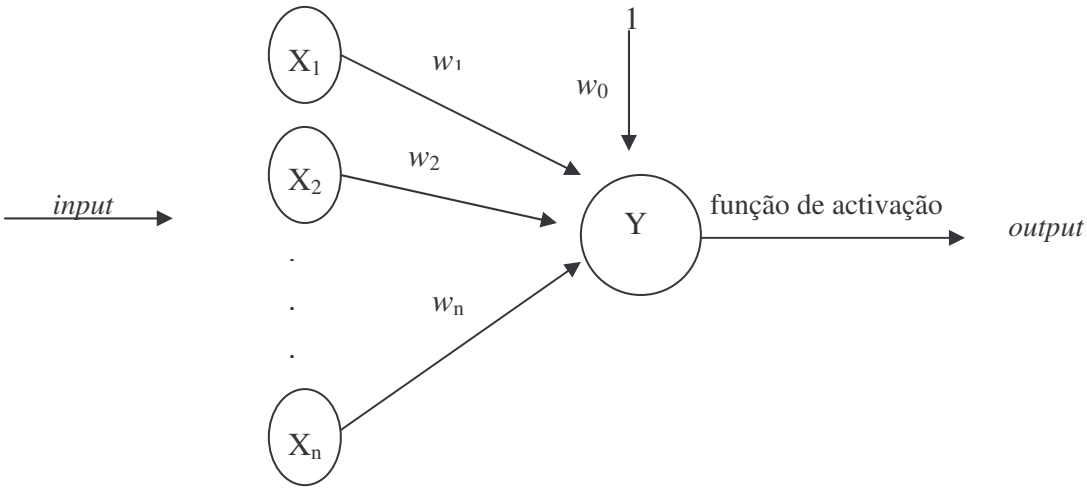


Figura 1.5- Rede neuronal com um neurónio escondido.

Poderá também incluir-se uma componente correspondente à ordenada na origem ou termo independente, apelidada por alguns de enviesamento (ver Friedman e Kandel, 1999, Cap. 8), bastando para tal que se adicione a constante 1 ao vector X. Designando por *f* a função de activação e por res o *output*, então res = $f(s)$ e $s = w^T X$ onde $\boldsymbol{w}^T = (w_0 , w_1 , ..., w_n)$ e $\mathbf{X} = (1 , X_1 , ..., X_n)$.

Conforme já foi focado é habitual e também, de uma forma geral, mais conveniente, tratar os neurónios em camadas. Assim dentro de cada camada todos os neurónios têm a mesma activação e o mesmo padrão de ligação. Por exemplo, se um neurónio da primeira camada tem um elo de ligação com um neurónio da segunda camada, então cada neurónio da primeira está ligado com todo o neurónio da segunda.

A função de activação, por vezes também designada por "*squashing function*", pode assumir várias formas estando naturalmente dependente da aplicação. Algumas são utilizadas frequentemente, a saber,

- A função de passo binário que atribui ao *output* o valor 1 caso X exceda determinado valor, designado na literatura por "*threshold"*, e 0 caso contrário,
- A função bipolar que atribui os valores +1 ou –1,
- A função sigmóide.

Esta última, descrita por $f(x) = \frac{1}{1+e^{-ax}}$ com a constante *a* geralmente igual a 1, é particularmente útil nas redes neuronais que são treinadas pelo método *backpropagation*. Nalguns casos torna-se ainda mais fácil lidar com a rede se a função sigmóide for "alisada" à custa da tangente hiperbólica, $\tanh(x) = \frac{e^{ax} - e^{-ax}}{e^{ax} + e^{-ax}}$.

As redes neuronais são caracterizadas e diferenciadas dos restantes métodos essencialmente pelo seu processo de aprendizagem, sendo com frequência úteis em problemas de classificação. Há inúmeras vantagens que podem ser apontadas a este tipo de abordagem. A aplicação desta metodologia permite generalizar a tomada de decisão sendo útil para a obtenção de boas previsões. Além disso fornece soluções eficazes para uma variedade de problemas tradicionais na área do r.p. tal como o reconhecimento de caracteres ou a modelação de sistemas onde os processos físicos não são bem compreendidos ou são bastante complexos. Podem e devem também ser aplicadas em problemas onde existem distorções nos dados pela forma como lidam com grande elasticidade contra ruídos, dados incertos ou mesmo dados omissos.

As redes neuronais também têm mostrado um bom desempenho na resolução de problemas demasiado complexos (por exemplo em problemas que não têm solução algorítmica), nomeadamente naqueles em que os humanos conseguem solucionar facilmente mas as técnicas tradicionais não. Isto porque as redes caracterizam-se não só pela capacidade de aprendizagem mas também pela simplicidade com que efectuam operações elementares e pelos curtos espaços de tempo que são necessários para operarem; não há necessidade de se desenvolver nenhum programa, podem ser

implementadas em *hardware* paralelo para uma grande velocidade de operação. Mais ainda, podem trabalhar com um grande número de variáveis ou parâmetros conseguindo contornar-se de uma forma simples o problema da relação entre o tamanho da amostra e o número de parâmetros. As densidades são estimadas directamente sem necessidade de se assumir qualquer modelo estatístico.

Digno de nota é a opinião de Smith (1996) acerca da vantagem da utilização das redes neuronais:

> *"Neural networks cannot do anything that cannot be done using traditional computing techniques, BUT they can do some things which would otherwise be very difficult. In particular, they can form a model from their training data (or possibly input data) alone… there may be an algorithm, but is not known, or has too many variables. It is easier to let the network learn from examples…"*

Para terminar as considerações que, no fundo, visam servir de suporte à escolha entre a abordagem estatística e a abordagem neuronal apresenta-se de seguida uma descrição dos aspectos positivos e negativos de cada abordagem, realizada recentemente por Raudys (2001). Raudys (1998, 2001) desenvolve uma nova abordagem que apelida de **abordagem integrada** e que vai de encontro à conjectura efectuada no último parágrafo da Secção 1.3 (pág. 20).

Raudys, (2001). *Statistical and Neural Classifiers: an Integrated Approach to Design*

Na abordagem estatística fazem-se suposições acerca das densidades condicionais às classes e escolhe-se o critério de *performance*. De seguida utiliza-se uma amostra de treino para construção de uma regra de decisão óptima.

Aspectos positivos da **abordagem estatística**:

- possibilidade de obtenção de algoritmos rigorosamente óptimos.

- possibilidade de usar informação contida nas hipótese estatísticas acerca das densidades condicionais às classes. No caso de existirem fortes indícios de que as hipóteses estão correctas, a amostra de treino pode ser de pequena dimensão.
- A p*erformance* em termos das relações entre os tamanhos das diferentes amostras de treino, para os algoritmos estatísticos de classificação mais populares, é conhecida.

Aspectos negativos da **abordagem estatística**:

- Em muitos casos práticos desconhecem-se as densidades condicionais às classes. O modelo escolhido pode estar muito longe do verdadeiro levando a um aumento significativo do erro de classificação.
- A técnica *plug-in* para obtenção de regras de classificação baseadas na amostra não é óptima no caso de amostras de treino de diferentes dimensões.
- A técnica preditiva de Bayes para obtenção de regras de classificação baseadas na amostra requer a escolha das distribuições de probabilidade *a priori* dos parâmetros desconhecidos.
- Em muitos modelos de dados multivariados, tanto as expressões analíticas que formam a regra de decisão como, em relação à *performance*, as relações com as dimensões das amostra de treino são muito complexas e não podem ser aplicadas de uma forma prática.
- Algoritmos estatísticos de classificação complexos requerem que se estime um grande número de parâmetros o que se traduz num grave problema para aplicações com elevado número de variáveis (*curse of dimensionality*).
- Para cada problema de r.p. particular tem de ser escolhido o melhor algoritmo de um vasto número de possíveis candidatos.

Na abordagem neuronal fazem-se suposições acerca do tipo de contorno de decisão e utiliza-se directamente a amostra de treino para determinar os coeficientes desconhecidos (pesos) da regra de classificação.

Aspectos positivos da **abordagem neuronal**:

- Não é necessário assumir qualquer hipótese acerca da densidades e pode utilizar-se a amostra de treino directamente por forma a determinar os coeficientes desconhecidos (pesos) da regra de decisão.

- As redes neuronais constituem aproximações universais.
- Durante o treino dos classificadores *perceptron* , MLP e RBF, poderá obter-se vários classificadores estatísticos de diferentes complexidades.
- Em alguns casos, *perceptron*, MLP e RBF podem ter boas propriedades para amostras de treino de pequenas dimensões.
- A informação contida em quase todos os pesos iniciais correctos do classificador *perceptron* pode ser guardada se pararmos, por obtenção da solução óptima, o processo de treino.

Aspectos negativos da **abordagem neuronal**:

- Tem de ser escolhida a arquitectura da rede e fixado o número de parâmetros.
- O treino tem de ser parado algum processo óptimo.
- Com excepção de alguns casos associados a diferentes níveis do treino do *perceptron* e para contornos errados irrealistas, a *performance* em termos das relações entre as dimensões das amostras de treino é desconhecida.
- O processo de treino é afectado pela singularidade dos dados, a inicialização dos pesos e os mínimos locais. Frequentemente o processo de treino é muito lento e muitos autores estão quase certos ao dizer que "o treino das redes neuronais não é uma ciência, é uma arte".

Raudys (2001) reafirma que tanto a abordagem estatística como a neuronal têm os seus aspectos positivos e negativos; sugere que em vez de nos questionarmos ou disputarmos os atributos de cada tipo de abordagem, será mais sensato explorar os aspectos positivos de cada uma delas por forma à criação de uma abordagem integrada conforme apresenta nas secções seguintes do Capítulo 5. Como conclusão sumaria em cinco pontos as principais razões para o sucesso da abordagem que apelida de integrada.

Voltando às redes neuronais, em que consiste afinal, a aprendizagem através de exemplos? Para que fique mais claro como é que este passo se processa na prática considere-se o exemplo da modelação da função lógica AND com utilização da regra Hebb como método de aprendizagem (adaptado de Friedman e Kandel, 1999, Cap. 8). Esta função é modelada por dois *inputs* X_1 e X_2 e um neurónio de *output* T, com activações bipolares x_1, x_2 e t dando origem à seguinte tabela de verdade:

x_1	x_2	*t*	T
1	1	1	1
1	-1	0	-1
-1	1	0	-1
-1	-1	0	-1

Tomando como zero o valor inicial para w_1 , w_2 , e b (peso anteriormente designado por w_0), e fazendo ,

$\Delta w_i = x_i T$, w_i (novo)=w_i (antigo)+ Δw_i para i=1,2 e Δb=T, b(novo)=b(antigo)+ Δb obtém-se o processo descrito na Tabela 1.1.

	Input			*Output*	Mudanças nos pesos			Pesos		
Passos	x_1	x_2	1	T	Δw_1	Δw_2	Δb	w_1	w_2	b
1	1	1	1	1	1	1	1	1	1	1
2	1	-1	1	-1	-1	1	-1	0	2	0
3	-1	1	1	-1	1	-1	-1	1	1	-1
4	-1	-1	1	-1	1	1	-1	2	2	-2

Tabela 1.1 - Exemplo da modelação da função lógica AND.

A linha de separação final é $2x_1+2x_2-2=0$, embora neste exemplo não fosse necessária a consideração dos quatro padrões de treino; após o terceiro passo já se obtinha a decisão final. A regra de decisão consiste em classificar uma nova observação (ou um novo objecto, no caso geral) na classe G_1 se $2x_1+2x_2-2\geq 0$ e em G_2 caso contrário.

Aqui se mostra como uma rede neuronal pode dividir o espaço de regiões em hiperplanos (note-se que nem sempre isso se verifica) por forma a que os pontos com a mesma categoria, digamos assim, caiam na mesma região. A questão que se prende com a localização dos hiperplanos, que efectuam tal divisão, é tomada recorrendo-se a um

algoritmo de treino até que se atinja a convergência ou até que uma medida de erro, previamente escolhida, seja suficientemente pequena. As redes neuronais podem demorar algum tempo neste processo de treino mas são geralmente bastante rápidas para práticas de classificação de padrões pelo facto do número de operações internas a ser efectuadas, ser geralmente inferior à quantidade de dados disponíveis (Alder, 1997)[8].

A regra de aprendizagem utilizada tem algumas limitações. No entanto, as limitações desta regra são compartilhadas com muitas outras regras de aprendizagem existindo mesmo condições suficientes para a convergência de outros processos iterativos. Por outro lado, e como é óbvio, este exemplo só foi referido para dar uma ideia da forma como se processa a aprendizagem, não interessando a utilização de um método eficiente mas sim de fácil compreensão.

Como se pode ver, o *design* de classificadores de redes neuronais obriga a numerosas escolhas que podem influenciar significativamente os resultados, quer na fase de treino, quer na fase operativa. De acordo com a aplicação, podem ser escolhidos vários tipos de rede, bem como o número de neurónios ou camadas nela inseridos e também a forma como irão estar relacionados(as). Relativamente aos neurónios incluídos, o seu número pode ser fixado à partida e manter-se inalterável durante a fase de treino ou pode ser modificado de acordo com técnicas apropriadas. Tal como é referido por Leondes (1998, pág. 167), no primeiro caso este número tem de ser estimado antes da fase de aprendizagem, com base num critério que depende da distribuição amostral no espaço das características; no segundo a escolha inicial é menos crítica mas os métodos até ao momento disponibilizados não podem ser considerados de aplicação geral uma vez que só foram testados em aplicações particulares. Muitos neurónios por camada podem causar sobreajustamento, correndo-se o risco da rede se tornar demasiado "especializada" na amostra. Pelo contrário, com poucos a rede pode não alcançar um reconhecimento "satisfatório" (mesmo para a amostra de treino).

[8] Conforme este autor explica, ao dividir o espaço em regiões, cada uma das quais prova que é convexa, pode-se concluir que se um dado ponto está cercado por outros, todos da mesma classe, então ele também pertence certamente a essa mesma classe.

Muito já foi aqui referido acerca da aprendizagem e das escolhas que têm de ser feitas neste contexto, não só em termos de algoritmos como também nas amostras ou até na própria definição do tempo óptimo para se proceder à paragem. Mais ainda, não existem apenas diferentes algoritmos como também diferentes condições iniciais, diferentes modalidades de treino (por exemplo supervisionada ou não) ou mesmo diferentes estratégias. Mesmo a ordem pela qual a amostra é processada deve ser controlada por forma a evitar efeitos que possam diminuir a *performance*.

O número de ciclos de aprendizagem poderá afectar a capacidade de generalização da rede ou seja, a sua habilidade em classificar correctamente amostras com algumas diferenças relativamente à amostra de treino. Mais uma vez, tal como na abordagem estatística, será conveniente recorrer a uma amostra de teste. Leondes (1998, pág. 168) afirma que enquanto a taxa de erro na amostra de treino decresce de uma forma monótona com o número de ciclos de aprendizagem, a taxa de erro na amostra de teste primeiro atinge um mínimo e depois aumenta; o mínimo corresponde ao número óptimo de ciclos de aprendizagem por forma a evitar sobretreino.

Resumindo, as arquitecturas de rede mais conhecidas são:

- MLP- *Multi-layer perceptron* (generalização da regra delta que por sua vez é descendente do famoso *perceptron*),
- RBF- *Radial basis functions,*
- LVQ- *learning vector quantization network,*
- SOM- *self organizing map,*
- ART- *adaptative resonance theory network,*
- PNN- *probabilistic neural network,*
- *Hopfield network.*

De acordo com a forma como os dados "circulam", desde que entram na rede como *input* até que saem na forma de *output*, pode ainda ser feita uma subdivisão em:

- FN- *Feedforward network*, onde os dados fluem por um único caminho.
- RN- *recurrent network*, onde os valores de *output* são utilizados novamente como *input*.

Além destes dois tipos poderá ainda existir uma variação intermédia que permite a existência de ligações entre neurónios da mesma camada a qual é designada por LC (*lateral connection).*

A Tabela 1.2 resume as arquitecturas de rede utilizadas com maior frequência a par da subdivisão agora apresentada, das respectivas modalidades de treino (S-supervisionada/N-não supervisionada) e do correspondente algoritmo de aprendizagem[9].

Arquitectura de rede	**Tipo de circulação de dados**	**Modalidade de treino**	**Algoritmo de aprendizagem**
MLP	FN	S	*Back-propagation*
RBF	FN	S e N	RBF
LVQ	LC	S ou N	LVQ
SOM	LC	N	Kohonen´s SOM
ART	LC	N	ART
PNN	FN	-	-
Hopfield	RN	N	Hebb

Tabela 1.2 – Algumas das arquitecturas de rede mais conhecidas (fonte: Leondes, 1998, pág. 169).

Em relação aos métodos ligados à classificação supervisionada, também se poderá fazer, resumidamente, algumas considerações. Tanto a rede MLP como a RBF são desenvolvimentos da regra delta (com origem no *perceptron*). A grande diferença entre estas duas arquitecturas é a função de activação utilizada nos neurónios escondidos. Enquanto que na primeira se utiliza a função sigmóide, na rede RBF a função de activação é Gaussiana. O número de neurónios escondidos da RBF é geralmente superior. Quanto à duração da fase de treino esta é geralmente inferior na RBF; enquanto que esta arquitectura possui apenas uma camada de neurónios escondidos a

[9] Se se pretendesse detalhar toda a metodologia associada à abordagem neuronal teria de existir mais rigor, como por exemplo deveria ser feita uma distinção entre aquilo a que se chama regra de aprendizagem e algoritmo de aprendizagem. Infelizmente, mesmo quem trata unicamente desta abordagem, ou de problemas específicos com ela relacionados, comete frequentemente um abuso de linguagem. Sugere-se que para mais detalhe se consulte Leondes (1998).

MLP poderá ter mais. Para considerações mais detalhadas sobre as diferenças entre estes dois tipos de rede ver, por exemplo, Bishop (1995, pág. 182).

O conjunto de métodos descritos nesta secção é de implementação relativamente fácil devido à grande evolução na área da computação. No entanto antes de se implementar métodos deste tipo há que em primeiro lugar pensar nas razões que justificam a necessidade deste tipo de abordagem. Será verdade, por exemplo, que a superfície de decisão é não linear? Será que a *performance* atingida com a utilização da abordagem neuronal é superior àquela que seria conseguida com a aplicação de outro tipo de abordagem? Webb (1999, Cap. 5, pág. 170) dá algumas sugestões, que designa por recomendações, entre as quais se destacam as mais gerais[10]:

- Uma técnica simples de r.p. (tal como por exemplo a discriminante linear) deve ser implementada como um "guia", antes de se avançar com uma rede neuronal.
- Há que ter cuidado na selecção do modelo para que não se corra o risco de sobretreino- sugere-se alguma forma de regularização no procedimento de treino ou a recolha de uma amostra de teste para determinação da ordem do modelo.
- Deverá ser utilizado todo o conhecimento acerca do processo de geração dos dados, incluindo ruídos, para o *design* da rede ou para o seu pre-processamento.

As aplicações das redes neuronais são quase ilimitadas e aparecem referenciadas em quase toda a literatura nesta área (desde as aplicações típicas até casos particulares) mas todas elas caem numa ou mais das seguintes categorias:

- Classificação (diagnóstico médico, reconhecimento de voz ou imagem, classificação de micróbios, verificação de assinaturas- aplicação de grande escala nos E.U.A.,...)
- Previsão (vendas futuras, indicadores económicos, necessidades energéticas, reacção de produtos químicos, riscos ambientais, meteorologia,...)
- Modelação (estruturas químicas, controlo de robots, sistemas dinâmicos,...)

[10] No sentido de não se entrar em pormenores sobre o que deverá ser feito em cada tipo de arquitectura de rede mas sim no contexto geral da abordagem neuronal.

Até à data, tem vindo a ser realizada, nesta área, uma pesquisa a nível mundial desde a metodologia básica, tal como a formulação de algoritmos de aprendizagem mais eficientes, até a redes que respondam a variações temporárias nos padrões ou técnicas que permitam implementar directamente as redes neuronais em silicone. A Universidade de Edimburgo, por exemplo, implementou um *chip* de rede neuronal e encontra-se a desenvolver processos de aprendizagem. A produção de um *chip* que seja capaz de aprender, irá permitir a aplicação desta tecnologia a um vasto conjunto de aplicações onde o preço de um PC e do respectivo *software,* deixa de ser justificado. Existe um interesse particular em aplicações sensoriais: redes que aprendem por forma a interpretar sensações reais e que aprendem com o ambiente que as rodeia.

Áreas de aplicação relativamente recentes são os Pen PC´s (PC´s onde se pode escrever numa placa e aquilo que é escrito é reconhecido e traduzido para código ASCII), sistemas de reconhecimento de visão e voz para incorporação em brinquedos, máquinas de lavar, etc., que permitem reconhecer entidades simples e que inclusivamente já estão a ser utilizados no Japão.

1.3.3. A abordagem estrutural

Embora a abordagem estrutural não seja de forma alguma objecto desta dissertação, julga-se que é importante tecer-se algumas considerações que permitam ter uma ideia genérica de como funciona este tipo de abordagem, visto que esta aparece referenciada num grande número de publicações na área do r.p.

Frequentemente a informação contida num padrão não é meramente a presença ou ausência de determinada característica, ou simplesmente um conjunto de valores numéricos. Pelo contrário, as inter-relações entre as várias medidas incluem uma informação estrutural importante que facilita a descrição ou até a classificação. Embora se tenha de extrair, quantificar e avaliar similaridades sob a informação relevante, a estrutura é a base da abordagem estrutural para o r.p.

Esta abordagem deverá ser utilizada quando o padrão a ser reconhecido está fortemente estruturado e/ou quando não há dados suficientes para o treino,

impossibilitando a utilização da abordagem estatística ou neuronal. Por conseguinte, a abordagem estrutural poderá ter alguma utilidade na resolução de problemas "particulares"[11]. Como alternativa à abordagem estrutural ainda se poderá optar pelos chamados métodos baseados em regras (*rule-based methods*) no caso em que as classes podem ser caracterizadas por relações gerais entre os objectos e não por si só (ver por exemplo Duda *et al.*, 2001- Secção. 8.8).

O aparecimento do reconhecimento estrutural de padrões data de meados dos anos 60 (do século XX) quando estava em voga a aprendizagem por meio de máquinas, sendo os primeiros conceitos introduzidos nesta área devidos a Kirsh, Ledley, Nararimhan e Shaw. Inicialmente tornou-se conhecida com a designação de reconhecimento sintáctico de padrões quando se começou a recorrer à teoria das linguagens formais para a resolução de problemas que envolviam um conjunto de amostras sintácticas. Desde o princípio dos anos 70 até meados dos anos 80 que Fu (ver referências várias em Schalkoff, 1992), foi dos que mais contribui com *papers* neste contexto. No entanto, rapidamente se concluiu que como o objectivo principal era o da análise de características e das suas relações, então a resolução deste tipo de problemas poderia ser realizada através de ferramentas nas quais se poderia incluir a utilização de *strings*, árvores ou correspondências gráficas. Foi assim que esta abordagem passou a ser conhecida como a do reconhecimento estrutural, designação introduzida por Pavlidis e que se julga mais adequada pelo seu generalismo.

Já foi aqui frisado que o objectivo principal deste tipo de abordagem é o reconhecimento de um objecto pela descrição. Esta é realizada sob as suas componentes básicas incluindo a comparação com outros possíveis objectos "armazenados em memória". Se o *input* corresponde a uma das descrições armazenadas então este é classificado nessa categoria, ou reconhecido como sendo desse tipo. Uma vez que se trabalha com dados nominais, o *input* já não é representado por vectores de números mas sim descrito por elementos básicos, designados por **primitivas** e por um conjunto de regras sintácticas especificando a forma como as primitivas podem ser concatenadas para formar um padrão legítimo dentro de cada classe de padrões. Pavlidis (1977,

[11] No sentido de serem específicos e não de casos restritos.

Secção 8.6) apresenta uma série de exemplos ilustrativos da forma como se podem transformar descrições na terminologia de linguagens naturais.

As descrições formais do r.p. estrutural utilizam muitos termos em comum com a teoria das linguagens formais. As características individuais podem ser consideradas como símbolos de um **alfabeto** que são agrupados sequencialmente numa *string* (sequência de símbolos nominais) ou frase (*string* gerada por um conjunto de regras).[12] Ao conjunto de regras que definem como se forma uma frase chama-se **gramática**. Assume-se que existe uma gramática para cada classe padrão construída com base num conjunto de padrões amostrais o que remete, consequentemente, para algo semelhante ao que se apelidou anteriormente de treino. A **linguagem** gerada pela gramática é o mecanismo que permite expressar ideias gerais[13] e a **sintaxe** a estrutura da linguagem. Claro que a utilização de gramáticas formais em aplicações deste tipo só é razoável se for possível uma decomposição do *input* num conjunto de primitivas.

A quantificação da estrutura a reconhecer, classificar ou descrever pode ser feita quer através de gramáticas, quer através de descrições relacionais, nomeadamente por métodos gráficos.

As gramáticas são compostas por quatro entidades fundamentais que correspondem à notação convencionada no sistema AnaGram e no BNF (*Backus-Naur Form*) para gramáticas ou para gramáticas de contexto livro, respectivamente (ver por exemplo glossário de termos neste contexto: *http://www.parsifalsoft.com/gloss.html#Grammar*).

- Um conjunto de primitivas (*T*), ou símbolos terminais, que não podem ser decompostos e que constituem um subconjunto do alfabeto.
- Um conjunto de símbolos não terminais (*NT*) utilizados como intermediários para se gerar o resultado final e que, por consequência, podem ser posteriormente decompostos.
- Um conjunto de produções (*P*), regras de produção ou regras de reescrita que permitam a "produção" de *strings* de uma dada linguagem, definindo a estrutura da gramática.
- Um símbolo de começo ou raiz (*S*).

[12] Apesar desta distinção entre *string* e frase, utilizar-se-á, ao longo do resto da exposição, um ou outro termo indiscriminadamente.
[13] Note-se que o termo linguagem significa apenas "palavras proferidas". Inclui, por exemplo, linguagens de programação ou linguagens utilizadas para descrever música (Shalkoff, 1992)

Os padrões complexos são transformados através de decomposições hierárquicas até que se obtenham as primitivas. Desta forma uma gramática pode ser definida por:

$$G=(T,NT,P,S)$$

e neste contexto uma linguagem gerada pela gramática é o conjunto de *strings* ou frases que satisfazem os seguintes requisitos (Shalkoff, 1992):

- Cada *string* é somente constituída por símbolos terminais.
- Cada *string* é produzida a partir de *S* utilizando *P* de *G*.

Consoante as restrições impostas, relativamente ao conjunto de símbolos terminais e não terminais, as gramáticas podem ser de vários tipos, a saber:

- Gramáticas livres (*free grammar*).
- Gramáticas de contexto sensitivo (*context-sensitive grammar*).
- Gramáticas de contexto livre (*context-free grammar*).
- Gramáticas de estados finitos ou regulares (autómatos finitos).

Estas últimas são bastante populares nomeadamente pela facilidade em se proceder à sua representação gráfica. Seja qual for o tipo de gramática utilizada, ela pode servir uma ou ambas das seguintes funções:

- Produzir uma *string*, de símbolos terminais, utilizando *P* ou seja, gerar uma frase na linguagem da gramática *G*.
- Dada uma frase e uma gramática *G*, determinar se a frase foi gerada por *G* e, caso afirmativo, a estrutura da mesma.

Sendo assim pode dizer-se que uma gramática pode ser utilizada de dois modos:

- Produção (*generative*)- tal como o próprio nome indica tem como objectivo criar uma *string* de primitivas por via das produções, *string* esta que é uma frase na linguagem considerada.

- Analítico (*analitic*)- dada uma frase avaliar se esta pode ser integrada na linguagem i.e., se pode ser gerada utilizando-se a gramática em causa e/ou, se a sua estrutura tem proveniência daquela linguagem.

Devido à possível e provável existência de ruídos que induzem ambiguidade, é comum a utilização de gramáticas estocásticas. Anteriormente não se incluía a possibilidade de ocorrência de erros, nas frases produzidas pela gramática, nem existia a possibilidade de se incorporar informação *a priori*. Por estas razões, a utilização de gramáticas estocásticas pode ser útil. A principal diferença entre estas e as gramáticas descritas anteriormente (gramáticas de *strings*) são as produções e o símbolo de começo: a noção de produção é generalizada por adição da correspondente probabilidade e a escolha de S é caracterizada por uma distribuição de probabilidade. A gramática passa assim a ser definida por um conjunto mais vasto $G=(T,NT,P,S,Q)$, onde Q é o conjunto das probabilidades associadas a cada uma das produções. Uma s*tring* produzida por G passa a ser vista como um resultado de uma sequência de ensaios e as descendências de uma frase como uma sequência de eventos probabilísticos.

Na maior parte das aplicações a gramática é "desenhada à mão". No entanto, como tem sido aqui bastante salientado (na medida em que se discute o r.p.) a aprendizagem assume um papel preponderante. Por outro lado, o introduzir-se o conceito de gramática estocástica é natural que também surja o conceito de inferência gramatical pelo que é desejável que as gramáticas possam ser construídas com base numa amostra de padrões. No entanto, além deste aspecto estar ligado a uma série de problemas de grande complexidade (ver Shalkoff, 1992, Duda, Hart e Stork, 2001 e Pavlidis, 1977), como é o caso da ambiguidade introduzida por algumas gramáticas (como se verá mais à frente pode não existir uma ligação única entre uma gramática e a correspondente linguagem), a utilização da inferência gramatical restringe-se a aplicações bastante específicas, para além da área do r.p., como é o caso da implementação de linguagens de programação. Por esta razão fica aqui apenas a referência de uma *homepage* que pretende ser uma fonte de informação centralizada da inferência gramatical e suas aplicações: *http://www.cs.iastate.edu/~honavar/gi/gi_main.html*.

Para o reconhecimento existem basicamente dois processos possíveis: *string matching*- processo heurístico em que não se têm em consideração modelos detalhados

nem a utilização de regras rígidas, e o *parsing*. O procedimento utilizado com maior frequência, para o reconhecimento, é via *parsing,* conceito fundamental neste contexto (quanto ao outro processo deixa-se como referência, por exemplo, Duda *et al.*, 2001, Secção 8.5). O *parsing* tem como objectivo primário verificar se uma dada *string* (*input*) está sintacticamente bem formada. Por outras palavras, é o processo que consiste em avaliar se o *input* é válido para uma dada linguagem e no contexto de determinada gramática[14]. A *string* pode ser decomposta de uma forma que permita um processamento posterior pelo que deve obedecer a uma determinada estrutura hierárquica.

O *parsing* é frequentemente utilizado para o processamento de programas de computadores já que estes devem ser alvos de uma inspecção que permita avaliar se está sintacticamente correcto e se a estrutura dos dados se reflecte na estrutura do próprio programa; se tal for verificado, poderá ser compilado numa linguagem a um nível inferior (as*sembler).* O *parsing* também assume um papel fundamental na interpretação de linguagens "naturais", tal como o Português. Os analisadores (*parsers*) apoiam-se na gramática que define as expressões válidas em determinada linguagem. Neste exemplo, será apenas a gramática de Português, para outros, tais como as linguagens de programação, será uma gramática que define as construções válidas nessa linguagem (com frequência, as gramáticas de contexto livre).

Mas o *parsing* de uma frase pode ir mais além do que a simples verificação da sua boa ou má constituição, poderá envolver toda a descrição estrutural, especificada pela gramática. O trabalho associado a todo este processo pode assim abranger vários aspectos tais como:

- Definição matemática das derivações de uma gramática e dos respectivos algoritmos,
- Cálculo das complexidades de tempo e espaço dos algoritmos em termos de comprimento da frase e tamanho das gramáticas (especialmente),
- Comparação de diferentes formalismos de gramáticas, mostrando sempre que possível as respectivas equivalências,
- Combinação da informação estatística e gramatical por forma a incrementar a eficiência dos analisadores.

[14] Note-se que uma linguagem pode ser gerada por mais do que uma gramática

Na linguagem formal das gramáticas a função dos analisadores (*parsers*) é reconhecer a **sintaxe** da linguagem. Qualquer estrutura que subsista à captação pelo analisador remete para a análise designada por **semântica**. Retomando o exemplo das linguagens de programação, e no caso de uma operação binária em Pascal, o facto do operador poder assumir dois argumentos (expressões) e de gerar uma expressão diz respeito à sintaxe; no entanto, o facto destes argumentos puderem ser números inteiros ou reais já diz respeito à semântica. Um outro exemplo contemplado na análise semântica diz respeito à declaração de uma variável antes desta ser utilizada; este tipo de dependência não pode ser capturada pelo analisador porque a utilização da variável não ocorre como parte da declaração. Só os programas sintacticamente correctos têm também semântica.

A sintaxe integra também os símbolos que permitem a construção de palavras bem como a respectiva estrutura de frases bem formadas. Voltando ao exemplo da linguagem Pascal, pode entender-se como *símbolos* as letras, dígitos e operadores, como *palavras* as palavras chave ou números e como *frases* as expressões ou declarações que formam os programas.

No contexto das linguagens de programação poder-se–á então dizer que a sintaxe se prende com o aspecto do programa, a respectiva estrutura e a definição matemática enquanto que a semântica está associada ao significado do próprio programa, sendo por isso difícil dar uma definição matemática.

Feitas estas considerações, pode sintetizar-se as quatro tarefas essenciais de um *parsing* da seguinte forma:

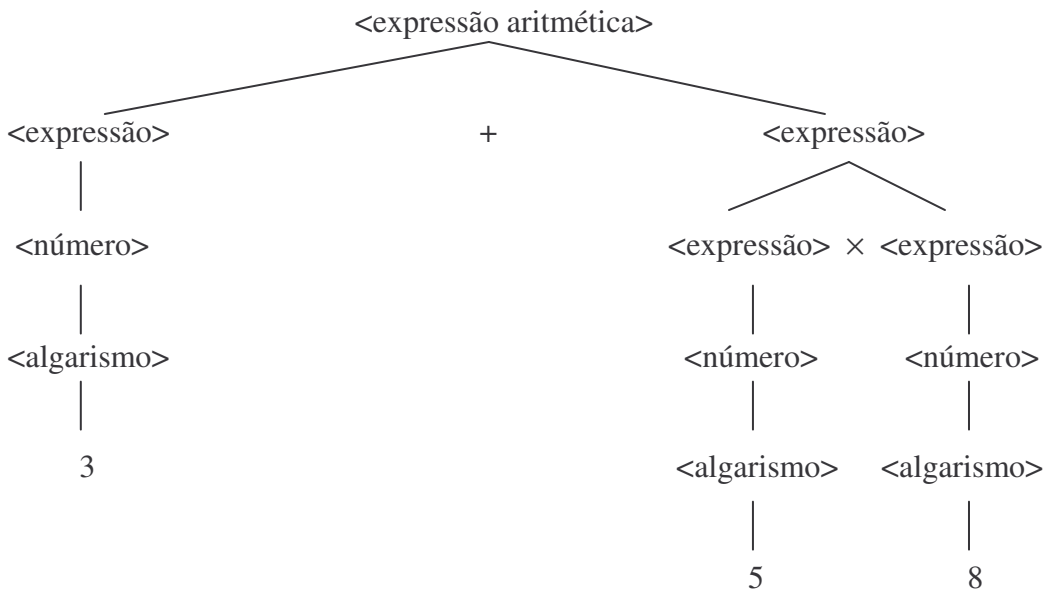
1. Leitura do texto: identificação da sequência de caracteres contida numa *string*.
2. Análise lexical: conversão do *input* numa sequência de sinais, ou seja, combinação dos caracteres em símbolos ou sinais (*tokens*) como por exemplo números reais ou inteiros, palavras chave,...
3. Análise sintáctica: processo de correspondência das produções com os sinais gerados anteriormente (verificar se uma dada sequência de sinais corresponde ao conjunto de produções da gramática- regras da gramática).

4. Criação da estrutura: se é encontrada uma correspondência cria-se uma determinada estrutura chamada *árvore sintáctica abstracta* que descreve como é que a *string* de *input* correspondeu, caso contrário é rejeitada.

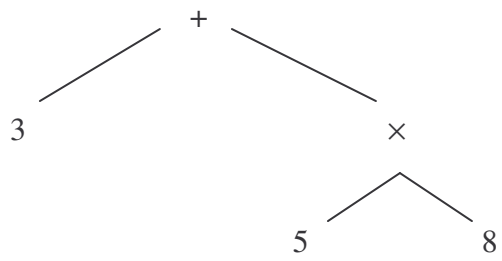
Realizando-se os passos acabados de apresentar, perde-se a informação acerca da forma como a sequência de sinais correspondeu, estruturalmente, às produções. Contudo, em geral esta informação é de grande importância. A ferramenta, definida neste contexto, que permite armazenar este tipo de informação é apelidada de **árvore de derivação**, uma forma mais completa da **árvore sintáctica abstracta**.

As representações gráficas constituem uma técnica bastante comum associada ao *parsing*. Quando não é necessário conhecer a produção a que se recorreu para a obtenção de determinada frase, ou seja, quando apenas se pretende captar a essência da produção (por exemplo algo do tipo "existe aqui uma adição") recorre-se à árvore sintáctica abstracta. Uma situação diferente surge quando existe necessidade de se conhecer a constituição exterior i.e., a estrutura da frase; nesse caso utiliza-se uma **árvore de análise** (*parse tree*).

Considere-se um exemplo muito simples onde o objectivo é o de analisar a expressão 3+(5×8) e verificar se esta é válida como expressão aritmética. Pode-se recorrer à construção da árvore de análise



ou ainda da árvore abstracta



Neste exemplo, + e × são símbolos terminais enquanto que 'expressão', 'número' e 'algarismo' constituem exemplos de símbolos não terminais. Para símbolo de começo toma-se 'expressão aritmética'.

Há, no entanto, que tornar clara a distinção entre árvore de análise e árvore de derivação. A primeira descreve a constituição externa da estrutura enquanto que a segunda representa a história da derivação, i.e., a descendência da análise (*parse*). As árvores de derivação são as frases reais da linguagem. Assim, fixada uma frase e uma gramática,

- Qualquer derivação corresponde a uma única árvore de análise,
- Uma árvore de análise pode corresponder a várias derivações.

Uma gramática é apelidada de ambígua se existir na forma de duas ou mais árvores de análise. Considere-se o seguinte exemplo de uma gramática (retirado do *site* da *internet* http://www.cse.msu.edu/~torng/Classes/360/lectures/lecture31)

$$G=(T,NT,P,S) \text{ onde } T=\{(,)\},\ NT=\{S\},\ S=S \text{ e } P=S\rightarrow SS\ /\ (S)\ /\ \lambda$$

Uma derivação possível para ( )(( )) é

$$S\Rightarrow SS\Rightarrow(S)S\Rightarrow(\ )S\Rightarrow(\ )(S)\Rightarrow(\ )((S))\Rightarrow(\ )((\ )).$$

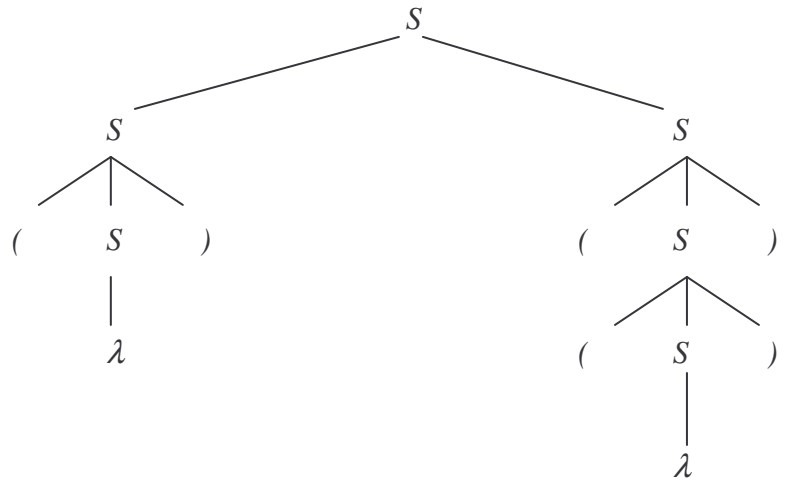
Esta derivação á apelidada de derivação mais à esquerda uma vez que é a variável que se encontra mais à esquerda que é expandida. Mas esta não é a única derivação possível, pode-se proceder à derivação mais à direita,

$$S \Rightarrow SS \Rightarrow S(S) \Rightarrow S((S)) \Rightarrow S(( )) \Rightarrow (S)(( )) \Rightarrow ( )(( ))$$

ou ainda,

$$S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)S \Rightarrow ( )(S) \Rightarrow ( )((S)) \Rightarrow ( )(( ))$$

Note-se que quer a derivação à esquerda, quer a derivação à direita são únicas. No entanto, qualquer que seja a derivação utilizada, a árvore de análise resultante é sempre a seguinte:



A estrutura gráfica de uma árvore de análise representa a informação semântica. Por vezes há quem distinga as descendências das árvores pela natureza dos seus arcos: uma linha tracejada é utilizada para a substituição e uma linha cheia para substituição por um símbolo terminal (ver por exemplo *http://www.cis.upenn.edu/~xtag/release-8.31.98-html/node1.html*).

A árvore tanto pode ser construída começando-se pelo topo- *top-down parsing*, como pela base- *bottow-up parsing*. No primeiro caso inicia-se com o chamado símbolo de começo ou raiz (designado anteriormente por *S*) e vai-se construindo a árvore de cima para baixo até se chegar à frase que está a ser alvo de análise. No segundo tipo de procedimento começa-se com o *input* e constroi-se a árvore de baixo para cima até se atingir o símbolo de começo *S*.

Já foi referido no início desta secção que a quantificação da estrutura a ser reconhecida pode ser feita quer através de gramáticas, quer através de descrições relacionais. As representações em árvore acabadas de apresentar, representam uma das alternativas subjacentes a estes métodos gráficos, a par dos grafos (árvores cíclicas) com particular relevância em questões de avaliação de similaridades e dos autómatos ou máquinas de estados finitos (árvores rotuladas recursivas). Para mais detalhes acerca dos métodos gráficos e do conjunto das técnicas associadas a esta abordagem aconselha-se vivamente Shalkoff (1992), numa perspectiva mais matemática, ou Pavladis (1977) na óptica da engenharia. Para uma abordagem mais superficial Friedman e Kandel (1999).

As descrições gráficas e especialmente as árvores, desempenham assim um papel fundamental quer neste tipo de abordagem, quer (como veremos mais adiante) na abordagem estatística onde irão surgir os denominados métodos baseados em árvores (*tree-based methods*) e que justificam a referência à abordagem estrutural nesta dissertação. Elas permitem que a informação seja armazenada através dos nodos que podem guardar informação sub-estrutural e através dos arcos que reflectem a informação relacional entre o nodo anterior e o sucessor.

Uma extensão, no contexto da abordagem estrutural, pode ser efectuada introduzindo-se a teoria dos conjuntos imprecisos- uma extensão das gramáticas estocásticas. Esta inovação permite a introdução de uma certa flexibilidade no espaço de localização. Um exemplo (dos típicos na área do r.p.), onde poderá ter interesse a sua introdução, é o do reconhecimento de caracteres. Bezdek (1992) justifica a que níveis e em que medida é que a teoria dos conjuntos imprecisos pode ser integrada na abordagem estrutural.

Por tudo aquilo que foi referido na Secção 1.3 e indo ao encontro das ideias proferidas por Leondes (1998), não há nenhum método que consiga ser superior aos restantes com respeito a todos os seguintes aspectos: generalização, precisão, tempo de aprendizagem, necessidades de memória e complexidade de implementação. Sendo assim, todos estes factores devem ser tomados em consideração antes de se proceder à escolha de determinado método, e não como é frequente, centrando-se exclusivamente nos factores precisão e generalização.

Capítulo 2

Classificação supervisionada

Neste capítulo trata-se do tema r.p. apenas do ponto de vista da classificação supervisionada e no contexto da abordagem estatística. Começa-se pelo tema da classificação apresentando-se a formulação de um problema típico e a notação básica. Aborda-se a questão da rejeição, descrevem-se os critérios de decisão existentes para a obtenção da regra de decisão e os critérios de avaliação da *performance* do classificador. O capítulo termina com uma apresentação e descrição dos métodos de classificação existentes, as tradicionais divisões efectuadas sobre estes métodos (paramétricos *versus* não paramétricos, estimação de densidades *versus* estimação de probabilidades *a posteriori* e outras subdivisões), bem como as correspondentes condições de aplicabilidade.

2.1 Introdução

Quando confrontados com um problema de classificação surgem-nos de imediato questões como: "Qual a melhor regra de classificação, ou qual o melhor classificador?". No entanto estas questões não são as mais adequadas pois diferentes problemas têm diferentes características e portanto diferentes soluções, o que é melhor para determinado problema pode não ser o mais adequado para outro. Então a questão a colocar deverá ser: "Qual a melhor regra de classificação para este problema particular?". E mais ainda, há que definir o conceito de melhor, ou seja, o que é realmente importante em termos de resposta ao problema, bem como o que se pretende optimizar.

Este conceito é geralmente interpretado no sentido de classificação precisa. No entanto a precisão é apenas um aspecto da *performance* do classificador, existem muitos outros que podem ser tanto ou mais importantes, como se verá mais adiante. Mas

mesmo que se tenha concluído que a precisão da classificação é o ou um dos factores mais importantes, há que ter cuidado com a sua definição; esta pode ter vários significados, consoante a formação de quem a utiliza e o contexto em que está inserida, o que pode envolver para além do mais, determinadas restrições.

A precisão de classificação pode estar relacionada com taxa de erro (percentagem de classificações incorrectas), custo total esperado, sensibilidade, especificidade, coeficiente de Gini (nomeadamente aquando da utilização de árvores de classificação), etc... Hand (1997) explica com detalhe em que consiste cada um destes critérios.

Ainda em relação à precisão há que tomar em consideração a própria medição: cálculos efectuados sob a amostra de treino conduzem a resultados geralmente diferentes daqueles efectuados sob a amostra de teste. Não deixa de ser comum, especialmente nas aplicações de redes neuronais ou nas aplicações de *machine learning*, que os dados de treino se ajustem perfeitamente e a *performance* na amostra de teste seja uma decepção total.

Que outros factores poderão também ser relevantes para a classificação?

- A facilidade com que é conseguida a interpretação da regra (há situações em que sendo um operador humano a fazê-la poderão ser cometidos erros catastróficos- é o caso do desastre de Chernobyl). Afirmações, citadas em http://www-bcf.usc.edu/~meshkati/chernobyl.html, tais como "*one of the defects of the system was that the designers did not foresee the awkward and silly actions of the operators*", "*human errors and problems with the man-machine interface*", entre muitas outras que apontam no mesmo sentido, justificam a origem deste acidente.
- A velocidade associada à construção da regra de classificação e mesmo à própria classificação. Em algumas circunstâncias a questão da velocidade poderá até ser o factor mais importante- um exemplo bastante simples é o da leitura automática de códigos postais. Um classificador com uma precisão de 90% pode ser preferível a outro com uma precisão de 95%, se for por exemplo, 100 vezes mais rápido- tais diferenças em escalas de tempo não são assim tão pouco comuns, nomeadamente quando se trabalha com redes neuronais que ao trabalharem em paralelo podem ser bastante rápidas, como

aliás já aqui foi referido. Ao contrário de outros métodos como, por exemplo, o K-NN que é muito lento devido à grande quantidade de pesquisas necessárias. Claro que o ser "rápido" ou "lento" é relativo-depende da aplicação em causa; o método K-NN até pode ser "rápido" se efectuar por exemplo a classificação num décimo de segundo! Este exemplo, referido por Hand (1997) permite também salientar a importância do tempo de aprendizagem. Por vezes torna-se mesmo necessário que se "aprenda" rapidamente a regra de classificação ou até mesmo que se façam ajustamentos de uma regra já existente em tempo real (para mais detalhes ver Michie *et al.*, 1994).

- A quantidade requerida de conhecimento *a priori*.
- A aptidão com que o método é capaz de lidar com valores omissos.
- A detecção das restrições associadas ao problema em causa.
- O "grau" de afinação do método, ou seja, o facto de todos os parâmetros poderem ser estimados directamente através da amostra de treino ou a necessidade de ter de se recorrer a um conjunto de afinação separado.
- O objectivo a ser alcançado, se a procura de um método para construção de um classificador a ser usado na prática ou meramente um método que demonstre o seu valor. Segundo Hand (1997) este aspecto está relacionado com o facto de se pretender explorar a *performance* condicional a uma dada amostra de treino (situação que aparece com frequência na prática) ou quando se está a tentar estabelecer níveis de *performance* gerais e portanto conclusões não condicionais que consagram o cruzamento de várias amostras de treino.
- Os custos das classificações provenientes de quais e quantas variáveis têm de ser medidas.
- A facilidade com que as variáveis podem ser medidas. Muitos problemas de classificação são motivados pelo desejo de substituir um processo de medida custoso/vagaroso por um processo barato/rápido (Hand, 1997). A perda de precisão da classificação é frequentemente um custo aceitável no sentido de se atingir o melhor compromisso entre a precisão de classificação e outros aspectos da *performance*.

Um dos aspectos acabados de referir e que tem particular relevância, prende-se com as restrições associadas a determinado problema[1], podendo ter uma influência bastante significativa, dependente da natureza do problema de classificação. Por exemplo, no caso particular da classificação de cromossomas, os tamanhos das classes são fixos. Ainda nesta área o termo *diagnosis* tem um significado diferente de *screening*; enquanto que no primeiro o objectivo é afectar um indivíduo à classe com maior probabilidade, o segundo já é um conceito ao nível da população pelo que o objectivo passa a ser a identificação da parte da população mais propensa a pertencer a determinada classe (ou a fracção da população mais propensa de sofrer de determinada doença, neste contexto). A distinção entre estes dois termos irá ter consequências nomeadamente na utilização da taxa de erro (Hand, 1997, Cap.6, Cap.8).

Outro exemplo bastante explorado na literatura é o do reconhecimento de discursos onde as probabilidades de ocorrência das palavras são influenciadas não só pelo contexto em que estão inseridas, mas também por toda a estrutura sintáctica e semântica da própria linguagem. Em problemas espaciais tais como a exploração de petróleo é suposto que determinadas classes se encontrem relacionadas com algumas classes "vizinhas". Muitos exemplos poderão ser encontrados em Hand (1997).

Esta exposição permite tornar claro o porquê da importância da construção de um procedimento de classificação, sendo a resposta sintetizada nos seguintes tópicos:

- A maior rapidez resultante da mecanização dos procedimentos de classificação (por exemplo na leitura de códigos postais);
- A tomada de decisão baseada apenas na informação relevante (os operadores humanos podem fazer surgir enviesamentos);
- A tentativa de explorar um conjunto de procedimentos alternativos àquele que poderá parecer o único caminho (por exemplo no diagnóstico médico poderá evitar-se que a operação seja o único caminho seguro, construindo procedimentos alternativos para se proceder a um diagnóstico preciso, apenas com sintomas externos).

[1] Entenda-se aqui restrições no sentido vasto podendo contemplar aspectos como o objectivo da classificação.

2.2 Formulação de um problema de classificação

Para que se possa avançar no desenvolvimento de resultados teóricos, passa-se de seguida à apresentação dos requisitos, objectivos e suportes afectos à formulação genérica de um problema de classificação.

Requisitos:

- Universo dos objectos (pacientes, caracteres,...);
- Classes a que os objectos poderão pertencer[2] (doente/não doente, 0/1/2/.../9,...);
- Conjunto de medidas (características) (idade/pressão sanguínea,..., imagem de 12*12 *pixels*,...).

Objectivo[3]*:*

- Predizer a identidade das classes, ou seja, afectar cada um dos objectos às classes correspondentes, com base nas características observadas.

Suportes:

- Assume-se disponível uma amostra de treino constituída por objectos sobre os quais se conhecem as suas características e a sua identidade (classe a que pertencem).

Formalmente considere-se um objecto (padrão) com vector de características $\mathbf{X}=(X_1,X_2,...,X_p)^T \in \chi$, um conjunto de c classes distintas $G_1, G_2, ..., G_c$ e uma amostra de treino $(\mathbf{x}_1, G_{i_1})$, $(\mathbf{x}_2, G_{i_2})$, ..., $(\mathbf{x}_n, G_{i_n})$ com $i_k=j$ se e só se a k-ésima observação pertence a G_j, j=1,2,...,c e k=1,2,...,n. O objectivo primário consiste em gerar uma **regra de decisão**, i.e., uma função que nos diz a decisão a tomar para cada novo objecto. A tarefa da classificação consiste então em estimar a verdadeira classe depois de observado **X**.

[2] Tal como já foi referido, no âmbito desta dissertação será apenas abordada a classificação supervisionada pelo que se assume que o conjunto das classes está previamente especificado.
[3] No sentido mais genérico da classificação, pois como também já aqui foi referido, poderá haver também interesse na descrição.

Naturalmente que no processo de classificação, baseado em **X**, pode optar-se pela rejeição, i.e. pela potencial recusa de decisão no caso em que o classificador não tem certeza suficiente. É no entanto, fundamental que se entenda que há duas diferentes razões para tal recusa

- a não evidência suficiente para se chegar a uma única decisão já que mais de uma das classes parece adequada para determinado objecto ou,
- o não reconhecimento, por parte do classificador de algum caso semelhante (que obviamente provém de um campo fora dos analisados e que portanto deve ser devolvido por resolver).

Sendo assim, de uma forma mais genérica pode-se dizer que um classificador é uma aplicação: $\chi \rightarrow \{G_1, G_2, ..., G_c, I, O\}$ onde I representa uma zona de indecisão e O uma zona de observações *outliers* onde a tarefa de classificação consiste em estimar a verdadeira classe associada a uma nova observação, caracterizada pelo vector $\mathbf{x}_{novo}$. Estas situações surgem quando as funções de decisão assumem valores muito idênticos ou aquando da existência de objectos muito diferentes dos restantes- observações *outliers*. Mais à frente apresentar-se-ão todos os detalhes acerca destas duas novas classes, considerando-se neste momento apenas o caso de c classes. Note-se, no entanto, que I e O constituem dúvida podendo, qualquer uma delas, ser estudada na forma de classe de rejeição[4].

O classificador divide o espaço das características em regiões de decisão por forma a que os pontos caiam numa região à qual se associa determinada classe. O aspecto crucial desta partição são as (hiper)superfícies, ou fronteiras, de decisão que representam os limites que fazem a separação entre as regiões de decisão associadas às várias classes. Estes limites são designados por **superfícies de decisão** uma vez que o movimento através da superfície modifica a decisão da classificação. As superfícies são expressas em termos das **funções de decisão** ou **funções discriminantes**.

Um aspecto que convém aqui salientar é o da diferença entre classificação e discriminação. Enquanto que nesta última se pretende descrever as diferenças entre

[4] A questão da rejeição no sentido da indecisão será abordada no Capítulo 3 e o problema das observações *outliers* no Capítulo 4.

vários objectos das diferentes classes e portanto encontrar discriminantes para a separação das mesmas, na classificação o que realmente tem interesse é a separação dos objectos e o encontrar de uma regra que possa ser usada para afectar novos objectos às classes pré-definidas. No entanto, em termos de funções é indiferente a utilização do termo função discriminante ou função de decisão (para mais detalhes consultar McLachlan, 1992).

Em termos genéricos, designando-se as funções de decisão por $d_i(\mathbf{x})$ i=1,2,...,c, a regra de decisão consiste em classificar um novo objecto, com vector de características **X**, na classe G_i, i=1,2,...,c, se $d_i(\mathbf{x})>d_j(\mathbf{x})$, $\forall j\neq i$, ou ainda, i = arg max d_j (**x**), j=1,2,...,c. Note-se que $d_i(\mathbf{x})=d_j(\mathbf{x})$ define a superfície de decisão entre as classes G_i e G_j. No caso particular de duas classes é equivalente a usar $d(\mathbf{x})=d_1(\mathbf{x})-d_2(\mathbf{x})$ e classificar em G_1 se $d(\mathbf{x})>0$ e em G_2 se $d(\mathbf{x})<0$ (geralmente, admite-se que quando $d(\mathbf{x})=0$ se procede a uma escolha aleatória razoável).

À parte o problema da selecção de características, a própria formulação das funções de decisão não é de todo trivial. O caso mais simples surge quando as classes são linearmente separáveis, i.e., $d(\mathbf{x})=w_0+w_1x_1+...+w_px_p=\sum_{i=1}^{p} w_i x_i + w_0 = \mathbf{w}^{\mathrm{T}}\mathbf{X}+w_0$ onde **w** é um vector de pesos. Claro que na prática estas funções são muitas vezes não lineares, estando disponíveis várias abordagens para a sua formulação, de acordo com a aplicação particular. Na Figura 2.1 apresentam-se várias situações esquemáticas, bidimensionais, com funções de decisão não lineares.

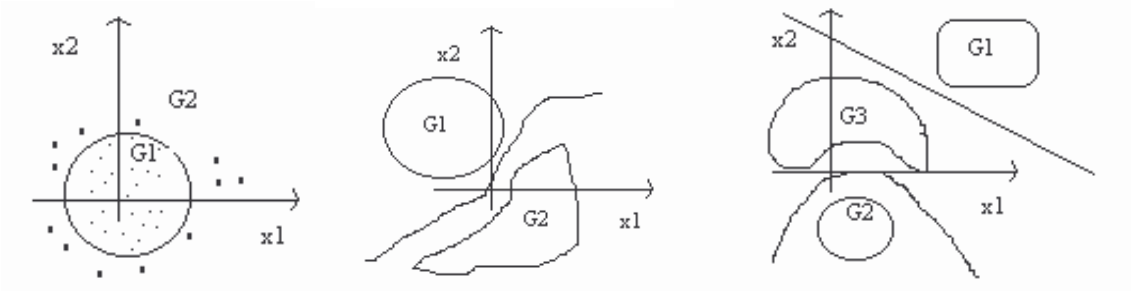


Figura 2.1- Exemplos de regiões de decisão separadas por superfícies não lineares.

Ainda a respeito das funções de decisão lineares há que optar por uma separação absoluta ou aos pares (*pairwise separation*). Se cada classe tiver uma função de decisão linear, $d_i(\mathbf{x})$, tal que $d_i(\mathbf{x}) = w_i^{\mathrm{T}}\mathbf{x} > 0$ se $\mathbf{x} \in G_i$, i=1,2,...,c, e 0 caso contrário (onde w_i é o vector de pesos associado a $d_i(\mathbf{x})$), então existe separação absoluta entre G_1, G_2, ..., G_c. Nesse caso a regra de decisão consiste em classificar um novo objecto, com vector de características $\mathbf{X}$, na classe G_i se $d_i(\mathbf{x})>0$ e $d_j(\mathbf{x})<0$ $\forall j \neq i$ e i,j=1,2,...,c; designe-se por D_i a correspondente região de decisão. Então a classe G_i é um subconjunto da região D_i. Este aspecto alerta-nos quanto à possível existência de regiões de "indefinição". Isto porque embora as classes padrão sejam geralmente limitadas, as respectivas regiões de decisão podem não o ser. Friedman e Kandell (1999, Cap.2, pág. 24-31) explicam e exemplificam com detalhe as diferentes situações que poderão ocorrer num ou noutro caso.

Na impossibilidade de separação absoluta pode ainda ser realizada uma separação parcial se cada par de classes for linearmente separável; é o caso que se refere como separação aos pares. Nesta situação, a cada par de classes G_i, G_j associa-se uma função de decisão linear $d_{ij}(\mathbf{x})$ tal que $d_{ij}(\mathbf{x})>0$ $\forall \mathbf{x} \in G_i$ e $d_{ij}(\mathbf{x})<0$ $\forall \mathbf{x} \in G_j$. Neste caso a regra consiste em classificar um novo objecto, com vector de características $\mathbf{X}$, na classe G_i se $d_{ij}(\mathbf{x})>0$ $\forall j \neq i$ e i,j=1,2,...,c. Note-se ainda que $d_{ji}(\mathbf{x}) = -d_{ij}(\mathbf{x})$.

Quer na separação absoluta quer na separação aos pares, não se obtém geralmente uma partição, i.e., a reunião das regiões de decisão não corresponde geralmente a $\mathbb{R}^p$. Consequentemente um objecto pode não ser classificado em nenhuma das classes. Mas mesmo que se consiga assegurar a inexistência de pontos em "terra ninguém", existem muitos problemas onde poderão ocorrer situações de indecisão ou de aparecimento de observações *outliers*. Isto reforça a necessidade da formulação do problema de classificação da forma genérica apresentada anteriormente e que contempla não c, mas c+2 classes.

As componentes básicas de um problema de classificação, no sentido deste estar definido de forma completa (por todos os elementos serem conhecidos) são:

- As probabilidades *a priori* das classes, as quais irão ser designadas por π_i e verificando, obviamente, $\pi_i \geq 0$ e $\sum_i \pi_i = 1$ para $i=1,2,...,c$.
- A função densidade de probabilidade (ou função de probabilidade) do vector **X** condicional à classe, $f_i(\mathbf{x}) = P(\mathbf{x}|G_i)$.[5] A função densidade da população total é dada por $f(\mathbf{x}) = \sum_i \pi_i f_i(\mathbf{x})$.
- A matriz de custos associada a uma classificação incorrecta (custos de má classificação), $C(i|j)$ ou C_{ji}, representando o custo de classificar erradamente um objecto na classe i, quando de facto pertence à classe j, tendo-se como é óbvio $C(i|i)=0$.
- Um critério de separação das classes: critérios de decisão apresentados em 2.3 e que são dependentes do conhecimento ou desconhecimento das três componentes anteriores.

Na prática o mais provável é que pelo menos uma das três primeiras componentes seja desconhecida. Tal como refere Pires (1995, Cap. 1, pág. 15), na posse de algumas o problema tem solução imediata recorrendo-se a um de vários critérios de decisão.

O desconhecimento das probabilidades *a priori* obriga à sua estimação, por exemplo através da frequência relativa com que cada classe ocorre na amostra de treino, ou à aplicação de algum critério onde estas não sejam necessárias. Quanto à f.d.p., ou se opta por algum dos muitos processos de estimação e depois utiliza-se esta estimativa como se fosse a verdadeira função, ou mais uma vez, se opta por um critério adequado à situação. Relativamente ao desconhecimento dos custos, ou se considera que o seu rácio é 1, ou terá de se enveredar por um critério de decisão que não necessite destes.

2.3 A questão da rejeição

Frequentemente é preferível não classificar, i.e, decidir pela rejeição, do que optar por uma decisão com elevada probabilidade de classificação incorrecta; são geralmente as observações "incertas" (nesta fase, entende-se por observações incertas aquelas para as

[5] Esta notação será utilizada arbitrariamente, quer a variável aleatória seja discreta ou contínua.

quais $\exists i,j,\ i\neq j$: $d_i(\mathbf{X})\approx d_j(\mathbf{X})$) que mais contribuem para o número de classificações incorrectas (ou má classificação). Os inconvenientes causados pela rejeição devem ser sempre julgados contra as consequências de uma má classificação embora na prática as vantagens ou desvantagens só possam ser avaliadas tomando em consideração a aplicação específica.

Há aplicações para as quais o custo de má classificação é muito grande e portanto uma rejeição de observações elevada é aceitável, tornando a taxa de má classificação o menor possível; exemplos típicos (Leondes, 1998) são aqueles que surgem no campo da medicina, tal como a apreciação de imagens para detecção de cancro. Noutras aplicações em que por exemplo há controlo humano posterior, é preferível classificar sempre numa classe, mesmo com risco de elevado número de más classificações. Um outro exemplo, que reforça e clarifica esta ideia, é o dos sistemas de detecção de aviões baseados em radar, descrito por Marques (1999, Cap.2, pág.28):

> *"... é obviamente menos grave um falso alarme do que uma falha de detecção que permita a um avião aproximar-se sem ser detectado"*

Uma escolha adequada da regra que conduz à rejeição permite que o comportamento do classificador se sintonize com a aplicação em causa. A ideia consiste em utilizar uma constante que actue como um limiar de segurança contra um número excessivo de erros de classificação e que poderá ser especificado pelo utilizador ou escolhido após a experimentação de diversos valores (se possível numa amostra de teste) obtendo-se estimativas da probabilidade de classificação incorrecta e da probabilidade de rejeição (**taxa de erro**, E, e da **taxa de rejeição**, R, respectivamente). Os detalhes de obtenção destas taxas serão fornecidos no Capítulo 3 quando se tratar da classificação com introdução da rejeição por indecisão.

Embora a opção de rejeição possa aliviar ou remover o problema do elevado número de más classificações, algumas observações correctamente classificadas podem ser convertidas em rejeição. Não esqueçamos contudo que rejeição pode (e deve) não significar exclusão mas apenas o suster da decisão para tratamento posterior (ou em

separado). Mais à frente, no Capítulo 4, irá retomar-se este assunto quando se tratar da questão da rejeição pela existência de observações *outliers*.

2.4 Critérios de decisão sem opção de rejeição

O r.p. do ponto de vista estatístico, centra-se essencialmente na decisão no sentido do desenvolvimento de estratégias de classificação. As regras de decisão são assim formuladas de acordo com diferentes critérios, como por exemplo, a conversão de probabilidades *a priori* em probabilidades *a posteriori* e da sua correspondente maximização ou ainda pela minimização de alguma medida de erro de classificação. Designem-se por D_1, D_2, ..., D_c as regiões de decisão associadas às c classes. Ver-se-á de seguida como é que uma ou mais das componentes básicas poderão ser combinadas para a construção de um critério de decisão.

2.4.1 Máxima verosimilhança

Começa-se pelo critério mais simples onde não são consideradas as probabilidades *a priori* e os custos de classificação incorrecta. Intuitivamente e tendo em conta o que já foi aqui mencionado, a proposta mais óbvia consiste em definir as regiões de decisão de acordo com a seguinte regra:

$$\mathbf{x} \in D_i \text{ se } f_i(\mathbf{x}) > f_j(\mathbf{x}) \quad \forall j \neq i \text{ e } i,j=1,2,...,c$$

ou equivalentemente,

$$\mathbf{x} \in D_i \text{ se } \frac{f_i(\mathbf{x})}{f_j(\mathbf{x})} > 1 \quad \forall j \neq i \text{ e } i,j=1,2,...,c.$$

Uma vez que o objectivo é comparar f.d.p. (ou estimativas destas) para se proceder à classificação, torna-se claro que a atenção se prenda, não com as funções individuais mas com a comparação das mesmas em termos de rácios ou como uma função deste; é o caso, por exemplo, da análise discriminante linear onde se toma o logaritmo.

Desta forma alguns pontos de $\mathbb{R}^p$ podem não pertencer a nenhuma região por se ter $f_i(\mathbf{x})= f_j(\mathbf{x})$ para algum i e j sendo habitual nestes casos proceder-se a uma atribuição aleatória razoável. Esta questão coloca-se para todos os critérios a seguir apresentados embora não seja muito importante em termos práticos uma vez que a probabilidade de aparecer um ponto sobre a fronteira é nula para vectores em que pelo menos uma variável é contínua.

2.4.2 Maximização da probabilidade *a posteriori*

Muitos métodos de classificação são baseados na comparação de f.d.p.. Outros, contudo, utilizam o teorema de Bayes para inverter o problema, focando a atenção não em $f_i(\mathbf{x})=P(\mathbf{x}|G_i)$ mas em $P(G_i|\mathbf{x})$. Para que tal objectivo seja concretizado, assumem-se como conhecidas, não só as f.d.p. como também as probabilidades *a priori* π_i. De acordo com o teorema de Bayes,

$$q_i(\mathbf{x}) \;=\; P(G_i \mid \mathbf{x}) = \frac{\pi_i f_i(\mathbf{x})}{\sum_{j=1}^{c} \pi_j f_j(\mathbf{x})} = \frac{\pi_i f_i(\mathbf{x})}{f(\mathbf{x})}, \qquad i=1,2,...,c.$$

Tem-se assim a seguinte regra de decisão[6]:

$$\mathbf{x} \in D_i \;\text{ se }\; q_i(\mathbf{x}) > q_j(\mathbf{x}) \;\; \forall j \neq i \;\text{ e }\; i,j=1,2,...,c.$$

ou equivalentemente (já que o denominador é comum),

$$\mathbf{x} \in D_i \;\text{ se }\; \pi_i f_i(\mathbf{x}) > \pi_j f_j(\mathbf{x}) \;\forall j \neq i \text{ e } i,j=1,2,...,c \qquad (2.1)$$

podendo ainda escrever-se como uma generalização da máxima verosimilhança,

$$\mathbf{x} \in D_i \;\text{ se }\; \frac{f_i(\mathbf{x})}{f_j(\mathbf{x})} > \frac{\pi_j}{\pi_i} \qquad \forall j \neq i \text{ e } i,j=1,2,...,c.$$

[6] Optou-se pela designação regra de decisão e não regra de classificação, como o fazem muitos autores, pelo facto deste termo ser mais abranjente no sentido de contemplar não só a classificação numa de c possíveis classes, como também a rejeição.

De notar que se as probabilidades *a priori* forem iguais este critério reduz-se ao anterior.

As regras de decisão baseadas no teorema de Bayes são óptimas- nenhuma outra regra tem taxa de erro esperada que seja inferior. Daí que, embora inacessível do ponto de vista prático (na medida em que pressupõe o conhecimento das três componentes), constitui a base de muitos algoritmos estatísticos.

2.4.3 Minimização da probabilidade de classificação incorrecta

A probabilidade de total de classificação incorrecta ou erro é dada por $\sum_{i=1}^{c} \pi_i P(\text{'erro'} \mid G_i)$ onde $P(\text{'erro'} \mid G_i)$ é a probabilidade de classificar incorrectamente objectos que pertençam a G_i, ou seja,

$$P(\text{'erro'} \mid G_i) = \int_{\overline{D_i}} f_i(\mathbf{x})\, d\mathbf{x}\,. \tag{2.2}$$

Logo a probabilidade total de classificação incorrecta é

$$P(\text{'erro'}) = \sum_{i=1}^{c} \pi_i \int_{\overline{D_i}} f_i(\mathbf{x})\, d\mathbf{x} = 1 - \sum_{i=1}^{c} \int_{D_i} \pi_i f_i(\mathbf{x})\, d\mathbf{x}\,. \tag{2.3}$$

Conclui-se como é óbvio que o critério da minimização da probabilidade de classificação incorrecta é equivalente ao da maximização da probabilidade de classificação correcta. Esta probabilidade é maximizada escolhendo-se D_i tal que $\mathbf{x}$ é afecto à classe para a qual o integrando é máximo. Obtém-se assim um critério equivalente ao da maximização da probabilidade *a posteriori* (fórmula 2.1).

Se for escolhida a classe para a qual a probabilidade *a posteriori* é máxima então o erro de Bayes é dado por:

$$e_B = 1 - \int_{\chi} \max_i \; \pi_i f_i(\mathbf{x})\, d\mathbf{x}\,. \tag{2.4}$$

Tal como refere Pires (1995, Cap.1, pág.17), este critério não é muito adequado quando existe uma grande disparidade nas probabilidades *a priori*. Nesse caso, pode ser mais adequado um critério do tipo minimax, onde o objectivo é a minimização do $\max_i P('\text{erro}'|G_i)$.

2.4.4 Minimização dos custos de classificação incorrecta ou regra de Bayes para minimização do risco

No critério anterior a regra de decisão seleccionava a classe para a qual a probabilidade *a posteriori*, $q_i(\mathbf{x})$, era máxima o que permitia a minimização da probabilidade de classificação incorrecta (ou de se cometer erro). Agora considera-se uma regra diferente que minimiza o custo ou perda esperada. Este novo factor é muito importante porque diferentes tipos de erro podem ter consequências variadas, tal como já foi salientado na Secção 2.3.

Não sendo a probabilidade de classificação incorrecta, um bom critério de avaliação do classificador em situações deste tipo, a solução passa pela atribuição de um custo a cada decisão. A maior parte da literatura sobre r.p. ignora a componente custo. São poucas as referências a algum tipo de custo e quando existem estão geralmente associadas a custos de classificação incorrecta, é por exemplo o caso de Webb (1999). Por esta razão, Turney (2000) apresenta uma catalogação em dez categorias, dos diferentes tipos de custos que poderão estar envolvidos na aprendizagem associada a um problema de classificação. O erro de classificação incorrecta, por exemplo, é referenciado como sendo do tipo 2 podendo ainda ser desagregado em 2.1 ou 2.2 consoante seja constante ou condicional e em 2.2.1 até 2.2.4 mediante o tipo de condicionamento. Já se torna comum hoje em dia, que apareçam *workshops* nesta área onde os vários artigos apresentados também são inseridos em categorias consoante o tipo de custos a que fazem referência. É exemplo disso o "*Workshop on Cost-Sensitive Learning*" realizado na Universidade de Stanford em 2000.

Na regra que se passará a apresentar assumem-se então conhecidas todas as componentes de um problema de classificação, definidas anteriormente. Para um dado

objecto **x**, o custo esperado de classificação incorrecta em G_i, também designado por perda esperada ou risco condicional é:

$$C^i(\mathbf{x}) = \sum_{j=1}^{c} C(i \mid j) P(G_j \mid \mathbf{x}) . \tag{2.5}$$

Então, o risco (médio) sob a região D_i é:

$$C^i = \int_{D_i} C^i(\mathbf{x})\, f(\mathbf{x})\, d\mathbf{x} .$$

Logo, o custo (ou perda) total esperado, ou *risco*, para objectos de todas as classes é:

$$\begin{aligned} r &= \sum_{i=1}^{c} C^i = \\ &= \sum_{i=1}^{c} \int_{D_i} C^i(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} = \\ &= \sum_{i=1}^{c} \int_{D_i} \left\{ \sum_{j=1}^{c} C(i \mid j) P(G_j \mid \mathbf{x}) f(\mathbf{x}) \right\} d\mathbf{x} \\ &= \sum_{i=1}^{c} \int_{D_i} \left\{ \sum_{j=1}^{c} C(i \mid j) f_j(\mathbf{x}) \pi_j \right\} d\mathbf{x}. \end{aligned} \tag{2.6}$$

A expressão do risco é minimizada se as regiões de decisão D_i forem escolhidas por forma a que,

$$\mathbf{x} \in D_i \text{ se } \sum_{j=1}^{c} C(i \mid j) f_j(\mathbf{x}) \pi_j < \sum_{j=1}^{c} C(k \mid j) f_j(\mathbf{x}) \pi_j \qquad \forall k \neq i \text{ e } i,k = 1,2,...,c.$$

ou, através das probabilidades *a posteriori*,

$$\mathbf{x} \in D_i \text{ se } \sum_{j=1}^{c} C(i \mid j) P(G_j \mid \mathbf{x}) < \sum_{j=1}^{c} C(k \mid j) P(G_j \mid \mathbf{x}) \qquad \forall k \neq i \text{ e } i,k = 1,2,...,c,$$

ou ainda,

$$\mathbf{x} \in D_i \text{ se } C^i(\mathbf{x}) < C^k(\mathbf{x}) \qquad \forall k \neq i \text{ e } i,k = 1,2,...,c.$$

Alguns autores preferem ainda apresentar a regra de decisão de uma outra forma, que nesta situação particular é a seguinte:

$$\mathbf{x} \in D_i \text{ se } i = \arg\min C^k(\mathbf{x}) \qquad k = 1,2,...,c\,.$$

Esta é a regra de Bayes para a minimização do risco, com risco de Bayes dado por:

$$r^* = \int_{\chi} \min_{i=1,2,...,c} \sum_{j=1}^{c} C(i \mid j) f_j(\mathbf{x}) \pi_j d\mathbf{x} = \int_{\chi} \min_{i=1,2,...,c} \sum_{j=1}^{c} C(i \mid j) P(G_j \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x}\,. \quad (2.7)$$

Para duas classes tem-se o caso particular,

$$\mathbf{x} \in D_1 \quad \text{se} \quad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{\pi_2 C(1 \mid 2)}{\pi_1 C(2 \mid 1)}.$$

Quando os custos são constantes ou seja, $C(i|j)=C_f \ \forall j \neq i$, a regra reduz-se àquela que se obtém com o critério da maximização das probabilidades *a posteriori*.

2.5 Avaliação de regras de classificação

Até agora ainda nada foi dito acerca da estimação das taxas de erro ou de outras medidas de *performance*. Contudo, este é um aspecto crucial que tem de ser tratado. Uma vez que o objectivo é o da classificação, o comportamento de determinada regra deve ser avaliado. Existem várias taxas e várias formas de as estimar. No entanto temos de estar cientes daquilo que realmente se pretende. Uma importante distinção é feita entre a *performance* de uma regra de classificação particular, desenvolvida sobre uma amostra de treino particular, e a *performance* do método de construção das regras de classificação. Nesta discussão o objectivo principal da avaliação de classificação é estimar até que ponto o classificador é bom na classificação de novos objectos. Se se pretender que a *performance* da classificação seja avaliada em termos da comparação de classificadores no sentido de se poder dizer que "algum é melhor que outro" então deverá tomar-se em conta outro tipo de conceitos tais como aqueles que são designados habitualmente por *inaccuracy*, *imprecision*, *inseparability* e *resemblance*. Todos os

detalhes sobre estes conceitos são descritos, por exemplo, em Hand (1997, Cap. 6, Cap. 8). Um exemplo do quão importante poderão ser estes conceitos (nomeadamente o termo "*imprecision*"- medida de quão bem são estimadas as probabilidades *a posteriori*) para a avaliação da regra, é também apresentado por Webb (199, Cap.10, pág. 328) embora este autor adopte uma designação diferente ("*reability"*).

Nesta secção começar-se-á por apresentar as diversas taxas de erro (não paramétricas) que permitem avaliar uma regra pela sua capacidade de bem classificar ou, de uma forma equivalente, de mal classificar. Como infelizmente, tanto em exemplos como na maior parte das situações reais, essas taxas são praticamente impossíveis de calcular, por desconhecimento de pelo menos uma das componentes básicas de um problema de classificação (ver Secção 2.2), segue-se uma descrição dos vários processos de estimação.

2.5.1 Tipos de taxas de erro

Dada uma regra de classificação, definida pela partição D_1, D_2, ..., D_c, as correspondentes probabilidades de classificação incorrecta, supondo conhecidas $f_i(\mathbf{x})$, $i = 1, 2, ..., c$ foram já introduzidas na Secção 2.4.3 (fórmulas 2.2 e 2.3). A **taxa de erro de Bayes**, também designada por **taxa de erro óptima**, é a taxa mínima de erro possível, supondo conhecidas $f_i(\mathbf{x})$ e π_i, $i = 1, 2, ..., c$ (daí a designação de taxa óptima). Proporciona um limite inferior para a taxa de erro que pode ser atingida por uma regra de classificação particular. A sua expressão também já foi introduzida na Secção 2.4.3 (fórmula 2.4).

Sendo por sua vez $\widehat{D_i}$, $i = 1, 2, ..., c$ as regiões de decisão estimadas (que se pretendem tão próximas da regiões óptimas quanto possível) definem-se as taxas de erro substituindo na fórmula (2.4) as regiões de decisão pelas correspondentes estimadas. Segundo Hand (1997) alguns autores chamam a esta taxa a verdadeira taxa de erro, outros a **taxa de erro actual ou condicional,** e_{act} , uma vez que é condicional à amostra de treino utilizada para construção do classificador. Pela própria definição de e_B tem-se $e_B \leq e_{act}$.

Pode também definir-se a **taxa de erro esperada ou não condicional** à amostra de treino, que é o valor esperado da taxa de erro actual. Dada uma amostra de treino (numa aplicação particular), poderá estar-se interessado nas taxas de erro condicionais das regras. No entanto, estando numa situação de ter de decidir que regra deve ser adoptada antes de se ver os dados, então interessará a taxa não condicional. Tem-se também que $e_B \leq \mathrm{E}[e_{act}]$.

Normalmente, num problema concreto, são as taxas actuais que assumem maior importância. A maior parte da pesquisa sobre a estimação das taxas de erro foca-se em e_{act}. Naquilo que se segue irá assumir-se que o objectivo primário será o da estimação da taxa de erro em termos globais ou totais (combinada sob as classes). Se, no entanto, o interesse recair nas taxas de erro individuais- por exemplo a proporção de objectos da classe G_1 que estão mal classificados- então estimativas correspondentes podem ser obtidas focando-se na classificação dos pontos de cada classe.

2.5.2 Estimação das taxas de erro actuais

Estimativas de substituição

Uma ideia bastante intuitiva consiste em substituir, nas taxas de erro actuais, as f.d.p., $f_i(\mathbf{x}),\ i=1,2,...,c$, pelas correspondentes estimativas $\hat{f}_i(\mathbf{x}),\ i=1,2,...,c$ em que se baseou a construção das regiões de decisão $\widehat{D_i}\ ,i=1,2,...,c$. Segundo Pires (1995, Cap. 3, pág. 66), estas estimativas não só dependem bastante da especificação do modelo (no caso paramétrico) como têm fracas propriedades e embora para certos métodos paramétricos tenham sido propostos processos de resolução, preferem-se actualmente outros procedimentos que podem ser chamados de não paramétricos e que se passam a apresentar.

Estimativas aparentes

Uma abordagem inicial óbvia para estimar a taxa de erro actual (já referida na Secção 2.2), que conduz à chamada taxa de erro aparente, consiste em reclassificar a amostra de treino a partir da qual a regra de classificação foi construída, e determinar a proporção de objectos mal classificados.

$$e_{ap} = \sum_{i=1}^{c} \pi_i \, e_{ap}(i),$$

onde $e_{ap}(i) = \frac{m_i}{n_i}$ sendo m_i o número de objectos mal classificados na classe i e n_i a dimensão da amostra pertencente à classe i.

Infelizmente, apesar de intuitiva, esta estimativa é enviesada com tendência a produzir valores optimistas, uma vez que são utilizados os mesmos objectos que entraram na construção da regra de classificação.

Método do conjunto teste

O caminho mais simples e óbvio, é a contagem dos objectos que a regra classifica mal, numa amostra de teste. Esta estimativa tem o mérito de ser fácil de obter e de não ser enviesada.

Em geral o número de observações disponíveis não é tão elevado que permita optar por este procedimento, e mesmo quando é, existe sempre uma certa relutância em fazê-lo, uma vez que surge a tentação de pensar que essa informação poderia ser utilizada para tornar mais eficiente a estimação da própria regra- quanto maior é a amostra melhor é a regra. Hand (1997, Cap. 7) alerta também para o cuidado que se deverá ter para garantir que a amostra de teste foi realmente desenhada da mesma distribuição que a amostra de treino e não é, por exemplo, distorcida de alguma forma, pelo processo de selecção ou divisão de uma amostra grande, na amostra de treino e de teste.

Um compromisso entre a partição nos dois conjuntos de dados e a simples utilização das taxas é a validação cruzada a qual irá ser apresentada de seguida. Este e os restantes métodos a ser apresentados envolvem a extracção de subconjuntos de dados para testar a *performance* dos classificadores construídos, repetindo esse procedimento para os diferentes subconjuntos e pesando os resultados.

Taxas de validação cruzada

O procedimento genérico do método de validação cruzada, também designado de rotação, é o seguinte[7]:

1. Estimar uma regra de classificação (a qual se pretende avaliar) com base na totalidade das observações da amostra de treino.
2. Separa a amostra de treino em duas subamostras, uma com *n-k* objectos e a outra com os restantes *k*. Com a primeira estima-se uma regra com a qual se classificam os restantes *k* objectos, registando-se o número de más classificações em cada classe (esta parte funciona assim como amostra de teste).
3. Repete-se o passo 2. um número elevado de vezes ou, de preferência, ${}^{n}C_{k}$, calculando-se a média das taxas de má classificação observadas em cada classe $e_c(i)$, e em seguida a estimativa habitual da taxa actual total,

$$e_c = \sum_{i=1}^{c} \hat{\pi}_i \, e_c(i) \, .$$

O valor mais usual para *k* é um, daí que o método também seja conhecido por *leave-one-out estimate* ou ainda por *U-method* ou *deleted estimate* (Webb, 1999, Cap.10, pág. 21). Alguns autores utilizam também incorrectamente (ver McLachlan, 1992, pág. 344) o termo *Jackknife*[8].

Uma das grandes desvantagens deste método é que pode envolver uma considerável quantidade de esforço computacional (nomeadamente no caso ideal em que a regra é formada ${}^{n}C_{k}$ vezes). No entanto é aproximadamente centrado (reduz substancialmente o enviesamento em relação às taxas aparentes) embora se traduza num aumento da variabilidade, especialmente no caso particular em que *k* é um. Webb (1999, Cap.10, pág. 321) refere, no entanto, que para regras de classificação baseadas na hipótese da normal multivariada, os cálculos computacionais podem ser reduzidos através da fórmula de Sherman-Morinsson (ver por exemplo McLachlan, 1992).

[7] Optou-se por apresentar este método de forma idêntica à proposta por Pires (1995, Cap.3, pág. 68).
[8] Para mais detalhes sobre este método (que antecede o *Bootstrap*) ver, por exemplo, Webb (1999, Cap.10) ou Hand (1997, Secção 7.6).

Com o intuito da redução do enviesamento a par da menor quantidade de dispersão surge um outro método baseado em *Bootstrap* o qual se passará a apresentar.

Taxas obtidas por *Bootstrap*

O *Bootstrap* refere-se a uma classe de procedimentos cuja base é a reamostragem (ou reutilização da amostra, segundo alguns) com reposição a partir da amostra de treino inicial. O objectivo é gerar conjuntos de observações *Bootstrap* que possam ser utilizadas para corrigir o enviesamento. Foi introduzido por Efron (1979) e tem recebido muita atenção na literatura nas últimas décadas. Proporciona estimativas não paramétricas do enviesamento e da variabilidade de um estimador e, como método de estimação da taxa de erro da estimação, tem provado ser superior a muitas outras técnicas (Webb, 1999, Cap. 10, pág. 326). Hoje em dia a literatura sobre este método é tão vasta que se torna praticamente impossível fazer um resumo de referências.

O procedimento pode ser descrito pelos seguintes passos:

1. Extrair, com reposição, uma nova amostra de tamanho *n*, da amostra de treino original, com $n = \sum_{i=1}^{c} n_i$ objectos referentes às *c* classes.
2. Seja n_i^* o número de objectos, da *i*-ésima classe, que se obtêm e calcule-se uma nova regra de classificação pelo processo que está a ser utilizado. Admita-se que m_i^* e m_i^{**} representam, respectivamente, o número de observações da nova amostra e da amostra original, na *i*-ésima classe, que são mal classificadas pela nova regra. Seja,
$$d_i = \frac{m_i^{**}}{n_i} - \frac{m_i^*}{n_i}.$$
3. Repita-se o processo um número elevado de vezes- recomenda-se no mínimo 100 (ver Pires, 1995, Cap. 3, pág. 69)- e seja $\overline{d}_i$ a média dos d_i ao longo dessas réplicas.
4. A estimativa *Bootstrap* da probabilidade total de má classificação é dada por,
$$e_{boot} = \sum_{i=1}^{c} \pi_i \, e_{boot}(i),$$

onde $e_{boot}(i) = \frac{m_i}{n_i} + \overline{d}_i$ e m_i é o número de objectos, da amostra de treino original que são mal classificados na *i*-ésima classe.

O *Bootstrap* pode ser utilizado para distribuições paramétricas nas quais as amostras são geradas de acordo com a forma paramétrica adoptada mas com os parâmetros substituídos pelas correspondentes estimativas. O procedimento não é limitado a procedimentos de redução do enviesamento da taxa de erro aparente e tem sido aplicado a outras medidas de precisão estatística. A sua aplicação com classificadores tais como as redes neuronais e árvores de classificação sofre também do problema do esforço computacional necessário associado à obtenção das réplicas. Nestes casos, no entanto, mais pela razão da existência de máximos locais (Webb, 1999, Cap.10, pág. 328).

Efron (1983) propõe ainda uma série de variantes do Bootstrap apelidadas de *double bootstrap*, *randomized bootstrap* e 0.632 *bootstrap*. Detalhes sobre estas três variantes são dados em McLachlan (1992, Cap.10).

Hand (1997, Secção 7.8) tece uma série de considerações acerca de que método deverá ser escolhido para avaliação da classificação em termos de taxas de erro, consoante aquilo que se tem disponível, as hipóteses que poderão ser assumidas, e os objectivos que se pretendem atingir.

Até agora, o assunto tratado nesta secção baseou-se no critério da minimização da probabilidade de classificação incorrecta ou do erro de má classificação. No entanto, tal como já aqui foi bastante frisado, é de todo conveniente introduzir custos associados aos diferentes tipos de erro e portanto calcular as taxas de erro baseadas na minimização dos custos de classificação incorrecta, ou seja, na minimização do risco.

Expressões idênticas às apresentadas podem ser obtidas introduzindo-se a componente custo. Aliás, fórmulas equivalentes às designadas por 2.2, 2.3 e 2.4 (ver Secção 2.4.3) foram já deduzidas na Secção 2.4.4 dando origem a 2.5, 2.6 e 2.7.

Infelizmente, dada a grande variedade e complexidade à volta do tema das taxas de erro, torna-se impraticável, nesta dissertação, aprofundar mais este assunto. Como refere Webb (1999, pág. 321),

"*A complete review of the literature on error rate estimation deserves a volume in itself...*"

2.6 Critérios de divisão dos métodos de classificação

A par das abordagens referidas no Capítulo 1, os métodos existentes podem ser classificados de acordo com um ou vários critérios, dependendo das propriedades que se pretendem realçar. Diferentes autores sugerem diferentes divisões sendo as mais comuns aquelas que distinguem os métodos paramétricos dos não paramétricos e os métodos de estimação de densidades (condicionais às classes) dos métodos de estimação directa das probabilidades *a posteriori*. Além destas duas grandes divisões, são realçadas outras tais como a divisão entre métodos de análise discriminante linear e métodos não lineares, métodos protótipo, métodos *subspace*, métodos de redução do número de variáveis, métodos de alisamento, métodos baseados em matrizes de covariâncias (Hand, 1997), métodos de partição recursiva, métodos não métricos, métodos baseados na expansão de funções base, etc. As designações dos vários métodos são apresentadas no Apêndice A .

A primeira grande divisão é a que faz a distinção entre **métodos paramétricos** e **métodos não paramétricos**. No primeiro conjunto de métodos é assumida uma forma funcional específica para a densidade do vector de características. Esta contém um número de parâmetros que são optimizados por ajustamento de um modelo à amostra de treino. São exemplos típicos a discriminante linear, LDA, a discriminante quadrática, QDA e suas variantes. Em contrapartida quando se faz referência a métodos não paramétricos não se assume uma forma particular para o vector de características, a forma da densidade é determinada directamente a partir dos dados: é por exemplo o caso do método de kernel, KDA, (também designado por método do núcleo ou método de Parzen) e do método dos *K* vizinhos mais próximos, K-NN.

Entre estes dois conjuntos de métodos há quem contemple ainda um terceiro, os **métodos semi-paramétricos** ou parcialmente paramétricos, que são referidos geralmente como sendo os que tentam encontrar um compromisso entre os outros dois, permitindo uma classe geral de formas funcionais nas quais se incluem um número

restrito de parâmetros funcionais dependentes da complexidade dos dados, podendo variar independentemente da amostra de treino mas aumentando o esforço computacional (ver por exemplo Leondes, 1998 ou Bishop, 1995). A maior parte dos autores que referem os métodos semi-paramétricos cingem-se aos modelos mistura no sentido abrangente, já que a par de proporcionarem técnicas potentes para a estimação das densidades também têm bastantes aplicações práticas noutros campos, como é o caso da configuração de funções base no contexto das redes neuronais ou na construção do modelo de misturas de *experts*, o qual será referido mais à frente. Como métodos de estimação de densidades os modelos mistura tornam-se mais flexíveis do que os modelos simples baseados em distribuições normais como a LDA, QDA ou mesmo RDA. Outros incluem neste conjunto mais métodos. É o caso de Leondes (1998, pág. 21) que procede a uma divisão que justifica como sendo a da perspectiva do utilizador de um sistema hipotético e não, como geralmente, do ponto de vista da teoria Matemática. Uma interpretação diferente é feita por autores como Mclachlan (1992) ou Duda, Hart e Stork (2001) que apenas os classificam como parcialmente paramétricos pelo facto de se admitir uma forma paramétrica para o quociente das densidades (como é o caso da discriminante logística).

A segunda grande divisão inclui os **métodos de estimação de densidades** (estimação do modelo probabilístico a partir dos dados, substituindo-o na regra de classificação de Bayes), os **métodos de estimação directa das probabilidades *a posteriori*** também chamados, por alguns, **métodos de regressão**[9] e aqueles que geralmente são designadas por "outros" uma vez que não resultam de critérios de decisão- no sentido da classificação (maximização da probabilidade *a posteriori* e minimização da probabilidade de erro ou do risco)- que não podem ser incluídos nos anteriores; inserem-se aqui os chamados **métodos protótipo** e por vezes os classificadores *subspace*. São exemplo de outros critérios o critério de separação máxima (de que o critério de Fisher é caso particular), os critérios de minimização de distâncias ou os critérios que impõem linearidade da função discriminante.

[9] Os métodos de regressão podem ser utilizados na classificação para estimação de probabilidades *a posteriori*, daí o abuso de linguagem. Sempre que haja necessidade pode-se definir o *output* de um problema de classificação como uma variável que toma o valor 1 se o *input* pertence a determinada classe e 0 caso contrário. Muitos dos assuntos que tem de ser tratados nos problemas de r.p. são comuns à classificação e à regressão. Métodos de regressão não paramétrica, utilizados com frequência nos nossos dias, são o MARS (*multivariate adaptative regression splines)*, PPR (*projection pursuit regression*), as árvores e as redes neuronais.

Nos métodos protótipo os dados são representados por um conjunto de pontos ou, de uma forma genérica, protótipos. Cada protótipo tem uma distribuição de classe associada e a classificação é feita de acordo com um voto de maioria do protótipo mais próximo. Os métodos diferem de acordo com o número e a forma como os protótipos são seleccionados mas têm em comum o princípio de manterem cópias em memória das amostras de treino e da classificação ser baseada nas distâncias entre os protótipos memorizados e o vector de *input*. São exemplos o método dos *k*- vizinhos mais próximos e o LVQ (*learning vector quantatization)*. A par destes incluem-se na categoria "outros" os métodos *subspace* que consistem na compressão e reconstrução de dados multidimensionais e baseiam-se na hipótese de que os dados provenientes de cada classe caem aproximadamente num subespaço de menor dimensão do que o espaço dos padrões. Inserem-se neste conjunto de métodos o CLAFIC (*class-featuring information compression*) e o ALSM (*average learning subspace method*) (ver por exemplo Leondes, 1998, pág. 32).

Quando é imposta a linearidade da função discriminante, onde o critério de Fisher é um caso particular, não se consideram hipóteses distribucionais, apenas se assume que as superfícies de decisão são lineares (inclui-se aqui também a discriminante logística). Pires (1995, Cap. 3, pág. 97) refere que todas as funções discriminantes em que a linearidade é imposta são de facto funções das projecções das observações numa dada direcção que é obtida por forma a maximizar (ou minimizar) determinado índice.

No contexto da análise discriminante não linear podemos incluir alguns métodos estatísticos e os **métodos neuronais**. Eles partilham o mesmo princípio, as funções discriminantes são construídas como funções lineares de funções não lineares[10]. A principal diferença entre as duas é que nos métodos neuronais de análise discriminante não linear as funções discriminantes são geralmente idênticas, no sentido de terem sempre a mesma forma paramétrica (logística para o MLP e normal para o RBF) e os dados serem utilizados para fixar os parâmetros das funções base; no caso do MLP são determinados como parte do processo de optimização e no RBF são geralmente obtidos através de um procedimento separado (os parâmetros lineares são obtidos por métodos como os mínimos quadrados ou a máxima verosimilhança). Por outro lado nos métodos

[10] As funções não lineares são geralmente apelidadas de funções base e designadas por ϕ.

estatísticos de análise discriminante não linear (como é o caso do PP, ACE e MARS), a forma de ϕ não é prescrita, sendo adaptada aos dados.

Entre os métodos lineares e não lineares situa-se a discriminante logística já que a discriminação é feita utilizando-se uma função linear sendo no entanto necessário um critério de optimização não linear para estimação dos parâmetros (e as probabilidades *a posteriori* são funções não lineares das variáveis). Esta discriminante tem a vantagem de não colocar restrições quanto ao tipo de variáveis embora a sua obtenção envolva um maior esforço computacional e dificuldades associadas à selecção das variáveis. Sob a hipótese da normalidade também é sabido que a classificação é inferior à efectuada com a LDA.

Webb (1999, Cap. 5) mostra como muitos métodos discriminantes são baseados em expansões de funções base. A RBF utiliza a soma pesada de funções univariadas de distâncias radiais, o MLP uma soma pesada de funções tipo sigmóide das projecções lineares, o PP usa uma soma de funções alisadas de regressões univariadas das projecções, o CART usa uma expansão em funções indicatrizes de rectângulos multidimensionais e o MARS é baseado numa expansão em somas de produtos de funções *spline* lineares univariadas.

Complementando o que já foi referido no Capítulo 1, haverá realmente necessidade de se proceder a uma separação entre a abordagem estatística, em termos genéricos, e a abordagem neuronal? Há na verdade alguma distinção a fazer? Webb (1999) é de opinião que existe uma grande sobreposição entre os modelos estatísticos e os neuronais e que o desenvolvimento de uma classe separada de métodos é apenas histórico. Leondes (1998) refere que é possível descrever estas duas abordagens através da utilização de termos comuns. Wan (1990) mostra como as redes neuronais estão relacionadas com os classificadores estatísticos proporcionando uma interpretação teórica do *output* do neurónio como uma estimativa da probabilidade *a posteriori*. O artigo de Cheng e Titterington (1994) e os respectivos comentários (nomeadamente o de Tibshirani[11]) são bastante elucidativos das ligações e diferenças entre as duas abordagens, enfatizando a necessidade do estatístico se interessar pelas redes neuronais e vice-versa e comparando técnicas incluídas em ambas. A diferença entre as duas

[11] Destaca em alguns pontos o que é que um estatístico pode aprender com um investigador em redes neuronais e vice-versa.

abordagens centra-se na forma como o problema é tratado; enquanto que o estatístico recorre a alguma técnica de estimação tal como a máxima verosimilhança, a inferência Bayesiana ou algum método não paramétrico, um especialista em redes neuronais treina a rede minimizando algum critério de erro. O comentário de Breiman ao artigo de Cheng e Titterington (1994) sintetiza os dois pontos fundamentais que justificam os bons resultados atingidos pelas redes neuronais. A saber, as propriedades desejáveis das funções base (funções alisadas de funções lineares bem limitadas) e o facto de poderem conter um grande número de parâmetros sem provocar sobreajustamento. Webb (1999, Cap.5, pág.170) sumaria uma série de aspectos que devem ser tomados em consideração aquando da utilização de redes neuronais.

A par dos métodos referidos há algumas generalizações que constituem avanços recentes como é o caso da PDA (*penalized discriminant analysis)* e da FDA (*flexible discriminant analysis)* (generalizações da LDA via regressão) devidas a Hastie *et al.* (1994) e Hastie *et al.* (1995) e apresentadas recentemente com grande detalhe em Hastie *et al.* (2001). Uma outra opção também recente, e que começa a ser utilizada com alguma frequência, é a combinação de vários classificadores em **comités,** resultando em técnicas como o *Bagging* (Breiman, 1996), *Stacking* (Wolpert, 1992), *Boosting* (Schapire, 1990, Freund, 1995, Freund e Shapire, 1997) e *Bumping* (Tibshirani e Knight, 1997). A **combinação de classificadores** é um procedimento que permite melhorar a *performance* da classificação na medida em que os vários classificadores fornecem informação complementar. Tal como referem Skurichina e Duin (2000), com frequência a combinação de classificadores dá melhores resultados que os classificadores individuais, já que se combinam na solução final as vantagens dos classificadores individuais. Woods *et al.* (1997) referem que embora na maior parte dos casos a combinação dê origem a menores erros de classificação, nem sempre é garantida. Leondes (1998, pág. 36) refere que além da melhoria na *performance* da classificação há outras boas razões para a utilização de comités de classificadores: os vectores de padrões podem ser compostos por componentes com origem em vários domínios e poderá não existir um processo óbvio de concatenação das várias componentes num único vector de padrões apropriado para um classificador de um só tipo. Noutras situações o esforço computacional poderá ser reduzido quer na fase de treino quer na fase do reconhecimento se a classificação for efectuada em várias fases.

Há geralmente duas formas básicas para a combinação de classificadores: ou são concatenados por forma a que o *output* de um torna-se o *input* de outro (ver Schurmann, 1996), ou os *outputs* dos múltiplos classificadores são combinados em paralelo- fusão de classificadores. Esta última variante é mais comum e pode ainda conter o caso em que todos os classificadores têm a mesma representação (por exemplo, ensaios de redes neuronais com a mesma arquitectura), trabalhando por isso com as mesmas características, ou o caso em que se tem uma representação diferente. Inclui-se aqui o caso em que os *outputs* dos vários classificadores são diferentes.

Depois de combinados os vários classificadores, referidos geralmente por ***combining experts*** (ou *ensamble classifiers*, *modular classifiers* ou *occasionally pooled classifiers*, segundo Duda *et al.*, 2001), se estes produzirem decisões/*labels*, a decisão é baseada nos vários *outputs* utilizando-se geralmente uma regra (heurística) de maioria onde o objecto é classificado na classe em que maior número de classificadores votaram[12] ou uma regra de maioria pesada onde são recolhidos os votos de todos os *experts* e o objecto é classificado na classe para a qual a soma dos votos, ponderada pela estimativa da fiabilidade do *expert* correspondente, é máxima[13]. Pode ainda ser utilizado um algoritmo para encontrar os pesos óptimos para combinar os *outputs* dos classificadores.

Em algumas situações, em que se consiga obter os valores das probabilidade *a posteriori*, é possível a utilização de outras regras para a decisão final. É o caso da obtenção da média dos coeficientes obtidos para cada classificador (*average*) ou da média, mediana, produto ou máximo das probabilidades *a posteriori*. Silva e Campilho (2000) apresentam um estudo experimental de alguns destes métodos de combinação de vários tipos de classificadores (classificador do máximo *a posteriori*, K-NN, MLP e RBF). Para o caso particular em que pelo menos um dos classificadores fornece apenas informação de categorias ordenadas ou o *label* de apenas uma categoria (*one-of-c outputs*) terá de se converter os *outputs* em valores discriminantes comparáveis. Em Duda *et al.* (2000, Cap. 9 , pág. 498) são apresentadas algumas heurísticas possíveis.

[12] Neste caso até poderá ser possível incluir a opção de rejeição da seguinte forma: escolher a classe para a qual a maioria vota ou rejeitar se houver pelo menos duas classes com o mesmo número de votos (Foggia *et al*., 1999).
[13] Técnica mais sofisticada já que se introduz uma medida de fiabilidade associada à resposta de cada *expert*.

Segundo Leondes (1998, pág. 37) os *outputs* dos vários classificadores de regressão podem também ser combinados linearmente, ou não, por forma a reduzir a variância das probabilidades *a posteriori*. Pode ainda utilizar-se uma amostra de treino comum para construir diferentes tipos de classificadores ou, alternativamente, usar diferentes amostras de treino para construir diferentes versões de um tipo de classificador.

Recentemente, o *Bagging* e o *Boosting* tornaram-se nos métodos mais populares de combinação de classificadores gerados pela mesma amostra de treino. Embora actuando de uma forma distinta, ambos modificam a amostra de treino, construindo classificadores baseados nestas transformações e culminando numa decisão final pela regra de maioria ou pela regra de maioria pesada. No *Bagging* (termo baseado nos conceitos apresentados por Breiman- *boostraping and aggregating concepts*) geram-se amostras *bootstrap* independentes, constroem-se os classificadores com base em cada uma delas e agrega-se pela regra de maioria na classificação final[14](para mais pormenores ver por exemplo Skurichina e Duin, 2000). Em contraste com este método em que as amostras de treino e os classificadores são independentes, no *Boosting* estes são obtidos sequencialmente em versão ponderadas da amostra de treino, através de um algoritmo. Inicialmente todos os objectos têm o mesmo peso e a primeira versão do classificador é construída com base na amostra de treino. Seguidamente os pesos são modificados de acordo com a *performance* do classificador; objectos mal classificados têm peso maior (para fazer com que o classificador trabalhe mais sobre esses pontos) e o classificador seguinte é reforçado (*boosted*) pela amostra de treino reponderada. Desta forma é obtida uma sequência de classificadores que são depois combinados por uma regra de maioria ou uma regra pesada de maioria (para mais pormenores ver Friedman e Kandel, 1999).

Skurichina e Duin (2000) chamam a atenção para o facto da escolha das regras de combinação de classificadores poder afectar a *performance* destes dois métodos; com base num estudo de simulação mostraram que a regra de maioria foi a pior escolha possível, enquanto que a regra de maioria pesada mostrou um bom desempenho quer para o *Bagging*, quer para o *Boosting*. As regras da média dos coeficientes obtidos para cada classificador (*average*) e da média e produto das probabilidades *a posteriori*

[14] Note-se que é construído uma classificador para cada amostra.

também funcionaram bem tendo até por vezes um comportamento superior ao da regra de maioria pesada (nomeadamente no *Bagging*).

Uma abordagem alternativa devida a Jacobs et al. (1991), designada por ***mixture-of-experts***, consiste em combinar linearmente o *output* dos vários classificadores. A motivação surge da pretensão de se construir um mecanismo que permita particionar a solução de um problema por vários classificadores, um classificador (*expert*) separado para determinar os parâmetros de cada função mais um classificador adicional (designado por *gating*) para determinar os coeficientes da mistura. De uma forma simples pode-se dizer que num modelo deste tipo há um número k de classificadores cada um dos quais produz, para um dado *input* $\mathbf{x}$, c valores discriminantes correspondendo a cada uma das classes. De seguida utiliza-se o sistema *gating* onde os *outputs* são pesados[15] e associados para a decisão final. A regra de decisão consiste em escolher a classe correspondente ao maior valor discriminante (para mais detalhes ver por exemplo Duda *et al.*, 2001, Cap. 9, pág. 496). Jordan e Jacobs (1994) generalizam esta ideia apresentando os modelos hierárquicos de mistura no qual cada *expert* é composto por um modelo de mistura de *experts* completo, com o seu próprio *gating* (ver Ripley, 1996, Cap. 8, pág. 283).

Muitas experiências relatadas na literatura são construídas para se efectuar a comparação de métodos (ver por exemplo Michie *et al*., 1994 ou Leondes, 1998). Este facto aliado às inúmeras divisões possíveis que surgem na área do r.p., faz com que seja bastante confuso proceder-se a uma catalogação precisa e exaustiva dos vários métodos. Até porque há dois aspectos que não podem ser descurados, por um lado o facto dos limites serem imprecisos (basta ver o caso da separação entre a própria abordagem estatística e neuronal) e por outro a forma como se actua no contexto do r.p; o mais importante não são os resultados teóricos associados a cada método mas sim a forma como o modelo se adapta aos dados. Há que tentar encontrar um compromisso entre a adaptação do modelo à complexidade da estrutura real e o ajustamento da estrutura à amostra de treino. De notar ainda que cada um dos métodos pode servir vários fins. Por estas razões sentiu-se uma grande dificuldade em agrupar logicamente os diversos métodos. Tendo em consideração que a catalogação foi feita tendo em conta o objectivo

[15] A inclusão de pesos torna este processo diferente do anterior designado por *combining experts*.

primário de cada um dos métodos, julga-se que a estrutura adoptada, embora discutível, é razoável.

	Paramétrico	**Não paramétrico**
Estimação directa de $P(G_i \mid \mathbf{x})$	Logística, MDA[16]	MLP, RBF, K-NN, LLR, MARS, PPR, ACE, Árvores
Estimação de $f_i(\mathbf{x})$	LDA, QDA, RDA e outras variantes[17]	KDA, *Nayve-Bayes*[18], PNN
Outros		*Piecewise*, LVQ

Tabela 2.1- Catalogação dos métodos mais utilizados no r.p.

De seguida apresentam-se algumas considerações breves sobre os diversos métodos. Os métodos não paramétricos são de difícil utilização em espaços de grande dimensão e recomenda-se que o K-NN seja utilizado como ponto de partida para a abordagem não paramétrica.

As redes neuronais (RBF e MLP) podem ser vistas como funções discriminantes generalizadas com funções ϕ_i flexíveis, cujos parâmetros têm de ser estimados ou fixados pela amostra de treino. Além de proporcionarem métodos de estimação de probabilidades *a posteriori*, também podem ser utilizadas para a estimação de densidades (Marques, 2000, Cap. 5, pág. 188). A RBF está muito relacionada quer com o método de kernel quer com o modelo de misturas de normais. Webb (1999, Cap. 5, pág. 142) mostra como é que a RBF pode ser obtida à custa de cada um destes dois métodos. Uma das suas propriedades que tem motivado a sua utilização numa vasta escala de aplicações é a sua universalidade de aproximação; sob certas condições é possível construir uma RBF que permite aproximar qualquer função ou implementar qualquer classificação (por muito precisa ou difícil que esta seja) embora possam existir problemas de sobreajustamento.

[16] Quando o número de parâmetros não excede um limite razoável que leva à utilização de procedimentos não paramétricos para a estimação
[17] Exemplos comuns são o SIMCA e DASCO (ver Mclachlan, 1992, pág. 141).
[18] Método de kernel simplificado que assume *inputs* independentes em cada classe.

Quanto aos métodos mais recentes, quer o ACE quer o MARS são recomendados para modelar dados que incluem variáveis de vários tipos embora o ACE seja mais adequado para conjuntos de dados que compreendem as medidas num pequeno conjunto de variáveis. Por seu lado o PP é mais apropriado para conjuntos de dados dispersos em espaços de grandes dimensões sendo necessário reiniciar algumas vezes o algoritmo associado ao PP antes de serem encontradas projecções apropriadas dos dados. O MARS tem-se revelado uma técnica promissora podendo ser considerado como uma generalização contínua da metodologia das árvores.

As árvores ou CART, são adequadas para problemas que incluem dados nominais onde não existe uma noção clara de similaridade ou métrica[19]. Método não métrico, segundo Duda, Hart e Stork (2001), a par dos métodos de reconhecimento via *strings* e gramáticas descritos em 1.3.3, as árvores podem ser vistas como um processo hierárquico de partição do espaço das variáveis: um processo de decisão de múltiplas fases. Em vez de se utilizar a amostra de treino completa e em conjunto para se tomar uma decisão, são usados diferentes subconjuntos de características em níveis diferentes da árvore. Este método permite a estimação directa das probabilidades *a posteriori*.

A árvore é construída recursivamente de cima para baixo desde o nodo no topo, o qual é designado por raiz, até que um nodo terminal ou folha, seja alcançado. A construção é simples e directa quando existe uma partição que classifica correctamente todos os objectos embora nesta situação se corra o risco de sobreajustamento (o que constitui uma desvantagem para propósitos de generalização). Quanto tal não acontece há duas estratégias possíveis, ou parar o crescimento mais cedo ou podar a árvore após a sua construção (procedimento *pruning*). O critério de partição mais utilizado dá pelo nome de índice de impureza. Considere-se o seguinte exemplo, suponhamos que um nodo contém 10 objectos dos quais 5 pertencem à classe A e os restantes 5 à classe B. Suponhamos ainda que todos os exemplos da classe A têm valores inferiores a 6 numa dada variável enquanto que todos os objectos de B assumem nessa mesma variável valores superiores a 6. Então este nodo diz-se impuro já que tem igual número de objectos em ambas as classes. Um novo objecto que surja e atinja este nodo tem então

[19] Este método é bastante popular e tem apresentado uma boa *performance* num grande conjunto de aplicações que não envolvem grandes conjuntos de dados permitindo modelar superfícies de decisão não lineares complexas, daí a importância que lhe é dada nesta secção.

uma probabilidade de 0,5 de pertencer a cada uma das classes. No entanto se procedermos a uma divisão deste nodo em duas classes, uma para valores superiores a 6 e outra para valores inferiores, obtemos dois nodos puros uma vez que cada um deles apenas contém exemplos de uma das classes. Se designarmos por *i(v)* uma medida qualquer de impureza de um dado nodo *v*, como por exemplo o índice de Gini, uma medida global de impureza da árvore vem dada por $I=\Sigma i(v)p(v)$ ao longo dos vários nodos terminais e onde $p(v)$ representa a proporção de objectos que caem no nodo terminal *v*.

Um regra simples para se proceder à paragem consiste em testar se a redução máxima no índice de impureza é inferior a determinado limiar previamente definido. No entanto esta abordagem é um pouco deficiente: se o limiar for pequeno a árvore apresenta-se com muitas folhas além de que se obtém uma variância grande (embora com fraco enviesamento); se o limiar for grande a variância é pequena mas tem-se um grande enviesamento (ver Ripley, 1997, Secção. 1.3). A alternativa, mais comum, à paragem consiste em podar a árvore. Claro que tal procedimento também arrasta outras decisões a ser tomadas tais como onde podar a árvore. Este processo acaba por ser análogo ao *forward* e *backward* da regressão. Existem muitas formas de o fazer sendo o corte realizado por forma a optimizar determinado critério de ajustamento. Webb (1999, Cap.7) apresenta o '*Cart pruning algorithm*' proposto por Breiman. Ripley (1996, Sec. 7.2) faz referência a mais alguns critérios. À parte da poda, outras abordagens para se encontrar uma boa estrutura em árvore incluem programação dinâmica, o método *branch and bound* e a *average tree*. Esta última é particularmente diferente por se focar no tipo de modelo construído e não no algoritmo de pesquisa; em vez de se procurar a "melhor" árvore constroem-se várias e toma-se a média das predições das classes (Hand, 1997). Classifica-se um objecto que atinge o nodo *v* na classe *k* se $P(k/v) = \max_{j=1,2,...,c} P(j/v)$. No caso em que temos probabilidades *a priori* diferentes das proporções relativas à amostra de treino, este método tem de ser ajustado.

A popularidade das árvores advém sobretudo da sua grande flexibilidade no tratamento de variáveis tanto pelo facto de permitirem lidar com interacções e variáveis de diferentes tipos, como pela facilidade de tratamento de valores omissos (ver Ripley, 1996, Cap.7). Existem outros métodos baseados em árvores que resultam de diferentes

escolhas dos critérios de paragem, da estratégia de poda ou do tratamento dos valores omissos. Os mais populares são o ID3 e o C4.5 (ver Duda, Hart e Stork, 2001 ou Michie *et al.*, 1994).

Seja qual for o método adoptado é importante que se entendam as ideias por detrás das várias técnicas por forma a saber como e quando utilizá-las; deve-se começar pelos métodos mais simples para que se possa avançar para os mais complicados. Também é importante que se avalie com precisão a *performance* do método ou seja, saber quão bem ou quão mau se espera que este funcione. A *performance* pode ser vista não só em termos de poder explicativo mas também em termos do compromisso erro/rejeição. Como é óbvio, diferentes pontos de vista dão origem a diferentes métodos.

Webb (1999) aponta algumas vantagens/desvantagens, limitações e recomendações dos métodos mais recentes (técnicas modernas segundo Michie *et al.*, 1994) comumente utilizados- ACE, PP e MARS bem como das redes neuronais e dos métodos não paramétricos tradicionais. O projecto *Statlog* (Michie *et al*., 1994) também é uma boa referência neste contexto, já que fornece uma vasta comparação de vários métodos de classificação sob vários conjuntos de dados.

Terminando com um comentário de Tuckey:

> *" ... it is often better to get an approximate solution to a real problem than a exact solution to an oversimplified one."*

Capítulo 3

Classificação supervisionada com opção de rejeição

Implicitamente em tudo o que já foi referido no Capítulo 2, há uma hipótese na qual estamos sempre interessados que é a da minimização dos erros de má classificação- taxa de erro ou taxa de má classificação. Embora esta medida seja muito popular e certamente uma das mais utilizadas, o facto é que raramente transparece aquilo que realmente se pretende. Hand (1997, Cap. 1, Sec. 1.2) aponta algumas fraquezas associadas a esta taxa sendo de especial destaque o facto de todos os tipos de má classificação serem tratados da mesma forma i.e. o se assumir uma simetria nos dados. Idealmente, como aliás já foi referido no Capítulo 2, a severidade da classificação deve ser quantificada em termos de custos devendo ser tomada em conta aquando da construção da regra de classificação.

Pode então escolher-se a regra de decisão que minimiza o custo total esperado (ou risco) de futuras más classificações em vez da minimização dos erros de má classificação (recorde-se que o critério de decisão correspondente à minimização da taxa de má classificação equivale a considerar custos de má classificação iguais).

Por outro lado a taxa de erro é limitada inferiormente por aquilo que foi designado no Capítulo 2 por erro de Bayes (que em praticamente todos os casos é superior a zero). Se existirem vectores de características que são comuns a mais do que uma classe de objectos, por muito esforço que haja na construção da regra de classificação, nunca se conseguirá reduzir a taxa de erro a zero. Por tudo isto tem interesse a introdução de uma opção de rejeição. Neste caso escolhe-se um limiar por forma a que apenas essas classificações nas quais se tem confiança (determinadas pelo limiar), são aceites. Isto significa que pode fazer-se com que a taxa de erro global seja tão pequena quanto se queira, aumentando, no entanto, o número de observações em relação às quais é declarada a indecisão. É óbvio que esta opção só terá interesse se houver outra possibilidade de classificação posterior (por exemplo manual) e/ou se o custo associado a essa operação não for demasiado elevado.

Chow foi o primeiro a introduzir a opção de rejeição num problema de classificação (Chow, 1957, 1970). No primeiro artigo formulou uma regra de decisão óptima. No segundo determinou uma relação fundamental entre as taxas de erro e rejeição e apresentou as **curvas de compromisso erro/rejeição**, que podem e devem ser utilizadas para descrever e comparar a *performance* empírica dos métodos de classificação.

Neste capítulo irá tratar-se o problema da classificação com inclusão da nova opção de rejeição por indecisão. Apresentar-se-á uma regra de decisão óptima, que pode ser escrita quer em termos das probabilidades *a posteriori*, quer em termos das densidades condicionais às classes, e alguns exemplos de aplicação. Embora esta regra seja baseada na teoria da decisão de Bayes clássica, é de facto uma generalização da regra original de Chow devido à introdução de uma estrutura de custos genérica que se julga nunca ter sido grandemente explorada nem mesmo utilizada do ponto de vista prático. Apresentam-se também expressões para o caso particular de duas classes e para finalizar o capítulo estabelece-se a ligação entre a abordagem proposta, baseada na regra óptima de Chow, e uma abordagem baseada em curvas ROC.

3.1 Critérios de decisão com opção de rejeição

3.1.1 Regra de Bayes para minimização do risco

Considerando agora uma classe de rejeição (associada à indecisão), $G_{c+1}=I$, activada sempre que o custo de classificação errada é suficientemente elevado, a regra de decisão é formulada de forma idêntica à que foi utilizada anteriormente, desde que à matriz de custos se adicione uma nova coluna. Nesta situação as regiões de decisão passarão a denotar-se D_1^*, D_2^*, ..., D_c^*, D_I. É habitual formular esta nova regra, que inclui a opção de rejeição, definindo previamente a denominada **função custo** que não é mais do que a definição dos custos de classificação e rejeição que formam a matriz de custos.

Considere-se a seguinte função de custos[1]:

[1] Note-se que t não é mais do que o custo de rejeição ou o custo de classificar manualmente um objecto, pelo que não faz sentido que dependa da classe a que pertence.

$$C(i \mid j) = \begin{cases} 0 & \text{se} \quad i = j \\ C_{ji} & \text{se} \quad i \neq j \quad i, j = 1, 2, ..., c. \\ t & \text{se} \quad i = I \end{cases}$$

O espaço das características χ é agora particionado numa região de aceitação, $A = \bigcup_{i=1}^{c} D_i^*$, e numa região de rejeição, D_I , da seguinte forma:

$$A = \left\{ \mathbf{x} : \min_i C^i(\mathbf{x}) \leq t \right\} = \left\{ \mathbf{x} : 1 - \max_i P(G_i \mid \mathbf{x}) \geq t \right\},$$

$$D_I = \left\{ \mathbf{x} : \min_i C^i(\mathbf{x}) > t \right\} = \left\{ \mathbf{x} : 1 - \max_i P(G_i \mid \mathbf{x}) < t \right\},$$

onde $C^i(\mathbf{x})$ é o risco condicional definido em (2.5). Seja $C^*(\mathbf{x}) = \min_i C^i(\mathbf{x})$. Então a regra de decisão consiste em,

aceitar o objecto se $C^*(\mathbf{x}) \leq t$

rejeitar o objecto se $C^*(\mathbf{x}) > t$.

Note-se que o objecto **x** só é rejeitado, ou classificado em D_I, se o custo de classificação nas restantes classes for superior a *t*, sendo também fácil verificar que quanto menor for *t*, maior é a região de rejeição. A regra de Bayes para minimização do risco com opção de rejeição passa a ser definida como:

$$\mathbf{x} \in Di^* \text{ se } \begin{array}{ll} C^i(\mathbf{x}) < C^k(\mathbf{x}) & \forall k \neq i \text{ e } i, k = 1, 2, ..., c \\ \text{e} & \\ C^i(\mathbf{x}) \leq t & \end{array}$$

$$\mathbf{x} \in D_I \quad \text{se} \quad C^*(\mathbf{x}) > t \,,$$

com risco de Bayes,

$$r^* = \int_A \min_{i=1,2,...,c} \sum_{j=1}^{c} C_{ji} P(G_j \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x} + \int_{D_I} t \sum_{j=1}^{c} P(G_j \mid \mathbf{x}) =$$

$$= \int_A \min_{i=1,2,...,c} \sum_{j=1}^{c} C_{ji} P(G_j \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x} + \int_{D_I} t f(\mathbf{x}) \; d\mathbf{x}.$$

Para o caso particular de duas classes obtém-se a seguinte regra de decisão:

$$\mathbf{x} \in D_1^* \text{ se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{C(1|2)-t}{t} \times \frac{\pi_2}{\pi_1} = t_1$$

$$\mathbf{x} \in D_2^* \text{ se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \leq \frac{t}{C(2|1)-t} \times \frac{\pi_2}{\pi_1} = t_2 \qquad (3.1)$$

$$\mathbf{x} \in D_I \quad \text{se} \qquad t_2 < \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} < t_1 .$$

Note-se que neste caso já não se verifica a equivalência a um problema sem custos (bastando conhecer a razão entre eles) tal como acontecia na Secção 2.4.4.

Para o caso particular em que os custos de classificação incorrecta são fixos a função de custos resume-se a:

$$C(i \mid j) = \begin{cases} 0 & \text{se} \quad i = j \\ C_f & \text{se} \quad i \neq j \quad i, j = 1, 2, ..., c \\ t & \text{se} \quad i = I \end{cases},$$

e a regra de decisão associada pode ser escrita à custa das probabilidades *a posteriori* da seguinte forma:

$$\mathbf{x} \in D_i^* \text{ se} \qquad \begin{array}{ll} P(G_i \mid \mathbf{x}) > P(G_k \mid \mathbf{x}) & \\ \text{e} & \\ P(G_i \mid \mathbf{x}) \geq 1 - \dfrac{t}{C_f} & \forall k \neq i \text{ e } i, k = 1, 2, ..., c \end{array}$$

$$\mathbf{x} \in D_I \quad \text{se} \quad \max_i P(G_i \mid \mathbf{x}) < 1 - \frac{t}{C_f}.$$

Note-se que se $t > C_f$ a regra resume-se à condição $P(G_i \mid \mathbf{x}) > P(G_k \mid \mathbf{x})\ \forall k \neq i$ e i,k=1,2,...,c o que significa que não há rejeição.

Para o caso particular de duas classes vem:

$$\mathbf{x} \in D_1{}^* \text{ se} \quad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{C_f - t}{t} \times \frac{\pi_2}{\pi_1},$$

ou equivalentemente

$$\mathbf{x} \in D_1{}^* \text{ se} \quad \begin{array}{l} P(G_1 \mid \mathbf{x}) > P(G_2 \mid \mathbf{x}) \\ \text{e} \\ P(G_1 \mid \mathbf{x}) \geq 1 - \frac{t}{C_f}. \end{array}$$

Particularizando ainda mais, i.e., fazendo C_f=1 o que corresponde a um caso particular abordado na literatura quando se inclui a opção de rejeição, pelo facto de ser equivalente à situação onde não se consideram custos (ver por exemplo Devijver e Kittler, 1982 ou Ripley, 1996), correspondendo à função de custos

$$C(i \mid j) = \begin{cases} 0 & \text{se} \quad i = j \\ 1 & \text{se} \quad i \neq j \quad i, j = 1, 2, ..., c \\ t & \text{se} \quad i = I \end{cases}, \tag{3.2}$$

obtém-se (particularizando a expressão apresentada anteriormente):

$$\mathbf{x} \in D_i{}^* \text{ se} \quad \begin{array}{ll} P(G_i \mid \mathbf{x}) > P(G_k \mid \mathbf{x}) & \\ \text{e} & \\ P(G_i \mid \mathbf{x}) \geq 1 - t & \forall k \neq i \text{ e } i,k = 1,2,...,c \end{array}$$

$$\mathbf{x} \in D_I \quad \text{se} \quad \max_i P(G_i \mid \mathbf{x}) < 1 - t,$$

ou ainda,

$$\mathbf{x} \in D_i{}^{*} \text{ se } \begin{array}{l} \pi_i f_i(\mathbf{x}) > \pi_k f_k(\mathbf{x}) \\ \text{e} \\ \pi_i f_i(\mathbf{x}) \geq (1-t) f(\mathbf{x}) \qquad \forall k \neq i \text{ e } i,k = 1,2,...,c \end{array} \tag{3.3}$$

$$\mathbf{x} \in D_{\mathrm{I}} \quad \text{se} \quad \pi_i f_i(\mathbf{x}) < (1-t) f(\mathbf{x}) .$$

Foi nesta última forma que foi originalmente apresentada a regra de indecisão denominada regra *t*-óptima (Chow, 1957).

Se, por outro lado, se partir da expressão do risco condicional, $C^i(\mathbf{x})$, vem que,

$$C^i(\mathbf{x}) = \sum_{\substack{j=1 \\ j \neq i}}^{c} P(G_j \mid \mathbf{x}) = 1 - P(G_i \mid \mathbf{x}).$$

$$C^I(\mathbf{x}) = \sum_{j=1}^{c} t \; P(G_j \mid \mathbf{x}) = t,$$

pelo que nesta situação particular, o risco condicional reduz-se à probabilidade condicional de erro de classificação em G_i e a regra de Bayes para minimização do risco reduz-se à expressão acima definida, em função das probabilidades *a posteriori*.

3.1.2 Regra de Bayes para minimização do risco com custos de classificação correcta

Até agora utilizou-se a estrutura que já tinha sido adoptada no capítulo anterior no que diz respeito aos custos isto é, apresentação da regra genérica com custos $C(i|i)=0$ e $C(i|j)=C_{ji}$ ($\forall i,j=1,2,...,c$) e depois a particularização para o caso de duas classes com custos $C(i|i)=0$ e, $C(i|j)= C_f$ ou $C(i|j)=1$, $i=1,2$ $i \neq j$; supôs-se em qualquer caso que não se incorria em perdas ou que não existiam lucros com a decisão correcta.

As questões que se consideram nesta secção são as seguintes: Será que a consideração destes custos faz sentido? E o que é que acontece quando se adiciona a opção de rejeição e por inerência o respectivo custo? Sem perda de generalidade trata-se de seguida apenas o caso particular de duas classes.

Na situação sem rejeição, poderiam ter sido considerados os quatro custos, $C(1|1)$, $C(1|2)$, $C(2|1)$ e $C(2|2)$. No entanto, feitas as contas com base nas fórmulas apresentadas na Secção 2.4.4, obtém-se

$$\mathbf{x}\in D_1 \quad \text{se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{C(1|2)-C(2|2)}{C(2|1)-C(1|1)}\times\frac{\pi_2}{\pi_1}$$

$$\mathbf{x}\in D_2 \quad \text{se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \leq \frac{C(1|2)-C(2|2)}{C(2|1)-C(1|1)}\times\frac{\pi_2}{\pi_1}$$

ou seja, a regra de decisão não depende dos quatro valores de custo (para o caso de duas classes) mas apenas das diferenças entre pares de custos pelo que para cada par (neste caso particular os pares $C(1|2)-C(2|2)$ e $C(2|1)-C(1|1)$) pode-se fixar arbitrariamente um dos custos como zero. E embora matematicamente essa escolha seja arbitrária, o que faz mais sentido é que se associe um custo nulo à classificação correcta ou seja aos custos $C(1|1)$ e $C(2|2)$. Note-se ainda que se os custos de classificar correctamente forem iguais e os custos de classificar incorrectamente também ($C(1|1)=C(2|2)$ e $C(1|2)=C(2|1)$) a regra de decisão reduz-se à obtida pelo critério de decisão da maximização das probabilidades *a posteriori* (ver Secção 2.4.2).

Por outro lado, quando se inclui a opção de rejeição (por indecisão), passam a ter-se três hipóteses: classificação correcta, classificação incorrecta ou rejeição. E tal como já foi aqui referido e será ainda mais reforçado, poderá haver mais interesse em descobrir correctamente uma das classes (por exemplo, doentes tal como acontece no Exemplo 3.1 a seguir apresentado). Claro que nem sempre isso acontece (imagine-se, por exemplo, o caso do reconhecimento de caracteres) mas essa opção com custos diferentes deverá sempre ser deixada em aberto para o problema particular em análise. No fundo aquilo que está aqui em causa e que deverá ser motivo de avaliação é o problema da (as)simetria das classificações, quer correctas, quer incorrectas. Neste caso (em que se inclui a opção de rejeição), obtém-se a seguinte regra de decisão

$$\mathbf{x} \in D_1^* \quad \text{se} \quad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{C(1|2)-t}{t-C(1|1)} \times \frac{\pi_2}{\pi_1} = t_1$$

$$\mathbf{x} \in D_2^* \quad \text{se} \quad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \leq \frac{t-C(2|2)}{C(2|1)-t} \times \frac{\pi_2}{\pi_1} = t_2$$

$$\mathbf{x} \in D_I \quad \text{se} \quad t_2 < \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} < t_1 .$$

Verifica-se assim que na situação com rejeição a regra depende das diferenças entre cada um dos quatro custos (para o caso de duas classes) e o custo de rejeição, pelo que pode fazer sentido (dependendo como se disse do problema em causa) a consideração dos custos (lucros) de classificação correcta.

3.1.3 Acerca do limiar de rejeição *t*

Devijver e Kittler (1982, Cap.2) referem que para o caso particular em que se trabalha com a função de custos (3.2) e para *c* classes, $0 \leq 1 - \max_i q_i(\mathbf{x}) \leq \frac{c-1}{c}$, o que é óbvio já que o limite superior corresponde ao acaso em que todas as classes têm igual probabilidade a *posteriori*. Sendo assim, para que a opção de rejeição seja activada é necessário que $0 \leq t \leq \frac{c-1}{c}$. No caso em que os custos de classificação são fixos, mas não necessariamente unitários, tem-se que $0 \leq t \leq C_f \frac{c-1}{c} \leq C_f$.

Quando os custos são quaisquer (situação apresentada na Secção 3.1.1), obtém-se $\exists_i$: $0 \leq t \leq \frac{1}{c} \sum_{\substack{j=1 \\ j \neq i}}^{c} C(i \mid j)$, (quando os custos são fixos, esta dupla desigualdade reduz-se à anterior).

De notar que, em qualquer dos casos, se *t*=0 todos os objectos são rejeitados e que se *t* igual ao limite superior da desigualdade não há rejeição pelo que é indiferente incluir a igualdade na aceitação ou na rejeição. De qualquer forma a probabilidade de se alcançar o limite superior das desigualdades é nula para vectores em que pelo menos

uma variável é contínua (no caso discreto pode-se optar por uma atribuição aleatória em caso de igualdade de condições, tal como já foi referido para o caso em que não se considerou a rejeição).

Para o caso mais comum em que existem duas classes e os custos são considerados iguais, a condição nunca é activada para $t>1/2$, $t=1/2$ significa inexistência de rejeição e quando $t=0$ tem-se 100% de rejeições.

3.1.4 Taxas de erro e rejeição

Na maior parte das tarefas de reconhecimento de padrões as distribuições de probabilidade subjacentes não são completamente conhecidas. Um procedimento comum poderá consistir em assumir, com base na informação *a priori* disponível, algumas formas funcionais e ajustar os parâmetros com base em dados empíricos (amostra de treino). Embora não seja simples validar as hipóteses subjacentes a tais estruturas assumidas, é sempre possível obter a curva de compromisso erro/rejeição e calcular a teórica com base nestas hipóteses. Uma comparação das curvas teóricas e empíricas pode sempre fornecer informação para iniciar o "melhoramento" do modelo teórico (Chow,1970).

Mas como podem ser obtidas as curvas de compromisso erro/rejeição, ou melhor, como podem ser obtidas as taxas de erro e rejeição? Como já foi referido, o espaço das características pode ser particionado numa região A, de aceitação e numa região D_I de rejeição definidas na Secção 3.1.1. Em termos dessas regiões, as taxas de erro e rejeição podem ser definidas da seguinte forma:

$$\mathrm{E} = \int_{A} \left[1 - \max_{i} P(G_i \mid \mathbf{x}) \right] f(\mathbf{x})\, d\mathbf{x},$$

$$\mathrm{R} = \int_{D_I} f(\mathbf{x})\, d\mathbf{x}.$$

Chow (1970) demonstrou que tanto a taxa de erro, E, como a taxa de rejeição, R, são ambas funções monótonas do limiar *t*. Segue de imediato que E aumenta e R diminui à medida que *t* aumenta. Em particular quando *t*=0, E=0 e quando *t*=1, R=0.

Segundo Chow (1970), uma descrição completa da *performance* de um sistema de r.p. é fornecida pelo compromisso erro/rejeição i.e., pela relação funcional entre E e R. E uma vez que tanto E como R são funções monótonas do limiar *t*, poderá representar-se graficamente o compromisso entre ambos, E(*t*) e R(*t*). Esta curva é monótona não crescente, uma vez que a rejeição de mais padrões ou reduz ou mantém a taxa de erro e poderá ser utilizada para fixar o limiar *t*. Um exemplo prático de aplicação das curvas de erro-rejeição é apresentado por Golfarelli, Maio e Maltoni (1997), no contexto dos sistemas de verificação biométrica (em particular na forma da mão e na face humana).

Mais ainda, o conhecimento de apenas uma destas taxas é suficiente para uma caracterização completa da *performance*. Esta afirmação deriva do facto das taxas de erro e rejeição estarem relacionadas da seguinte forma:

$$\mathrm{E}(t) = -\int_0^t t \, d\mathrm{R}(t),$$

sendo que E(*t*) é o integral de Stieltjes de *t* relativamente a R(*t*). De salientar também que o limiar de rejeição *t* constitui um limite superior da taxa de erro.

3.2 Exemplos

O principal objectivo de todo o trabalho teórico consiste na aplicação prática da metodologia desenvolvida. Por outro lado, acontece com frequência que é apenas na sua aplicação que nos apercebemos de determinados aspectos que devem ser tomados em consideração. É esta fase que nos permite clarificar o problema em mãos, alertando-nos para algumas dificuldades e motivando-nos para a prossecução do estudo sobre novos aspectos.

Assim, considera-se importante apresentar um conjunto de exemplos, neste caso os tradicionais da análise discriminante (linear, quadrática e logística) e a sua aplicação a dois conjuntos de dados que passarão a ser designados por Exemplo 3.1 e Exemplo 3.2. Os aspectos computacionais associados a esta análise são incluídos no Apêndice B.

Exemplo 3.1 (Detecção da doença coronária: dados CHORON)

Os dados considerados neste exemplo encontram-se pormenorizadamente descritos em Macieira-Coelho *et al.* (1990). Os dados originais referem-se a uma amostra de 113 indivíduos nos quais se observaram 9 variáveis: 4 clínicas (idade, sexo, factores de risco e dor precordial) e 5 variáveis obtidas na prova de esforço (depressão de ST, aparecimento de dor precordial durante o teste, aparecimento de arritmias, variações nos valores da pressão arterial e variações na amplitude da onda R). Como refere Pires, (1995, pág. 9), este exame que é do tipo invasivo permite detectar a doença coronária com elevada sensibilidade mas acarreta custos elevados e não é isento de riscos, pelo que é desaconselhável a sua utilização rotineira.

Todos os indivíduos foram submetidos à coronariografia tendo-se concluído que 30 não sofriam de doença coronária (classe G_1) e que 83 sofriam de doença coronária (classe G_2). Salienta-se a existência de variáveis de vários tipos (binário, ordinal e contínuo). Para uma descrição mais detalhada das variáveis ver tabela apresentada por Macieira-Coelho *et al.* (1990).

No trabalho desenvolvido começou-se por fazer uma análise baseada na amostra completa estando, no entanto, cientes de que o número de observações relativamente ao número de variáveis era bastante pequeno. Posteriormente, pelas razões apontadas por Pires (1995, cap. 6), foram então retiradas duas variáveis (x_4: dor precordial e x_8: variações na amplitude da onda R) passando a trabalhar-se apenas com 7 variáveis.

Exemplo 3.2 (Detecção de diabetes em mulheres Índias com mais de 21 anos da tribo 'Pima', residentes no Arizona: dados PIMA)

Os dados considerados neste exemplo foram retirados de Ripley (1996)[2] e referem-se a uma amostra de treino de 200 mulheres nas quais se observaram 5 variáveis: número de vezes que esteve grávida, concentração de glucose no sangue, concentração de insulina, índice de massa muscular e idade.

Nos dados originais disponha-se de 7 variáveis (ver Ripley, 1996, Cap.1, pág.15) mas uma selecção de variáveis pelo método *stepwise* (Ripley, 1996, Cap.3, pág. 114) rejeitou duas passando-se a trabalhar apenas com 5. Das 200 mulheres que compõem a amostra de treino 132 tinham diabetes (classe G_1) e as restantes 68 não apresentavam diabetes (classe G_2).

3.2.1 Regras de classificação para as discriminantes linear, quadrática e logística

Considere-se o caso particular de duas classes e tome-se por base a regra *t*-óptima de Chow apresentada na Secção 3.1.1 ou em Santos-Pereira e Pires (2001) com custos de má classificação unitários, isto é, $C(1|2)=C(2|1)=1$. Designando por $d(\mathbf{x})$ uma função de decisão genérica, a regra de decisão que servirá de suporte às discriminantes linear, quadrática e logística pode ser sintetizada da seguinte forma:

$$\begin{aligned} \mathrm{x} \in D_1^{*} \ \text{se} \quad & d(\mathbf{x}) \geq t_1 \\ \mathrm{x} \in D_2^{*} \ \text{se} \quad & d(\mathbf{x}) \leq t_2 \\ \mathrm{x} \in D_I \ \text{se} \quad & t_2 < d(\mathbf{x}) < t_1 \end{aligned}$$

onde t_1, t_2 , e $d(\mathbf{x})$ podem ser especificados para cada uma das discriminantes, obtendo-se as expressões que se passam a indicar.

[2] Os conjuntos de dados tratados por Ripley (1996) encontram-se disponíveis no seguinte endereço da Internet: *http://www.stats.ox.ac.uk/~ripley/Prbook*.

Discriminante Linear (LDA)

Suponha-se que $\mathbf{X} \mid G_i \sim N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma})$ i=1,2, ou seja que

$$f_i(\mathbf{x}) = (2\pi)^{-p/2} \left|\boldsymbol{\Sigma}\right|^{-1/2} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^{\mathrm{T}} \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right], \ \mathbf{x} \in \mathbb{R}^p .$$

Designado, como habitualmente, por $d(\mathbf{x})$ o $\ln\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right)$ obtém-se,

$$d(\mathbf{x}) = \ (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2) .$$

Tendo em conta a fórmula (3.1), tem-se directamente,

$$t_1 = \ln\left[\frac{\pi_2}{\pi_1} \times \frac{1-t}{t}\right], \quad t_2 = \ln\left[\frac{\pi_2}{\pi_1} \times \frac{t}{1-t}\right].$$

Discriminante Quadrática (QDA)

No caso em que as classes continuam a ter distribuição normal mas com matrizes de covariâncias distintas, $\mathbf{X} \mid G_i \sim N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ i=1,2, $d(\mathbf{x})$ pode ser escrita da seguinte forma:

$$d(\mathbf{x}) = - D_1(\mathbf{x}) + D_2(\mathbf{x}),$$

onde $D_i(\mathbf{x}) = (\mathbf{x}-\boldsymbol{\mu}_i)^{\mathrm{T}} \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)$ representa o quadrado da distância de Mahalanobis entre $\mathbf{x}$ e a média da classe G_i, i=1,2. Note-se que neste caso o quociente das densidades é dado por

$$\frac{\left|\boldsymbol{\Sigma}_1\right|^{-1/2}}{\left|\boldsymbol{\Sigma}_2\right|^{-1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^{\mathrm{T}} \boldsymbol{\Sigma}_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) + \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^{\mathrm{T}} \boldsymbol{\Sigma}_2^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)\right],$$

pelo que

$$\ln\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right) = \ d(\mathbf{x}) \ - \ \frac{1}{2} \ \ln\left(\frac{\left|\boldsymbol{\Sigma}_1\right|}{\left|\boldsymbol{\Sigma}_2\right|}\right).$$

Obtêm-se então os seguintes valores para t_1 e t_2:

$$t_1 = 2\ln\left[\frac{\pi_2}{\pi_1}\times\frac{1-t}{t}\right]-\ln\left(\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|}\right), \qquad t_2 = 2\ln\left[\frac{\pi_2}{\pi_1}\times\frac{t}{1-t}\right]-\ln\left(\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|}\right).$$

Discriminante Logística

Neste caso admite-se que o logaritmo do quociente das densidades condicionais às classes (ou equivalentemente o logaritmo da razão entre as probabilidades *a posteriori*) é linear pelo que,

$$d(\mathbf{x}) = \alpha_0 + \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{x} = \ln\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right).$$

Pelas razões anteriormente apontadas obtêm-se os seguintes valores para t_1 e t_2,

$$t_1 = \ln\left[\frac{\pi_2}{\pi_1}\times\frac{1-t}{t}\right], \quad t_2 = \ln\left[\frac{\pi_2}{\pi_1}\times\frac{t}{1-t}\right].$$

3.2.2 Avaliação das regras de classificação

Descrevem-se em seguida os processos utilizados para a obtenção das taxas de erro e rejeição (E e R, respectivamente), definidas anteriormente na Secção 3.1.4, para cada uma das discriminantes. No caso particular de duas classes tem-se,

$$\begin{aligned} \mathrm{E} = P(\text{'erro'}) &= P(\mathbf{x}\in G_2,\ d(\mathbf{x})\geq t_1) + P(\mathbf{x}\in G_1,\ d(\mathbf{x})\leq t_2) = \\ &= \pi_2 P(d(\mathbf{x})\geq t_1 \mid \mathbf{x}\in G_2) + \pi_1 P(d(\mathbf{x})\leq t_2 \mid \mathbf{x}\in G_1). \end{aligned}$$

$$\mathrm{R} = P(\text{'rejeição'}) = \pi_1 P(t_2 < d(\mathbf{x}) < t_1 \mid \mathbf{x}\in G_1) + \pi_2 P(t_2 < d(\mathbf{x}) < t_1 \mid \mathbf{x}\in G_2).$$

Até agora admitiu-se que as probabilidades *a priori* e as densidades condicionais às classes eram conhecidas. No entanto, tanto nos exemplos apresentados como na maior parte das situações reais, este não é o caso. Sendo conhecida a forma das distribuições condicionais às classes, embora os parâmetros sejam desconhecidos, está-se perante um

problema de decisão paramétrica pelo que o procedimento utilizado foi o habitual, que consiste na estimação dos parâmetros desconhecidos resultando no chamado "classificador de Bayes *plug-in*".

Para estimação das probabilidades *a priori* optou-se pelo cálculo da frequência relativa de cada classe no seio da amostra de treino, i.e. $\hat{\pi}_i = \frac{n_i}{n}$, $i = 1, 2$, onde n_i representa o número de objectos na amostra de treino que pertencem à *i*-ésima classe e *n* é a dimensão da amostra de treino.

Quanto à estimação dos parâmetros que caracterizam cada uma das discriminantes aqui abordadas, utilizou-se como estimativa da média da *i*-ésima classe, a média amostral. No que diz respeito à matriz de covariâncias, no caso da discriminante linear, utilizou-se a matriz de covariâncias combinada usual,

$$\hat{\mathbf{\Sigma}} = \mathrm{S} = \frac{(\mathrm{n}_1 - 1)\mathbf{S}_1 + (\mathrm{n}_2 - 1)\mathbf{S}_2}{\mathrm{n} - 2},$$

onde $\mathbf{S}_1$ e $\mathbf{S}_2$ representam as matrizes de covariâncias amostrais de cada classe.

No caso das discriminantes quadrática e logística as matrizes de covariâncias de cada classe, $\mathbf{\Sigma}_i$, foram estimadas através das correspondentes matrizes amostrais.

De seguida descrevem-se os procedimentos adoptados para estimar as taxas de erro e rejeição para cada uma das regras discriminantes em consideração.

Discriminante Linear (LDA)

Nesta situação é possível calcular as taxas de erro e rejeição teóricas. Os valores de E e R podem ser facilmente estimados invocando-se um resultado básico da estatística multivariada:

$$\text{Se } \mathbf{x} \in G_1 \text{ então } d(\mathbf{x}) \sim N\left(\frac{1}{2}\Delta, \Delta\right),$$

$$\text{Se } \mathbf{x} \in G_2 \text{ então } d(\mathbf{x}) \sim N\left(-\frac{1}{2}\Delta, \Delta\right),$$

onde $\Delta = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$.

Havendo necessidade de comparar estas taxas com as correspondentes para outros métodos discriminantes nomeadamente, neste caso, a discriminante quadrática (descrita a seguir) onde não é viável a obtenção de resultados teóricos, é sempre possível calcular as correspondentes taxas de erro e rejeição para a amostra de treino ou para a amostra de teste (caso esta exista).

Discriminante Quadrática (QDA)

Como é de conhecimento geral não se conseguem obter expressões simples (como no caso da discriminante linear) para os integrais da função densidade numa dada região. Na impossibilidade de se recorrer a qualquer resultado teórico, pode optar-se por estimar E e R através da amostra teste, caso esta exista ou estimar E e R através de uma amostra simulada (conjunto de teste simulado) a partir de uma distribuição normal multivariada, com parâmetros iguais aos valores amostrais das médias e matrizes de covariâncias associadas a cada classe

Discriminante Logística

Caso se possa admitir que $d(\mathbf{x})$ segue aproximadamente uma distribuição Normal então, tal como na discriminante linear, é possível estimar as taxas de erro e rejeição teóricas recorrendo-se ao resultado que se segue para a estimação de E e R,

$$\text{Se } \mathbf{x} \in G_i \text{ , } d(\mathbf{x}) \sim N\ (\alpha_0 + \boldsymbol{\alpha}^{\mathrm{T}} \mu_i, \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\Sigma}_i \boldsymbol{\alpha}) .$$

Caso contrário tem de se proceder como na discriminante quadrática.

Note-se que um teste formal para a hipótese de normalidade pode ser conduzido calculando-se o coeficiente de correlação entre os *scores* discriminantes e os seus valores esperados sob a hipótese de normalidade (ver Neter *et al*., 1996, Cap. 3, pág. 111). A Tabela B.6 (Neter *et al*., pág. 1348) contém valores críticos (percentis)

associados a vários tamanhos da amostra, para a distribuição daquele coeficiente de correlação. Este teste pode ser acompanhado da representação gráfica dos *scores* discriminantes em papel de probabilidade normal (*qqplot*). Para dimensões amostrais não existentes na tabela poderá sempre recorrer-se a uma interpolação linear.

3.2.3 Alguns resultados

Dados CHORON

Um aspecto que já foi aqui bastante referido e que deve, sempre que possível, ser tomado em linha de conta, é o carácter não simétrico das consequências das classificações, e neste exemplo em particular, as consequências das classificações erradas. Mas neste exemplo, tal como na maior parte das aplicações, há uma grande dificuldade e subjectividade inerente à quantificação destes custos. Assim, como o objectivo neste momento é o da ilustração da regra de decisão com inclusão da opção de rejeição por indecisão, e por uma questão de simplificação, irá assumir-se a situação particular em que a função de custos utilizada é a apresentada em (3.1).

Recordando aquilo que já foi referido, neste exemplo está-se perante uma amostra de treino de 113 observações. No entanto, não está disponível qualquer amostra de teste e havendo necessidade de se proceder a uma comparação das três discriminantes, optou-se por criar uma amostra "simulada" de 1000 observações para cada classe[3]. Sendo assim, decidiu-se pelo cálculo por simulação dos integrais da função densidade, estimando-se o vector de médias e a matriz de covariâncias da amostra de treino para cada classe e gerando-se as observações a partir de uma distribuição normal com esses parâmetros.

Apresentam-se em seguida, nas Figuras 3.1 e 3.2 as curvas de compromisso erro/rejeição em função do limiar *t*, referidas na Secção 3.1.4. Na Figura 3.1 os valores das taxas de erro e rejeição são os fornecidos pela amostra de treino e na Figura 3.2 apresentam-se as taxas de erro e rejeição "teóricas", no caso das discriminantes linear e logística, e no caso da discriminante quadrática os valores correspondentes fornecidos pela amostra "simulada".

[3] Note-se que é importante que se mantenha a mesma precisão em cada classe.

Considerações acerca da forma como estes valores foram obtidos são feitas no Apêndice B onde também se apresentam os programas (S-Plus) respectivos. Para a discriminante logística foi validada a hipótese de normalidade através dos procedimentos descritos anteriormente.

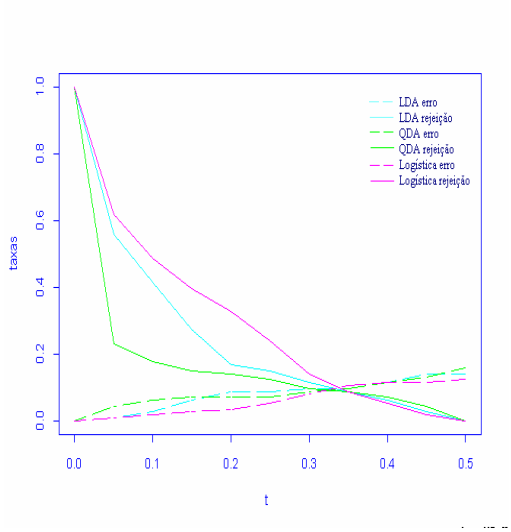


Figura 3.1- Taxas de erro e rejeição para os dados CHORON. Amostra de treino.

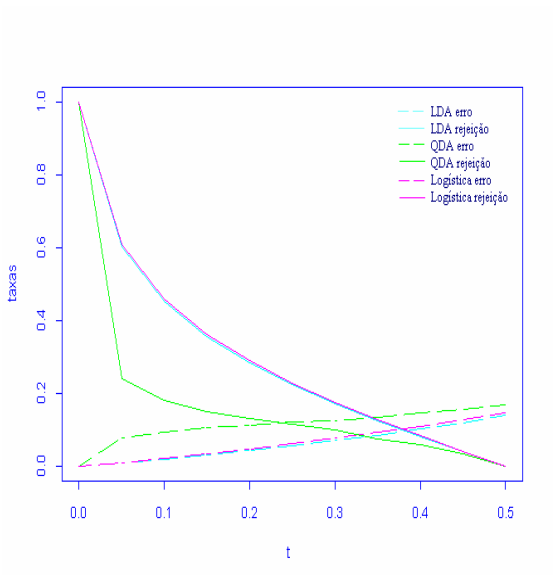


Figura 3.2- Taxas de erro e rejeição para os dados CHORON. Discriminantes linear e logística: "teóricas", discriminante quadrática: amostra "simulada".

A análise da Figura 3.1 permite concluir que há uma evidência bastante forte de que a taxa de erro é inferior para a discriminante logística com a agravante (já esperada) de induzir um aumento na taxa de rejeição nomeadamente em relação às outras duas

discriminantes para valores de *t* até 0.35; a partir deste valor a discriminante logística é claramente a melhor discriminante (menor taxa de erro e rejeição). No entanto, na Figura 3.2, as taxas de erro e rejeição são bastante idênticas no caso das discriminantes logística e linear.

Por seu lado a discriminante quadrática é a que apresenta piores resultados em termos de taxa de erro mas a que acarreta uma menor probabilidade de rejeição incorrecta para qualquer valor de *t*.

Um outro aspecto importante que é oportuno salientar (ver Tabela 3.1) é que os valores da taxa de erro para a situação em que não se inclui opção de rejeição são sempre superiores aqueles que se obtêm quando se introduzem custos de rejeição. As taxas de erro e rejeição para cada uma das três discriminantes aqui consideradas encontram-se no Apêndice B.

t	***t*1**	***t*2**	**E**	**Taxa aparente Amostra de treino**	**R**	**Taxa aparente Amostra de treino**
0	+∞	-∞	0	0	1	1
0.05	3.962	-1.927	0.007	(0+1) 0.009	0.603	0.557
0.10	3.215	-1.179	0.018	(2+1) 0.026	0.453	0.416
0.15	2.752	-0.717	0.030	(6+1) 0.062	0.356	0.274
0.20	2.404	-0.037	0.043	(9+1) 0.088	0.283	0.168
0.25	2.116	-0.081	0.056	(9+1) 0.088	0.224	0.150
0.30	1.865	0.170	0.071	(10+1) 0.097	0.172	0.115
0.35	1.637	0.399	0.086	(10+1) 0.097	0.125	0.088
0.40	1.423	0.817	0.102	(11+2) 0.115	0.080	0.062
0.45	1.218	0.612	0.119	(13+3) 0.142	0.040	0.026
0.50	1.018	1.018	0.139	**(13+3) 0.142**	0	0

Tabela 3.1- Taxas de erro e rejeição "teóricas" e estimadas (amostra de treino) para a discriminante linear.

Dados PIMA

Neste caso encontra-se disponível uma amostra de teste constituída por 332 observações. Contudo, foi também construída uma amostra "simulada" de 1000 observações em cada classe, para que fosse possível fazer uma comparação entre as taxas de erro e rejeição "teóricas" associadas às discriminantes linear e logística e as estimativas mais aproximadas, possíveis de obter, para a discriminante quadrática (ver Figura 3.4). Tal como no exemplo anterior, para o caso da discriminante logística foi validada a hipótese de normalidade dos scores.

Na Figura 3.3 apresenta-se a curva de compromisso erro/rejeição, em função do limiar *t*, obtidas através da amostra de teste para as três discriminantes.

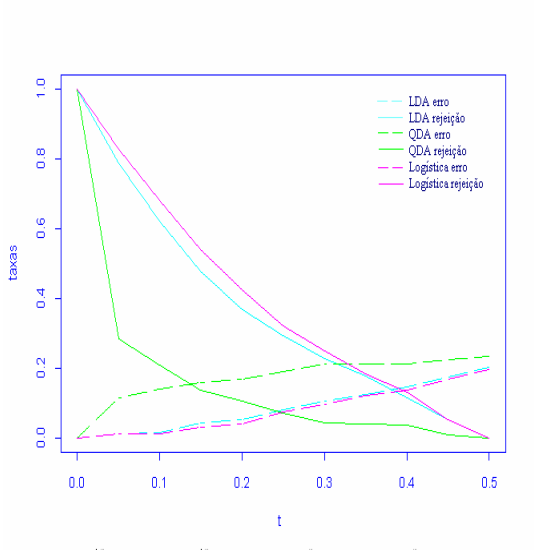


Figura 3.3- Taxas de erro e rejeição para os dados PIMA. Amostra de teste.

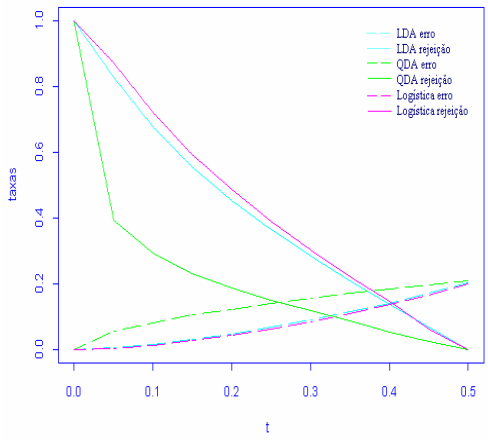


Figura 3.4- Taxas de erro e rejeição para os dados PIMA. Discriminantes linear e logística: "teóricas", discriminante quadrática: amostra "simulada".

Fixando *t* pode-se fazer uma leitura dos valores de E (taxa de erro) e R (taxa de rejeição). A título de exemplo, com a amostra teste (Figura 3.3) e na discriminante logística, para *t*=0.3 reduz-se a taxa de erro para cerca de metade: passa de 19,6% para 9,6%, mas temos uma taxa de rejeição de 25%. Se permitirmos apenas que cerca de 15% das observações sejam rejeitadas, já baixamos a taxa de erro de 19,6% para 13,5%.

Um outro aspecto que deve ser tomado em consideração e que ajudará a encontrar o melhor valor de *t* é o conhecimento dos custos relativos do erro e rejeição.

As taxas de erro e rejeição para cada uma das três discriminantes aqui consideradas encontram-se no Apêndice B. De salientar que embora se pretende-se "validar" a amostra "simulada" i.e., ver até que ponto fez sentido gerar uma amostra a partir da amostra de treino comparando os valores obtidos com os da amostra de teste, tal não foi possível. E isto porque, tal como se pode observar quer pelos valores da tabela correspondente, quer pela Figura 3.4, a discriminante quadrática não funciona muito bem com este conjunto de dados.

3.3 Relação com o método baseado em curvas ROC

As curvas ROC tiveram origem em 1950 no contexto da teoria de decisão estatística para avaliação da detecção de sinais em radar e na psicologia sensorial e desde aí têm sido utilizadas com bastante frequência. É muito vasta a literatura sobre este tema nomeadamente em aplicações ligadas à medicina (ver por exemplo Braga, 2001). No âmbito do r.p. supervisionado, Tortorella (2000) apresenta uma regra de rejeição binária óptima, no sentido da maximização de uma função de utilidade, baseada na curva ROC.

O objectivo desta secção é fazer a ligação entre a abordagem do problema da rejeição (para duas classes) através da curva ROC (Tortorella, 2000) e a generalização da abordagem de Chow (1970), mostrando que as duas abordagens são teoricamente equivalentes. Em Santos-Pereira e Pires (2004b) apresenta-se uma versão mais completa desta secção na medida em que é tida em conta não só a ligação entre as duas abordagens como uma versão compacta da generalização da regra de Chow apresentada ao longo das secções anteriores.

Tortorella (2000, pág. 612) afirma assumir que o classificador fornece, para cada observação multivariada (**x**), um valor $x \in [0,1]$ que exprime o grau de confiança com que a observação **x** pertence à classe *P* (na notação adoptada neste trabalho, G_1). Para concretizar a regra de classificação (sem rejeição) é necessário definir um ponto limiar, $u \in [0,1]$ (a que Tortorella chama "*confidence threshold*" e denota por *t*), tal que um objecto de origem desconhecida é classificado em G_1 se $x>u$. É imediato reconhecer que Tortorella está a admitir que o classificador fornece uma estimativa da probabilidade *a posteriori* da observação pertencer à classe G_1. Uma vez que se pretende estabelecer a equivalência teórica entre as duas abordagens vai-se admitir que o classificador fornece não uma estimativa mas a verdadeira probabilidade *a posteriori* da observação pertencer à classe G_1, ou seja $x(\mathbf{x})=P(G_1|\mathbf{x})$. Recorde-se que

$$P(G_1 \mid \mathbf{x}) = \frac{\pi_1 f_1(\mathbf{x})}{\pi_1 f_1(\mathbf{x}) + \pi_2 f_2(\mathbf{x})}.$$

Para considerar a rejeição o que Tortorella faz é considerar que é necessário determinar não um mas dois limiares de rejeição, tais que: **x** é classificada em G_2 se $0 \le x < u_1$, em G_1 se $u_2 < x \le 1$ e não é classificada (ou seja, é rejeitada) se $u_1 \le x \le u_2$.

Tortorella considera "lucros" associados a cada possibilidade de classificação e vai maximizar a utilidade esperada. Na abordagem de Chow consideram-se custos e minimiza-se o risco. Neste aspecto não há dúvida que há equivalência entre as duas abordagens. Os custos considerados são os que se têm vindo a ser utilizados: $C_{11}=C(1|1)$ e $C_{22}=C(2|2)$ (custos de classificação correcta e como tal ≤0); $C_{21}=C(1|2)$ e $C_{12}=C(2|1)$ (custos de classificação incorrecta, logo >0); e finalmente, o custo de rejeição, que se passará a designar por $C_r>0$, por questões de simplificação (*t* na notação de Chow, 1970, e na notação até agora adoptada). A função de custos adoptada passará então a ser

$$C(i \mid j) = \begin{cases} C_{ii} & \text{se} \quad i = j \\ C_{ji} & \text{se} \quad i \neq j \quad i, j = 1, 2, ..., c. \\ C_r & \text{se} \quad i = I \end{cases} \qquad (3.3)$$

A curva ROC (teórica) é definida implicitamente em função do limiar *u*. Assim, para cada $u \in [0,1]$, calcule-se *TPR* ("*true positive rate*") e *FPR* ("*false positive rate*")

$$TPR(u) = \int_u^1 f_P(x)dx \quad \text{e} \quad FPR(u) = \int_u^1 f_N(x)dx, \tag{3.4}$$

onde $f_P(x)$ e $f_N(x)$ representam a densidade de $P(G_1|\mathbf{x})$, encarada como uma v.a. que é uma função do vector aleatório **x**, quando este tem distribuição condicionada a cada classe, respectivamente. A curva ROC propriamente dita é a representação gráfica dos pontos $\{(FPR(u), TPR(u)),\ u \in [0,1]\}$. Na Figura 3.5 apresenta-se um exemplo de uma curva ROC teórica. Note-se que embora, em geral, $u \in [0,1]$, há casos para os quais os valores assumidos por $P(G_1|\mathbf{x})$ são apenas um subconjunto deste intervalo, seja $[u_{\min}, u_{\max}]$. Nestes casos os pontos da curva ROC para $u \in [0, u_{\min}]$ estão todos coincidentes com o ponto do canto superior direito da curva ROC e os pontos tais que $u \in [u_{\max}, 1]$ estão todos coincidentes com o ponto do canto inferior esquerdo da curva ROC.

As taxas definidas em (3.4) podem ser escritas de uma outra forma mais directa sem necessidade de calcular as densidades $f_P(x)$ e $f_N(x)$. Para isso note-se que

$$x > u \Leftrightarrow P(G_1 \mid \mathbf{x}) > u \Leftrightarrow \frac{\pi_1 f_1(\mathbf{x})}{\pi_1 f_1(\mathbf{x}) + \pi_2 f_2(\mathbf{x})} > u \Leftrightarrow \mathbf{x} \in R_u = D_1(u),$$

onde $R_u \subset \mathbb{R}^p$ representa, para cada $u \in [0,1]$, a usual região de classificação (anteriormente designada por D_1) associada à classe G_1 no espaço das variáveis originais mas que agora se representa desta forma para indicar a dependência de *u*. Note-se que $u = 0 \Rightarrow R_u = \mathbb{R}^p, u = 1 \Rightarrow R_u = \varnothing$ e $\forall_{u_1,u_2} : 0 \le u_1 < u_2 \le 1, R_{u_1} \supset R_{u_2}$. Admite-se, como é habitual, que o conjunto $\{P(G_1 \mid \mathbf{x}) = u\}$ tem medida de probabilidade nula para qualquer uma das classes e para qualquer $u : u_{\min} < u < u_{\max}$. Tem-se então

$$TPR(u) = \int_{R_u} f_1(\mathbf{x})d\mathbf{x} \quad \text{e} \quad FPR(u) = \int_{R_u} f_2(\mathbf{x})d\mathbf{x}, \tag{3.5}$$

e, por outro lado,

$$f_P(u) = -\frac{d}{du}TPR(u) \quad \text{e} \quad f_N(u) = -\frac{d}{du}FPR(u).$$

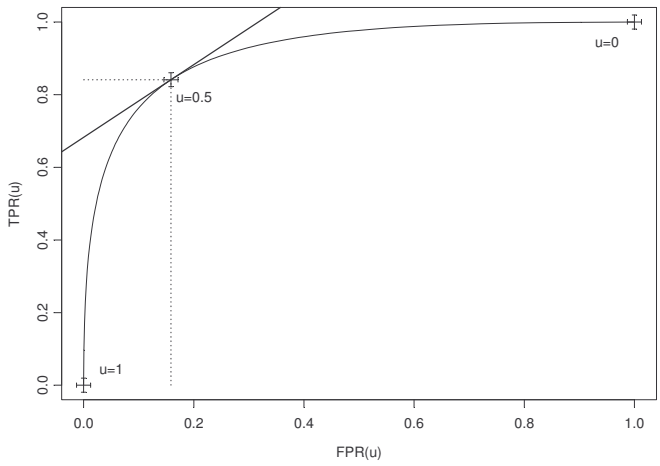


Figura 3.5- Exemplo de uma curva ROC teórica (X|G_1~N(0,1), X|G_2~N(2,1), $\pi_1 = \pi_2 = 0.5$) com linha tangente em u=0.5.

O que Tortorella mostra é que o lucro é maximizado (ou como se viu, o risco é minimizado) se se escolherem limiares u_1 e u_2 tais que a derivada da curva ROC associada a cada um desses limiares é, respectivamente

$$m_1 = \frac{\pi_2}{\pi_1} \times \frac{C_r - C_{22}}{C_{12} - C_r} \quad \text{e} \quad m_2 = \frac{\pi_2}{\pi_1} \times \frac{C_{21} - C_r}{C_r - C_{11}}. \tag{3.5}$$

Tortorella propõe depois um processo geométrico para determinação dos limiares a partir de curvas ROC empíricas.

Por outro lado, como se viu na Secção 3.1.2 aquando da generalização da abordagem de Chow e de acordo com a função de custos definida em (3.3), conclui-se que[4]

$$\mathbf{x} \in D_I \qquad \text{se} \qquad \frac{C_r - C_{22}}{C_{12} - C_r} \times \frac{\pi_2}{\pi_1} \leq \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \leq \frac{C_{21} - C_r}{C_r - C_{11}}, \tag{3.6}$$

mas como

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq m \Leftrightarrow P(G_1 \mid \mathbf{x}) \geq u, \text{ se e só se } m = \frac{\pi_2}{\pi_1} \times \frac{u}{1-u},$$

conclui-se que as duas abordagens são teoricamente equivalentes se e só se a derivada da curva ROC for igual a

$$\frac{\pi_2}{\pi_1} \times \frac{u}{1-u}.$$

É o que se mostra na proposição seguinte.

Proposição: Considere-se um vector aleatório **x** com distribuições distintas em duas classes, G_1 e G_2, caracterizadas respectivamente pelas densidades $f_1(\mathbf{x})$ e $f_2(\mathbf{x})$, e com probabilidades *a priori* π_1 *e* π_2. Dada a curva ROC, $\{(FPR(u), TPR(u)),\ u \in [0,1]\}$, com $FPR(u)$ e $TPR(u)$ definidas em (3.5), tem-se que

$$\left.\frac{dTPR(u)}{dFPR(u)}\right|_{u=u_0} = \frac{\pi_2}{\pi_1} \times \frac{u_0}{1-u_0},\ u_{\min} \leq u_0 < u_{\max}.$$

onde $u_{\min} = \min_{\mathbf{x}} P(G_1 \mid \mathbf{x})$ e $u_{\max} = \max_{\mathbf{x}} P(G_1 \mid \mathbf{x})$.

Demonstração: Em primeiro lugar note-se que

[4] Note-se que para certas combinações de custos a opção de rejeição pode não ser activada. Facilmente se verifica que uma condição necessária, mas não suficiente, para a sua activação é que $C_r < C_{21}$ e $C_r < C_{12}$.

$$\left.\frac{dTPR(u)}{dFPR(u)}\right|_{u=u_0} = \frac{\left.\frac{dTPR(u)}{du}\right|_{u=u_0}}{\left.\frac{dFPR(u)}{du}\right|_{u=u_0}} = \frac{\lim_{h\to 0}\frac{TPR(u_0+h)-TPR(u_0)}{h}}{\lim_{h\to 0}\frac{FPR(u_0+h)-FPR(u_0)}{h}} = \frac{\lim_{h\to 0}\frac{\int_{R_{u_0+h}} f_1(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_1(\mathbf{x})\,d\mathbf{x}}{h}}{\lim_{h\to 0}\frac{\int_{R_{u_0+h}} f_2(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_2(\mathbf{x})\,d\mathbf{x}}{h}} =$$

$$= \lim_{h\to 0}\frac{\int_{R_{u_0+h}} f_1(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_1(\mathbf{x})\,d\mathbf{x}}{\int_{R_{u_0+h}} f_2(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_2(\mathbf{x})\,d\mathbf{x}}.$$

Considere-se agora separadamente os casos $h>0$ e $h<0$. Para $h>0$ tem-se $R_{u_0+h} \subset R_{u_0}$ pelo que

$$\int_{R_{u_0+h}} f_i(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_i(\mathbf{x})\,d\mathbf{x} = -\int_{R_{u_0}\setminus R_{u_0+h}} f_i(\mathbf{x})\,d\mathbf{x} \quad ,i=1,2.$$

Por outro lado para $h<0$ tem-se $R_{u_0+h} \supset R_{u_0}$ e

$$\int_{R_{u_0+h}} f_i(\mathbf{x})\,d\mathbf{x} - \int_{R_{u_0}} f_i(\mathbf{x})\,d\mathbf{x} = \int_{R_{u_0+h}\setminus R_{u_0}} f_i(\mathbf{x})\,d\mathbf{x} \quad ,i=1,2.$$

Então as derivadas laterais são dadas por

$$\left.\frac{dTPR(u)}{dFPR(u)}\right|_{u=u_0^+} = \lim_{h\to 0^+}\frac{-\int_{R_{u_0}\setminus R_{u_0+h}} f_1(\mathbf{x})\,d\mathbf{x}}{-\int_{R_{u_0}\setminus R_{u_0+h}} f_2(\mathbf{x})\,d\mathbf{x}}$$

$$\left.\frac{d\mathrm{TPR}(u)}{d\mathrm{FPR}(u)}\right|_{u=u_0^-} = \lim_{h\to 0^-}\frac{\int_{R_{u_0+h}\setminus R_{u_0}} f_1(\mathbf{x})\,d\mathbf{x}}{\int_{R_{u_0+h}\setminus R_{u_0}} f_2(\mathbf{x})\,d\mathbf{x}}.$$

Mas

$$\text{se } \mathbf{x}\in R_{u_0}\setminus R_{u_0+h},\ u_0 \le P(G_1\mid\mathbf{x}) \le u_0+h$$
$$\text{se } \mathbf{x}\in R_{u_0+h}\setminus R_{u_0},\ u_0+h \le P(G_1\mid\mathbf{x}) \le u_0,$$

e à medida que $h \to 0^+$ ($h \to 0^-$), a região $R_{u_0} \setminus R_{u_0+h}$ ($R_{u_0+h} \setminus R_{u_0}$) é cada vez mais "estreita" e é tal que $P(G_1|\mathbf{x})$ tende para u_0 e $P(G_2|\mathbf{x})$=1- $P(G_1|\mathbf{x})$ tende para 1- u_0.

Então para $h \to 0^+$ tem-se,

$$\lim_{h\to 0^+} \frac{\int_{R_{u_0} \setminus R_{u_0+h}} f_1(\mathbf{x}) d\mathbf{x}}{\int_{R_{u_0} \setminus R_{u_0+h}} f_2(\mathbf{x}) d\mathbf{x}} = \lim_{h\to 0^+} \frac{\int_{R_{u_0} \setminus R_{u_0+h}} \frac{f(\mathbf{x}) P(G_1 \mid \mathbf{x})}{\pi_1} d\mathbf{x}}{\int_{R_{u_0} \setminus R_{u_0+h}} \frac{f(\mathbf{x}) P(G_2 \mid \mathbf{x})}{\pi_2} d\mathbf{x}} = \frac{\pi_2}{\pi_1} \times \frac{u_0}{1-u_0} \lim_{h\to 0^+} \frac{\int_{R_{u_0} \setminus R_{u_0+h}} f(\mathbf{x})}{\int_{R_{u_0} \setminus R_{u_0+h}} f(\mathbf{x})} = \frac{\pi_2}{\pi_1} \times \frac{u_0}{1-u_0},$$

e do mesmo modo quando $h \to 0^-$.

Notas

- Esta proposição pode ser relacionada com um resultado bem conhecido da análise de curvas ROC (ver por exemplo van Trees, 1968, pág. 44, Propriedade 3).
- Embora se tenha acabado de demonstrar que os dois métodos são equivalentes isso não significa que num caso concreto o sejam (obviamente por questões de estimação).
- Poderiam ainda ser apresentadas outras demonstrações para casos particulares (por exemplo, quaisquer distribuições univariadas, distribuições normais com médias diferentes e variâncias iguais, distribuições normais com médias iguais e variâncias diferentes, etc.). A título de exemplo considere-se a situação seguinte.

Exemplo 3.3

Considere-se $\mathrm{X}|G_1 \sim N_2(\mathbf{0},\mathbf{I})$ e $\mathrm{X}|G_2 \sim N_2(\boldsymbol{\mu}_1,\mathbf{I})$ com $\boldsymbol{\mu}_1=(2,0)^{\mathrm{T}}$.

$$P(G_1 \mid \mathbf{x}) \geq t \Leftrightarrow \frac{\pi_1}{\pi_1 + \pi_2 \exp(2x_1 - 2)} \geq t \Leftrightarrow x_1 < \frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right) + \ln\left(\frac{1-t}{t}\right)\right] + 1.$$

Tem-se então

$$\mathrm{FPR}(t) = \int_{R_t} f_2(x)\,dx = \int_{-\infty}^{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]+1} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-2)^2}{2}}\,dx = \Phi\left(\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]-1\right).$$

$$\mathrm{TPR}(t) = \int_{R_t} f_1(x)\,dx = \int_{-\infty}^{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]+1} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}\,dx = \Phi\left(\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]+1\right).$$

Logo

$$\frac{d\,\mathrm{TPR}(t)}{d\,\mathrm{FPR}(t)} = \frac{\frac{d\,\mathrm{TPR}(t)}{dt}}{\frac{d\,\mathrm{FPR}(t)}{dt}} = \frac{\varphi\left\{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]+1\right\}\times\left\{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]+1\right\}'}{\varphi\left\{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]-1\right\}\times\left\{\frac{1}{2}\left[\ln\left(\frac{\pi_1}{\pi_2}\right)+\ln\left(\frac{1-t}{t}\right)\right]-1\right\}'} =$$

$$= \frac{\pi_2}{\pi_1}\times\frac{t}{1-t}.$$

Algumas considerações

Com a abordagem de Chow:

- não é necessário recorrer a qualquer procedimento geométrico para obtenção dos limiares de classificação.
- é indiferente utilizar probabilidades *a posteriori* ou densidades.
- pode trabalhar-se com qualquer número de classes.
- não se está limitado a um conjunto discreto de valores de *u* (como em Tortorella) o que pode trazer desvantagens no desempenho do método baseado em curvas ROC, conforme se irá ver na aplicação a seguir apresentada.
- o resultado também é relevante mesmo quando não se considera a rejeição pois permite a obtenção do chamado "*optimal operating point*" (valor de corte que permite a separação entre duas classes) por outra via que não a utilização da curva ROC (ver Halpern *et al*., 1996), com a vantagem de se adaptar a qualquer número de classes.

3.3.1 Aplicação

De acordo com uma das notas acabadas de referir, embora os dois métodos sejam teoricamente equivalentes, isso não significa que num caso concreto o sejam. Daí a importância da comparação dos resultados obtidos pelos dois métodos quando aplicados a um conjunto de dados reais.

Para levar a cabo este estudo experimental com o objectivo de comparar o método apresentado, com o método proposto por Tortorella (2000), foi considerado um conjunto mais vasto dos dados PIMA, referidos no exemplo 3.2. Esses dados encontram-se disponíveis em *http://www.ics.uci.edu/~mlearn/MLSummary.html* e dizem respeito a uma amostra de 768 mulheres entre as quais 268 têm diabetes (classe G_1 ou *P*) e 500 não apresentam diabetes (classe G_2 ou *N*), observadas relativamente a 8 variáveis. As probabilidades *a priori* estimadas são portanto $\hat{\pi}_1 = 0.35$ e $\hat{\pi}_2 = 0.65$.

Tal como no artigo de Tortorella, a amostra foi dividida aleatoriamente em três conjuntos[5]: amostra de treino à qual correspondem 614 observações ($\simeq 80\%$), 77 casos observações ($\simeq 10\%$) para a construção das curvas ROC e as restantes 77 ($\simeq 10\%$) que irão constituir a amostra de teste.

Foram escolhidos três classificadores: discriminante linear (LDA), discriminante logística (LD) e redes neuronais cuja estimação, a partir da amostra de treino, permitiu a obtenção de três regras discriminantes.

No caso particular das redes neuronais escolheu-se o MLP com 8 unidades da camada de *input*, 4 unidades na camada escondida e 1 unidade na camada de *output*. Esta opção teve como objectivo adoptar o mesmo tipo de rede adoptado por Tortorella. Para utilizar esta metodologia no S-Plus, recorreu-se ao módulo *nnet* tendo-se incluído os argumentos opcionais *rang*=0.1, *decay*=0.01, *maxit*=20000. O funcional do custo adoptado é o utilizado em problemas de classificação binária que se baseiam no critério da entropia (Silva, 2000) e os restantes parâmetros são os utilizados habitualmente neste tipo de situações (Venables e Ripley, 1999).

[5] Procedimento habitual quando se utilizam redes neuronais.

Foram também adoptadas as sete combinações de custos escolhidas por Tortorella (2000) e reproduzidas na Tabela 3.2.

Caso	*CFN* $C(2\|1)$	*CFP* $C(1\|2)$	*CTN* $C(2\|2)$	*CTP* $C(1\|1)$	*CR* C_r
a	50	25	-200	-400	12.5
b	50	25	-100	-200	12.5
c	50	25	-50	-100	12.5
d	50	25	-25	-50	12.5
e	100	50	-25	-50	12.5
f	200	100	-25	-50	12.5
g	400	200	-25	-50	12.5

Tabela 3.2- Combinações de custos.

Quando a opção de rejeição não é activada, a regra de classificação é a que se segue, qualquer que seja a combinação de custos considerada:

$$\text{Classificar } \mathbf{x} \text{ em } G_1 \text{ (ou } P\text{) se } P(G_1 \mid \mathbf{x}) \geq \frac{CFP - CTN}{CFP - CTN + CFN - CTP} = \frac{1}{3},$$

e em G_2 (ou N) caso contrário. Para cada caso e cada classificador estimou-se, com base na amostra de teste, as taxas de erro *FPR* (*false positive rate*), *FNR* (*false negative rate, FNR*= 1-*TPR*) e as taxas de classificação correcta *TPR* (*true positive rate*) e *TNR* (*true negative rate, TNR*= 1-*FPR*).

Sendo activada a opção de rejeição têm de se calcular, para cada um dos métodos, dois limiares, u_1 e u_2, por forma a que uma observação seja rejeitada se $u_1 < \hat{P}(G_1 \mid \mathbf{x}) < u_2$.

Para o método proposto, baseado na abordagem de Chow, estes limiares são independentes do classificador e dados por

$$u_1 = \frac{CR - CTN}{CFN - CTN}, \quad u_2 = \frac{CFP - CR}{CFP - CTP}.$$

Para o método baseado nas curvas ROC obtém-se, para cada classificador, uma estimativa da curva ROC independente, utilizando-se o segundo conjunto de observações e onze limiares candidatos equidistantes u=0, 0.1, ...,1.

De seguida obtém-se o invólucro convexo da curva ROC e através deste os limiares finais, u_1 e u_2, tais que os correspondentes declives são tão próximos quanto possível dos valores dados por m_1 e m_2 definidos em (3.5). Os valores dos pontos e declives do invólucro convexo são apresentados na Tabela 3.3. A Figura 3.6 apresenta as curvas ROC, estimadas pelos três classificadores, bem como os respectivos invólucros convexos.

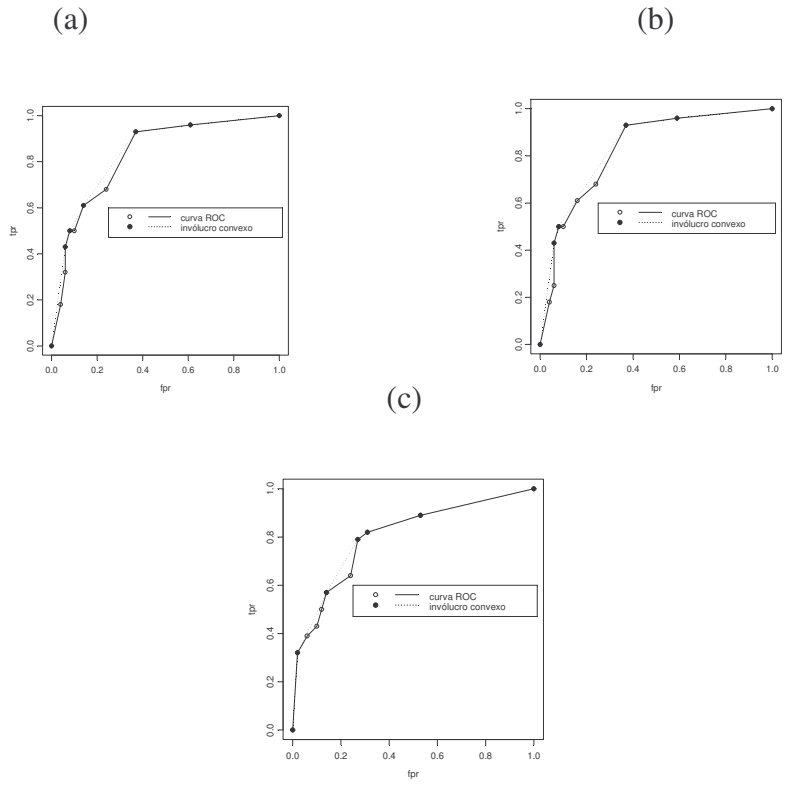


Figura 3.6- Curvas ROC estimadas para os dados PIMA e três classificadores. (a) LDA, (b) LD, (c) MLP.

	Vértices do invólucro convexo	u	Declives do invólucro convexo
LDA	(0.00,0.00)	1.0	∞
	(0.06,0.43)	0.7	7.17
	(0.08,0.50)	0.6	3.50
	(0.14,0.61)	0.4	1.83
	(0.37,0.93)	0.2	1.39
	(0.61,0.96)	0.1	0.13
	(1.00,1.00)	0	0.10
LD	(0.00,0.00)	1.0	∞
	(0.06,0.43)	0.7	7.17
	(0.08,0.50)	0.6	3.50
	(0.37,0.93)	0.2	1.72
	(0.59,0.96)	0.1	0.14
	(1,1)	0	0.10
MLP	(0.00,0.00)	1.0	∞
	(0.02,0.32)	0.9	16.0
	(0.14,0.57)	0.5	2.08
	(0.27,0.79)	0.3	1.69
	(0.31,0.82)	0.2	0.75
	(0.53,0.89)	0.1	0.32
	(1.00,1.00)	0.0	0.23

Tabela 3.3- Vértices e respectivos declives do invólucro convexo das curvas ROC.

Depois de obtidos u_1 e u_2 pelos dois métodos, classificam-se as observações da amostra de teste e estimam-se novamente *FPR*, *FNR*, *TPR* e *TNR*, tal como dantes, e ainda as probabilidades de rejeição de um caso negativo, *RN* e de um caso positivo, *RP*. A Tabela 3.4 sumaria os resultados. Note-se que para as quatro primeiras combinações de custos (casos **a**, **b**, **c** e **d**) a opção de rejeição não é activada. Tal facto não se deve à falha da condição necessária de activação da rejeição apresentada em (3.6) mas sim à ocorrência de $u_2<u_1$.

Classificador Método	Caso	*FPR*	*TPR*	*FNR*	*TNR*	*RP*	*RN*	u_1	u_2
	a b c d	0.34	0.83	0.17	0.66	-	-	-	-
LDA	e	0.23	0.80	0.17	0.54	0.03	0.23	0.30	0.38
Chow	f	0.02	0.67	0.03	0.34	0.30	0.64	0.17	0.58
	g	0.02	0.47	0.00	0.15	0.53	0.83	0.09	0.75
	a b c d	0.34	0.83	0.17	0.66	-	-	-	-
LDA	e	0.62	0.9	0.10	0.38	-	-	0.20	0.20
ROC	f	0.02	0.67	0.10	0.38	0.23	0.60	0.20	0.60
	g	0.02	0.63	0.10	0.38	0.27	0.60	0.20	0.70
	a b c d	0.28	0.77	0.23	0.72	-	-	-	-
LD	e	0.15	0.77	0.20	0.68	0.03	0.17	0.30	0.38
Chow	f	0.02	0.67	0.13	0.40	0.20	0.58	0.17	0.58
	g	0.02	0.40	0.00	0.19	0.60	0.79	0.09	0.75
	a b c d	0.28	0.77	0.23	0.72	-	-		
LD	e	0.57	0.83	0.17	0.43	-	-	0.20	0.20
ROC	f	0.02	0.63	0.14	0.45	0.23	0.53	0.20	0.60
	g	0.02	0.50	0.13	0.45	0.37	0.53	0.20	0.70
	a b c d	0.28	0.70	0.30	0.72	-	-	-	-
MLP	e	0.21	0.70	0.30	0.68	0.00	0.11	0.30	0.38
Chow	f	0.11	0.53	0.14	0.55	0.33	0.34	0.17	0.58
	g	0.00	0.37	0.07	0.36	0.56	0.64	0.09	0.75
	a b c d	0.28	0.70	0.30	0.72	-	-	-	-
MLP	e	0.32	0.70	0.30	0.68	-	-	0.30	0.30
ROC	f	0.07	0.00	0.57	0.2	0.73	0.43	0.20	0.90
	g	0.07	0.00	0.00	0.00	0.93	1.00	0.00	0.90

Tabela 3.4- Resultados da classificação e limiares de rejeição para os dados PIMA, com base na amostra de teste.

Finalmente, estima-se para cada combinação de custos e classificador, o risco total associado (simétrico da função utilidade definida por Tortorella, 2000). Para a situação sem rejeição o risco pode ser obtido através de

$$\widehat{RT} = \hat{\pi}_1(CTP.TPR + CFN.FNR) + \hat{\pi}_2(CTN.TNR + CFP.FPR),$$

enquanto que na situação com inclusão de rejeição

$$\widehat{RT_{rej}} = \hat{\pi}_1(CTP.TPR + CFN.FNR + CR.RP) + \hat{\pi}_2(CTN.TNR + CFP.FPR + CR.RN).$$

Estes resultados encontram-se na Tabela 3.5 . Sendo o objectivo o da minimização do risco, o menor (melhor) valor de cada linha é representado a ***bold***. Dos três classificadores escolhidos a LDA é o que dá melhor resultados. Por outro lado, comparando os resultados com e sem rejeição conclui-se que para as combinações de custos em que a opção de rejeição é activada, o risco com rejeição é geralmente inferior ao risco sem rejeição (a LDA/Chow para o caso **e** e o MLP/ROC para o caso **f** constituem a excepção). Para terminar pode concluir-se que neste exemplo o método de rejeição baseado na regra de Chow permite atingir menores riscos do que o método de rejeição baseado na curva ROC.

	LDA			LD			MLP		
Caso	Sem rejeição	Chow	ROC	Sem rejeição	Chow	ROC	Sem rejeição	Chow	ROC
a	**-193.5**	-	-	-192.8	-	-	-181.8	-	-
b	**-92.5**	-	-	-92.1	-	-	-86.0	-	-
c	**-42.0**	-	-	-41.8	-	-	38.1	-	-
d	**-16.8**	-	-	-16.6	-	-	-14.2	-	-
e	-8.3	-7.4	-	-8.0	**-11.1**	-	-4.4	-5.1	-
f	8.8	**-7.3**	-3.7	9.1	-2.2	-1.9	15.3	2.9	47.9
g	42.8	**1.0**	5.5	43.4	1.6	10.7	54.5	5.1	21.3

Tabela 3.5- Estimativas do risco total, com e sem rejeição, para os dados PIMA, com base na amostra de teste.

Capítulo 4

Rejeição de *outliers* em dados multivariados

O objectivo deste capítulo é desenvolver um método eficaz para a detecção de *outliers* com o intuito de abordar o problema da classificação com inclusão da opção de rejeição por existência de observações deste tipo.

Embora não se pretenda desenvolver demasiado sobre os procedimentos utilizados no tratamento de *outliers*, em especial o caso univariado e os conjuntos de dados normais- não porque não seja importante, mas porque já foi objecto de trabalhos anteriores, nomeadamente em Barnett e Lewis (1994), alem de se desviar um pouco do âmbito deste trabalho- torna-se, contudo, necessária uma breve introdução e apresentação de alguns conceitos que permitem a contextualização do tema abordado.

4.1 Introdução

4.1.1 Motivação

Um assunto que, desde cedo, interessou os estatísticos foi o problema dos *outliers*. Embora a primeira discussão sobre *outliers* apareça em 1777 com Bernoulli, a primeira pessoa a especificar um critério de rejeição de *outliers,* propondo-se solucionar o problema das observações discordantes (em sentido lato, qualquer observação que surpreenda o investigador), foi Peirce em 1852. Como referem Barnett e Lewis (1994), já nessa altura Peirce afirmava que "Na maioria das séries de observações, encontram-se algumas que diferem tanto das outras que ... espantam e induzem em erro o investigador". Assim, dada a vulgaridade com que aparecem nos dados, saber a forma adequada de lidar com essas observações discrepantes das restantes tem sido uma preocupação dos investigadores do passado e do presente. Durante o século passado, a abordagem aos *outliers* foi, gradualmente, sendo feita de um modo mais rigoroso e com objectivos bem definidos; essa investigação foi frutífera pois foi construído um vasto conjunto de instrumentos de análise, permitindo uma ruptura com o senso comum. Por exemplo, note-se que o comentário de Peirce (1852), embora aponte a característica

básica dos *outliers* é, actualmente, considerado um pouco obsoleto e restritivo porque "observações que diferem tanto das outras",

(i) não necessariamente "espantam e induzem em erro o investigador";

(ii) podem não ser "más" ou "erróneas";

(iii) podem indicar algo de extraordinário, de fora de normal relativamente ao que seria de esperar.

Nos dias de hoje, evitam-se posições extremas: a simples rejeição das observações sem tratamento posterior (correndo-se o risco de perder informação genuína), ou a inclusão dessas observações (correndo-se o risco de contaminação). Conforme se fará referência mais à frente, é habitual a utilização de procedimentos robustos que utilizam todas as observações mas que minimizam a influências dos *outliers*. A inferência robusta é na sua essência pouco sensível a grandes ou pequenos desvios sobre as hipóteses assumidas como por exemplo a normalidade ou a independência. Genericamente entende-se por procedimento robusto aquele que consegue lidar com situações que envolvem *outliers*. Hampel (2000, 2001) tece alguns comentários sobre a inferência robusta, o porquê da sua necessidade- comparando-a com os procedimentos não robustos- focando também o tema dos *outliers* neste contexto e salientando a importância do ponto de rotura[1]. De realçar a grande importância destes procedimentos nomeadamente em dados p-variados com $p>2$ pela dificuldade em recorrer à inspecção visual.

Surgindo da necessidade de se encontrarem observações anómalas que integram a tarefa de pré-processamento associado à aplicação de qualquer método multivariado, por forma a preservar os resultados dos desvios provocados por tais observações, a detecção de *outliers* é pois bastante importante nas mais variadas aplicações da análise multivariada. De destacar o especial interesse na classificação supervisionada (ou análise discriminante) se, ao predizer as classes, pretendermos classificar uma observação como não pertencente a nenhuma das classes pré-definidas.

[1] Percentagem de contaminação arbitrária a partir da qual podem ser produzidos também valores arbitrários da estimativa. Um estimador é tanto mais robusto quanto maior for o seu ponto de rotura (na prática não fazem sentido valores superiores a 50%).

Mas o que se entende por *outliers*? *Outliers* são observações consideradas não consistentes com o resto dos dados: segundo Webb (1999) existem duas hipóteses. Podem ser observações genuínas (observações discordantes de acordo com Beckman e Cook, 1983), que são surpreendentes para o observador, podendo até ser as mais valiosas (ver por exemplo Davies e Gather, 1993) ao revelar uma certa estrutura nos dados que indicam desvios da normalidade, como por exemplo numa distribuição de caudas longas. Em alternativa podem ser fruto dos erros causados pela cópia ou transferência de dados ou seja, observações que não pertencem à distribuição alvo (observações contaminantes de acordo com Beckman e Cook, 1983). Hampel (2002, pág. 5) refere que o tratamento de *outliers* de diferentes tipos pode ser feito exactamente da mesma forma na maior parte das situações, a sua interpretação é que difere. Rosado (1984) define *outlier* (no sentido de observação discordante) da seguinte forma: "é aquela observação que, após rejeição da homogeneidade nas observações, for responsável por essa rejeição".

Na realidade é com frequência que nos deparamos com situações onde os *outliers* são realmente as observações mais valiosas de toda a amostra, indicando um novo efeito ou uma descoberta de que não se suspeitava. Hampel (2000) faz referência ao exemplo da descoberta do buraco de ozono sob a Antárctida (onde os dados são claramente *outliers*). Aqui se cita um bom exemplo que reforça o que foi referido no Capítulo 2 quando se abordou a questão da rejeição, provando-se que há situações onde não é mesmo aconselhável "livrarmo-nos" simplesmente dos dados considerados *outliers* sem uma análise posterior. Segundo este autor o rumor de que os Americanos não foram os primeiros a descobrir o buraco de ozono devido ao seu equipamento altamente sofisticado em que os *outliers* eram automaticamente rejeitados foi bastante contestado; na realidade, os dados eram automaticamente marcados como necessitados de atenção especial, depois comparados com outros (séries de observações totalmente discordantes) e só depois considerados seriamente duvidosos. Mas o que se passou afinal foi que estas séries continham um erro sistemático medindo algo bastante diferente do pretendido sendo portanto essa a causa de engano dos cientistas. Este facto verídico constitui um exemplo claro de que a identificação de *outliers* deve ser realizada para proveito da ciência e não por puras razões estatísticas (como por exemplo, a eficiência).

Até agora (Capítulos 2 e 3), admitiu-se que a distribuição das observações ou as probabilidades *a posteriori* de cada classe eram conhecidas. Sendo essa hipótese válida qual seria o desempenho do classificador atrás definido, se o conjunto de dados contivesse observações consideradas *outliers*?

Comece-se por analisar o que aconteceria se apenas se comparassem as probabilidades *a posteriori* de cada classe. Sem perda de generalidade considere-se um exemplo simples com uma variável e duas classes em que $X|G_1 \sim N(-2,1.3)$, $X|G_2 \sim N(5,1.5)$, $\pi_1 = \pi_2 = 0.5$.

Como é bem sabido, em dados reais há determinado número de observações que não são bem descritas pelo modelo probabilístico escolhido: é por exemplo o caso de padrões que ocorrem onde a densidade de probabilidade é próxima de zero. Imagine-se então que no exemplo em causa existiam duas observações nessas condições: x=1 e x=11. Na Figura 4.1 representam-se as funções de densidade de probabilidade condicionais às classes e aquelas duas observações. Observando a figura conclui-se que provavelmente, faria sentido seria classificar a observação x=1 (representada por um triângulo) numa zona de indecisão e a observação x=11 (representada por uma cruz) como *outlier*.

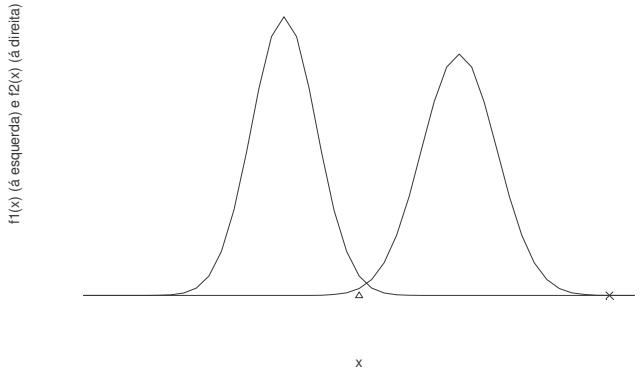


Figura 4.1- Funções densidade de probabilidade condicionais às classes ($X|G_1 \sim N(-2, \sigma=1.3)$, $X|G_2 \sim N(5, \sigma=1.5)$, $\pi_1 = \pi_2 = 0.5$) e duas observações x=1 e x=11 (representadas por Δ e $\times$, respectivamente).

Observem-se agora as probabilidades *a posteriori*, representadas na Figura 4.2, correspondentes ao exemplo acabado de descrever[2]. Facilmente se verifica que de acordo com o correspondente critério de decisão (maximização das probabilidades *a posteriori*), a observações localizada em x=1 seria classificada na classe G_1 e a observação localizada em x=11 seria classificada em G_2. Além disso, também se julga importante salientar que para observações muito distantes (como é o caso da que foi designada por ×) uma das probabilidades *a posteriori* é aproximadamente 1 o que dá uma indicação errada de que a observação deverá ser classificada em determinada classe quando de facto, seria mais natural considerá-la um *outlier*.

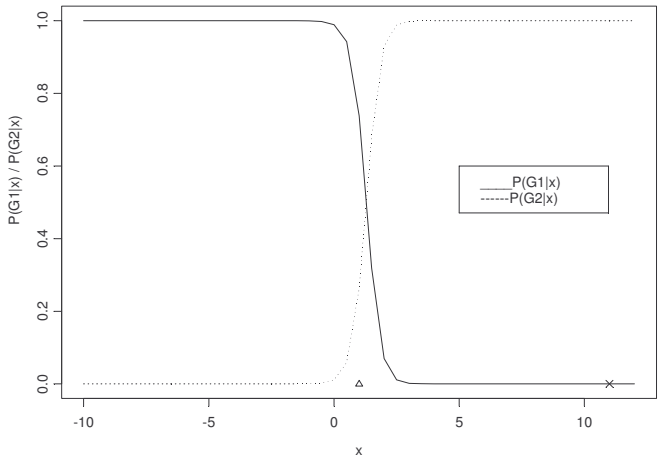


Figura 4.2- Probabilidades *a posteriori* associadas às classes G_1 e G_2 (X|G_1~N(-2, σ=1.3), X|G_2~N(5, σ=1.5), $\pi_1 = \pi_2 = 0.5$).

O que realmente se verifica na maior parte das situações é que o conhecimento das probabilidades *a posteriori* é insuficiente para a classificação duma observação como *outlier*. Por outro lado as densidades são geralmente desconhecidas e mesmo que se pensasse num processo de estimação certamente que num exemplo deste tipo, nos pontos considerados, os correspondentes valores também seriam pequenos e portanto a informação fornecida também seria insuficiente. Ripley (1996, Cap. 2, pág. 24) sugere duas técnicas para a detecção deste tipo de situações, em que os dados contêm *outliers*. A primeira corresponde à introdução de uma classe adicional baseada num modelo para

[2] Dado que se está numa situação de variâncias diferentes a regra discriminante óptima é do tipo quadrático o que no caso de uma variável corresponde a um intervalo. Tem-se assim que $D_1 = [-47.46, 1.21]$ e $D_2 =]-\infty, -47.46[\cup]1.21, +\infty[$. Isto significa que na região não representada no gráfico existe uma outra inversão da probabilidade *a posteriori*.

os *outliers*. A segunda hipótese consiste na comparação da densidade de probabilidade estimada com um limiar pré-definido, concluindo-se que uma observação é *outlier* se a densidade nesse ponto for inferior ao limiar. No entanto, tal como o próprio autor também refere, ambas as sugestões apresentam grandes dificuldades. Nomeadamente em relação à primeira obviamente que se já é tão difícil e já existe tanta controvérsia em se chegar a um consenso quanto à definição do termo *outlier*, muito pior ainda será conseguir encontrar a respectiva distribuição de probabilidade pelo que esta hipótese, excepto em casos muito particulares, é praticamente inviável.

Em relação à segunda sugestão do Ripley, embora numa primeira análise a ideia pareça boa, colocam-se de imediato algumas dúvidas no que diz respeito à escolha do limiar de decisão. Em primeiro lugar, há que ter em conta que existe uma grande dependência em relação ao modelo e em relação ao número de variáveis consideradas; e isto, claro, partindo-se do princípio que se conseguem obter estimativas fiáveis das densidades. Para ilustrar o que se acaba de referir, considere-se o caso mais simples de uma distribuição normal estandardizada e veja-se, na Tabela 4.1, o que acontece aos valores da densidade nos pontos $\mathbf{x}$ para os quais $\mathbf{x}=0$ onde $f(\mathbf{x})=(2\pi)^{-p/2}$, e $\mathbf{x}=1$ onde $f(\mathbf{x})=(2\pi e)^{-p/2}$, em função do aumento do número de variáveis, p.

p	$f(0)$	$f(1)$
1	0.3981	0.2420
2	0.1592	0.0585
5	0.0101	0.0008
10	10^{-4}	6.8×10^{-7}

Tabela 4.1- Alguns valores para a densidade da normal estandardizada em função do número de variáveis.

Ressalta de imediato que os valores da densidade são pequenos mesmo para um número reduzido de variáveis o que permite deduzir que os valores fornecidos pela densidade não podem ser utilizados directamente. No entanto, para o caso particular da distribuição normal, esse problema pode ser resolvido recorrendo-se aos contornos de igual densidade. Como é de conhecimento geral, se $\mathbf{X}=(X_1,X_2,...,X_p)^T$ tem densidade dada por

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} e^{-(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})/2},$$

ou seja, se $\mathbf{X} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, então os contornos de igual densidade são elipsóides definidos por

$$(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) = c^2,$$

sendo os elipsóides centrados em $\boldsymbol{\mu}$ e com eixos $\pm c\sqrt{\lambda_i}e_i$, $i = 1, 2, ..., p$ onde $\boldsymbol{\lambda}, \boldsymbol{e}$ representam os valores e vectores-próprios de $\boldsymbol{\Sigma}$, respectivamente. Por outro lado, sabe-se que

$$(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) \sim \chi_p^2$$

pelo que a probabilidade de $(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) < \chi_p^2(1-\alpha)$ vale $1-\alpha$ (ver por exemplo, Johnson e Wichern, 1992, Cap. 4)[3].

Sendo assim, no contexto da distribuição normal, a ideia funciona embora não de uma forma directa. Porém, havendo necessidade de se proceder à estimação da densidade será que se consegue seguir a ideia sugerida por Ripley (1996)? Infelizmente isso não se verifica, como se verá com detalhe na Secção 4.3. De qualquer modo, considera-se ainda ter interesse reafirmar, que qualquer que seja a regra que se adopte, se existir um modelo para os dados, os pontos a rejeitar serão sempre os de menor densidade.

4.1.2. Breve resenha histórica

As técnicas inicialmente propostas para a detecção de *outliers* em amostras normais multivariadas resultam na sua maioria de generalizações directas das técnicas univariadas existentes (Beckman e Cook, 1983). Existe uma vasta literatura sobre a rejeição de *outliers* (de destacar Barnett e Lewis, 1994) tendo sido propostas uma infinidade de regras de rejeição. Muitos dos procedimentos que foram surgindo foram

[3] Repare-se que a expressão representada no primeiro membro da igualdade corresponde ao quadrado da tradicional distância de Mahalanobis entre a observação **x** e a média da população.

alvo de um grande impulso dado por Gnanadesikan e Kettenring (1972), baseados em parte numa extensão directa de uma das primeiras tentativas de identificação de *outliers* realizada por Wilks (1963). Gnanadesikan e Kettenring encorajaram a criação de vários testes e a procura de procedimentos genéricos capazes de lidar com a complexidade associada ao caso multivariado. Hampel (2000) "simplifica" o problema ao referir que o paradigma principal resume-se, por exemplo, a um teste do tipo H_0: *os dados são exactamente normais* contra H_1: *dois outliers à direita*; mas claro que esta hipótese é raramente satisfeita, os dados nunca são exactamente normais. Por outro lado, apesar da existência de bastantes métodos de detecção de *outliers* esporádicos em dados multivariados (Hawkins, 1980), o problema da detecção de nuvens de *outliers* apresenta-se bastante difícil (Rocke e Woodruff, 1996). Hoje em dia, a par do desenvolvimento de métodos eficazes de detecção de *outliers* estudam-se métodos robustos de estimação da localização e escala preferentemente invariantes para transformações afins, computacionalmente praticáveis e com elevado ponto de rotura; com frequência a detecção de *outliers* e a estimação robusta são discutidas em conjunto. Dos métodos para detecção de *outliers*, recorrendo-se ou não a estimação robusta, destacam-se Campbell (1978, 1982), Atkinson (1994), Barnett e Lewis (1994), Gnanadesikan e Kettenring (1972), Hawkins (1980), Maronna e Yohai (1995), Rocke e Woodruff (1996), Rousseeuw e van Zomeren (1990) e Kosinski (1999).

A literatura estatística dedicada à detecção de *outliers* é tão vasta, que os procedimentos podem mesmo ser subdivididos dando origem à seguinte classificação proposta por Gather e Becker (1997): métodos gráficos, procedimentos em bloco, testes consecutivos e métodos heurísticos, ou informais, (que podem ser quantitativos ou gráficos mas que não têm suporte em qualquer distribuição teórica) que em grande parte dos casos têm revelado um óptimo desempenho. Procedimentos de identificação de *outliers* construídos para a detecção simultânea num só passo com respeito a um modelo de referência, nomeadamente a distribuição normal, têm também sido sugeridos por vários autores, quer para a situação univariada quer para a multivariada (Rosado, 1984, Barnett e Lewis, 1994, Davies e Gather, 1993, Rousseeuw e van Zomeren, 1990).

Um método mais recente que merece especial atenção é o BACON (*Blocked Adaptative Computationally Efficient Outlier Nominators*) devido a Billor, Hadi e

Velleman (2000). Este consiste numa abordagem genérica baseada no método de Hadi (1992, 1994). São apresentadas duas versões equivariantes para transformações afins (com ponto de rotura grande ou moderado consoante a versão) tendo como principal vantagem a eficiência computacional e a boa *performance* em conjuntos de dados de grandes dimensões, chegando mesmo a igualar a *performance* dos estimadores mais populares (e provavelmente mais respeitados), MCD e MVE em todos os problemas teste publicados[4]. No entanto, tal como todos os outras propostas, peca pela dependência de aplicabilidade a dados elipticamente simétricos (ou aproximadamente).

4.2 Estimadores, distâncias e outros aspectos relevantes

O procedimento clássico para a identificação de *outliers* em dados multivariados é baseado no cálculo das distâncias de Mahalanobis (observações que apresentam distâncias "grandes" são consideradas *outliers*). Enquanto que é relativamente simples detectar um único *outlier* recorrendo a esta distância, este procedimento não é suficiente para múltiplos *outliers*. Esta e outras abordagens que apareceram anteriormente, podem falhar por dois motivos: o efeito *masking* e o efeito *swamping*. O primeiro corresponde à incapacidade de um teste identificar mesmo que um único *outlier*, na presença de outros valores suspeitos. A presença de outras observações mascara a detecção do impacto real de uma única observação; múltiplos *outliers* na mesma nuvem distorcem as estimativas quer pela atracção da média amostral quer por inflacionar a matriz de covariâncias amostral na sua direcção, levando à obtenção de valores menores para a distância de Mahalanobis. No segundo efeito, o *swamping*, as observações podem parecer *outliers* quando de facto não o são (ver Davies e Gather, 1993). Refere-se ao efeito que um agregado (nuvem) de *outliers* pode ter num conjunto de observações consistentes com a maioria; o agregado pode causar distorções na matriz de covariâncias pelo que grandes valores de distâncias podem traduzir-se em observações que na realidade não são *outliers*. Grosso modo, diz-se que o efeito de *swamping* ocorre quando, observações regulares são rotuladas como sendo *outliers*. Rocke e Woodruff (1996) dão especial relevância ao porquê da detecção de *outliers* multivariados ser tão difícil e ao porquê desta dificuldade aumentar com a dimensão dos dados.

[4] Nas secções seguintes estes estimadores serão abordados com mais detalhe.

Uma forma de contornar estes problemas consiste na utilização de estimadores robustos do vector de médias e da matriz de covariâncias. Este tema tem sido alvo de intensa investigação nos últimos anos e constitui um grande desafio tendo sido considerado um dos problemas mais difíceis da estatística robusta. Aliás, conforme já aqui foi referido, a existência de *outliers* nos dados é das razões mais óbvias para que necessitemos de algum tipo de procedimento robusto.

Do conjunto de estimadores disponíveis destacam-se o MVE (*Minimum Volume Ellipsoid*) e o MCD (*Minimum Covariance Determinant*) de Rousseeuw (Rousseeuw, 1985; Rousseeuw e Leroy, 1987) e versões reponderadas dos mesmos, mais eficientes (ver Croux e Haesbroeck, 1999). No caso do MCD o objectivo é o de encontrar h pontos que fazem com que a matriz de covariâncias tenha determinante mínimo enquanto que no MVE se procura o elipsóide que contém os h pontos que perfazem volume mínimo. O MCD é geralmente mais utilizado porque supera o MVE tanto em termos de eficiência estatística como em velocidade computacional (Rousseuw e Van Driessen, 1999). Destaque-se ainda que o estimador MVE não tem distribuição assintótica normal e não é $\sqrt{n}$ consistente (Davies, 1992). Estes estimadores são resistentes aos *outliers* porque estes não se envolvem nos cálculos efectuados para a estimação da localização e da escala: as distâncias robustas baseadas no MCD são mais precisas quando há contaminação do que as baseadas no MVE permitindo uma melhor detecção dos *outliers*.

Tanto o estimador MCD como o estimador MVE possuem elevado ponto de rotura (arbitrariamente próximo de 50% para qualquer n), são equivariantes, mas pouco eficientes próximo do modelo multinormal colocando até há bem pouco tempo, grandes problemas de cálculo (ver por exemplo Pires, 1995). No entanto, este problema foi ultrapassado em 1999 com o aparecimento de um novo algoritmo (Rousseeuw e van Driessen, 1999) que tornou o MCD disponível originando o aparecimento de uma rotina que, por exemplo, constitui parte integrante do *software* S-Plus. O novo estimador passou a chamar-se "fast MCD" e tomou a designação dada pela sigla FMCD ou RMCD, no caso da versão reponderada. Muitos outros autores, como por exemplo Atkinson (1994), e Rocke e Woodruff (1996) utilizaram também estes estimadores (MCD e MVE) nos seus métodos de detecção de *outliers*.

Importa também fazer menção a outros estimadores tais como os estimadores-S (Davies, 1987), os estimadores-M de Maronna (1976) e os outros estimadores baseados em projecções com especial destaque para o estimador Stahel-Donoho proposto por Stahel (1981) e Donoho (1982) e mais tarde estudado por Maronna e Yohai (1995). Ao contrário dos estimadores-M, todos os outros possuem elevado ponto de rotura. Hampel (2002), referindo-se a estes estimadores, tece alguns comentários interessantes acerca das matrizes de covariâncias robustas.

Recentemente, Maronna e Zamar (2002) propõem um novo estimador baseado numa versão modificada do estimador robusto de Gnanadesikan e Kettering que designam por OGK (*orthogonalised Gnanadesikan-Kettnering estimate*). Como estes autores referem, embora tenham aparecido muitas abordagens para lidar com o problema da sensibilidade da matriz de covariâncias à presença dos *outliers* (destacando-se o MCD), a obtenção destes estimadores requer tempos de processamento substanciais, mesmo no caso do FMCD, principalmente para amostras de grande dimensão. Neste artigo dá-se relevo a alguns obstáculos associados à consistência e ao enviesamento do estimador MCD sendo apresentado um método genérico para obtenção de uma matriz de dispersão robusta, definida positiva e aproximadamente equivariante para transformações afins. Este estimador tem também a vantagem de requerer tempos de processamento inferiores a estimadores robustos recentes (tais como o FMCD e o estimador de Stahel-Donoho). Com base num conjunto de simulações com dados normais contaminados mostra-se que o estimador OGK é quase tão bom como o estimador de Stahel-Donoho e claramente melhor que o FMCD sendo mais rápido que ambos especialmente para grandes dimensões. Quando aplicado a dados reais também mostrou ser tão bom, ou melhor, que o estimador de Stahel-Donoho e o FMCD.

Pelas razões apontadas decidiu-se incluir neste trabalho o estimador OGK já que poderá vir a revelar-se uma proposta competitiva com o MCD pelo que se julga conveniente dedicar um pouco mais de atenção à descrição das ideias básicas do mesmo.

De grosso modo, o estimador de localização-escala OGK, é baseado numa versão modificada do estimador de Gnanadesikan-Kettenring, através da seguinte relação entre variâncias e covariâncias:

$$\text{Cov(X,Y)}= \frac{1}{4}(\sigma(X+Y)^2 - \sigma(X-Y)^2)$$

onde σ representa o desvio-padrão usual e *X*, *Y* um par de variáveis aleatórias. Estes autores propõem a definição de uma "matriz de covariâncias robusta" através da utilização de uma escala robusta (como por exemplo o "trimmed standard deviation"). Contudo, com base nesta relação, a matriz de localização-dispersão multivariada resultante não é equivariante para transformações afins e não se pode garantir que seja definida positiva.

Recorde-se que sendo V a matriz de covariâncias do vector **x** *p*-dimensional e σ o desvio-padrão, então $\sigma(a^{\mathrm{T}}\mathbf{x})^2=a^{\mathrm{T}}\mathrm{V}a, \forall_{a\in\mathbb{R}^p}$. Maronna e Zamar (2002) propõem uma modificação, forçando esta igualdade para um conjunto de "direcções principais" e obrigando a que a matriz se torne definida positiva. Esta modificação assenta no facto de que os valores próprios da matriz de covariâncias correspondem às variâncias ao longo das direcções dadas pelos respectivos vectores próprios. Para tal definem uma matriz de dispersão V(**X**) e um vector de localização t(**X**) que podem e devem, ser iterados e que são obtidos à custa de quatro passos sendo o primeiro o que torna o estimador equivariante para a escala e os restantes uma espécie de "componentes principais" com a substituição dos valores próprios (que podem ser negativos) por "variâncias robustas" das correspondentes direcções, conforme referem estes mesmos autores. O estimador pode também ser melhorado depois de um passo de reponderação. Sendo assim, o que é proposto neste artigo é o seguinte. Tome-se o quadrado da distância de Mahalanobis usual, $d_i = d(x_i) = (x_i - \mathrm{t})^{\mathrm{T}}\mathrm{V}^{-1}(x_i - \mathrm{t})$, com matriz de dispersão V=V(**X**) e vector de localização t=t(**X**). Seja *W* a função de pesos e t_W, V_W a matriz de dispersão e o vector de localização pesados correspondentes, onde cada x_i tem peso $w_i=W(d_i)$. Considere-se então a função de pesos mais simples- rejeição abrupta (*hard rejection*)- com $W(d)=I(d\leq d_0)$, onde *I*(.) é uma função indicatriz (unitária se $d\leq d_0$) e tome-se $d_0 = \dfrac{\chi^2_p(\beta)\,\mathrm{med}(d_1,...,d_n)}{\chi^2_p(0.05)}$ onde $\chi^2_p(\beta)$ é o β- quantíl de uma distribuição Qui-quadrado com *p* graus de liberdade e med a notação utilizada para referir a mediana.

Uma vez que é conveniente a utilização de estimadores robustos e eficientes para a escala e localização, Maronna e Zamar (2002) sugerem para σ a "τ-*scale*" de Yohai e Zamar (1988)- que é um desvio *standard* truncado – e para estimador de μ uma versão "pesada" da média. Para tal definam-se as seguintes funções:

$$W_c(x) = \left(1 - \left(\frac{x}{c}\right)^2\right)^2 I(|x| \le c), \quad \rho_c(x) = \min(x^2, c^2).$$

Seja $\mathrm{X} = \{x_1, ..., x_n\}$ uma amostra univariada e faça-se,

$$\sigma_0 = \mathrm{MAD}(X) = \mathrm{med}(|X - \mathrm{med}(X)|, \quad w_i = W_{c_1}\left(\frac{x_i - \mathrm{med}(X)}{\sigma_0}\right).$$

Então os estimadores de localização e dispersão são definidos como

$$\mu(\mathrm{X}) = \frac{\sum_i x_i w_i}{\sum_i w_i}, \quad \sigma^2(\mathrm{X}) = \frac{\sigma_0^2}{n}\sum_i \rho_{c_2}\left(\frac{x_i - \mu(\mathrm{X})}{\sigma_0}\right).$$

Para combinar robustez e eficiência Maronna e Zamar (2002) sugerem que se tome c_1=4.5 e c_2=3 que permite, tanto para dados normais como de Cauchy, uma eficiência de aproximadamente 80% para localização e dispersão univariada. O programa que permite o cálculo deste estimador encontra-se no Apêndice C (este programa é uma tradução para a linguagem S-Plus da versão original escrita para a linguagem GAUSS, gentilmente cedida por Maronna).

A 12 de Janeiro de 2004 Pagnotta deu um seminário no Instituto Superior Técnico de Lisboa, onde abordou o tema " *Approximation algorithms for the estimate of the MCD and a new proposal*". A sua intervenção veio reforçar aquilo que já se suspeitava pelo que se tinha observado nalgumas situações. O estimador FMCD funciona muito bem quando a configuração dos dados contém nuvens de *outliers* com dispersão superior à das chamadas "boas" observações[5], caso contrário apresenta algumas desvantagens nomeadamente quando o número de variáveis aumenta e quando também

[5] Mais atenção às chamadas "boas" observações irá ser dada na Secção 4.4.

aumenta o número de *outliers* que se concentram[6] em determinada localização. Recentemente Pagnotta encontra-se a desenvolver uma versão modificada do FMCD a que chama CC-MCD e que se espera que introduza alguns melhoramentos nas propriedades do estimador bem como na diminuição do número de subconjuntos amostrais. Pagnotta (2003) esboça o algoritmo associado ao FMCD e analisa as respectivas fontes de falha apresentando a sua proposta alternativa CC-MCD.

Para terminar chama-se a atenção para um outro aspecto de importância relevante nos procedimentos de detecção de *outliers*: os métodos devem ser equivariantes para transformações afins, ou seja, as alterações de escala, localização e orientação não devem alterar o comportamento destes métodos, podendo ser ignoradas. É óbvio que esta propriedade é desejável tanto para o modelo como para as medidas de discrepância.

Note-se que a distância de Mahalanobis é um dos poucos critérios de identificação de *outliers* invariante para transformações afins. Por essa razão não são aqui referidas e não serão experimentadas (no estudo de simulação apresentado no Capítulo 5 e na aplicação a conjuntos de dados reais no Capítulo 6) outras propostas de distâncias, como por exemplo a distância às medianas utilizada por Billor, Hadi e Velleman (2000, pág. 285).

4.3 Motivação do método proposto

Relembre-se que nos critérios de decisão apresentados no Capítulo 3 se assumiram conhecidas as densidades condicionais às classes, ou equivalentemente, as probabilidades *a posteriori*. Nos exemplos apresentados nesse mesmo Capítulo, procedeu-se à classificação trabalhando-se com populações conhecidas (distribuições normais com matrizes de covariâncias idênticas, distribuições normais com matrizes de covariâncias distintas e distribuições aproximadamente normais dos *scores* discriminantes sob a hipótese de linearidade do logaritmo do quociente das densidades). No entanto, tal como é bem sabido, mesmo estas últimas situações são um pouco

[6] Em geral consta da literatura que o problema da detecção de *outliers* se agrava quando os *outliers* se encontram concentrados; *outliers* dispersos são relativamente fáceis de detectar (ver por exemplo Rocke e Woodruff, 1996).

irrealistas. Na melhor das hipóteses, o que geralmente se tem num problema real é apenas uma amostra de treino.

Por outro lado, como já foi referido neste mesmo capítulo, mesmo o conhecimento das estimativas das densidades, ou das probabilidades *a posteriori*, por si só não é suficiente para a classificação de objectos de origem desconhecida incluindo a detecção de *outliers*.

Mais ainda, mesmo que antes de se proceder à classificação, se tente utilizar um procedimento de detecção de *outliers*, os métodos existentes não se têm mostrado totalmente eficientes na detecção dos mesmos em determinadas situações. Os procedimentos clássicos sofrem, quanto mais não seja, do efeito *masking* e os estimadores robustos, para obtenção da tradicional distância de Mahalanobis, não funcionam bem quando a maior parte dos dados não tem uma distribuição elipticamente simétrica.

No Capítulo 3 propôs-se um método que permite classificar as observações numa de c+1 classes distintas, G_1, G_2, ..., G_c, I. A aplicação desse procedimento pode ser realizada ou na inexistência de *outliers* ou tendo em conta a classificação prévia dos *outliers* numa classe de rejeição. No que diz respeito à definição dessa tal classe, já designada anteriormente por O, embora o conhecimento das densidades condicionais às classes ou das correspondentes probabilidades *a posteriori* seja insuficiente, a sua estimação poderá e deverá ser utilizada, em certo sentido, desde que se consiga conciliar com outro tipo de procedimento complementar que resulte num método eficaz para detecção dos *outliers*. E o objectivo principal é que esse método funcione não só nas situações usuais mas que consiga ir mais além, permitindo a classificação em contextos não elipticamente simétricos. É nesta linha que se irá trabalhar nas próximas secções.

Para clarificar algumas das ideias já mencionadas e alguns dos problemas que possam advir da utilização dos métodos de detecção de *outliers* existentes, começa-se por considerar dois exemplos simples onde a detecção falha no cálculo das distâncias de Mahalanobis (no Exemplo 4.1 com a utilização de estimadores clássicos e no Exemplo 4.2 até mesmo com a utilização de estimadores robustos). Estes e outros exemplos servem de motivação à construção de um método (ou regra) que funcione para além das

situações "tradicionais". No entanto, julga-se importante que fique também claro que nem sempre isto acontece. Como já se referiu e se volta a reforçar, com a apresentação do exemplo que se segue, há situações onde basta a utilização de métodos robustos para haver uma melhoria significativa no número de *outliers* detectados.

Exemplo 4.1 Considere-se uma simulação de 150 observações com distribuição $t_{2,3}(\boldsymbol{\mu},\boldsymbol{\Sigma})$, com $\boldsymbol{\mu}=(0,12)^T$, $\boldsymbol{\Sigma}$=diag(1,0.3) e 20 observações *outliers* $t_{2,3}((-2,6)^T,0.01\mathbf{I})$.

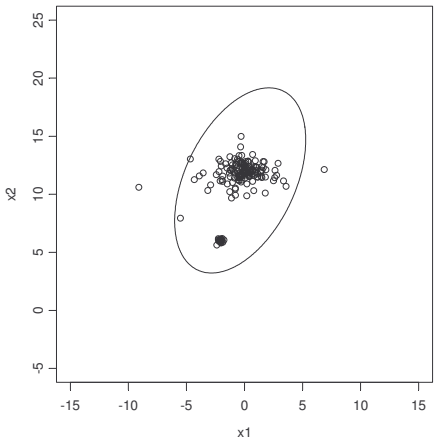


Figura 4.3- Dados do Exemplo 4.1 e elipse a 90% de confiança com limite de detecção assintótico (Qui-quadrado) e distâncias baseadas nos valores amostrais do vector de médias e da matriz de covariâncias clássica.

A análise da Figura 4.3 permite concluir que neste exemplo, a utilização de estimadores clássicos não é suficiente para a detecção dos *outliers*. No entanto, a Figura 4.5 demonstra como a utilização de estimadores robustos (neste caso o RMCD com ponto de rotura aproximado de 25%) já é suficiente para a detecção dos mesmos. Porém, esta detecção não é totalmente eficiente. Conforme também se pode verificar, há valores que saem fora do contorno e que portanto seriam classificados como *outliers* mesmo não tendo sido fabricados para o serem. Será que então não poderá ser criado algum processo de melhorar a detecção numa situação como esta ou em situações ainda mais complexas? É o que se irá ver mais à frente.

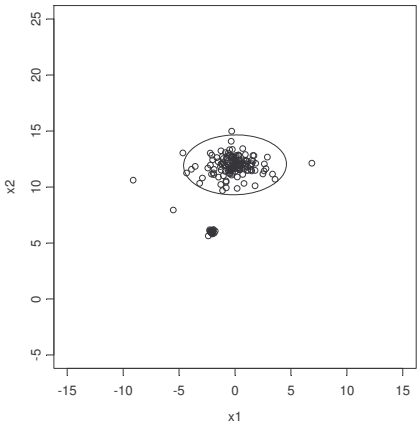


Figura 4.4- Dados do Exemplo 4.1 e elipse a 90% de confiança com limite de detecção assintótico determinado através de uma simulação com 10000 conjuntos de dados normais e distâncias baseadas no vector de médias e matriz de covariâncias fornecidas pelo estimador RMCD.

Note-se que o contorno desenhado na Figura 4.4 poderia também ter sido calculado com base na distribuição Qui-quadrado. Optou-se, no entanto, pelo procedimento descrito onde a obtenção do limite de detecção é baseada num conjunto de simulações (embora neste momento aquilo que se diz ainda seja pouco esclarecedor) pois como se justificará na Secção 4.4 (e posta em prática no Cap. 5), ele conduz a melhores resultados. Se se optasse por definir o contorno pelo procedimento usual não haveria qualquer melhoria nos resultados. Como referem Rousseeuw e van Zomeren (1991), a utilização de limites baseados nos quantis do Qui-quadrado leva com frequência à identificação de demasiados *outliers*. Pela razões apontadas irá ser utilizado o mesmo procedimento no Exemplo 4.2 que se passa a apresentar.

Exemplo 4.2. Considere-se agora uma outra situação onde se geraram 150 observações bivariadas uniformemente distribuídas na intersecção do exterior de uma circunferência de raio 10 com o interior de uma circunferência de raio 12 bem como um conjunto de 20 observações *outliers* $N_2((25,0)^T,0.1\mathbf{I})$- esta configuração passará a ser apelidada de configuração DONUT (com *outliers*).

Na Figura 4.5 encontra-se representado este conjunto de dados e a elipse a 90% de confiança que permite a detecção dos *outliers* com base na utilização da distância de Mahalanobis com estimadores clássicos.

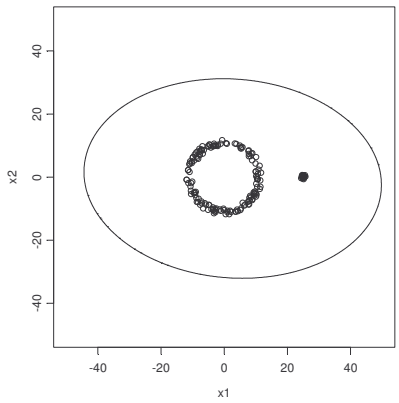


Figura 4.5- Dados do Exemplo 4.2 e elipse a 90% de confiança com limite de detecção assintótico e distâncias baseadas nos valores amostrais do vector de médias e da matriz de covariâncias clássica.

Pela análise da Figura 4.5 é fácil concluir que por este meio não se consegue detectar nenhum dos 20 *outliers*. Mas será que a utilização da distância de Mahalanobis com estimadores robustos, neste caso o RMCD com um ponto de rotura aproximado de 25%, a situação se altera? Observe-se o que acontece na Figura 4.6.

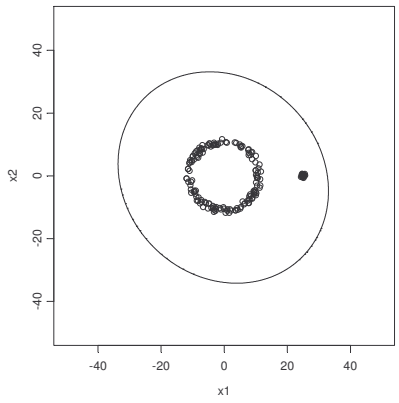


Figura 4.6 - Dados do Exemplo 4.2 e elipse a 90% de confiança com limite de detecção assintótico determinado através de uma simulação com 10000 conjuntos de dados normais e distâncias baseadas no vector de médias e matriz de covariâncias fornecidas pelo estimador RMCD.

Os resultados obtidos correspondem ao esperado. Conforme se pretendia aqui demonstrar, há determinadas situações, tal como a apresentada no Exemplo 4.2, onde nem mesmo a utilização de estimadores robustos, por si só, conseguem uma detecção eficaz dos *outliers*.

Resumindo, por aquilo que foi visto na Secção 4.1.1 e nos Exemplos 4.1 e 4.2, pode-se concluir que se a distribuição for normal é sempre possível encontrar um limite de detecção para os *outliers*. Caso contrário, e mesmo no caso em que a distribuição das observações é conhecida, pelo facto de não existir nenhum resultado equivalente ao da normal, tanto podem existir situações onde é possível encontrar esse limite com base na utilização de estimadores clássicos e/ou robustos como poderão existir outras situações onde nem mesmo o recurso a estimadores robustos será viável para a resolução do problema da obtenção de um limite de detecção de *outliers*.

Depois de feitas algumas reflexões básicas necessárias e de introduzidos os conceitos mais relevantes chegou então o momento de fazer menção aos aspectos que servirão de fundamento à introdução da regra de detecção de *outliers* a ser formalmente apresentada na Secção 4.5.

Ainda numa tentativa de seguir a linha apresentada por Ripley (1996), referida na Secção 4.1.1, coloca-se agora a questão da estimação das densidades e das possíveis consequências que possam derivar desta necessidade. Como já foi abordado com algum detalhe no Capítulo 2, a estimação poderá ser feita por exemplo, com recurso a um método não paramétrico, ou à utilização de modelo mistura embora este último tenha como condicionante problemas inerentes à obtenção de máximos locais devido ao grande número de parâmetros a estimar. McLachlan e Basford (1988) e mais tarde McLachlan e Peel (2000), apresentam um método de detecção de *outliers* para misturas de densidades multivariadas que consiste em comparar uma nova observação com cada uma das componentes estimadas da mistura através da utilização de uma determinada medida, tal como a distância de Mahalanobis, com respeito à média de cada componente; se esta medida apresentar valores grandes para todas as componentes da mistura então a nova observação pode ser considerada um *outlier.* No entanto, conforme alertam Wang *et al*. (1997), esta abordagem tem pouco controlo sob o nível de significância global, excepto em casos muito particulares. Como alternativa Wang *et al*. (1997) propõem uma modificação simples do teste da razão de verosimilhanças que compara a hipótese da nova observação ser proveniente do modelo mistura postulado contra a hipótese alternativa de não pertencer, recorrendo a técnicas *bootstrap* para obtenção da distribuição da estatística de teste. Tanto Wang *et al*.(1997) como Sain *et al*.(1999) procederam a este estudo com o objectivo de determinar se uma observação

sísmica se tratava de uma explosão nuclear, de um terramoto ou de um outro qualquer distúrbio sísmico.

Regressando aos métodos não paramétricos e tomando por exemplo o método de Kernel o que se verifica e que se irá demonstrar de seguida, é que emerge a impossibilidade de se obterem as constantes que definem os contornos de detecção de *outliers* ao contrário do que acontecia no contexto da distribuição normal.

Para efeito de ilustração considere-se o exemplo seguinte (Figura 4.8), baseado na configuração DONUT (já apresentada no Exemplo 4.2), mas agora com um conjunto de 1200 observações (com o intuito de se obter estimativas bem fiáveis)[7].

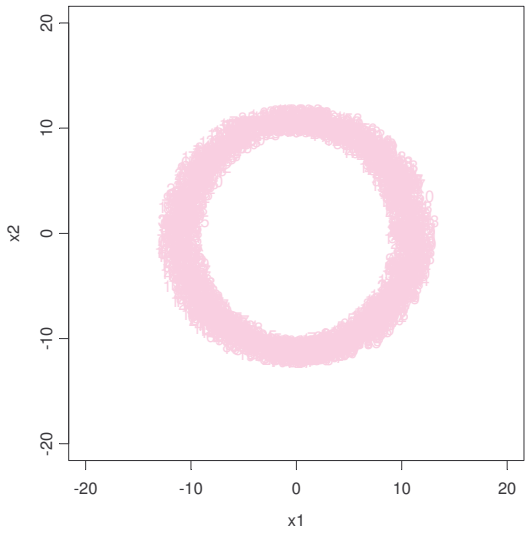


Figura 4.7- Dados da configuração DONUT com 1200 observações e sem *outliers*.

Para estimação da densidade que melhor se ajusta às observações representadas na Figura 4.7 utilizou-se o estimador de Kernel com núcleo normal ou *Gaussiano* e h=2 de acordo com:

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^2}\sum_{i=1}^{n} \mathrm{K}_p\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right),$$

[7] Mais á frente quando se proceder ao estudo de simulação, esta configuração irá ainda ser alvo de um estudo mais detalhado.

onde

$$\mathrm{K}_p(\mathbf{x}) = (2\pi)^{-1} \exp(-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{x}) .$$

Note-se que embora *h* seja bidimensional, para permitir que o parâmetro de alisamento seja diferente consoante a direcção, é usual na prática, admitir-se que ambos os valores são iguais (ou que todos os valores são iguais no caso em que o número de variáveis é superior a dois), isto é razoável desde que as escalas das variáveis sejam semelhantes, como é o caso.

O valor de *h* foi obtido de acordo com o critério de minimização do erro quadrático médio integrado, MISE (ver por exemplo Soares, 1997, Cap. 2, pág. 39), no qual

$$h_{opt} = 1.06\sigma n^{-1/5} .$$

Experimentou-se ainda diminuir o valor de *h*, seja *h*=1, numa tentativa de melhorar a estimação. Os dados (representados a rosa) bem como os contornos de igual densidade para a estimativa de kernel (representados a verde) com *h*=2 e *h*=1 são apresentados nas Figuras 4.8 e 4.9, respectivamente. Quando se passa de *h*=2 para *h*=1 verifica-se, conforme era de esperar, uma melhoria na proximidade mas à custa de um aumento da irregularidade (ver Figura 4.9). É sabido que a proximidade do estimador de *f* à verdadeira função de densidade depende em grande parte da escolha do parâmetro de alisamento, *h*, utilizado devendo este ser escolhido por forma a se obter um alisamento que seja um compromisso aceitável entre o enviesamento que pode ser tolerado e a flutuação aleatória (embora dependendo também do critério escolhido e do núcleo utilizado). No entanto, no contexto em que este estimador é utilizado esta questão é irrelevante pelo que não se entra em mais detalhes.

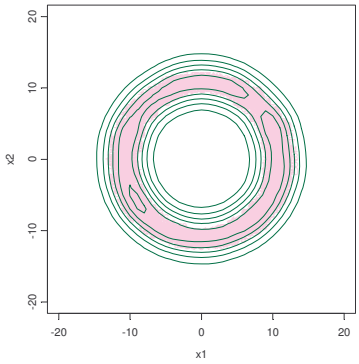


Figura 4.8- Contornos de igual densidade para a estimativa de Kernel com *h*=2.

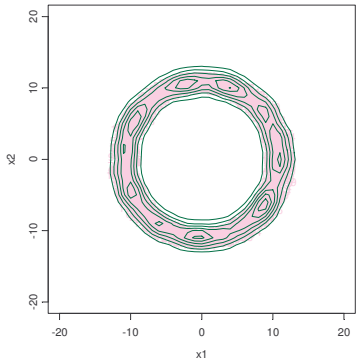


Figura 4.9- Contornos de igual densidade para a estimativa de Kernel com *h*=1.

Como se poderá então estabelecer um contorno que funcione como limite de detecção de *outliers*? Por analogia com a distribuição normal estandardizada (tal como foi feito anteriormente), e tomando $\alpha = 0.1$ e *n*=1200, o que é equivalente a tomar $a_N = 0.9999122$, obtém-se $\chi_0^2 = 18.6809$ pelo que o a densidade deverá ser considerada pequena quando o seu valor for inferior a 1.4×10^{-5} dando lugar aos contornos (a laranja) representados na Figura 4.10 a par da estimativa de kernel (para o caso em que se tomou *h*=2).

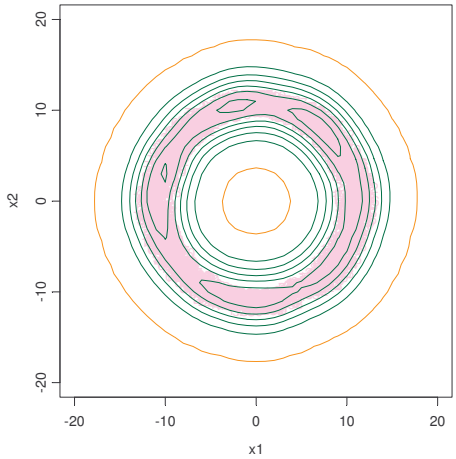


Figura 4.10- Contornos de detecção baseados num valor baixo da estimativa da função densidade ("valor baixo" obtido por analogia com a distribuição normal).

Para dados sem *outliers* parece que tudo funciona bem. No entanto, verifica-se de seguida, que mesmo que se conseguisse, ainda que duma forma heurística, estabelecer algum tipo de contorno, provavelmente ele não funcionaria como limite de detecção de *outliers*. Isto porque, no passo seguinte utilizou-se o mesmo limiar que teria sido obtido no caso particular em que os dados têm distribuição normal numa tentativa de se encontrar uma solução e verificou-se que embora se tenha conseguido encontrar um determinado contorno, não é possível que com esse mesmo contorno, e no caso em que tem de se recorrer à estimação da densidade (neste caso através do método de kernel) esse procedimento seja eficaz na detecção de *outliers*- veja-se o que acontece, na Figura 4.12, quando se adicionam dois novos pontos localizados em $(x_1, x_2) = (15,15)$. Mais uma vez representam-se, tal como anteriormente, os contornos de igual densidade para a estimativa de Kernel (a verde) e os contornos baseados na distribuição normal (a laranja). Na Figura 4.11 apresenta-se o gráfico tridimensional correspondente.

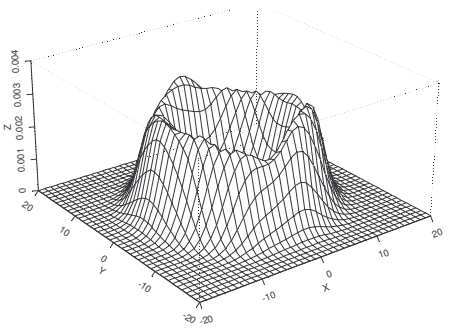


Figura 4.11- Gráfico tridimensional para a configuração DONUT com 2 *outliers*.

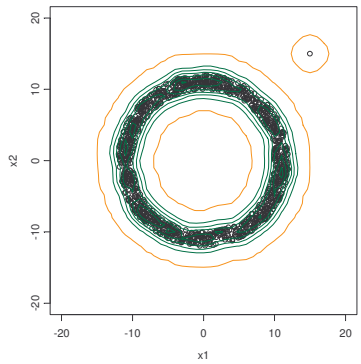


Figura 4.12- Contornos de detecção baseados num "valor baixo" da estimativa da função densidade (por analogia com a distribuição normal) para a configuração DONUT com 2 *outliers*.

Com os conhecimentos adquiridos com base nos resultados obtidos e enfrentando-se a dificuldade da obtenção de tais limites, o que se tentou fazer de seguida foi algo de intermédio entre a estimação pela distribuição normal e o método não paramétrico do núcleo com distribuição normal. Surgiu então a ideia de se pensar nos métodos de

detecção de *outliers* aplicados a observações agrupadas em nuvens. Em primeiro lugar esta divisão poderá ajudar na compreensão da estrutura dos dados e na identificação dos *outliers*. Por outro lado, como se irá ver, acresce a vantagem de por esta via ser possível a obtenção dos limites de detecção. Ao segmentar-se as n observações em k nuvens consegue-se criar elipses que impõe determinados limites de detecção podendo ainda, este mesmo procedimento, ser visto como uma tentativa de estimação de densidades[8] como se mostra de seguida. Quanto aos limites de detecção que irão ser calculados o problema fica resolvido já que os mesmos podem ser obtidos à custa da distribuição normal isto porque, embora os dados não sejam necessariamente normais ao se proceder à divisão em nuvens no fundo acaba-se por ficar mais próximo de uma mistura de normais. Esta questão será retomada mais à frente.

Sendo assim, seguindo o caminho do encontrar um método eficaz de detecção de *outliers* mostra-se agora na Figura 4.13 os contornos obtidos com base numa partição do conjunto das observações em 4 nuvens após a qual se procedeu à estimação da densidade pelo método de kernel- designe-se por **método de kernel adaptado**- de acordo com a seguinte expressão:

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{k} \frac{n_i}{n} f_N(\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Sigma}}_i) ,$$

ou

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{k} \frac{n_i}{n} (2p)^{-1} \left| \hat{\boldsymbol{\Sigma}}_i \right|^{-1/2} \exp\left[\frac{-(\mathbf{x} - \hat{\boldsymbol{\mu}}_i)^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_i (\mathbf{x} - \hat{\boldsymbol{\mu}}_i)}{2} \right],$$

onde k representa o número de nuvens e n_i a dimensão da i-ésima nuvem, i=1,2,...,k. Neste exemplo, uma vez que o objectivo se concentra apenas na ilustração das possíveis vantagens da utilização de nuvens, por uma questão de simplificação, utilizaram-se nuvens de igual dimensão ou seja, n_i / n= constante, i=1,2,...,k .

[8] Daí a afirmação de se estar a fazer algo de intermédio entre os dois procedimentos descritos.

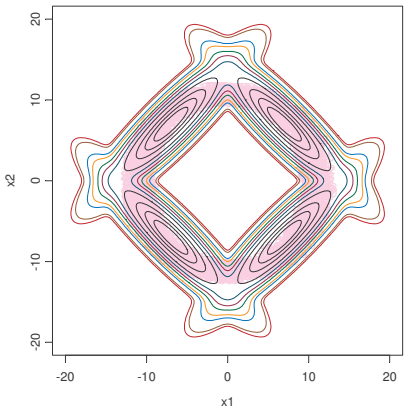


Figura 4.13- Contornos de igual densidade obtidos pelo método de Kernel adaptado com 4 nuvens.

Embora sejam visíveis alguns "bicos" isto é apenas uma limitação do pequeno número de nuvens consideradas conforme se pode ver na Figura 4.14 quando se utiliza o mesmo procedimento com 8 nuvens.

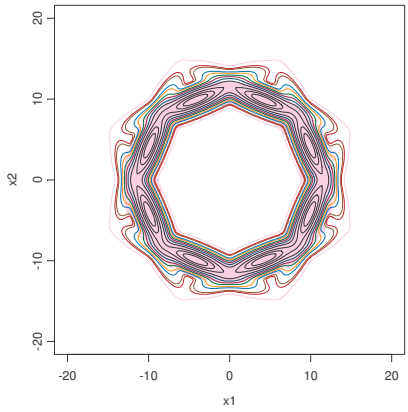


Figura 4.14- Contornos de igual densidade obtidos pelo método de Kernel adaptado com 8 nuvens.

Considera-se ainda ter interesse reflectir um pouco sobre este aspecto que o exemplo apresentado suscita. O que fazer com estes "bicos"? será que eles afectam o desempenho do método? Poderia pensar-se nalgum tipo de regularização no meio da distribuição mas para efeitos de discriminação e detecção de *outliers* é óbvio que os "bicos interiores" não afectam a decisão, basta que nas caudas a aproximação seja boa o que reforça a necessidade da utilização de um número relativamente "grande" de nuvens

(obviamente, desde que haja observações em número suficiente). Relembre-se que quaisquer que sejam as observações a rejeitar, não há dúvidas de que serão aquelas onde a densidade assume valores pequenos pelo que sendo respeitada esta premissa, esta aproximação poderá não ser tão boa em termos de estimação da densidade, mas poderá funcionar na definição dos limites de detecção. Para clarificar melhor a afirmação considere-se ainda um exemplo simples de uma distribuição uniforme univariada, U[0,1]. Na Figura 4.15 apresenta-se a função densidade bem como as estimativas dessa mesma distribuição com duas, quatro e dez nuvens. Qualquer que seja o número de nuvens escolhido, é evidente a irregularidade ("bicos") mas constata-se que conforme o número de nuvens aumenta, a aproximação nas caudas vai melhorando.

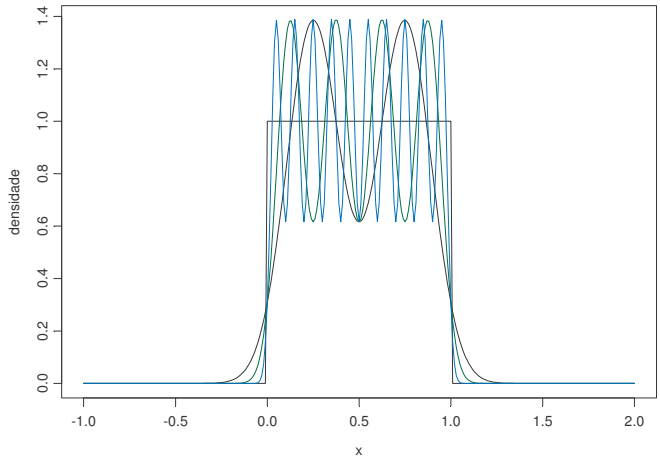


Figura 4.15- Estimativas da densidade obtidas pelo método de Kernel adaptado com 2, 4 e 10 nuvens para dados de uma distribuição uniforme U[0,1].

Em suma, como se irá ver de seguida, o método proposto de rejeição de *outliers* equivale a proceder à estimação da densidade em torno dos pontos com a vantagem de se poder calcular os contornos que permitem a detecção dos *outliers*.

4.4 Formulação do problema de identificação de *outliers*

Como já aqui foi referido, a detecção de *outliers* tem sido largamente investigada na literatura sendo o termo *outlier* bastante flexível. Uma possível definição formal é apresentada por Davies e Gather (1993). A sua definição consiste em considerar que os *outliers* têm uma distribuição diferente das restantes observações definindo-os em

termos da sua posição relativa ao modelo das chamadas "boas observações"; algo semelhante ao que apresenta Kosinski (1999) ao basear-se no pressuposto dos dados estarem divididos em duas classes, aquela a que apelida de "boas observações" (que corresponde à maioria dos casos) e reflecte a distribuição da população e uma segunda classe que contém os *outliers* (se estes existirem). Segundo este autor, o objectivo de qualquer método de detecção de *outliers* é o de encontrar a verdadeira partição e por consequência, separar os *outliers* das observações consideradas "boas". Esta abordagem é também seguida por outros autores como por exemplo, Rocke e Woodruff (1996).

Segundo Davies e Gather (1993) o problema da identificação de múltiplos *outliers* corresponde ao problema da identificação das observações que caiem na chamada região de *outliers* out(α, μ, σ^2) podendo ser formulado da seguinte forma: dada uma variável aleatória X, univariada, com distribuição normal,

$$P(X \in \mathbb{R} \setminus \text{out}(\alpha, \mu, \sigma^2))=1-\alpha$$

para um dado $\alpha \in]0,1[$. No caso em que μ e σ^2 são conhecidos a região de *outliers* pode ser definida da seguinte forma:

$$\text{out}(\alpha, \mu, \sigma^2)=\left\{ x : |x-\mu| > z_{1-\frac{\alpha}{2}}\, \sigma \right\}$$

onde z_q corresponde ao q-ésimo quantil de uma distribuição $N(0,1)$.

Supondo agora uma amostra de dimensão n, $\mathbf{X}_n=(X_1, ..., X_n)$, com distribuição normal e assumindo-se μ e σ^2 conhecidos, a região de *outliers* pode ser obtida por forma a que

$$P(X_i \in \mathbb{R} \setminus \text{out}(\alpha_n, \mu, \sigma^2), \forall i=1,2,\ldots,n)=1-\alpha$$

para um dado α, o que é equivalente a tomar $\alpha_n=1-(1-\alpha)^{1/n}$ passando o problema da identificação de *outliers* a ser o da especificação das observações que caiem nessa região particular. Note-se que, como é referido pelos mesmos autores, embora se use a distribuição normal como distribuição de referência, pode ser tomada qualquer outra distribuição simétrica e unimodal sendo que a essência dos resultados não se altera.

Em Becker e Gather (1999) este conceito é generalizado ao caso multivariado e é dada uma justificação formal para a utilização de estimadores robustos mostrando que no caso em que estes possuem ponto de rotura elevado, permitem uma melhor prevenção do efeito de *masking* comparativamente aos estimadores clássicos. Para um dado $\alpha \in \ ]0,1[$ e com respeito a uma $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ uma observação $\mathbf{x} \in \mathbb{R}^p$ é considerada *outlier* se cai na região de *outliers* out($\alpha, \boldsymbol{\mu}, \boldsymbol{\Sigma}$) onde

$$\text{out}(\alpha, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \left\{\mathbf{x} \in \mathbb{R}^p : (\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu}) > \chi^2_{p;1-a}\right\}.$$

Tal como no caso univariado, o tamanho da região de *outliers* pode ser ajustado consoante a dimensão da amostra. Para uma amostra de dimensão *n* pode definir-se out($\alpha_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}$) por forma a que:

$P_{N(\boldsymbol{\mu},\boldsymbol{\Sigma})}$ (nenhuma observação de uma amostra "limpa" de dimensão $n \in$ out($\alpha_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}$))=1-$\alpha$

pelo que,

$$\text{out}(\alpha_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \left\{\mathbf{x} \in \mathbb{R}^p : (\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu}) > \chi^2_{p;1-\alpha_n}\right\},$$

para um dado $\alpha \in \ ]0,1[$ onde, tal como anteriormente, α_n=1-(1-α)$^{1/n}$.

Sendo os parâmetros $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$ desconhecidos podem tomar-se os correspondentes estimadores equivariantes para transformações afins, $\hat{\boldsymbol{\mu}}$ e $\hat{\boldsymbol{\Sigma}}$. Para encontrar os múltiplos *outliers* numa dada amostra $\mathbf{x}_n$=($\mathbf{x}_1$,$\mathbf{x}_2$,...,$\mathbf{x}_n$) de dimensão *n*, $\mathbf{x}_i \in \mathbb{R}^p$, pode estimar-se out($\alpha_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}$) pela região correspondente *OR* ($\mathbf{x}_n$, α_n) onde,

$$OR\ (\mathbf{x}_n, \alpha_n) = \left\{\mathbf{x} \in \mathbb{R}^p : (\mathbf{x}-\hat{\boldsymbol{\mu}})^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}^{-1} (\mathbf{x}-\hat{\boldsymbol{\mu}}) > \mathrm{c}(p, n, \alpha_n)\right\},$$

e c(p, n, α_n) é escolhida por forma a que seja garantida uma condição do tipo

$P_{N(\boldsymbol{\mu},\boldsymbol{\Sigma})}$(não sejam identificados α_n *outliers* numa amostra de dimensão *n*)=1-α, (4.1)

com α_n=1-(1-α)$^{1/n}$. Esta condição é equivalente a manter o nível de significância global α na aplicação de teste múltiplos. Tomando por exemplo o caso univariado, tome-se uma variável aleatória X com distribuição normal e suponha-se que se pretende,

$$P(x \in]\mu-\varepsilon, \mu+\varepsilon[) = 1-\alpha .$$

Tomando agora uma amostra de dimensão *n* de variáveis aleatórias i.i.d e pretendendo manter-se o grau de confiança, vem:

$$P(x_1 \in]\mu-\varepsilon, \mu+\varepsilon[, ..., x_n \in]\mu-\varepsilon, \mu+\varepsilon[) = 1-\alpha_n$$

ou seja,

$$P(x_i \in]\mu-\varepsilon, \mu+\varepsilon[, i = 1, ..., n) = 1-\alpha_n$$

Quanto deverá valer o α_n para que não se altere o grau de confiança i.e., que grau deverá ser tomado em cada intervalo individual pretendendo-se manter o grau de confiança global? Como por hipótese as observações são independentes isto é equivalente a dizer que $(1-\alpha_n)^n = 1-\alpha$ donde resulta que α_n=1-(1-α)$^{1/n}$. Fazendo o desenvolvimento em série de McLaurin de ordem um vem que 1-(1-α)$^{1/n} \cong \frac{\alpha}{n}$ i.e., a utilização deste valor para α_n mostra a relação com a correcção de Bonferroni (1936) para testes múltiplos. Rencher (1998, Cap. 3, pág. 77) apresenta o desenvolvimento de Bonferroni para a obtenção de intervalos de confiança conjuntos (vários parâmetros) com um coeficiente especificado global 1-α, mostrando que na construção de *g* intervalos de confiança é suficiente que o grau de confiança iguale 1-α/*g*. De salientar que há muitos autores que não têm o cuidado de proceder a esta correcção (e também que existem outros métodos de correcção).

Em Becker (2000) e Becker e Gather (2001) investiga-se a *performance* de vários procedimentos de identificação de *outliers* multivariados baseados na utilização de estimadores robustos de localização e escala. São calculadas por simulação as constantes c(*p*, *n*, α_n) para os estimadores MVE e MCD de acordo com a condição (4.1), usando-se os algoritmos de Rousseeuw e Van Driessen (1999) e Rousseeuw e van

Zomeren (1990) implementados no S-Plus (versão 4.5) para n=20, 50 e p=2, 3, 4 com base em 10000 observações e α=0.1.

Para terminar é importante salientar que com a definição de Davies e Gather (1993) o problema da identificação de *outliers* deixa de ser o de determinar as observações *outliers* que são contaminantes para ser o da especificação das observações que caiem na região out ou OR (conforme os parâmetros sejam conhecidos ou desconhecidos, respectivamente). De ressalvar que não há consenso relativamente aos termos utilizados para distinguir as observações consideradas *outliers*. As definições de observações contaminantes e discordantes não são simples nem precisas- é um campo um tanto polémico. Por esta razão ressalta-se que todo o trabalho que daqui para a frente irá ser desenvolvido será baseado na definição de *outlier* proposta por Davies e Gather (1993).

4.5 Regra de detecção de *outliers* multivariados (RRO)

Depois de apresentada a definição formal de *outlier* a ser utilizada neste trabalho, chegou o momento de ser apresentada a regra (ou método) de rejeição de *outliers* no contexto multivariado e na detecção de múltiplos *outliers*.

Tal como Ripley (1996) refere, o conceito de *outlier* não se ajusta totalmente no trabalho da teoria da decisão estatística; neste supõe-se uma descrição clara do problema e os *outliers* sugerem especificações incorrectas. Por analogia, se considerarmos o espaço dos padrões como um grande conjunto de terra (um grande continente) então aquilo que se pretende é encontrar as regiões habitadas, onde serão especificadas as várias classes. Fora destas zonas existem vastas áreas de "terra incógnita" onde se encontram os *outliers*. Este argumento abona a favor da utilização de métodos heurísticos de detecção de *outliers*.

Embora a utilização dos estimadores robustos, referidos na Secção 4.2, tenha sido utilizada nomeadamente para se evitar o efeito *masking* (Rousseeuw e Leroy, 1987; Rousseeuw e van Zomeren, 1990; Rocke e Woodruff, 1996; Becker e Gather, 1999), o desempenho destes métodos ainda está bastante dependente da adaptação da distribuição normal multivariada ao conjunto de dados. O objectivo do método que irá

aqui ser proposto será o de enfraquecer esta dependência contribuindo para a construção de um método promissor para a detecção de *outliers* multivariados em situações não padrão.

Já aqui foi sugerido, na Secção 4.3, que a utilização de métodos de detecção de *outliers* aplicados a dados agrupados em nuvens poderá revelar-se um procedimento perspicaz para a compreensão da estrutura dos dados. É por esta razão que a regra a seguir apresentada começa pela utilização desta ideia numa tentativa de melhorar o conhecimento da estrutura dos dados, na esperança de que cada nuvem pareça "mais normal" do que a original simplificando o procedimento de detecção das observações denominadas por *outliers*.

RRO (**R**egra de **R**ejeição de ***O****utliers*)

Considere-se um conjunto de dados multivariado com n observações e p variáveis.
As ideias básicas podem ser descritas em quatro passos:

1. Segmentar as n observações (possivelmente com forma complicada) em k nuvens através da utilização de um método de partição *clustering*.
2. Aplicar a regra de detecção de *outliers* multivariados a cada nuvem através do cálculo de distâncias de Mahalanobis, à partida robustas[9], de cada observação a cada nuvem. Uma observação é considerada *outlier* se é *outlier* para todas as nuvens. Todas as observações de uma nuvem podem também ser consideradas *outliers* se o tamanho relativo da nuvem é pequeno (propõe-se $n<2p+2$ já que para um número menor de observações as estimativas da matriz de covariâncias não são fiáveis[10]).
3. Remover as observações detectadas em 2. e repetir 1. e 2. até que não sejam detectadas mais observações.

[9] Embora seja bem sabido que a utilização de estimadores robustos melhora os resultados na medida em que minimizam a influência dos *outliers*, contornando o efeito *masking*, este comportamento poderá alterar-se pela prévia divisão dos dados em nuvens. É o que se irá verificar mais à frente quando se proceder aos estudos de simulação.

[10] Neste trabalho fixou-se o ponto de rotura em 25% numa tentativa de tornar os estimadores mais eficientes. Por esta razão, o valor assumido para as nuvens serem consideradas pequenas, resulta do facto de por defeito, *quan* (número de observações que determinam o estimador MCD) ser no mínimo $(n+p+1)/2$. Como se trabalha com $quan=0.75n$ deverá ter-se $n\geq 2p+2$.

4. A decisão final sobre considerar todas as observações duma dada nuvem (não removida anteriormente, isto é, com mais do que $2p+1$ pontos) como *outliers* é baseada numa tabela de distâncias tipo Mahalanobis entre nuvens.

De seguida discute-se mais detalhadamente cada um dos passos.

Acerca do método de partição *clustering* e do número de nuvens (Passo 1)

A análise de *clusters* consiste no particionar dos dados em subgrupos com significado, quando o número de subgrupos- designados por nuvens (*clusters*) e outra informação acerca da composição dos mesmos pode ser desconhecida. Uma das vantagens da aplicação de um método de *clustering* aos dados é que posteriormente cada uma das nuvens pode ser vista como vinda de uma única população o que faz com que a regra de detecção de *outliers* possa ser aplicada individualmente a cada nuvem.

Os métodos de partição inserem-se num vasto leque, desde os heurísticos até aos procedimentos mais formais baseados em modelos estatísticos. No entanto, ambos seguem geralmente uma estratégia iterativa na qual as observações vão sendo colocadas nas possíveis nuvens.

A escolha do método de partição *clustering* é crucial e pode depender de uma prévia análise exploratória de dados devendo ter-se em conta aspectos como a forma dos dados e a dimensão da amostra. Jiang *et al.* (2001), frequentes utilizadores dos métodos de *clustering*, realçam essa mesma ideia- não existe um procedimento que possa ser considerado o melhor, o sucesso da técnica de *clustering* escolhida depende do tipo de dados e do objectivo da investigação. Outros autores que corroboram e reforçam esta ideia são Jain *et al.* (2000) ao referir que estudos recentes em análise de *clusters* sugere que se tenha sempre em atenção (i) que todos os algoritmos de *clustering* irão encontrar nuvens num determinado conjunto de dados, quer existam quer não pelo que os dados devem ser sujeitos a testes para análise das tendência de *clustering* antes da aplicação do algoritmo seguido de uma validação das nuvens geradas pelo mesmo; (ii) não existe o "melhor" algoritmo de *clustering*; para um dado conjunto de dados, aconselha-se a que sejam experimentados vários. Não é por acaso que Kaufman e Rousseeuw (1990) e

Rousseeuw, Kaufman e Trauwaert (1996) se referem à arte de encontrar grupos nos dados.

Formalmente, um método de partição *clustering* pode ser formulado como se segue: Dados *n* padrões num espaço *p*-dimensional, determinar a partição dos padrões em *k* nuvens, por forma a que os padrões pertencentes à mesma nuvem sejam mais similares aos padrões dessa nuvem do que aos padrões das restantes. O valor de *k* pode ou não ser especificado tendo, contudo, de ser adoptado um critério de *clustering*, global ou local (Jain *et al*., 2000). Estes mesmos autores sumariam os algoritmos de *clustering* mais conhecidos evidenciando as suas propriedades e tecendo alguns comentários que poderão revelar-se úteis.

O método *k-means* (devido a Forgy, 1965) tende a produzir *clusters* hiperesféricos pelo que se a hipótese de nuvens esféricas não for adequada ou se não estiverem bem separadas no espaço das variáveis deve ser utilizado um método mais robusto como por exemplo o método *mclust* (*model-based and heuristic hierarchical agglomerative clustering*) de Banfield e Raftery (1992). Fraley e Raftery (1999, 2002) referem vários aspectos deste método e indicam como deve ser utilizado e como proceder em determinadas situações particulares, no S-Plus, nas versões 3.4 para UNIX e 4.5 para Windows no primeiro artigo, e na versão 6 para UNIX e Windows no segundo. No entanto, o *mclust*, pode não ser adequado em determinadas estruturas para grandes conjuntos de dados (ver Fraley e Raftery, 1999, pág. 13). Nesse caso o método *clara* de Kaufman e Rousseeuw (1990) pode ser mais apropriado. Um método similar ao *k-means* mas onde se usa "medoids" em vez de "centroids" é o *pam* (*partitioning around medoids* de Kaufman e Rousseeuw, 1990). Uma outra vantagem deste método relativamente ao *k-means* (ver por exemplo, *help* do S-Plus) nomeadamente o facto de ser mais robusto por minimizar uma soma de dissimilaridades em vez do tradicional quadrado das distâncias euclideanas.

Os métodos *k-means*, *pam* e *clara* impõe uma especificação inicial do número de nuvens. Em contraste existem os chamados métodos hierárquicos como por exemplo o *mclust*, já aqui referido, ou o *fanny* (também devido a Kaufman e Rousseeuw, 1990) tido como um método dito difuso (*fuzzy*) e onde ao contrário dos restantes apresentados, onde um objecto é classificado numa só nuvem, neste caso a cada observação associa-se

uma função de pertença p_i ($0< p_i<1$ e $\Sigma p_i=1$); as funções p_i medem o grau de pertença de qualquer observação **x** a cada uma das classes. Mais detalhes são dados em Rousseeuw, Kaufman e Trauwaert (1996); neste artigo são apresentados vários métodos que generalizam o *k-means* difuso mas que não se restringem a nuvens esféricas permitindo a existência de nuvens elipsoidais com várias orientações. Comparado com outros métodos difusos, o *fanny* tem a vantagem de ser robusto à hipótese de *clusters* esféricos. Mais considerações acerca do *background* dos vários métodos de partição *clustering*, desde os procedimentos heurísticos até aos procedimentos mais formais, são referidas, por exemplo, em Fraley e Raftery (1998).

Uma outra questão, não menos importante, prende-se com a escolha do número de nuvens, *k*. À partida não é necessário fixar previamente *k*; sugere-se que se experimentem vários valores e que se observem os resultados. Mas claro que há um limite para *k* que depende do número de observações e do número de variáveis. Como se deverá então proceder? Naturalmente que *k* deverá ser no mínimo 2, $k=1$ corresponde à situação sem nuvens. Por outro lado, se o conjunto dos dados "limpo" (sem *outliers*) for aproximadamente normal o método funciona com qualquer número de nuvens, sujeito obviamente a restrições de compromisso entre o número de observações e o número de variáveis. Caso o conjunto de dados "limpo" não seja normal, e não havendo limites para o número de observações, os resultados devem estabilizar a partir de determinado *k* (dependente, obviamente, da estrutura dos dados), conforme se exemplificará mais adiante no Cap. 5, com casos práticos.

Considere-se então a seguinte conjectura:

> "Admita-se que não há limite para o número de observações i.e.,
> que se pode ter tantas observações quantas se queira."

Então, dependendo da estrutura dos dados "limpos":

- existe um *k´* a partir do qual os resultados não se alteram.
- para uma estrutura de dados "limpos" elipticamente simétrica e unimodal (como por exemplo a distribuição normal ou *t*-Student) $k´=2$ ($k=1$ corresponde à situação sem nuvens).

- pode-se ir aumentando o *k* até encontrar uma valor a partir do qual haja "estabilidade", no sentido de não se verificarem grandes alterações nos *outliers* detectados (claro que não se pode garantir que haja estabilidade para todos os valores a partir de *k´* mas pelo menos consegue-se obter um valor de referência).

É evidente que na prática não se consegue aumentar *k* indefinidamente. Como encontrar então uma regra útil para a escolha desse valor? Pelas razões já apontadas deverá tomar-se o 2 como mínimo. Por outro lado é habitual considerar-se que uma nuvem de pontos deverá ter no mínimo entre 5 e 10 observações por variável[11] pelo que será sensato considerar-se,

$$\frac{n}{10p} < k_{\max} < \frac{n}{5p}$$

De acordo com esta regra, se por exemplo n=150 e p=2,3,4 , obtêm-se os seguintes valores de referência para $k_{\max}$:

p=2	7.5	15
p=3	5	10
p=4	3.75	7.5

Tabela 4.2- Alguns valores de referência para *k*.

Uma outra possibilidade, mais fundamentada, para o cálculo do número de nuvens é a utilização do critério da informação de Akaike, AIC[12], que se define como:

AIC= -2 log (máxima verosimilhança) + 2 (número de parâmetros estimados),

ou ainda do critério BIC (*Bayesian Information Criterion*)[13] definido como:

[11] Jain e Chandrasekaran (1982) referem que num problema de r.p. é aceitável e considerado de boa prática que se tome um número de observações 5 a 10 vezes maior que o número de variáveis. Mais tarde, Jain *et al.* (2000) reforçam esta mesma ideia afirmando "*It is generally accepted that using at least ten times as many training samples per class as the number of features is a good practice to follow the classifier design*".
[12] Ver Sakamoto *et al.* (1986)
[13] Devido a Schwarz (1978).

BIC= -2 log (máxima verosimilhança) + (n.º de parâmetros estimados)*log(n.º de observações).

A ideia de usar um destes critérios baseia-se numa sugestão de McLachlan e Peel (2000, Cap. 6, pág. 175) no contexto da obtenção do número de componentes de um modelo mistura. A referência ao critério AIC é bastante comum na literatura dedicada à detecção de *outliers* múltiplos no contexto dos modelos normais (ver por exemplo, Barnett e Lewis, Cap. 4, pág. 105 ou Beckman e Cook, pág. 131). Fraley e Raftery (1998) também sugerem a utilização do critério BIC como um critério bem sucedido na obtenção do número de nuvens. Note-se que embora teoricamente, os parâmetros devessem ser estimados pelo método da máxima verosimilhança, esta adaptação do AIC (AIC "livre") é utilizada com frequência e está associada à obtenção de bons resultados. Aliás como acabou de ser referido, também o próprio critério BIC é usado como um BIC "livre" i.e. acaba por ser um critério baseado na mistura de normais independentemente da distribuição dos dados.

Por outro lado, Hardin *et al.* (2000) consideram admissível que o número de nuvens varie entre 2 e 10 mesmo para elevados conjuntos de dados (no exemplo tratado neste artigo são consideradas 7000 observações).

Uma alternativa recente ao método *k-means* tradicional, foi proposta por Kovesi, Boucher e Saoudi (2001) e designada por SKA (*Stochastic K-means Algorithm*). Este algoritmo *k-means* estocástico tem a vantagem de não estar dependente da escolha inicial dos centros das nuvens ao contrário do algoritmo *k-means* usual (designado por KMA no trabalho destes autores).

Para finalizar este assunto, uma última alternativa que poderá ser interessante mas que não irá ser alvo de análise já que pressupõe o recurso à simulação e que portanto iria tornar o método aqui proposto computacionalmente ainda mais intensivo, é a estatística GAP devida a Tibshirani *et al*. (2000). Estes autores propõem um método para estimar o número de nuvens num conjunto de dados através da utilização do *output* de qualquer método de *clustering* usual (tal como o *k-means* ou um qualquer método hierárquico), comparando a dispersão da nuvem com a dispersão esperada sob um modelo de referência apropriado. Porém, uma vez que sob o modelo de referência

uniforme esta estatística ultrapassou o desempenho de outros métodos de estimação do número de nuvens nas simulações efectuadas, considera-se ter interesse facultar a sua referência.

Acerca da regra de detecção de *outliers* multivariados (Passo 2)

Na sequência do que foi referido na Secção 4.4, e com base no trabalho de Davies e Gather (1993), a regra de detecção de *outliers* multivariados é tal que para uma amostra normal multivariada de dimensão n_j, a probabilidade de não existirem observações detectadas como sendo *outliers* é 1-α_j para j=1,...,k. Se α_j=1-(1-α)$^{1/k}$ pode ser garantido um nível de significância global α para misturas de k distribuições normais multivariadas[14]. Uma observação **x** tal que:

$$d^2 = (\mathbf{x} - \hat{\boldsymbol{\mu}}_j)^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_j^{-1} (\mathbf{x} - \hat{\boldsymbol{\mu}}_j) \geq c(p, n_j, \alpha_j),$$

é considerada *outlier* relativamente à j-ésima nuvem. A constante $c(p,n_j,\alpha_j)$ é assintoticamente $\chi^2_{p;\beta}$ onde $\beta = (1-\alpha_j)^{1/n_j}$. No entanto, pelo facto de para amostras não muito grandes e dependendo dos estimadores $\hat{\boldsymbol{\mu}}_j$ e $\hat{\boldsymbol{\Sigma}}_j$ poderem existir grandes diferenças relativamente aos valores assintóticos deve-se recorrer à simulação para obtenção de constantes mais fiáveis, tal como sugerido por Becker e Gather (2001).

Para estimadores de $\hat{\boldsymbol{\mu}}_j$ e $\hat{\boldsymbol{\Sigma}}_j$ pode tomar-se o vector de médias amostral clássico e a matriz de covariâncias amostral clássica ou, para maior protecção contra o efeito de *masking*, estimadores robustos correspondentes.

Acerca da decisão final (Passo 4).

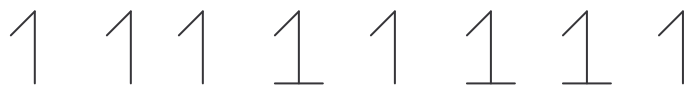
Sejam $\mathbf{x}_{ij}$ i=1,...,n_j, j=1,...,k , as observações finais, onde os n_j representam os tamanhos das k nuvens finais (note-se que $\sum_j n_j$ é em geral menor do que n) e defina-se

$$D_{lm} = \min_{i=1,\ldots,n_l} (\mathbf{x}_{il} - \hat{\boldsymbol{\mu}}_m)^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_m^{-1} (\mathbf{x}_{il} - \hat{\boldsymbol{\mu}}_m) \qquad l, m = 1, \ldots, k \ .$$

[14] Note-se que embora a obtenção das constantes seja baseada na distribuição normal, os dados não têm de ser normais. Ao se proceder à divisão em nuvens estamos como que a tomar a mistura de várias normais como foi exemplificado aquando da apresentação da motivação na Secção 4.3.

Diz-se que uma dada nuvem, *l,* é desligada dos restantes dados se $D_{lm} > c(p,n_j,\alpha_j)$ para todo o $m \neq l$ e ligada caso contrário. Doutra forma, uma dada nuvem *l* é desligada se o mínimo das distâncias de Mahalanobis das observações da nuvem *l* em relação ao centro da classe *m* ainda for maior que a constante correspondente. Nuvens desligadas são suspeitas de conterem apenas *outliers* (a decisão final deve depender do tipo de dados).

Apesar de ser lícito a introdução deste passo poderão existir situações onde será conveniente a sua desactivação. Considere-se o tão famoso exemplo do reconhecimento de caracteres[15]. Será válido, neste caso, rejeitar todas as observações, pertencentes a nuvens de dimensão superior a $2p+1$, mesmo que esta seja catalogada como uma nuvem desligada? Esta questão não é tão inverosímil quanto poderá parecer à partida. Basta para isso que se pense em algo tão simples quanto o seguinte exemplo:



Será que a 4ª, 6ª e 7ª observações devem ser consideradas *outliers*? Ou será que as restantes é que devem ser consideradas *outliers*? Obviamente que a resposta a ambas essas questões é não. O que se passa com este exemplo, e com certeza com alguns outros, é que se poderá estar interessado em que a mesma classe tenha observações em localizações diferentes pelo que se deve ter especial cuidado com a sua utilização.

Nota

O procedimento proposto é invariante para transformações afins (ou seja, as observações identificadas como *outliers* não mudam quando os dados são submetidos a transformações daquele tipo) se os estimadores de localização e escala forem equivariantes e se o método de *clustering* for invariante, para o mesmo tipo de transformações. Note-se que o método *k-means* não tem esta propriedade.

[15] Problemas que surgem neste contexto e soluções/ metodologias para a sua resolução são descritas, por exemplo, em Suen *et al*. (1992).

4.6 Alguns procedimentos alternativos

Nos últimos anos tem-se registado bastante actividade de investigação nesta linha. Como referem Jain *et al*. (2000, pág. 79) desde os anos 80 que o r.p. estatístico tem experimentado um rápido crescimento tendo esta rápida expansão várias causas. Uma delas prende-se com o aumento da interacção entre várias diferentes disciplinas que forçaram a fomentação de novas ideias, metodologias e técnicas enriquecendo o paradigma do r.p. estatístico tradicional. Outra causa importante é o aparecimento de processadores rápidos, da *internet* e da possibilidade de aumentar o armazenamento em memória acrescido de um menor custo. O avanço da tecnologia computacional tornou possível a implementação de algoritmos de optimização e aprendizagem complexos impossíveis há algumas décadas atrás. Não menos importante foi também o aparecimento de aplicações tais como o *data mining* que trouxeram novos desafios e que favoreceram a renovação do interesse pela investigação na área do r.p. estatístico.

Entende-se importante, antes de mais, fazer menção que a par da ideia que motivou este trabalho e que se baseia antes de mais na partição dos dados em nuvens, Rocke e Woodruff (1999) também sugerem que o reconhecimento das ligações entre os estimadores para nuvens e os *outliers*, pode vir a dar um grande impulso nos procedimentos de detecção de *outliers*. Embora este trabalho e o destes autores tivesse sido desenvolvido de forma independente e que é facilmente justificável pela reflexão que inicia esta secção, este aspecto pode ser encarado de forma positiva já que ao que tudo indica deparamo-nos com uma ideia boa e certamente com futuro. De salientar, no entanto dois aspectos. Por um lado, em 1999, Rocke e Woodruff (1999, pág. 3) assumem, como tantos outros, que a população principal obedece a uma distribuição elipticamente simétrica. Por outro lado, como estes mesmos referem, o seu método está dependente da descoberta da verdadeira forma das nuvens pelo que consideram dados difíceis aqueles em que a forma dos dados principais e a forma do conjunto de dados completo difere substancialmente. Mostra-se ao longo deste trabalho que o método proposto é capaz de lidar eficazmente com situações deste tipo.

Entretanto, já este trabalho tinha sido desenvolvido quando se encontrou na *internet* o *abstact* de um artigo de Hardin, Rocke e Woodruff submetido ao CAMDA-2000 (*Critical Assessment of Microarray Data Analysis*). Posteriormente ficou disponível a

versão completa (Hardin *et al*., 2000) como apresentação efectuada nessa mesma conferência, embora até à data nunca tenha chegado a ser publicado. O objectivo inerente a esse artigo é basicamente o mesmo desta fase do trabalho de investigação. Porém, além do trabalho ter sido desenvolvido de uma forma completamente independente, existem várias diferenças em relação à forma como o problema é abordado, das quais se realçam as seguintes:

- a aplicação que apresentam é realizada para *clusters* de variáveis,
- não fazem testes múltiplos,
- não consideram nuvens desligadas,
- relativamente à escolha de *k*: ao contrário da regra aqui apresentada onde, como se verá mais adiante, se sugere e considera a utilização do critério AIC, estes autores não indicam qualquer tipo de critério,
- relativamente à constante que serve de limite de detecção: enquanto que na regra aqui apresentada se opta pela simulação ou pelo Qui-quadrado, estes autores utilizam uma constante fornecida pela distribuição *F*.

Ainda este ano aparece um artigo de Hardin e Rocke (2004) que segue a linha do último referido (embora de um ponto de vista mais teórico, sem aplicação prática como no anterior onde o próprio nome "*Robust model-based clustering of genes in microarray data: are there gene clusters?*" diz quase tudo) e que consiste de uma forma genérica na apresentação de um método robusto de *clustering* em conjunto com um método de identificação de *outliers* sendo mais uma vez utilizada a distribuição *F* para várias nuvens de diferentes tamanhos e formas.

Jiang, Tseng e Su (2001) apresentam um processo de *clustering* em duas fases para a detecção de *outliers*. Na primeira fase modificam o algoritmo *k-means* tradicional com base na seguinte premissa: "se um novo padrão de *input* está suficientemente distante de todas as nuvens então este passa a ser afectado a uma nova nuvem na qual ele passa a constituir o centro". Isto faz com que pontos de uma mesma nuvem possam ser todos *outliers* ou todos não *outliers*. Numa segunda fase aplica a teoria das árvores, nomeadamente a MST (*minimum spanning tree*).

Peña e Prieto (2001) também apresentam um procedimento de detecção de *outliers* multivariados e um estimador robusto da matriz de covariâncias baseado na análise de

projecções dos pontos da amostra num conjunto de $2p$ direcções, onde p representa a dimensão do espaço amostral. Estas direcções são obtidas pela maximização e minimização do coeficiente de curtose das projecções. Estes autores referem que este procedimento pode ser visto como um caminho rápido e bem sucedido de implementação do algoritmo de Stahel-Donoho. Para exemplificação do procedimento proposto efectuaram um extenso estudo de simulação para distribuições normais contaminadas com vários níveis de contaminação e diferentes números de variáveis que conforme estes mesmos autores referem, mostrou um comportamento satisfatório especialmente em amostras de grandes dimensões e em contaminações concentradas.

Fraley e Raftrey (1998, 2002) sugerem o tratamento dos *outliers* através de um modelo mistura com inclusão de um termo, representativo do ruído, através de um processo de Poisson de primeira ordem. Esta solução exige, no entanto, que seja obtida uma boa estimativa inicial para o ruído. São feitas algumas sugestões e apresenta-se um exemplo ilustrativo, no sentido da classificação, em Fraley e Raftrey (2002) para os célebres dados Iris (Fisher, 1936).

Para terminar, de referir que o que é feito nesta tese é algo no espírito do que é proposto no artigo de Fraley e Raftrey (2002) no que confere à divisão dos dados em nuvens e à possibilidade de estimação via modelos mistura mas indo mais longe na medida em que são utilizados estimadores robustos e em que se inclui a detecção de *outliers* no seguimento destas mesmas ideias.

Capítulo 5

Estudo de simulação

5.1 Introdução

Tal como na maior parte dos artigos onde se procede à identificação de *outliers*, inclui-se neste trabalho um estudo de simulação. Estes estudos são hoje em dia largamente aceites e fazem parte do conteúdo de grande parte dos artigos de análise multivariada. Neste contexto têm como objectivo avaliar a *performance* do método descrito anteriormente a proceder à sua comparação com o método usual baseado no cálculo das distâncias de Mahalanobis[1]. Pretende-se destacar a necessidade da utilização de métodos robustos, em determinadas situações, e as vantagens que advêm da aplicação conjunta dos métodos de partição *clustering*. Concretizam-se as ideias apresentadas, no Capítulo 4, para a detecção de *outliers* em contextos não padrão, nomeadamente em regiões não elipsoidais (i.e., não necessariamente normais ou elipticamente simétricas). Como referem Rocke e Woodruff (1999, pág. 16): "*difficult data sets ... are those where the shape of the main data and the shape of the entire dataset differ substantially*."

Naturalmente, a forma como os *outliers* são gerados é de importância fundamental neste tipo de estudos. Como alertam Davies e Gather (1993) a boa *performance* do procedimento de simulação pode simplesmente ser devida ao facto de não ocorrerem as situações onde este falha, por causa do método usado para a geração dos *outliers*. O comportamento de uma regra de identificação de *outliers* deve, por consequência, ser teoricamente analisado por forma a se descobrirem as condições nas quais este tem sucesso ou falha.

As três primeiras secções deste capítulo são dedicadas à apresentação de um pequeno estudo de simulação onde são analisadas oito situações distribucionais correspondentes a duas configurações de dados (normais e não normais) e comparados

[1] A comparação entre diversos métodos é extremamente difícil do ponto de vista teórico. Assim, a maior parte das conclusões associadas a este tipo de estudos baseiam-se em resultados obtidos em estudos de simulação pelo método de Monte Carlo (ver por exemplo Gentle, 1998).

dois tipos de estimadores (clássicos e MCD). Segue-se um estudo mais detalhado (Secção 5.5) onde se incluem mais sete configurações de dados (algumas com maior percentagem de contaminação e com amostras de dimensão superior), mais um estimador robusto, OGK, e onde se entra em mais detalhes particularmente no que diz respeito à escolha do valor de k e à forma de sintetizar a informação fornecida pelos indicadores de simulação. O Capítulo termina com a Secção 5.7 onde se procede à apresentação dos aspectos computacionais que serviram de suporte ao estudo de simulação realizado.

5.2 Delineamento

Reafirmando as considerações feitas anteriormente, avança-se com um estudo preliminar com o intuito da obtenção de uma primeira indicação do comportamento do método, avançando-se depois para um estudo mais alargado. Para se avançar com o estudo de simulação e com a implementação foram realizadas as seguintes escolhas:

- Métodos de *clustering*
  - *k-means,*
  - *pam* (*partition around medoids* de Kaufman e Rousseeuw, 1990)[2],
  - *mclust* (*model based clustering* de Banfield e Raftery, 1992),

  cada um deles com k=3,4,5.

- Dois pares de estimadores de localização-escala:
  - Clássicos $\left(\bar{\mathbf{x}}, \mathbf{S}\right)$ com limites de detecção assintóticos,
  - Robustos, RMCD25 (*Reweighted Minimum Covariance Determinant* com um ponto de rotura aproximado de 25% que tem maior eficiência do que o correspondente com ponto de rotura de 50% (que é o máximo)[3] com limites de detecção determinados previamente através de uma simulação com 10000 conjuntos de dados normais,

[2] Este método foi também experimentado coma opção *metric*="manhattan" (disponível no S-plus) em vez da distância euclideana, que é a utilizada por defeito, mas não se obteve qualquer melhoria nos resultados.
[3] Como é referido em Rousseeuw e van Driessen (1999), na certeza dos dados conterem menos de 25% de contaminação um bom compromisso entre o ponto de rotura e a eficiência estatística é obtido fazendo $h=\lceil 0.75n \rceil$. Para o cálculo computacional foi utilizado o comando cov.mcd (do S-plus) com a opção *quan=floor*(0.75n).

- Nível de significância global: α=0.1,

- Classificar as nuvens desligadas (no passo 4 da RRO) como *outliers*,

- Oito situações distribucionais:
  1. Normal, p=2, sem *outliers*, 150 observações $N_2(\mathbf{0},\mathbf{I})$.
  2. Normal, p=2, com *outliers*, 150 observações $N_2(\mathbf{0},\mathbf{I})$ e 10 observações *outliers* $N_2((10.51,10.51)^T,0.1\mathbf{I})$.
  3. Normal, p=4, sem *outliers*, 150 observações $N_4(\mathbf{0},\mathbf{I})$.
  4. Normal, p=4, com *outliers*, 150 observações $N_4(\mathbf{0},\mathbf{I})$ e 10 observações *outliers* $N_4((8.59,8.59)^T,0.1\mathbf{I})$.
  5. Não-Normal, p=2, sem *outliers*, 50 observações $N_2(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 50 observações $N_2(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$ e 50 observações $N_2(\mathbf{0},\boldsymbol{\Sigma}_1)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9).
  6. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_2((-2,6)^T,0.01\mathbf{I})$.
  7. Não-Normal, p=4, sem *outliers*, 50 observações $N_4(\boldsymbol{\mu}_3,\boldsymbol{\Sigma}_3)$, 50 observações $N_4(\boldsymbol{\mu}_4,\boldsymbol{\Sigma}_4)$ e 50 observações $N_4(\mathbf{0},\boldsymbol{\Sigma}_3)$, com $\boldsymbol{\mu}_3=(0,12,0,0)^T$, $\boldsymbol{\Sigma}_3$=diag(1,0.3,1,1), $\boldsymbol{\mu}_4=(1.5,6,0,0)^T$ e $\boldsymbol{\Sigma}_4$=diag(0.2,9,1,1).
  8. Não-Normal, p=4, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_4((-2,6,0,0)^T, 0.01\mathbf{I})$.

Para cada combinação foram efectuadas 1000 simulações no S-plus. Com vista a tornar este procedimento automático (entrada do conjunto de dados e do número de nuvens e como saída o conjunto de dados "limpo" depois das várias iterações) foram construídos os programas que se listam no Apêndice E e aos quais se dedica mais atenção na Secção 5.7.

Um aspecto tido em conta foi o da localização dos *outliers*. Para o caso normal, Rocke e Woodruff (1996) definem o significado de *outliers* próximos e afastados ("close" e "far" *outliers*). Na segunda secção do seu artigo desenvolvem a teoria que permite caracterizar as classes de dados com *outliers* que são, num dado sentido, as mais difíceis de encontrar investigando as dificuldades associadas à localização de

outliers multivariados. Como resultado concluem que estes devem ser colocados à distância dQ_p onde $Q_p=\sqrt{\chi^2_{p;0.999}/p}$ fazendo d=2 (*close outliers*) ou d=4 (*far outliers*). Neste último caso, obtêm-se assim os valores 8.59 e 10.51, para p=2 e p=4 respectivamente.

Uma outra questão que impera e que deve ser aqui esclarecida diz respeito ao cálculo das constantes designadas por $c(p,n_j,\alpha_j)$. Naturalmente que se deve ter em conta não só o número de nuvens e o nível de significância, α, como também a dimensão de cada uma. Considerando por exemplo α=0.1 então para três nuvens é como se se pretendesse uma probabilidade de $0.9^{1/k}$. Segue-se o cálculo de α_N tendo em conta a dimensão das nuvens (ver programa no Apêndice E). Considera-se ainda necessário fazer referência a uma questão técnica associada à obtenção destas constantes. Para minimizar o tempo, que se revelava incomportável (várias semanas para cada valor de p, em função das várias dimensões das nuvens) condicionando o andamento do estudo de simulação, optou-se por proceder apenas ao cálculo dos valores de dimensões de nuvens correspondentes a múltiplos de 5; caso contrário tomou-se o valor correspondente à dimensão de nuvem mais próxima (por exemplo, para uma nuvem de dimensão 27 tomou-se a constante correspondente a uma nuvem de dimensão 25. Nesta fase do estudo de simulação procedeu-se à obtenção dos valores das constantes para k=1,3,4,5,6 e 7, dimensões de nuvens, n_k, entre 5 e 150 (múltiplos de 5), α=0.01,0.1 e p=2,4 para o estimador RMCD25 e com base em dados gerados para 10000 observações. Mais tarde foram ainda calculados outros valores destas constantes tendo-se abandonado α=0.01 (na detecção dos *outliers*) já que este valor restringia muito os resultados no sentido de existirem muitos *outliers* não detectados. Decidiu-se adoptar α=0.1 (de acordo com Becker e Gather, 2001) e calcular as constantes não só para este valor mas também para valores maiores de α, caso mais tarde fosse necessário.

As Figuras 5.1 e 5.2 mostram um dos conjuntos de dados gerados para a situação 6 com os respectivos contornos de detecção.

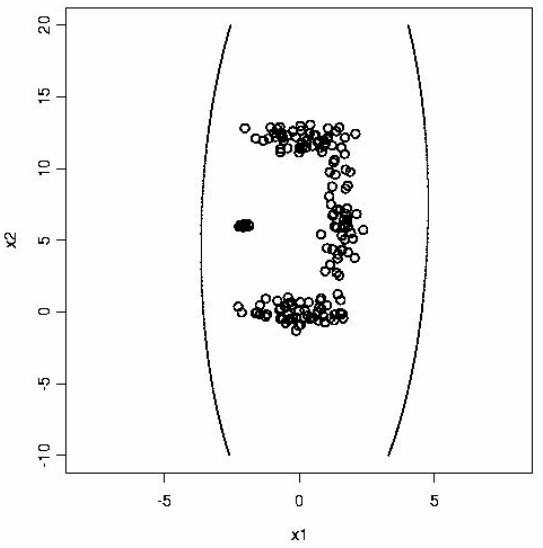


Figura 5.1- Contornos de detecção (α=0.1) calculados com a distância de Mahalanobis robusta (RMCD25).

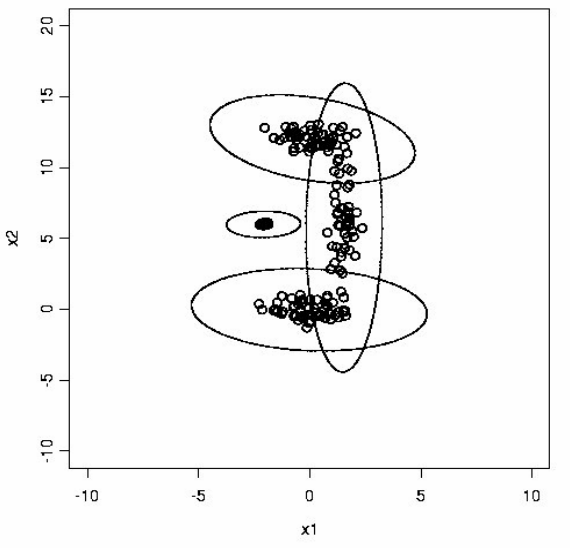


Figura 5.2- Contornos de detecção (α=0.1) calculados com o *mclust* para k=4 e RMCD25.

Repare-se que na Figura 5.1 nenhuma observação é detectada como *outlier* enquanto que na Figura 5.2 pode constatar-se que existem três nuvens ligadas e uma desligada. Neste caso é lícito concluir-se que é mais vantajoso proceder-se à detecção dos *outliers* após a aplicação de um método de partição.

5.3 Indicadores para o estudo de simulação

Na sequência daquilo que já foi referido torna-se agora necessário adoptar uma medida de avaliação da qualidade do método proposto. Inicialmente pensou-se em calcular a média e o desvio-padrão do número de *outliers* correctamente detectados e do número de *outliers* incorrectamente detectados mas rapidamente se abandonou essa ideia. Além destas mesmas medidas não serem robustas a informação que forneciam revelava-se escassa. Optou-se então pela utilização dos indicadores de simulação (p_1, p_2 e p_3) propostos por Kosinski (1999) mais um indicador p_4 (baseado nestes três): para um conjunto de dados com g pontos "bons" e b *outliers* a fracção de contaminação (fracção de *outliers*) é dada por b/N onde $N=b+g$. Sendo gerados S conjuntos de dados a *performance* do método vai ser avaliada com base nos seguintes indicadores:

p_1: proporção de simulações que resultaram na identificação correcta de todos os *outliers*

$$p_1 = \frac{1}{S}\sum_{i=1}^{S} I(m_i = 0)\,.$$

p_2: média das proporções de *outliers* não identificados em cada simulação

$$p_2 = \frac{1}{S}\sum_{i=1}^{S} \frac{m_i}{b}\,.$$

p_3: média das proporções de observações incorrectamente classificadas como *outliers* em cada simulação

$$p_3 = \frac{1}{S}\sum_{i=1}^{S} \frac{s_i}{g}\,.$$

p_4: proporção de simulações em que não houve observações incorrectamente classificada como *outliers*

$$p_4 = \frac{1}{S}\sum_{i=1}^{S} I(s_i = 0)\,,$$

$$\text{onde } I(m_i = 0) = \begin{cases} 1, & \text{se } m_i = 0 \\ 0, & \text{caso contrario} \end{cases}, \quad I(s_i = 0) = \begin{cases} 1, & \text{se } s_i = 0 \\ 0, & \text{caso contrario} \end{cases},$$

m_i: número de observações que são *outliers* e que não foram identificadas como sendo e s_i: número de observações que não são *outliers* e que foram identificadas como sendo .

Obviamente que quando b=0, p_1 e p_2 não estão definidos (daí a notação "nd" nas Tabelas 5.1 e 5.2 e nas tabelas inseridas no Apêndice D). Um bom método deve ter $p_1 \cong 1$, $p_2 \cong 0$, $p_3 \cong 0$ e $p_4 \cong 1\text{-}\alpha = 0.9$.

Outros critérios de comparação dos métodos utilizados nas simulações efectuadas poderiam ter sido adoptados (ver Becker, 2000). Destaque-se o recente critério de *performance* intitulado "tamanho do maior *outlier* não identificável" (*size of the largest nonidentifiable outlier*) com autoria de Becker e Gather (2001) para o caso univariado. A generalização deste conceito para o caso multivariado foi realizada por Becker e Gather (2001, Sec. 3).

5.4 Análise dos resultados preliminares

Nesta fase inicial apenas irão ser feitas algumas considerações enfatizando certos aspectos que se destacam da análise das figuras e tabelas e que servirão de suporte e fundamento às questões estudadas em pormenor e com mais precisão no estudo alargado.

São apresentadas três figuras referentes a: p_1 para dados normais e não normais com *outliers* (Figura 5.3), p_4 para dados normais com e sem *outliers* (Figura 5.4) e p_4 para dados não normais com e sem *outliers* (Figura 5.5). Optou-se por não apresentar qualquer gráfico relativo a p_2, uma vez que, nestas situações, este é praticamente igual a 1- p_1, e a p_3 e não proporciona nenhuma informação adicional relevante. A seguir são apresentados em forma de tabela (Tabelas 5.1 e 5.2) os resultados finais associadas às 1000 simulações efectuadas no S-plus.

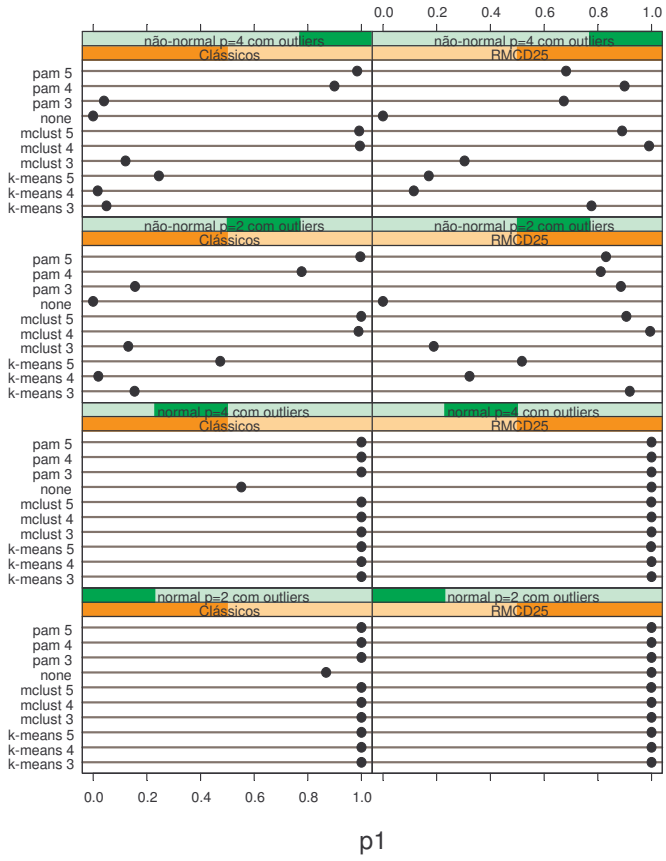


Figura 5.3- Resultados de p_1 para dados normais e não normais com *outliers.*

Analisando a Figura 5.3 pode constatar-se, como aliás era de esperar, que para situações normais sem nuvens, o estimador RMCD25 é melhor do que o clássico. Relativamente às situações não-normais destacam-se dois aspectos: quer o método sem *clustering*, quer o *k-means*, falham. De certa forma fica já aqui exemplificada a vantagem da utilização de métodos de *clustering* em situações não-normais.

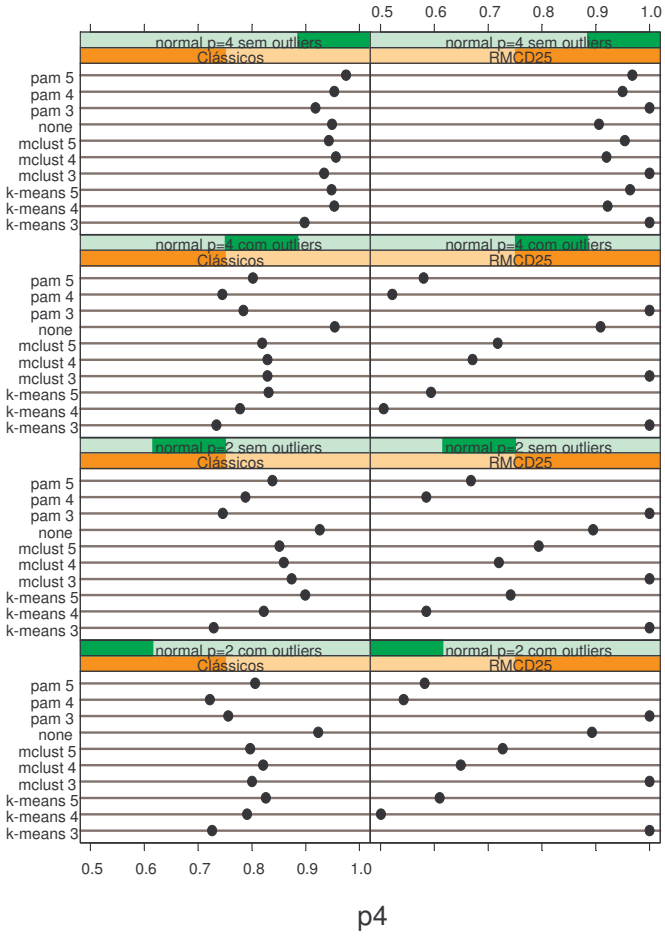


Figura 5.4- Resultados de p_4 para dados normais com e sem *outliers.*

Relembrando que este indicador deveria ser próximo de 0.9, da análise da Figura 5.4 destaca-se a sua instabilidade para o estimador RMCD25.

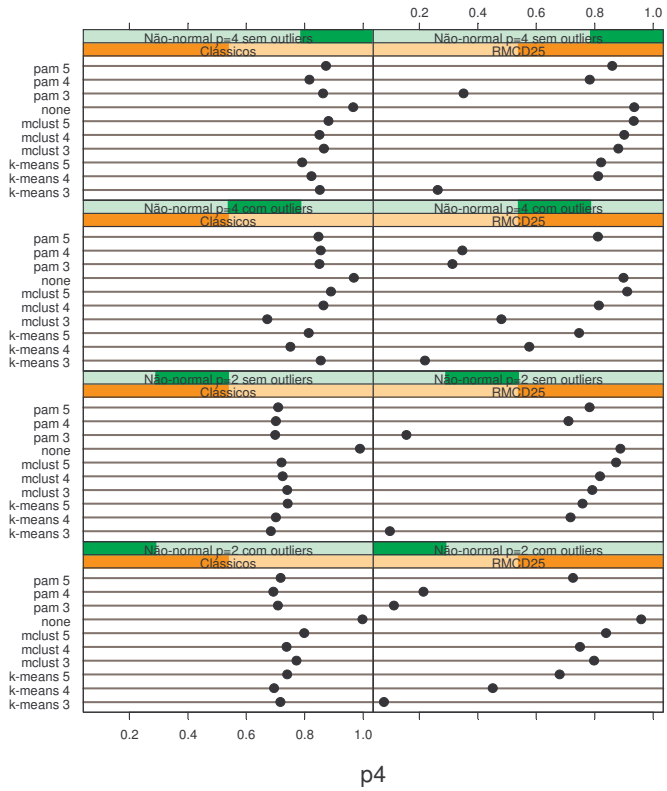


Figura 5.5- Resultados de p_4 para dados não normais com e sem *outliers*.

De certo modo, na Figura 5.5 verifica-se que os estimadores clássicos parecem ter um desempenho superior aos robustos. No entanto, analisando bem a Tabela 5.1 e vendo os resultados de uma forma genérica também se constata que o resultados estão bastante dependentes do método de partição escolhido; a melhor *performance* parece ser alcançada com o *mclust*.

			Robustos				Clássicos			
		k	p_1	p_2	p_3	p_4	p_1	p_2	p_3	p_4
Normal	*k-means*	3	nd	nd	0.000	1.00	nd	nd	0.002	0.73
p=2	"	4	nd	nd	0.005	0.58	nd	nd	0.002	0.82
Sem	"	5	nd	nd	0.003	0.74	nd	nd	0.001	0.90
outliers	*pam*	3	nd	nd	0.000	1.00	nd	nd	0.002	0.75
	"	4	nd	nd	0.005	0.58	nd	nd	0.002	0.79
	"	5	nd	nd	0.003	0.67	nd	nd	0.001	0.84
	mclust	3	nd	nd	0.000	1.00	nd	nd	0.001	0.87
	"	4	nd	nd	0.003	0.72	nd	nd	0.001	0.86
	"	5	nd	nd	0.002	0.79	nd	nd	0.001	0.85
	sem nuvens	1	nd	nd	0.001	0.90	nd	nd	0.001	0.93
Normal	*k-means*	3	1.00	0.00	0.000	1.00	1.00	0.00	0.002	0.73
p=2	"	4	1.00	0.00	0.007	0.50	1.00	0.00	0.002	0.79
com	"	5	1.00	0.00	0.004	0.61	1.00	0.00	0.001	0.83
outliers	*pam*	3	1.00	0.00	0.000	1.00	1.00	0.00	0.002	0.76
	"	4	1.00	0.00	0.005	0.54	1.00	0.00	0.002	0.72
	"	5	1.00	0.00	0.005	0.58	1.00	0.00	0.002	0.81
	mclust	3	1.00	0.00	0.000	1.00	1.00	0.00	0.002	0.80
	"	4	1.00	0.00	0.004	0.65	1.00	0.00	0.001	0.82
	"	5	1.00	0.00	0.003	0.73	1.00	0.00	0.002	0.80
	sem nuvens	1	1.00	0.00	0.001	0.89	0.87	0.13	0.001	0.92
Normal	*k-means*	3	nd	nd	0.000	1.00	nd	nd	0.001	0.90
p=4	"	4	nd	nd	0.001	0.92	nd	nd	0.000	0.95
sem	"	5	nd	nd	0.001	0.96	nd	nd	0.002	0.95
outliers	*pam*	3	nd	nd	0.000	1.00	nd	nd	0.001	0.92
	"	4	nd	nd	0.000	0.95	nd	nd	0.000	0.95
	"	5	nd	nd	0.000	0.97	nd	nd	0.000	0.98
	mclust	3	nd	nd	0.000	1.0	nd	nd	0.000	0.93
	"	4	nd	nd	0.001	0.92	nd	nd	0.000	0.96
	"	5	nd	nd	0.001	0.95	nd	nd	0.000	0.94
	sem nuvens	1	nd	nd	0.001	0.91	nd	nd	0.000	0.95
Normal	*k-means*	3	1.00	0.00	0.000	1.00	1.00	0.00	0.002	0.73
p=4	"	4	1.00	0.00	0.006	0.51	1.00	0.00	0.002	0.78
com	"	5	1.00	0.00	0.004	0.59	1.00	0.00	0.001	0.83
outliers	*pam*	3	1.00	0.00	0.000	1.00	1.00	0.00	0.002	0.78
	"	4	1.00	0.00	0.006	0.52	1.00	0.00	0.002	0.75
	"	5	1.00	0.00	0.005	0.58	1.00	0.00	0.001	0.80
	mclust	3	1.00	0.00	0.000	1.00	1.00	0.00	0.001	0.83
	"	4	1.00	0.00	0.004	0.67	1.00	0.00	0.001	0.83
	"	5	1.00	0.00	0.003	0.72	1.00	0.00	0.001	0.82
	sem nuvens	1	1.00	0.00	0.001	0.91	0.55	0.45	0.000	0.95

Tabela 5.1. Resultados do estudo de simulação para dados normais (1000 simulações).

			Robustos				Clássicos			
		k	p_1	p_2	p_3	p_4	p_1	p_2	p_3	p_4
	k-means	3	nd	nd	0.209	0.10	nd	nd	0.046	0.68
Não	"	4	nd	nd	0.018	0.72	nd	nd	0.030	0.70
Normal	"	5	nd	nd	0.007	0.76	nd	nd	0.031	0.74
p=2	*pam*	3	nd	nd	0.200	0.16	nd	nd	0.046	0.70
sem	"	4	nd	nd	0.018	0.71	nd	nd	0.035	0.70
outliers	"	5	nd	nd	0.011	0.78	nd	nd	0.049	0.71
	mclust	3	nd	nd	0.051	0.79	nd	nd	0.039	0.74
	"	4	nd	nd	0.012	0.82	nd	nd	0.041	0.72
	"	5	nd	nd	0.004	0.87	nd	nd	0.044	0.72
	sem nuvens	1	nd	nd	0.001	0.89	nd	nd	0.000	0.99
	k-means	3	0.92	0.08	0.213	0.08	0.16	0.85	0.040	0.72
Não	"	4	0.32	0.68	0.107	0.45	0.02	0.98	0.059	0.70
Normal	"	5	0.52	0.48	0.022	0.68	0.47	0.53	0.037	0.74
p=2	*pam*	3	0.89	0.11	0.208	0.11	0.16	0.84	0.042	0.71
com	"	4	0.81	0.19	0.181	0.21	0.78	0.22	0.051	0.69
outliers	"	5	0.83	0.17	0.015	0.73	1.00	0.01	0.029	0.72
	mclust	3	0.19	0.81	0.053	0.80	0.13	0.87	0.039	0.77
	"	4	1.00	0.01	0.042	0.85	0.99	0.01	0.039	0.74
	"	5	0.91	0.09	0.012	0.84	1.00	0.00	0.030	0.80
	sem nuvens	1	0.00	1.00	0.000	0.96	0.00	1.00	0.000	1.00
	k-means	3	nd	nd	0.171	0.26	nd	nd	0.013	0.85
	"	4	nd	nd	0.011	0.81	nd	nd	0.020	0.82
Não	"	5	nd	nd	0.008	0.82	nd	nd	0.031	0.79
Normal	*pam*	3	nd	nd	0.153	0.35	nd	nd	0.015	0.86
p=4	"	4	nd	nd	0.014	0.78	nd	nd	0.020	0.82
sem	"	5	nd	nd	0.003	0.86	nd	nd	0.020	0.87
outliers	*mclust*	3	nd	nd	0.029	0.88	nd	nd	0.013	0.87
	"	4	nd	nd	0.008	0.90	nd	nd	0.018	0.85
	"	5	nd	nd	0.004	0.93	nd	nd	0.013	0.88
	sem nuvens	1	nd	nd	0.000	0.94	nd	nd	0.000	0.97
	k-means	3	0.78	0.22	0.195	0.22	0.05	0.95	0.014	0.86
	"	4	0.11	0.88	0.062	0.58	0.02	0.98	0.054	0.75
Não	"	5	0.17	0.83	0.017	0.75	0.25	0.76	0.030	0.81
Normal	*pam*	3	0.67	0.33	0.175	0.31	0.04	0.96	0.011	0.85
p=4	"	4	0.9	0.10	0.138	0.35	0.90	0.10	0.011	0.86
com	"	5	0.68	0.32	0.010	0.81	0.98	0.02	0.018	0.85
outliers	*mclust*	3	0.30	0.70	0.326	0.48	0.12	0.88	0.17	0.67
	"	4	0.99	0.01	0.021	0.81	0.99	0.01	0.01	0.86
	"	5	0.89	0.11	0.005	0.91	0.99	0.01	0.02	0.89
	sem nuvens	1	0.00	1.00	0.001	0.90	0.00	1.00	0.000	0.97

Tabela 5.2. Resultados do estudo de simulação para dados não normais (1000 simulações).

Na sequência do que foi dito e apresentado até agora, surgem as seguintes conclusões gerais acerca do estudo de simulação:

- Para os dados normais todos os métodos demonstraram um bom comportamento com excepção da distância de Mahalanobis clássica onde se observa algum efeito de *masking* (o que já era esperado).
- Para dados não normais a melhor *performance* foi alcançada com o *mclust* para k=4 e k=5, sem diferenças significativas entre os estimadores clássicos e robustos.
- Para dados não normais destaca-se também a falha do método *k-means*, justificada pela notória não esfericidade das distribuições escolhidas.

Neste momento, onde à partida a RRO já parece dar alguns frutos, pode levantar-se uma questão acerca deste método que é a seguinte: será que existindo um só *outlier* isolado ele é detectado por este método? Experimentou-se uma situação idêntica à 6 mas onde se colocou mais um *outlier* com localização no ponto (5,14). A utilização deste método permitiu verificar que esse *outlier*, embora sendo um só isolado (à parte dos outros 10) também foi detectado pelo método *k-means* para k=4 e pelo *mclust* e *pam* para k=4 e k=5.

Um outro aspecto alvo de pesquisa foi a utilização do critério AIC (referido na Secção 4.5). Para exemplificação da utilização deste critério recorreu-se à estimação da densidade pelo método de kernel adaptado (sugerido na Secção 4.3) obtendo-se os valores do AIC para a configuração não normal, situações 5 e 6 (sem e com *outliers*), com estimadores clássicos e método de partição *k-means*. Os resultados obtidos, pelo o método *k-means* e estimadores clássicos, encontram-se nas Figuras 5.6 e 5.7. Comparando os resultados obtidos com os valores dos indicadores de simulação apresentados anteriormente, verifica-se que as conclusões estão de acordo com as expectativas, isto é, os menores valores associados ao critério AIC correspondem aos "melhores" valores obtidos com os resultados das simulações. Na situação 6, por exemplo, o AIC atinge os menores valores para os casos em que se consideram 5, 6 e 7 nuvens e os resultados das simulações também apresentam melhores resultados no caso em que se consideram 5 nuvens.

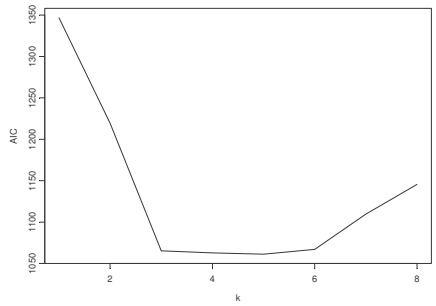


Figura 5.6- Resultados do critério AIC, em função do número de nuvens para a situação 5.

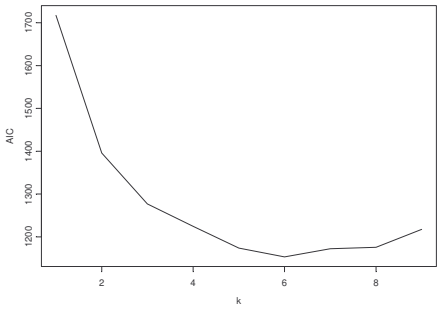


Figura 5.7- Resultados do critério AIC, em função do número de nuvens para a situação 6.

Deste estudo de simulação preliminar parece válido concluir que o método proposto é promissor. No entanto ainda é necessário afinar alguns aspectos. Tentar-se-á experimentar outros estimadores robustos à partida mais eficientes e efectuar um estudo de simulação mais extenso, criando-se novas configurações de dados, para investigar a instabilidade do estimador p_4 e avaliar a *performance* destes métodos em conjuntos de dados maiores e/ou com maior percentagem de contaminação.

5.5 Estudo alargado

Salientada a necessidade de se ir mais longe nos estudos de simulação, decidiu-se introduzir uma notação adequada às várias situações correspondentes às diferentes configurações de dados criadas. Deste modo foram definidas as seguintes designações:

- N / NN para as situações normais e não normais, respectivamente.
- CO/ SO para as situações com e sem *outliers*, respectivamente.
- 2v/3v/4v para as situações com duas, três ou quatro variáveis, respectivamente.

Em relação às configurações, uma vez que se optou por criar um conjunto mais vasto, foram escolhidas as designações (que se procurou serem elucidativas), correspondentes às situações que se passam a apresentar, complementando-se com as representações gráficas correspondentes a uma simulação efectuada para as situações de duas variáveis com *outliers*:

- observações correspondentes a três normais dispostas em forma de lua, correspondendo à situação 6 do primeiro estudo de simulação efectuado (**LUAN3**) e *outliers* com distribuição normal (muito concentrada).

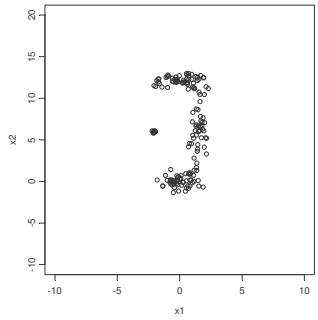


Figura 5.8- Exemplo de uma simulação da configuração LUAN3.

- observações dispostas em cruz e *outliers* com distribuição uniforme (**CRUZU**).

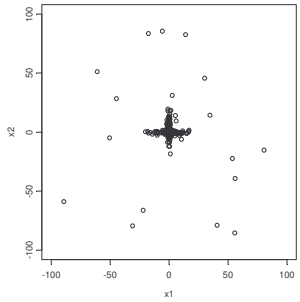


Figura 5.9- Exemplo de uma simulação da configuração CRUZU.

- observações dispostas em cruz e *outliers* com distribuição normal (**CRUZN**).

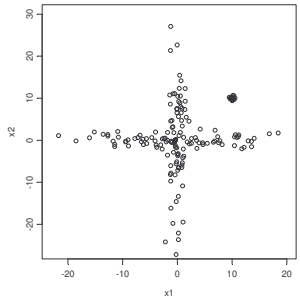


Figura 5.10- Exemplo de uma simulação da configuração CRUZN.

- observações correspondentes a três *t*-Student bivariadas com 3 graus de liberdade, dispostas em forma de lua (**LUAT3**) e *outliers* com distribuição do mesmo tipo mas parâmetros diferentes.

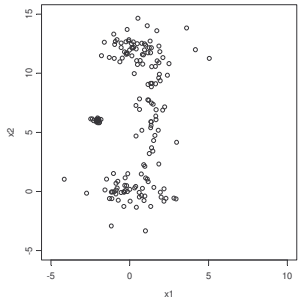


Figura 5.11- Exemplo de uma simulação da configuração LUAT3.

- observações uniformemente distribuídas numa esfera (**ESFERA**) e *outliers* com distribuição normal.

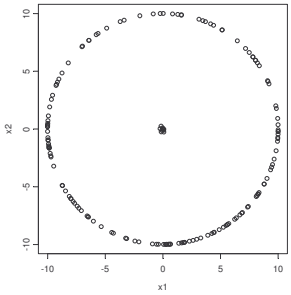


Figura 5.12- Exemplo de uma simulação da configuração ESFERA.

- observações distribuídas uniformemente num "donut" (**DONUT**) e *outliers* com distribuição normal.

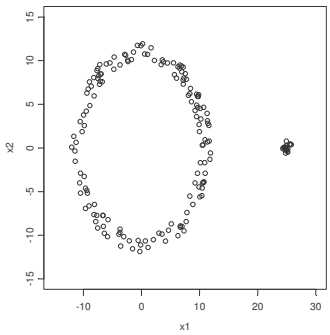


Figura 5.13- Exemplo de uma simulação da configuração DONUT.

- observações com distribuição *t*-Student bivariada com 3 graus de liberdade (**T1**) e *outliers* com distribuição do mesmo tipo mas parâmetros diferentes.

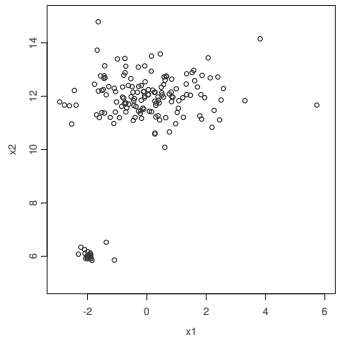


Figura 5.14- Exemplo de uma simulação da configuração T1.

- observações correspondentes a duas t-Student com 3 graus de liberdade (**T2**) e *outliers* com distribuição do mesmo tipo mas parâmetros diferentes.

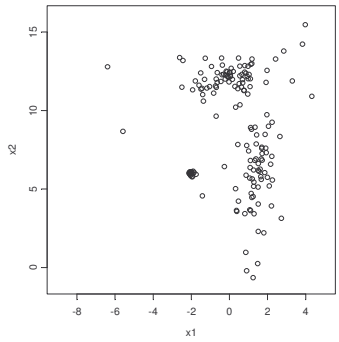


Figura 5.15- Exemplo de uma simulação da configuração T2.

Apresenta-se agora a descrição detalhada de todas as situações distribucionais consideradas neste estudo de simulação com as correspondentes designações:

1. Normal, p=2, sem *outliers*, 150 observações $N_2(\mathbf{0},\mathbf{I})$: **N_SO_2v**.

2. Normal, p=2, com *outliers*, 150 observações $N_2(\mathbf{0},\mathbf{I})$ e 10 observações *outliers* $N_2(\mathbf{10.51},0.1\mathbf{I})$: **N_CO_2v**.

3. Normal, p=4, sem *outliers*, 150 observações $N_4(\mathbf{0},\mathbf{I})$: **N_SO_4v**.

4. Normal, p=4, com *outliers*, 150 observações $N_4(\mathbf{0},\mathbf{I})$ e 10 observações *outliers* $N_4(\mathbf{8.59},0.1\mathbf{I})$: **N_CO_4v**.

5. Não-Normal, p=2, sem *outliers*, 50 observações $N_2(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 50 observações $N_2(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$ e 50 observações $N_2(\mathbf{0},\boldsymbol{\Sigma}_1)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9): **NN_SO_2v_LUAN3**.

6. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_2((-2,6)^T,0.01\mathbf{I})$: **NN_CO_2v_ LUAN3**.

7. Não-Normal, p=4, sem *outliers*, 50 observações $N_4(\boldsymbol{\mu}_3,\boldsymbol{\Sigma}_3)$, 50 observações $N_4(\boldsymbol{\mu}_4,\boldsymbol{\Sigma}_4)$ e 50 observações $N_4(\mathbf{0},\boldsymbol{\Sigma}_3)$, com $\boldsymbol{\mu}_3=(0,12,0,0)^T$, $\boldsymbol{\Sigma}_3$=diag(1,0.3,1,1), $\boldsymbol{\mu}_4=(1.5,6,0,0)^T$ e $\boldsymbol{\Sigma}_4$=diag(0.2,9,1,1): **NN_SO_4v_ LUAN3**.

8. Não-Normal, p=4, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_4((-2,6,0,0)^T, 0.01\mathbf{I})$: **NN_CO_4v_ LUAN3**.

9. Não-Normal , p=2, sem *outliers*, 75 observações $N_2(0,\boldsymbol{\Sigma}_5)$, com $\boldsymbol{\Sigma}_5$=diag(1,81), 75 observações $N_2(0,\boldsymbol{\Sigma}_6)$, com $\boldsymbol{\Sigma}_6$=diag(81,1): **NN_SO_2v_CRUZU**.

10. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* U_2 ([-90,90]× [-90,90]): **NN_CO_2v_ CRUZU**.

11.Não-Normal, p=4, sem *outliers*, 75 observações $N_4(0,\boldsymbol{\Sigma}_7)$, com $\boldsymbol{\Sigma}_7$=diag(1,81,1,1), 75 observações $N_4(0,\boldsymbol{\Sigma}_8)$, com $\boldsymbol{\Sigma}_8$=diag(81,1,1,1): **NN_SO_4v_ CRUZU**.

12. Não-Normal, p=4, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* U_4 ($[-10,10]^2\times [-90,90]^2$) : **NN_CO_4v_ CRUZU**.

13. Não-Normal , p=2, com *outliers*, 75 observações $N_2(0,\boldsymbol{\Sigma}_5)$, com $\boldsymbol{\Sigma}_5$=diag(1,81), 75 observações $N_2(0,\boldsymbol{\Sigma}_6)$, com $\boldsymbol{\Sigma}_6$=diag(81,1) e 20 observações *outliers* $N_2(\mathbf{10},0.1\mathbf{I})$: **NN_CO_2v_CRUZN**.

14.Não-Normal, p=4, com *outliers*, 75 observações $N_4(0,\boldsymbol{\Sigma}_7)$, com $\boldsymbol{\Sigma}_7$=diag(1,81,1,1), 75 observações $N_4(0,\boldsymbol{\Sigma}_8)$, com $\boldsymbol{\Sigma}_8$=diag(81,1,1,1) e 20 observações *outliers* $N_4((10,10,0,0)^T,0.1\mathbf{I})$: **NN_CO_4v_ CRUZN**.

15. Não-Normal , p=2, sem *outliers*, 50 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 50 observações $t_{2,3}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$ e 50 observações $t_{2,3}(\mathbf{0},\boldsymbol{\Sigma}_1)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9): **NN_SO_2v_ LUAT3**. ($t_{p,\upsilon}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ representa a distribuição t-student p-variada com υ graus de liberdade, parâmetro de localização $\boldsymbol{\mu}$ e parâmetro de dispersão $\boldsymbol{\Sigma}$).

16. Não-Normal , p=2, com *outliers*, 50 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 50 observações $t_{2,3}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$ e 50 observações $t_{2,3}(\mathbf{0},\boldsymbol{\Sigma}_1)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9) e 20 observações *outliers* $t_{2,3}((-2,6)^T,0.01\mathbf{I})$: **NN_CO_2v_ LUAT3**.

17. Não-Normal, p=4, sem *outliers*, 50 observações $t_{4,3}(\boldsymbol{\mu}_3,\boldsymbol{\Sigma}_3)$, 50 observações $t_{4,3}(\boldsymbol{\mu}_4,\boldsymbol{\Sigma}_4)$ e 50 observações $t_{4,3}(\mathbf{0},\boldsymbol{\Sigma}_3)$, com $\boldsymbol{\mu}_3=(0,12,0,0)^T$, $\boldsymbol{\Sigma}_3$=diag(1,0.3,1,1), $\boldsymbol{\mu}_4=(1.5,6,0,0)^T$ e $\boldsymbol{\Sigma}_4$=diag(0.2,9,1,1): **NN_SO_4v_LUAT3**.

18. Não-Normal, p=4, com *outliers*, 50 observações $t_{4,3}(\boldsymbol{\mu}_3,\boldsymbol{\Sigma}_3)$, 50 observações $t_{4,3}(\boldsymbol{\mu}_4,\boldsymbol{\Sigma}_4)$ e 50 observações $t_{4,3}(\mathbf{0},\boldsymbol{\Sigma}_3)$, com $\boldsymbol{\mu}_3=(0,12,0,0)^T$, $\boldsymbol{\Sigma}_3$=diag(1,0.3,1,1), $\boldsymbol{\mu}_4=(1.5,6,0,0)^T$ e $\boldsymbol{\Sigma}_4$=diag(0.2,9,1,1) e 20 observações *outliers* $t_{4,3}((-2,6,0,0)^T$, $0.01\mathbf{I})$: **NN_CO_4v_ LUAT3**.

19. Não-Normal, p=2, sem *outliers*, 150 observações uniformemente distribuídas numa esfera de raio 10: **NN_SO_2v_ESFERA**.

20. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_2(\mathbf{0},0.01\mathbf{I})$: **NN_CO_2v_ ESFERA**.

21. Não-Normal, p=4, sem *outliers*, 150 observações uniformemente distribuídas numa hiperesfera de raio 10: **NN_SO_4v_ ESFERA**.

22. Não-Normal, p=4, com *outliers*, 150 observações idênticas à situação anterior e 10 observações *outliers* $N_4(\mathbf{0},0.01\mathbf{I})$: **NN_CO_4v_ ESFERA**.

23. Não-Normal, p=2, sem *outliers,* 150 observações uniformemente distribuídas na intersecção do exterior de uma circunferência de raio 10 com o interior de uma circunferência de raio12 ("donut"): **NN_SO_2v_DONUT**.

24. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* $N_2((25,0)^T,0.1\mathbf{I})$: **NN_CO_2v_ DONUT**.

25. Não-Normal, p=3, sem *outliers*, 150 observações em que as duas primeiras variáveis têm distribuição idêntica à situação anterior e uma terceira variável independente destas com distribuição U [-1,1]: **NN_SO_3v_ DONUT**.

26. Não-Normal, p=3, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* $N_3((25,0,0)^T,0.1\mathbf{I})$: **NN_CO_3v_ DONUT**.

27. Não-Normal, p=4, sem *outliers*, 150 observações em que as duas primeiras variáveis têm distribuição idêntica a **23.** e mais duas variáveis independentes com distribuição U_2 $([-1,1]^2)$: **NN_SO_4v_DONUT**.

28. Não-Normal, p=4, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* $N_4((25,0,0,0)^T,0.1\mathbf{I})$: **NN_CO_4v_DONUT**.

29. Não-Normal, p=2, sem *outliers*, 150 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3): **NN_SO_2v_T1**.

30. Não-Normal, p=2, com *outliers*, 150 observações idênticas à situação anterior e 20 observações *outliers* $t_{2,3}((-2,6)^T,0.01\mathbf{I})$: **NN_CO_2v_T1**.

31. Não-Normal, p=2, sem *outliers*, 75 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 75 observações $t_{2,3}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9): **NN_SO_2v_T2**.

32. Não-Normal, p=2, com *outliers*, 75 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, 75 observações $t_{2,3}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$, com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9) e 20 observações *outliers* $t_{2,3}((-2,6)^T,0.01\mathbf{I})$: **NN_CO_2v_T2**.

A escolha destas configurações não obedeceu a nenhum critério definido à partida. A única preocupação com a sua escolha foi a de fixar claramente o objectivo de averiguar se o método proposto consegue lidar eficazmente com distribuições não elipticamente simétricas. Exceptua-se o caso da normal que surge como opção natural com o objectivo apenas de evidenciar que o método também funciona em situações usuais. As ideias que deram forma às configurações adoptadas foram surgindo naturalmente com a evolução do trabalho de investigação. Assim, por exemplo, a ideia da configuração CRUZU foi retirada do trabalho de Fraley e Raftery (1999) e por consequência também surgiu a configuração CRUZN. A configuração LUAT3 surgiu de uma sugestão de Christian Hennig quando lhe foi pedido que se pronunciasse sobre o artigo Santos-Pereira e Pires (2002). Transcreve-se de seguida parte do seu comentário: "*…Perhaps one should think about worst case behavior, test data from mixture distributions of other than Normal shape (multivariate t, something skew...) and so on. There should be a comparison of the method to the "noise component" approach of Fraley, Raftery and co-workers, and perhaps also to the mixtures of t-distributions with outlier identification as explained in the mixture model book of McLachlan and Peel*".

Para terminar, quer a ideia da ESFERA quer o DONUT surgiram do trabalho de Billor *et al*. (2000). Depois de escolhidas as configurações, optou-se por variar alguns aspectos tais como o número de observações, o número de variáveis e diferentes percentagens de contaminação.

Nesta segunda fase do estudo de simulação interessa, em primeiro lugar, destacar as principais alterações efectuadas relativamente ao estudo preliminar:

- redução do número de simulações de 1000 para 100.
- introdução do novo estimador robusto: OGK.
- criação de um indicador que permitirá reconhecer qual a melhor combinação em termos de método de *clustering*, número de nuvens e estimador.

Foram também feitas outras tentativas, nomeadamente:

- simulação do dobro dos dados.
- aumento das percentagens de contaminação em algumas configurações.
- experimentação de outra constante de detecção.
- utilização do critério AIC para a escolha da "melhor combinação".

Apesar de ser vantajoso efectuar o maior número de simulações possível por forma a obter estimativas mais fiáveis, pondo a questão de outro modo parece ser lícito a redução proposta de 1000 para 100. Isto porque, nos vários casos apresentados no estudo preliminar (situações 1 a 8) foram efectuadas 100 (ver Santos-Pereira e Pires, 2002) e 1000 simulações tendo-se verificado que os resultados eram semelhantes. Mais ainda, ao efectuar 100 simulações para cada situação diminuiu-se o tempo de cálculo de cada uma permitindo considerar mais situações e diferentes configurações de dados. Note-se ainda que o erro padrão das estimativas das proporções (indicadores de simulação), cerca de 3%, associado às 100 simulações e ao valor de $\alpha = 0.1$, também parece razoável.

A escolha dos estimadores é óbvia e vem no seguimento do estudo preliminar e das considerações feitas na Secção 4.2 do Capítulo 4. Passarão a ser objecto de comparação, os estimadores clássicos e, como estimadores robustos, o OGK e RMCD25. Um outro argumento que presidiu à escolha de um outro estimador robusto é fruto dos resultados do estudo de simulação preliminar onde em determinadas situações os estimadores clássicos apresentaram um melhor comportamento. Claro que esses resultados não são tão inverosímeis quanto poderão parecer à primeira vista pelo facto de se estar a utilizar o RMCD25 a par da partição dos dados em nuvens.

A grande quantidade de tabelas e/ou gráficos obtidos a partir das simulações associadas às várias configurações de dados adoptadas (que justifica a sua colocação no Apêndice D), bem como a grande quantidade de informação resultante, apela agora para a necessidade de adopção de um determinado critério, ou medida, que permita condensar toda essa informação e que ao mesmo tempo permita comparar a *performance* das várias combinações, em termos de método de *clustering*, número de nuvens e estimador para cada par de situações, com e sem *outliers* (pois à partida desconhece-se se estes existem ou não). De salientar que o indicador que irá proposto e adoptado nas simulações para satisfazer este objectivo permite, não só, comparar as várias combinações como também validar o desempenho do próprio método.

Originalmente começou por se pensar num indicador que condensasse a informação dos quatro indicadores de simulação. Feitos alguns testes abandonou-se a ideia. A par de algumas outras deficiências- a principal das quais a incapacidade deste indicador traduzir a "qualidade" das composições- não faria sentido juntar quantidades diferentes já que os indicadores p_2 e p_3 representam médias e p_1 e p_4 proporções. Por estas razões e também pela grande instabilidade do indicador p_4, como já se teve oportunidade de demonstrar, propõe-se o seguinte indicador, daqui em diante designado por DIF:

$$\text{DIF} = \begin{cases} \dfrac{(p_2 - 0.05)_+ + (p_3 - 0.05)_+}{2} \times 100 & \text{para dados com } \textit{outliers} \\ (p_3 - 0.05)_+ \times 100 & \text{para dados sem } \textit{outliers}. \end{cases}$$

$$\text{em que } (x)_+ = \begin{cases} 0 & \text{se } x \le 0 \\ x & \text{se } x > 0 \end{cases}.$$

Note-se que este indicador varia entre 0 e 95 e uma boa combinação deve resultar num DIF=0. Importa também chamar a atenção para o facto de só se poder considerar "boa", uma combinação que forneça um bom resultado para os dados com e sem *outliers*. O valor 0.05 não tem que ser o valor de α, uma vez que se relaciona apenas com p_2 e p_3, representa sim uma tolerância que se considera razoável em relação ao valor 0.

Os exemplos seguintes ajudam a compreender a instabilidade do indicador p_4.

Exemplo 5.1 Considere-se uma amostra de 100 observações sem *outliers*. Admita-se que se realizaram 100 simulações entre as quais em 10 o método identificou 1 *outlier* e nas restantes 0 *outliers*. Neste caso obtém-se p_3=0.001 e p_4=0.9.

Exemplo 5.2 Considere-se uma amostra de 100 observações sem *outliers*. Admita-se que se realizaram 100 simulações entre as quais em 10 o método identificou 10 *outliers* e nas restantes 0 *outliers*. Neste caso obtém-se p_3=0.01 e p_4 mantém-se em 0.9.

Exemplo 5.3 Considere-se uma amostra de 100 observações sem *outliers*. Admita-se que se realizaram 100 simulações e que em cada uma deles o método identificou 1 *outlier*. Neste caso obtém-se p_3=0.01 e p_4=0.

Estes exemplos ajudam a entender determinados valores que resultam das simulações efectuadas e justificam a adopção de 0.05 em vez do valor adoptado para α. Embora se tenha adoptado α=0.10 os exemplos demonstram que por vezes este valor pode ser bastante menor.

Em relação à constante que funciona como limite de detecção verificou-se que em determinadas situações o Qui-quadrado tinha o defeito de detectar *outliers* a mais (o que aliás vem de encontro ao trabalho de Rousseeuw e van Zomeren, 1991) e a constante $c(p,n,\alpha_n)$ de Davies e Gather (1993) *outliers* a menos. Foi então experimentada uma outra constante[4] c^*, baseada na distribuição F, referida em Becker e Gather (1999, pág. 948) e que segundo estas autores constitui uma escolha apropriada para amostras de pequena dimensão. No entanto a utilização desta constante não permitiu qualquer melhoria nos resultados. Pelo contrário, piorou na medida em que aumentou o número de *outliers* detectados já que os seus valores se apresentam inferiores aos do Qui-quadrado. Experimentou-se também a distribuição *F* sugerida por Hardin e Rocke (1999, pág. 12) mas os resultados foram idênticos.

[4] $c^* = \dfrac{p(n-1)^2 F_{p,n-p-1;1-\alpha/n}}{n(n-p-1+pF_{p,n-p-1;1-\alpha/n})}$

A introdução do estimador robusto OGK neste estudo levou a que tivessem também de ser calculadas novas constantes baseadas na proposta de Becker e Gather (2001) referida na Secção 4.4 do Capítulo 4. À partida, parecia que se tinha encontrado a solução ideal já que os seus valores se encontravam entre os do Qui-quadrado e os das constante associadas ao RMCD25. E realmente, para dimensões de amostras pequenas (inferiores a 50, aproximadamente) é isso que acontece. Por exemplo para n=20, p=2, k=1 e α=0.1 obtém-se $\chi^2_{p,1-\alpha}$ = 10.49746, c_{MCD} =85.58786 e c_{OGK} =24.84670. No entanto, para dimensões superiores, as constantes associadas ao OGK são ligeiramente superiores às do RMCD25. Na prática isto traduz-se num bom desempenho do estimador OGK embora não superior à *performance* do RMCD25, conforme se verá de seguida na análise dos resultados associados às várias configurações de dados. Com o estimador OGK foram calculadas constantes para α=0.1, k=1,2,...,7 , p=2,3,4 e n_k=5,10,...,250 e com o RMCD25[5] foram efectuados mais alguns cálculos tendo-se acesso ás constantes associadas a α=0.1,0.15,0.2, k=1,2,...,7, p=2,3,4 e n_k=5,10,...,250. Na prática acabou por não se trabalhar com α=0.15 e α=0.20 uma vez que os resultados com 0.1 já foram satisfatórios. Por essa razão alguns dos valores calculados não foram incluídos nos programas. Foi também introduzida nos programas, que envolvem estimadores robustos, uma opção que para valores de n_k superiores a 250 considera o valor da aproximação à distribuição Qui-quadrado.

Ainda em relação ao estimador OGK, importa aqui destacar que a versão utilizada foi a reponderada com duas iterações, $\mathrm{OGK}_{(2)}(0.9)$, que segundo Maronna e Zamar (2002) é uma versão melhor e "mais equivariante", tendo demonstrado um bom desempenho, quer nas simulações, quer nos dados reais, por aqueles autores utilizados. O programa que permite calcular as estimativas correspondentes encontra-se no Apêndice C, conforme já referido no Capítulo 4. Note-se que, embora uma das grandes vantagens deste estimador seja o seu tempo de processamento, não se teve neste trabalho a preocupação de optimizar o programa que permite a sua obtenção pelo que a comparação dos tempos de cálculo entre este estimador e o RMCD25 além de ser irrelevante para o estudo em causa, não faria qualquer sentido.

[5] Nalgumas partes desta secção substituir-se-á a sigla RMCD25 apenas por MCD por uma questão de simplificação.

Na Secção 5.4 exemplificou-se a utilização do critério AIC, com a configuração LUAN3, com estimadores e método de *clustering* fixo, apenas para comparação dos vários valores de *k*. Nesta secção vai-se mais longe experimentando-se também o critério AIC, mas neste caso para a detecção da melhor combinação em termos de método de *clustering*/estimador/número de nuvens, a título de exemplo, para a configuração LUAN3. A utilização deste critério poderá revelar-se bastante útil nomeadamente para escolha da "melhor" combinação em dados reais. Note-se que, enquanto que em dados simulados se pode recorrer a indicadores de simulação, como por exemplo os p_i´s apresentados na Secção 5.3 ou o indicador DIF proposto na Secção 5.5, com dados reais isso não é possível.

Um outro aspecto que merece alguma atenção, na medida em que é tido muito em conta neste tipo de estudos e pode influenciar os resultados, é a percentagem de contaminação. Neste trabalho optou-se por utilizar percentagens de contaminação no seguimento da linha traçada por Hampel *et al*. (1986, Cap. 1, pág. 26) que se traduz na seguinte afirmação: "*Crudely speaking, one has to distinguish between high-quality data with no or virtually gross errors, and routine data, with about 1-10% or more gross errors*". Assim, para a maior parte das situações considerou-se uma percentagem de contaminação de 6,25% ou de 11,8%. Para verificar se existiam alterações nos resultados, com o aumento da percentagem de contaminação, foram escolhidas duas configurações nas quais se experimentaram três valores de contaminação diferentes. Julga-se no entanto importante referir que não é a percentagem de contaminação por si só que pode influenciar o desempenho do método utilizado. Tem também que se ter em conta a forma e a localização dos *outliers*. Só como pequeno exemplo, na posse de um conjunto de dados normais ou mesmo não normais, em que os *outliers* se encontram posicionados a uma grande distância dos restantes, é sempre possível conseguir-se uma detecção dos mesmos. Na prática é habitual dizer-se que existe uma grande percentagem de contaminação quando esta varia entre 35-45% (ver por exemplo Kosinski, 1999 ou Rocke e Woodruff, 1996). No entanto situações deste tipo são consideradas irrealistas pelo que não se considerou ter interesse entrar em detalhes desse tipo, assumindo-se no geral situações rotineiras. Porém, embora se tenham considerado contaminações da ordem da grandeza referida, a título de curiosidade experimentou-se para a configuração LUAN3 uma contaminação de cerca de 29% (60 *outliers* num total de 210 observações) tendo-se verificado que os resultados não se alteravam. Concluiu-

se assim que os valores de contaminação adoptados, correspondentes a contaminações realistas habituais, são suficientes para demonstrar o bom desempenho do método, independentemente das percentagens de contaminação.

Para terminar, de referir que também se experimentou o método de *clustering* designado por *fanny* (referido na Secção 4.5) mas como não deu bons resultados para as configurações adoptadas, optou-se por não incluir neste trabalho os valores obtidos.

5.6 Análise dos resultados

Apresentam-se de seguida, para cada configuração de dados, os resultados do indicador de qualidade da combinação, DIF, e respectivas conclusões, gráficos correspondentes a uma simulação para a "melhor" e "pior" combinação encontrada no caso bivariado e outros aspectos relevantes. A escolha da "melhor"/"pior" combinação baseou-se em primeiro lugar na informação fornecida pelo indicador DIF. Em caso de empate atendeu-se aos indicadores de simulação p_1, p_2, p_3 e p_4. Em situações idênticas foi escolhida uma combinação arbitrariamente. Nas tabelas apresentadas a "melhor" combinação é representada a ***bold*** e a "pior" combinação também a *bold* mas com a célula correspondente a **sombreado.** Para não sobrecarregar demasiado esta secção, criou-se o Apêndice D no qual se encontram as Tabelas D.1 a D.14 correspondentes aos valores obtidos para os indicadores de simulação, relativos às várias configurações e algumas variações também consideradas neste trabalho.

Considera-se também ter interesse justificar o número de casas decimais adoptadas para os indicadores de simulação. Como p_1 e p_4 representam percentagens naturalmente que os valores destes indicadores aparecem com duas casas decimais. Por outro lado, visto que na maior parte das situações (a configuração CRUZU constitui a excepção) p_2 é aproximadamente igual a 1- p_1, decidiu-se considerar também duas casas decimais para este indicador. No entanto no caso do indicador p_3 optou-se por adoptar mais uma casa decimal uma vez que os seus valores são geralmente baixos pelo que a sua aproximação a duas casas decimais conduziria muitas vezes a um arredondamento ao valor 0 perdendo-se alguma informação.

NORMAL

Conforme se constata por observação das Tabelas 5.3 e D.1, relativamente a esta configuração, praticamente que não há nada a acrescentar relativamente à análise efectuada no estudo preliminar. A inclusão do estimador OGK no estudo de simulação não alterou as conclusões já obtidas. Qualquer que seja o estimador, método de *clustering* e número de nuvens utilizado, o método mostra um bom desempenho para a configuração normal. Exclui-se a situação onde não são consideradas nuvens e se opta pela utilização de estimadores clássicos.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	0.0	0.0	0.0
N	"	**4**	0.0	0.0	0.0	**N**	"	**4**	0.0	0.0	0.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	0.0	0.0	0.0
2v	***pam***	**3**	0.0	0.0	0.0	**2v**	***pam***	**3**	0.0	0.0	0.0
	"	**4**	0.0	0.0	0.0		"	**4**	0.0	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.0	0.0	0.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	**0.0**	0.0	0.0
	"	**4**	0.0	0.0	0.0		"	**4**	0.0	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.0	0.0	0.0
	sem nuvens	**1**	0.0	0.0	0.0		**sem nuvens**	**1**	0.0	0.0	**5.5**
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	0.0	0.0	0.0
N	"	**4**	0.0	0.0	0.0	**N**	"	**4**	0.0	0.0	0.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	0.0	0.0	0.0
4v	***pam***	**3**	0.0	0.0	0.0	**4v**	***pam***	**3**	0.0	0.0	0.0
	"	**4**	0.0	0.0	0.0		"	**4**	0.0	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.0	0.0	0.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	0.0	0.0	0.0
	"	**4**	0.0	0.0	0.0		"	**4**	0.0	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.0	0.0	0.0
	sem nuvens	**1**	0.0	0.0	0.0		**sem nuvens**	**1**	0.0	0.0	**17.5**

Tabela 5.3- Resultados do indicador DIF para a configuração normal.

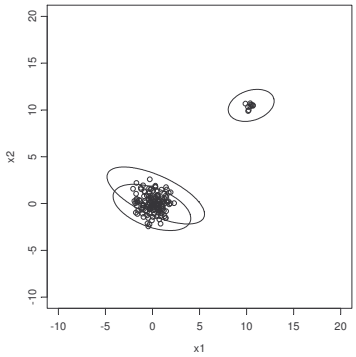


Figura 5.16- Representação da "melhor" combinação para a configuração Normal.

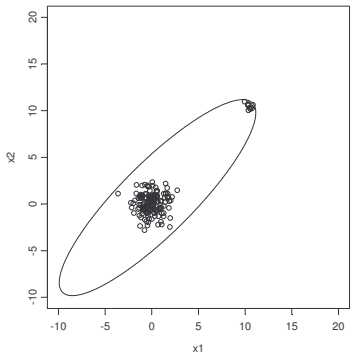


Figura 5.17- Representação da "pior" combinação para a configuração Normal.

LUAN3

Dos resultados apresentados nas Tabelas 5.4 e D.2 constata-se que de uma forma genérica nada poderá ser dito acerca do tipo de estimador que apresenta um melhor desempenho nesta configuração; todos os três tipos apresentam-se como melhores em determinadas situações particulares como por exemplo, na situação bivariada, o MCD se for utilizado o *mclust* com 3 nuvens, o OGK se for utilizado o *pam* com 4 nuvens ou o clássico para o caso do *k-means* com 5 nuvens. No entanto, tendo em conta que a situação deve ser analisada com e sem *outliers* (pois à partida desconhece-se se estes existem ou não), o MCD parece ser o estimador com pior desempenho. No caso particular da situação com 4 variáveis os estimadores clássicos apresentam um melhor desempenho que os robustos. Bem evidente é a superioridade da RRO aquando da utilização dos métodos de *clustering* e por consequência a partição dos dados em nuvens.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	14.9	11.3	1.3		***k-means***	**3**	8.7	13.4	40.5
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	31.2	27.0	48.4
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	22.5	25.0	22.0
2v	***pam***	**3**	14.3	7.4	0.0	**2v**	***pam***	**3**	10.9	21.8	39.5
LUAN3	"	**4**	0.0	0.0	0.0	**LUAN3**	"	**4**	11.5	7.2	8.6
	"	**5**	0.0	0.0	0.0		"	**5**	6.0	9.5	0.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	37.2	43.5	39.0
	"	**4**	0.0	0.0	0.0		"	**4**	**0.0**	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.5	3.0	0.0
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	**47.5**	47.5	47.5
	k-means	**3**	13.1	0.1	0.0		***k-means***	**3**	17.3	35.9	44.0
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	41.0	46.0	48.2
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	39.5	43.0	34.5
4v	***pam***	**3**	7.4	0.1	0.0	**4v**	***pam***	**3**	20.4	38.5	45.5
LUAN3	"	**4**	0.0	0.0	0.0	**LUAN3**	"	**4**	7.6	3.0	2.5
	"	**5**	0.0	0.0	0.0		"	**5**	10.5	21.0	0.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	48.1	49.9	44.6
	"	**4**	0.0	0.0	0.0		"	**4**	**0.0**	0.0	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	0.5	9.0	0.0
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	47.5	47.5	**47.5**

Tabela 5.4- Resultados do indicador DIF para a configuração LUAN3.

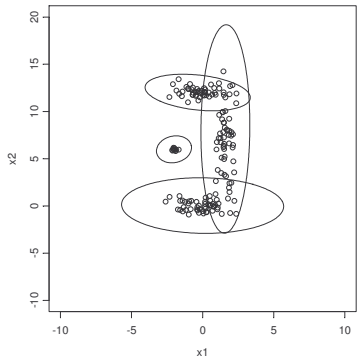


Figura 5.18- Representação da "melhor" combinação para a configuração LUAN3.

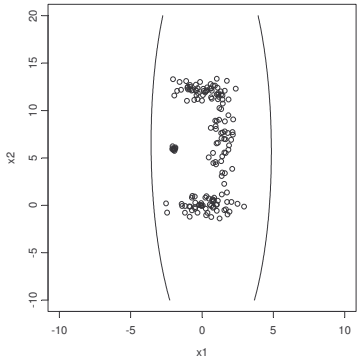


Figura 5.19- Representação da "pior" combinação para a configuração LUAN3.

Na Secção 5.4 experimentou-se o critério AIC para esta configuração com o intuito de se encontrar a melhor combinação em termos de método de *clustering*/estimador/ número de nuvens. No entanto, apenas se procedeu a uma simulação pelo que as conclusões obtidas não poderão ser consideradas fiáveis. Por outro lado, como também foi referido, em situações reais onde não é possível a utilização de indicadores de simulação surge a necessidade de alguma forma se proceder a um "validação" do procedimento AIC. Para isso calculou-se para a configuração LUAN3 na situação de 2 variáveis com *outliers*, a média dos valores do AIC correspondentes às 100 simulações, tendo-se obtido os resultados que se passam a apresentar na Tabela 5.5.

	k	MCD	Clássicos
k-means	**2**	1868.662	1395.609
	3	1675.131	1276.821
"	**4**	1327.665	1224.542
"	**5**	1187.077	1173.442
	6	1480.513	1153.179
	7	1349.737	1172.183
	8	1471.515	1175.775
pam	**2**	1907.77	1438.401
	3	1684.203	1278.618
"	**4**	1173.771	1135.177
	5	1288.149	1100.274
	6	1385.863	1158.871
"	**7**	1251.556	1143.372
	8	1336.663	1161.403
mclust	**2**	1555.53	1364.72
	3	1646.997	1286.301
"	**4**	1270.661	1179.698
"	**5**	1171.925	**1088.645**
	6	1234.926	1162.001
	7	1184.255	1168.04
	8	1317.725	1155.206
sem nuvens	**1**	1505.829	1717.534

Tabela 5.5- Resultados do critério AIC para a situação NN_CO_2v_LUAN3.

Comparando os resultados fornecidos pela Tabela 5.5[6] com os resultados apresentados na Tabela D.2, correspondente à mesma situação mas onde são fornecidos os p_i´s constata-se que as conclusões são idênticas; à parte uma ou outra excepção, os menores valores obtidos no critério AIC correspondem praticamente aos menores valores correspondentes ao indicador DIF. Na necessidade de se proceder a uma escolha da melhor combinação, o critério AIC indicaria o método *k-means* com estimadores clássicos e 5 nuvens o que corresponde a uma das hipótese possíveis de acordo com o

[6] Exclui-se aqui o estimador OGK por não se achar relevante para o que se pretende "validar".

indicador DIF. Aliás, tendo em conta não só o valor de DIF como os próprios valores dos indicadores de simulação, esta seria com certeza a melhor combinação. Feitas estas considerações, e tendo em conta que se está a trabalhar com dados simulados, julga-se que este pequeno exemplo poderá "validar" de certa forma a utilização do critério AIC para situações reais. Insere-se no Apêndice D um dos programas elaborados para obtenção do valor do AIC para a situação onde é utilizado o *k-means* com estimadores robustos MCD.

CRUZU

Dada a configuração em causa e mais precisamente pela disposição dos *outliers* ser uniforme, os valores de p_1 já eram esperados: é praticamente impossível existir uma simulação em que sejam detectados todos os *outliers*, nomeadamente para a situação bivariada. No entanto, em dimensão quatro, como já era de esperar, a situação melhora. Pela mesma razão, nesta configuração particular em que os *outliers* não estão concentrados (como em todas as outras configurações aqui apresentadas) os valores do indicador p_2 também já não se encontram próximos de $1-p_1$; assim como é natural que não sejam detectados todos os *outliers*, também é provável que a média das proporções de *outliers* não identificados em cada simulação também seja maior que o habitual pelo que numa situação como esta até é de esperar que os valores de p_1 e p_2 sejam relativamente próximos ao contrário das outras configurações em que são aproximadamente complementares.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	11.9	13.6	5.5
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	4.8	1.4	0.5
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	3.7	0.4	**0.0**
2v	***pam***	**3**	0.0	0.0	0.0	**2v**	***pam***	**3**	20.6	19.2	15.6
CRUZU	"	**4**	0.0	0.0	5.0	**CRUZU**	"	**4**	8.9	8.5	14.4
	"	**5**	0.0	0.0	0.0		"	**5**	7.1	10.1	9.8
	mclust	**3**	0.0	0.0	5.0		***mclust***	**3**	**33.6**	30.7	32.5
	"	**4**	0.0	0.0	0.0		"	**4**	31.5	31.3	27.6
	"	**5**	0.0	0.0	0.0		"	**5**	29.6	27.0	18.7
	sem nuvens	**1**	16.9	1.7	0.0		sem nuvens	**1**	8.9	1.1	0.6
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	0.0	0.0	0.0
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	0.0	0.0	0.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	0.3	0.0	**0.0**
4v	***pam***	**3**	0.0	0.0	0.0	**4v**	***pam***	**3**	0.9	0.0	2.5
CRUZU	"	**4**	0.0	0.0	0.0	**CRUZU**	"	**4**	0.0	0.0	4.8
	"	**5**	0.0	0.0	0.0		"	**5**	0.0	0.0	4.1
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	21.8	22.2	19.2
	"	**4**	0.0	0.0	0.0		"	**4**	**26.6**	20.8	16.7
	"	**5**	0.0	0.0	0.0		"	**5**	24.0	20.7	8.0
	sem nuvens	**1**	7.6	0.0	0.0		sem nuvens	**1**	3.4	0.0	0.0

Tabela 5.6- Resultados do indicador DIF para a configuração CRUZU.

De realçar também que os estimadores clássicos demonstram melhor desempenho do que os robustos. À partida não é o que se esperaria dada a complexidade e irregularidade da própria configuração, em oposição, por exemplo, à configuração LUAN3. No entanto esta constatação fica de certa forma explicada pelo facto de se estar a trabalhar com os dados agrupados em nuvens. E embora pela análise isolada da Tabela 5.6 esta afirmação possa parecer um pouco despropositada, os resultados apresentados na Tabela D.3 evidenciam um melhor desempenho dos métodos robustos para detecção dos *outliers* na situação sem nuvens, relativamente aos estimadores clássicos bastando para isso que se observem os valores de p_1 e p_2.

Um outro aspecto que importa também realçar é a superioridade do desempenho do estimador OGK relativamente ao MCD. De acordo com os resultados da Tabela 5.6 verifica-se que o desempenho do OGK se situa entre o dos estimadores clássicos e o do MCD.

Uma vez que esta configuração difere bastante das outras, relativamente à disposição dos *outliers*, experimentaram-se também outras percentagens de contaminação para avaliação do desempenho do método. Para o caso particular de duas variáveis os resultados dos estimadores robustos pioram um pouco. Por exemplo, enquanto que na situação com 6,25% de contaminação, com o OGK se conseguia um bom desempenho escolhendo o *k-means* com 3 nuvens, aumentando essa percentagem para cerca de 16,7% o resultado altera-se completamente. No entanto quer os resultados dos estimadores clássicos quer os da situação em que se trabalha com quatro variáveis, não parecem ser significativamente abaladas pela percentagem de contaminação.

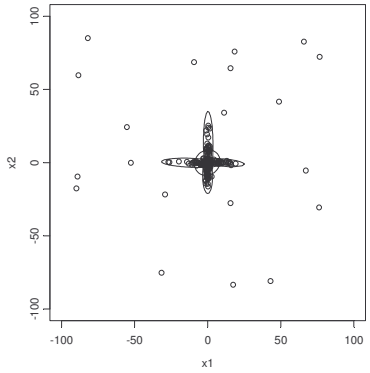


Figura 5.20- Representação da "melhor" combinação para a configuração CRUZU.

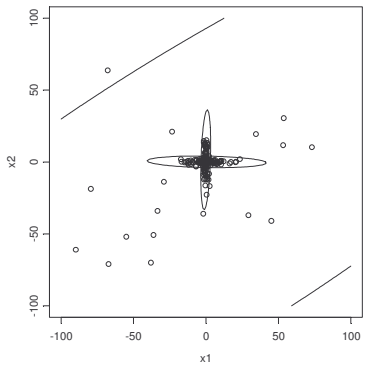


Figura 5.21- Representação da "pior" combinação para a configuração CRUZU.

A título de exemplo, apresentam-se os tempos que os programas demoraram a correr para 100 simulações com o método de *clustering pam* com 3, 4 e 5 nuvens, para os estimadores utilizados (clássicos, RMCD25 e OGK) num computador com processador Pentium 4 1.8 GHz com 512 MB de memória no programa S-plus, versão 4.5. De salientar, no entanto, que tal como já foi referido na secção anterior, os programas não foram optimizados por forma a se minimizar o tempo de processamento. A razão de neste momento se optar por referir os tempos prende-se não com a comparação entre estimadores mas como uma referência ao tempo gasto, contribuindo para cimentar a justificação da necessidade de se proceder a um menor número de simulações, em benefício do enriquecimento do trabalho efectuado pela utilização de maior número de configurações e da referência a outros aspectos relevantes.

	pam* com 10 *outliers	***pam* com 30 *outliers***
Clássicos	40m	42m 19s
MCD	44m 17s	1h 7m 23s
OGK	48m	2h 20m 1s

Tabela 5.7- Alguns tempos de simulação para a configuração CRUZU.

CRUZN

Neste caso os resultados da aplicação da RRO estão fortemente dependentes do método de *clustering* escolhido. No que toca ao número de variáveis, duas ou quatro, verifica-se que em ambas as situações quer o *k-means*, quer o *pam* funcionam mal. No caso de duas variáveis o *mclust* apenas fornece resultados satisfatórios para os estimadores robustos. Quando o número de variáveis aumenta para quatro a situação já não se mantém, a RRO consegue já ter boa *performance* tanto em combinações onde intervêm os estimadores clássicos como os robustos.

Mais uma vez se confirma a superioridade do método RRO relativamente às situações em que não se consideram nuvens.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	47.0	47.5	47.5
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	45.0	44.2	46.5
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	39.1	39.0	43.5
2v	***pam***	**3**	0.0	0.0	0.0	**2v**	***pam***	**3**	47.5	47.5	47.5
CRUZN	"	**4**	0.0	0.0	5.0	**CRUZN**	"	**4**	46.0	44.7	46.5
	"	**5**	0.0	0.0	0.0		"	**5**	32.5	40.0	42.5
	mclust	**3**	0.0	0.0	5.0		***mclust***	**3**	**0.0**	2.0	9.0
	"	**4**	0.0	0.0	0.0		"	**4**	3.0	2.0	10.5
	"	**5**	0.0	0.0	0.0		"	**5**	4.0	0.5	5.5
	sem nuvens	**1**	16.9	0.9	0.0		sem nuvens	**1**	41.5	47.5	**47.5**
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	47.5	47.5	47.5
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	47.5	47.5	47.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	46.0	45.5	44.0
4v	***pam***	**3**	0.0	0.0	0.0	**4v**	***pam***	**3**	47.5	47.5	47.5
CRUZN	"	**4**	0.0	0.0	0.0	**CRUZN**	"	**4**	46.0	46.5	47.0
	"	**5**	0.0	0.0	0.0		"	**5**	46.0	44.5	44.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	**0.0**	3.0	6.0
	"	**4**	0.0	0.0	0.0		"	**4**	7.5	3.5	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	19.0	4.0	0.0
	sem nuvens	**1**	7.6	1.0	0.0		sem nuvens	**1**	47.5	47.5	**47.5**

Tabela 5.7- Resultados do indicador DIF para a configuração CRUZN.

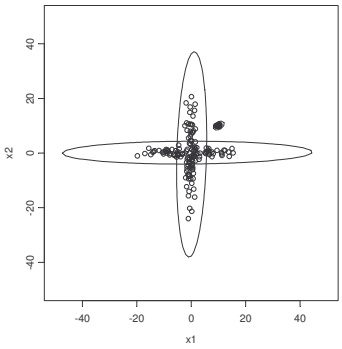


Figura 5.22- Representação da "melhor" combinação para a configuração CRUZN.

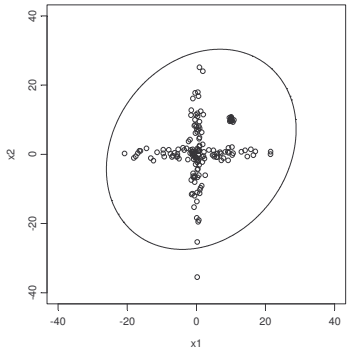


Figura 5.23- Representação da "pior" combinação para a configuração CRUZN.

LUAT3

Há que ter em atenção que os indicadores de simulação aqui apresentados, em especial p_3 e p_4, não são comparáveis com os correspondentes utilizados para a configuração LUAN3. Estes indicadores não são possivelmente os mais indicados para traduzir esta configuração pois não é possível saber-se, à partida, quantos e quais serão os *outliers* "fabricados" e os *outliers* gerados pela própria distribuição. Basta reparar, por exemplo, nos resultados dos indicadores p_3 e p_4 dos estimadores clássicos para a situação NN_SO_2v_LUAT3 para se concluir que existem *outliers* mesmo quando não são

"fabricados". Tal constatação implica evidentemente uma diminuição do indicador p_1 e um aumento de p_2, indicadores que por sua vez já são comparáveis com os correspondentes na situação LUAN3.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	8.4	7.9	0.0		***k-means***	**3**	49.5	39.5	**47.0**
NN	"	**4**	2.1	0.5	0.0	**NN**	"	**4**	37.9	40.6	45.7
SO	"	**5**	0.9	0.5	0.0	**CO**	"	**5**	34.2	34.0	27.5
2v	***pam***	**3**	0.0	1.6	0.0	**2v**	***pam***	**3**	58.9	43.3	45.8
LUAT3	"	**4**	0.0	0.0	0.9	**LUAT3**	"	**4**	10.9	12.8	10.0
	"	**5**	0.0	0.0	0.0		"	**5**	27.1	24.6	16.7
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	43.9	48.3	49.4
	"	**4**	0.0	0.0	0.0		"	**4**	**0.3**	1.6	20.1
	"	**5**	0.0	0.0	0.0		"	**5**	36.7	32.3	28.1
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	47.5	47.5	47.5
	k-means	**3**	1.1	2.4	0.0		***k-means***	**3**	48.0	60.2	48.8
NN	"	**4**	2.7	3.4	1.1	**NN**	"	**4**	49.7	50.2	45.1
SO	"	**5**	0.0	5.1	5.2	**CO**	"	**5**	49.5	51.1	38.5
4v	***pam***	**3**	0.0	0.0	0.0	**4v**	***pam***	**3**	54.0	**62.5**	50.5
LUAT3	"	**4**	0.4	2.7	0.0	**LUAT3**	"	**4**	27.6	12.4	11.5
	"	**5**	0.1	2.3	0.0		"	**5**	41.0	32.1	16.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	46.5	48.4	50.0
	"	**4**	0.0	0.0	0.0		"	**4**	22.6	15.3	34.0
	"	**5**	0.0	0.0	0.0		"	**5**	46.0	42.3	39.5
	sem nuvens	**1**	1.4	0.9	0.0		sem nuvens	**1**	49.7	49.3	47.5

Tabela 5.8- Resultados do indicador DIF para a configuração LUAT3.

Pelo facto de só existir uma combinação onde a RRO funciona bem, embora o próprio valor do DIF ultrapasse um pouco a condição imposta à partida, decidiu-se experimentar outras percentagens de contaminação (consideração de 10, 20 ou 30 *outliers*). Claro que numa situação como esta, onde é natural que exista uma confusão entre o que são *outliers* "fabricados" e os reais (no fundo entre os *outliers* que são observações contaminantes e discordantes, no sentido mais usual dos termos), estes resultados já eram previstos.

Para o caso em que o número de variáveis é quatro, a RRO tem um mau desempenho em qualquer uma das combinações e para qualquer uma das três percentagens de contaminação experimentadas (6,25%, 11,8% e 16,7%). Na situação bivariada, para qualquer um dos três níveis de contaminação adoptados, apenas existe uma combinação que se destaca das restantes: *mclust*/*k*=4 com estimadores robustos. Os diferentes níveis de contaminação não parecem trazer nada de significativo relativamente às conclusões associadas à *performance* do método. Para uma contaminação de 6,25% a RRO não funciona. No entanto com 11,8% ou 16,7% já funciona relativamente bem para a combinação referida havendo apenas uma "troca"

relativamente ao estimador robusto a ser utilizado: MCD ou OGK consoante se tem 11,8% ou 16,7% de contaminação, respectivamente.

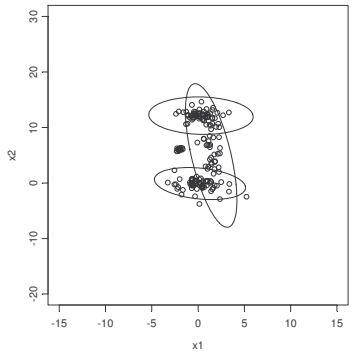


Figura 5.24- Representação da "melhor" combinação para a configuração LUAT3.

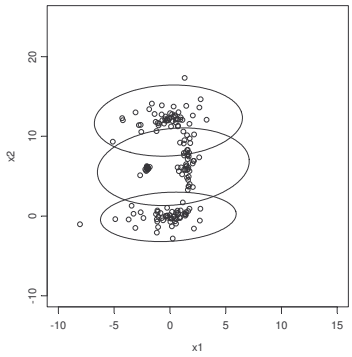


Figura 5.25- Representação da "pior" combinação para a configuração LUAT3.

ESFERA

Como à partida também já seria de esperar, para esta configuração a RRO não deu bons resultados. Está-se perante uma situação muito difícil e aliás muito irrealista. Para que o método funcionasse talvez fosse necessário aumentar o número de nuvens consideradas. Mas, por outro lado, o número de nuvens a considerar cresce exponencialmente com a dimensão.

Ainda numa tentativa de melhorar os resultados experimentou-se o método proposto com o dobro dos dados (apenas para o caso de duas variáveis pela razão apontada no parágrafo anterior) mas embora os valores do indicador DIF tivessem diminuído ligeiramente, esse decréscimo não foi de forma alguma suficiente a ponto de tornar os resultados aceitáveis, como se pode verificar na Tabela D.11.

Este exemplo confirma o outro comentário do Cristian Hennig acerca do método proposto "*I think that the proposed method may be valuable.... but I suspect that the performance of the method will depend extremely on the constellation of the clusters and the outliers...*".

Para terminar resta justificar que os traços representados na Tabela D.10 correspondem a situações onde o programa não funcionou porque se atingiu uma situação de rotura na medida em que os dados foram agrupados sucessivamente em nuvens e excluídos os *outliers* até à completa extinção das observações.

DONUT

Embora no geral existam várias situações onde não se obtêm bons resultados, consegue-se que a regra funcione bem numa ou noutra situação para os três tipos de dimensões aqui fixados (repare-se que nesta configuração decidiu-se experimentar uma nova dimensão, p=3). Como facilmente se percebe, observando a representação gráfica da configuração para o caso bivariado (ver Figura 5.26), a combinação onde se verifica um bom desempenho é a que corresponde a uma divisão em 5 nuvens com estimadores clássicos e neste caso independentemente do método de *clustering* considerado. No caso em que não são utilizadas nuvens o método não funciona e os valores obtidos são iguais para os três conjuntos de estimadores.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	26.6	19.1	0.0		***k-means***	**3**	36.0	34.0	38.0
NN	"	**4**	9.1	1.3	0.0	**NN**	"	**4**	30.1	27.1	17.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	9.0	6.0	**0.0**
2v	***pam***	**3**	28.2	18.9	0.0	**2v**	***pam***	**3**	34.0	32.4	33.5
DONUT	"	**4**	14.3	0.9	0.0	**DONUT**	"	**4**	25.4	14.6	0.0
	"	**5**	0.0	0.0	0.0		"	**5**	8.1	6.9	0.0
	mclust	**3**	17.7	10.7	0.0		***mclust***	**3**	20.9	30.4	20.5
	"	**4**	7.9	0.4	1.9		"	**4**	23.3	19.6	5.5
	"	**5**	0.0	0.0	0.0		"	**5**	7.1	8.0	0.0
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	**47.5**	47.5	47.5
	k-means	**3**	22.6	5.2	0.0		***k-means***	**3**	34.1	39.5	36.5
NN	"	**4**	3.5	0.0	0.0	**NN**	"	**4**	26.9	25.8	23.0
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	14.9	18.0	**0.0**
3v	***pam***	**3**	21.5	1.2	0.0	**3v**	***pam***	**3**	33.8	36.0	33.5
DONUT	"	**4**	5.1	0.0	0.0	**DONUT**	"	**4**	20.1	14.8	5.5
	"	**5**	0.0	0.0	0.0		"	**5**	10.9	12.0	0.0
	mclust	**3**	12.4	0.0	0.0		***mclust***	**3**	29.9	33.0	29.5
	"	**4**	1.4	0.0	0.0		"	**4**	24.2	17.0	10.0
	"	**5**	0.0	3.1	0.0		"	**5**	9.0	13.0	1.5
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	47.5	47.5	**47.5**
	k-means	**3**	9.2	0.0	0.0		***k-means***	**3**	33.7	44.0	35.5
NN	"	**4**	0.0	0.0	0.0	**NN**	"	**4**	29.1	25.5	27.5
SO	"	**5**	0.0	0.0	0.0	**CO**	"	**5**	22.0	16.5	**0.0**
4v	***pam***	**3**	10.1	0.0	0.0	**4v**	***pam***	**3**	37.9	39.0	36.5
DONUT	"	**4**	0.0	0.0	0.0	**DONUT**	"	**4**	21.0	17.5	6.0
	"	**5**	0.0	0.0	0.0		"	**5**	20.0	18.5	0.0
	mclust	**3**	2.6	0.0	0.0		***mclust***	**3**	30.0	39.5	30.5
	"	**4**	0.0	0.0	0.0		"	**4**	23.1	19.5	12.0
	"	**5**	0.0	0.0	0.0		"	**5**	20.5	14.5	3.0
	sem nuvens	**1**	0.0	0.0	0.0		sem nuvens	**1**	47.5	47.5	**47.5**

Tabela 5.9- Resultados do indicador DIF para a configuração DONUT.

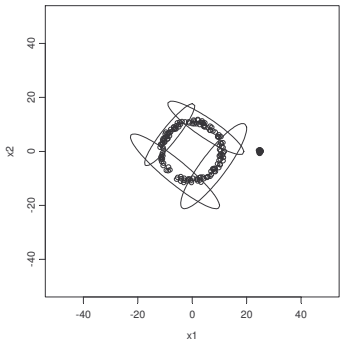


Figura 5.26- Representação da "melhor" combinação para a configuração DONUT.

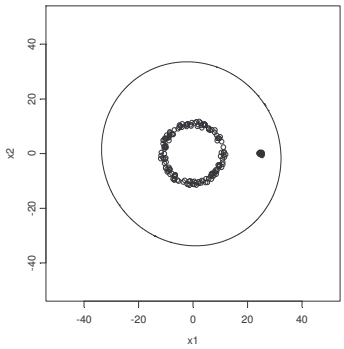


Figura 5.27- Representação da "pior" combinação para a configuração DONUT.

T1

Quer nesta configuração, quer na seguinte, optou-se por considerar apenas o caso bivariado uma vez que o estudo já vai longo. Os resultados apresentados na Tabela 5.10 evidenciam mais uma vez o desempenho intermédio do estimador OGK relativamente ao MCD e aos clássicos que na sua maioria apresentam um melhor desempenho.

		k	**MCD**	**OGK**	**Clássicos**			***k***	**MCD**	**OGK**	**Clássicos**
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	1.5	1.5	**0.0**
NN	"	**4**	0.7	0.7	0.0	**NN**	"	**4**	9.7	5.4	2.5
SO	"	**5**	0.1	0.5	0.0	**CO**	"	**5**	14.7	8.1	0.0
2v	***pam***	**3**	0.0	0.0	0.0	**2v**	***pam***	**3**	0.0	0.0	0.0
T1	"	**4**	0.0	0.0	0.0	**T1**	"	**4**	1.3	1.2	2.0
	"	**5**	0.0	0.0	0.0		"	**5**	13.3	7.8	0.0
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	10.8	7.4	10.0
	"	**4**	0.0	0.0	0.0		"	**4**	10.5	10.5	14.0
	"	**5**	0.0	0.0	0.0		"	**5**	20.0	14.0	18.0
	sem nuvens	**1**	0.7	0.6	0.0		sem nuvens	**1**	0.3	0.5	**47.5**

Tabela 5.10- Resultados do indicador DIF para a configuração T1.

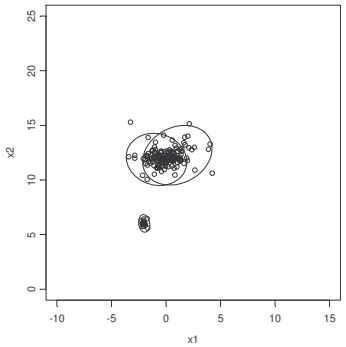


Figura 5.28- Representação da "melhor" combinação para a configuração T1.

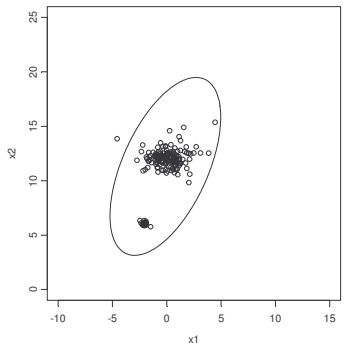


Figura 5.29- Representação da "pior" combinação para a configuração T1.

T2

Nesta configuração os resultados são em geral pouco satisfatórios mas mais uma vez consegue-se encontrar uma combinação que desempenha um bom papel na detecção dos *outliers*. Novamente é evidente o fracasso do método quando não são consideradas nuvens.

		k	MCD	OGK	Clássicos			*k*	MCD	OGK	Clássicos
	k-means	**3**	0.0	0.0	0.0		***k-means***	**3**	40.8	32.8	47.5
NN	"	**4**	0.6	0.7	0.0	**NN**	"	**4**	33.7	30.8	43.0
SO	"	**5**	0.7	1.0	0.0	**CO**	"	**5**	30.8	27.7	19.5
2v	***pam***	**3**	0.0	0.0	0.0	**2v**	***pam***	**3**	31.9	33.1	20.0
T2	"	**4**	0.0	0.0	0.0	**T2**	"	**4**	21.8	17.7	16.0
	"	**5**	0.0	0.0	0.0		"	**5**	35.0	31.2	15.5
	mclust	**3**	0.0	0.0	0.0		***mclust***	**3**	**0.0**	1.2	9.6
	"	**4**	0.0	0.0	0.0		"	**4**	23.5	16.8	34.6
	"	**5**	0.0	0.0	0.0		"	**5**	33.9	34.8	36.0
	sem nuvens	**1**	0.4	0.0	0.0		sem nuvens	**1**	40.1	45.5	**47.5**

Tabela 5.11- Resultados do indicador DIF para a configuração T2.

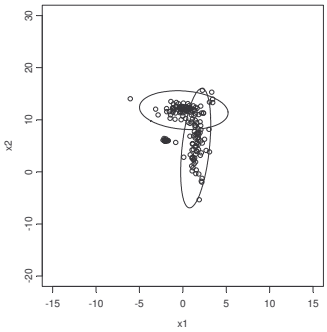


Figura 5.30- Representação da "melhor" combinação para a configuração T2.

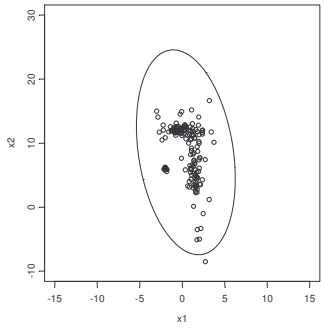


Figura 5.31- Representação da "pior" combinação para a configuração T2.

5.6.1 Conclusões

Para todas as configurações que foram aqui tratadas conclui-se que há pelo menos algumas situações onde, com a RRO, o problema dos *outliers* fica resolvido. Exclui-se, neste estudo a configuração ESFERA, como aliás acontecia com o trabalho de outros autores (ver por exemplo Billor *et al*., 2000).

O que é aqui importante realçar é que este método além de lidar eficazmente com as situações "tradicionais" consegue detectar eficazmente os *outliers* em situações mais

complexas em que são violadas as hipóteses dos dados serem elipticamente simétricos ou da maioria ter distribuição normal. Sintetizando,

- O método proposto funciona nas mesmas situações em que o método usual funciona.
- Há muitas situações onde o método proposto funciona e o método usual não funciona.
- Há casos onde o método proposto não funciona.

Para a generalidade das configurações destacam-se também as seguintes conclusões:

- os estimadores clássicos com nuvens têm melhor desempenho que os estimadores robustos com nuvens[7].
- o estimador OGK tem, em geral, um melhor desempenho que o estimador MCD.
- dos métodos de *clustering* adoptados o *mclust* é o que apresenta um melhor desempenho na maior parte das situações.

5.7 Aspectos computacionais

Nesta secção faz-se uma breve descrição da estrutura e funcionamento do programa (inserido no Apêndice E), que implementa a RRO e permitiu efectuar o estudo de simulação levado a cabo neste capítulo. Tece-se também alguns comentários acerca da geração de números aleatórios.

5.7.1 Programas

Os programas foram feitos no S-plus[8] e estiveram a correr, arbitrariamente, num computador com processador Pentium 4 1.8 GHz com 512 MB de memória e num computador com processador Pentium 2- MMX 500 MHz com 128 MB de memória com as versões 4.5 e 2000, respectivamente.

[7] Esta conclusão não é de todo surpreendente uma vez que ao se proceder à divisão dos dados em nuvens se pretende que essa nuvem seja "mais normal" que o conjunto de dados inicial.
[8] O livro de Venables e Ripley (1999) teve o seu contributo na construção destes programas.

Os diagramas de estrutura dos programas encontram-se descritos nas Figuras 5.32 e 5.33 conforme se trata da implementação da RRO (apresentada e descrita na Secção 4.5) para um conjunto de dados reais (ou apenas um conjunto de dados simulado), ou para um estudo de simulação, RROSIM.

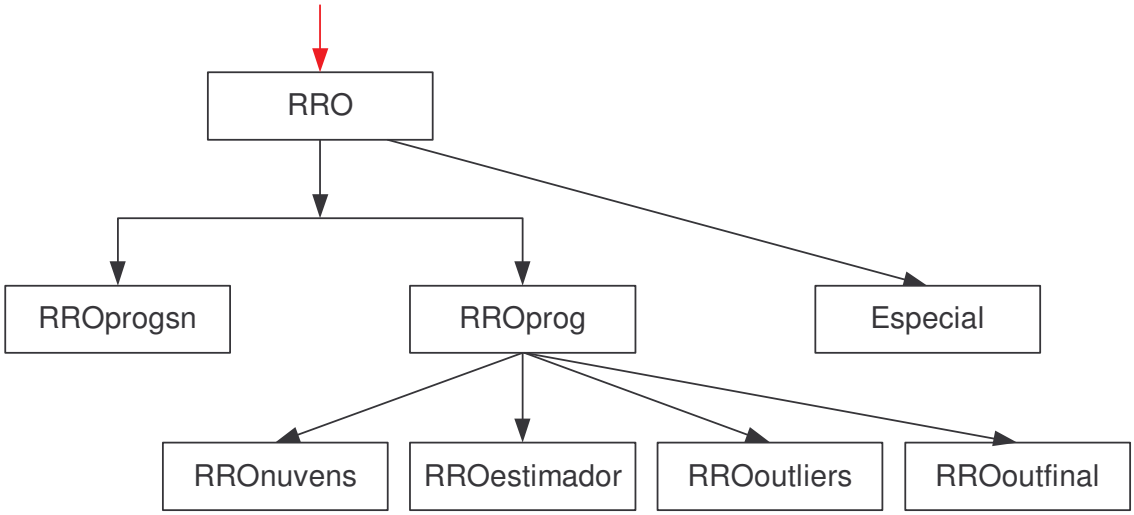


Figura 5.32- Diagrama de estrutura do programa que implementa a RRO.

Este último (RROSIM) funciona em articulação com as funções definidas para a implementação da RRO sendo que a saída, a vermelho, dá entrada na estrutura da Figura 5.32 (onde se encontra a seta com a mesma cor).

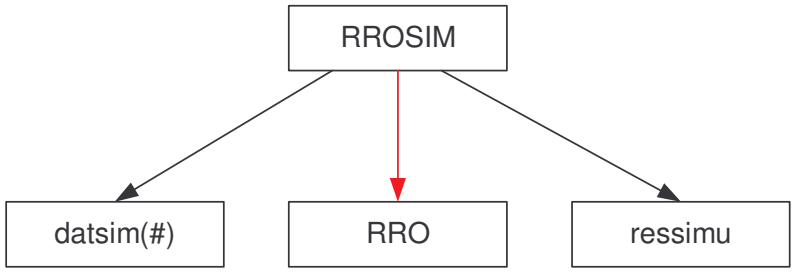


Figura 5.33- Diagrama de estrutura do programa que implementa a RRO para um conjunto de simulações.

Detalha-se em seguida o funcionamento destes dois programas principais, progRRO e progRROSIM, bem como das subrotinas associadas a cada um deles.

Funcionamento do programa RRO:

1) Pede, como *input*:

- ficheiro de dados (idat),
- método de *clustering* (imetodo)
- tipo de estimador de localização/escala (iestimador),
- tipo de constantes (itipconstantes)
- constantes (iconstantes)
- número de nuvens (inumnuvens),
- alfa (ialfa),

2) Implementa a regra de rejeição de *outliers*, RRO,

2.1) caso geral: começa por fazer uma chamada à função **rroprog** que por sua vez implementa os passos 1) (**rronuvens**) e 2) (**rroestimador** e **rrooutliers**). Após a realização destes passos o controlo retorna ao programa RRO que vai efectuar iterações, passo 3), terminando a chamar a função **rrooutfinal** que implementa o passo 4). A par destas funções foi também criada a **especial** que é chamada após a execução dos passos 1) e 2) mas antes de uma nova iteração sendo também utilizada aquando a realização do passo 4). Esta função cria uma vector composto por uns e zeros consoante uma observação é ou não *outlier*.

2.2) caso sem nuvens: começa por fazer uma chamada à função **rroprogsn** que por sua vez implementa os passos 1) e 2) da RRO considerando o caso particular em que o número de nuvens é 1, i.e. em que os dados não são particionados em nuvens (numnuvens=1). Após a realização destes passos o controlo retorna ao programa RRO que vai efectuar iterações, passo 3). É também chamada a função **especial**, após a execução dos passos 1) e 2) mas antes de uma nova iteração.

3) Fornece como *output*

- um vector constituído por uns e zeros conforme se trata de uma observação *outlier*, ou não.
- o número de *outliers*.
- o índice das observações que não são *outliers*.

Os dados de *input*, com as designações descritas em **1)** têm as opções que se passam a apresentar:

Designações	**Opções**
dat	qualquer matriz $n \times p$
metodo	0- sem nuvens 1- *k-means* 2- *pam* 3- *mclust*
estimador	1- clássicos 2- MCD 3- OGK
tipconstantes	1- Qui-quadrado 2- constantes cMCD 3- constantes cOGK
constantes	0 (vector nulo) cmcd cogk cmcdsn cogksn
numnuvens	1 (situação sem nuvens) 3 4 5
alfa	0.10 0.15 0.20

Tabela 5.12- Designações correspondentes ao *input* do programa RRO e respectivas opções.

Funcionamento das funções afectas ao programa RROSIM

Descreve-se em seguida, de uma forma genérica, o funcionamento do programa **RROSIM**, que auxilia o estudo de simulação efectuado neste capítulo e que funciona em articulação com o programa **RRO**, como mostram as Figuras 5.32 e 5.33.

1) Pede como input:

- método de *clustering* (imetodo)
- tipo de estimador de localização/escala (iestimador),
- tipo de constantes (itipconstantes)
- constantes (iconstantes)
- número de nuvens (inumnuvens),
- alfa (ialfa),
- número de simulações a efectuar (inrep)
- opção associada às características dos dados a gerar[9] (iopcao)

2) Começa por gerar os dados simulados, consoante a opção efectuada para **datsim**, chamando de seguida o programa RRO, já descrito, e terminando a executar a função **ressimu** que auxilia nos cálculos dos indicadores de simulação, repetindo estes três passos tantas vezes quantas o número de simulações fixado (inrep).

3) Fornece como *output* os valores associados aos indicadores de simulação p_1, p_2, p_3 e p_4 (descritos na Secção 5.3).

Embora o objectivo principal na construção destes programas não fosse o da minimização das linhas de código, teve-se alguma preocupação com a simplificação de alguns cálculos, recorrendo-se sempre que possível a funções estatísticas que integram o *software*, das quais se destacam:

- **cov.mcd**: função genérica que retorna estimativas da localização robusta, da matriz de covariâncias robusta e da matriz de correlações robusta baseada no MCD. A função **.mcd** permite entre outras a inclusão da opção **quan** que indica o número de observações que entram no cálculo das estimativas.
- **mahalanobis**: vector das distâncias de Mahalanobis para as *n* observações que constituem os dados.
- **rmvnorm/ runif/ rchisq**: funções de geração de números aleatórios com determinada distribuição (descritas na secção seguinte)

[9] O número da opção corresponde ao adoptado para cada uma das situações distribucionais definidas na Secção 5.5.

- **qchisq**: vector de quantis da distribuição Qui-quadrado.
- **k-means/ pam/ mclust/ clara/ fanny**: métodos de *clustering*.

5.7.2 Geração de números aleatórios

A geração de números aleatórios é um dos aspectos com maior importância num estudo de simulação. Neste trabalho foram utilizadas as funções do S-plus que permitem a geração de números aleatórios com distribuição Normal multivariada (função **rmvnorm**), com distribuição uniforme (função **runif**) e distribuição Qui-quadrado (função **rchisq**).

Para obter amostras com distribuição *t*-Student, fez-se uso da seguinte caracterização de distribuições elipticamente simétricas (Pires, 1995, Cap. 3, pág. 108). Assim, diz-se que o vector aleatório $\mathbf{Y}$ de dimensão p tem distribuição elipticamente simétrica de parâmetros $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}^0$ se existir uma variável aleatória unidimensional não negativa V tal que $\mathbf{Y} = \mathrm{V}^{-1}\mathbf{X} + \boldsymbol{\mu}$, com $\mathbf{X} \sim \mathrm{N}_p(\mathbf{0}, \boldsymbol{\Sigma}^0)$ independente de V. Para a distribuição $t_{p,\upsilon}$ verifica-se que $\mathrm{V} \sim \sqrt{\chi^2_\upsilon / \upsilon}$. Na prática basta partir de uma amostra $N_p(\mathbf{0}, \boldsymbol{\Sigma})$ e gerar um número aleatório, independente, da distribuição unidimensional V para obter a amostra pretendida. Sendo assim, no caso da configuração T1, para a situação 29 (ver Secção 5.5) foram efectuadas 100 simulações, em cada uma das quais se gerou 150 observações normais bivariadas de média $(0,0)^{\mathrm{T}}$ e $\boldsymbol{\Sigma} = diag(1, 0.3)$ e 150 observações Qui-quadrado univariadas com 3 graus de liberdade, sendo criado V, tendo-se de seguida efectuado uma translação de valor $\boldsymbol{\mu}$=$(0,12)^{\mathrm{T}}$, obtendo-se $\mathbf{Y}$.

5.8 Alguns incentivos e sugestões para o desenvolvimento deste trabalho

Contribuíram para o desenvolvimento desta fase do trabalho várias pessoas, entre as quais se gostaria de destacar os contributos que se passam a apresentar:

Organização do COMPSTAT 2002

From: CompStat 2002 [info@compstat2002.de]
Sent: sexta-feira, 14 de Dezembro de 2001 09:38
To: carla@upt.pt
Subject: Compstat 2002 - Ranking of your paper

Dear Dra Carla Santos Pereira,
The reviewing and ranking process of all abstracts has finished and due to the high quality of your submitted paper Detection of outliers in multivariate data, the Scientific Programme Committee of Compstat 2002 would like you to present your paper in a full contributed session (ca. 25min presentation +5 min discussion) at our conference. Therefore we would like to ask you to contribute a long version (6 pages) of your paper until February 10th 2002 (or earlier). For details please visit our webpage: http://www.compstat2002.de

Best regards
Prof. Dr. Wolfgang Härdle
Compstat2002

Claudia Becker

From: Claudia Becker [c.becker@wiwi.uni-halle.de]
Sent: quarta-feira, 04 de Dezembro de 2002 16:29
To: carla@mail.uportu.pt
Subject: Re: detection of outliers in multivariate data: finally!

Dear Carla,
now I finally have finished reading your paper and as promised, will give you my impression.
First of all, as I already told you at the conference, I am really impressed how well your procedure works. There is a very good, intuitive interpretation of an observation being outlying if it is outlying with respect to every cluster in the data.
To my opinion, the crucial point in your procedure seems to be the choice of an appropriate number of clusters. In the paper you mention that you do not need to fix the

number of clusters in advance but try several values of k and decide based on the results. For further research it would be good to discuss this point in more detail. How do you decide? Which criteria do you look at to finally choose k? And how are these criteria themselves influenced by outliers? I can imagine that it is possible to increase the chosen number k if there are some single separate outliers which then tend to build clusters of their own. In this case you will not be able to identify them as outliers later (only if you do not allow for clusters of size 1, or declare such as outliers). You will usually find them to be disconnected clusters and then have to decide on each by separate inspection, if I got it right.

This leads me to the second point I want to address: the problem of disconnected clusters. Looking at your figure 1, I think that for this situation the result of your procedure is very convincing: the three groups are characterized by the detection contours, which are connected, and there is the fourth group of a few observations which is clearly separated from the rest and can then be investigated further. (Although I am not sure if this last cluster should be called outlying. One can and should surely discuss this. This would lead to the definition of a concept of outlyingness for clusters, which is also an interesting topic.) But I think that in many clustering situations you will see a different picture, namely many disconnected clusters (where there is clear separation) and only some which are like in your example. And wouldn't it be very time-consuming if you had to judge about each of these clusters "by hand"? Maybe I got something wrong, and you can easily correct me with this impression.

Finally, when looking at figure 1, I immediately thought of fuzzy-clustering (didn't I also mention this in Porto? I do not remember, so I just tell you again). There you do not assign every observation to a specific cluster but give the observations so-called membership values between 0 and 1 for each cluster. A value near 0 means that the observation does probably not belong to the cluster, a high value represents a high belief for this observation belonging to the cluster. I could imagine that in the outlier situation it may come out that an outlying observation gets low membership values for all clusters. This would be similar to your approach with being identified as outlying with respect to all clusters. Maybe this would be a direction for a comparison of your procedure with a different approach.

I hope these comments will be of a certain help for you. If you do not understand what I mean at some points, please do not hesitate to ask me, and I will try to put it differently.

It would be good to hear from you and your progress in the topic. Please, do excuse my late response.
Warm regards,
Claudia

****

Prof. Dr. Claudia Becker
Lehrstuhl für Statistik
Wirtschaftswissenschaftliche Fakultät
Martin-Luther-Universität Halle-Wittenberg
06099 Halle

Tel.: 0345 55 23396
Fax: 0345 55 27191
****

Capítulo 6

Método genérico e tratamento de exemplos

No Capítulo 2 introduziu-se o problema de classificação genérico onde se referiu para além da classificação nas c classes habituais, a introdução de uma opção de rejeição, por indecisão ou por existência de observações *outliers*, pelo que de uma forma geral o problema de classificação passou a consistir na escolha de uma entre c+2 classes.

No Capítulo 3 foi tratada a rejeição por indecisão. No Capítulo 4 propôs-se um método de detecção de *outliers* que permite a rejeição por existência desse tipo de observações. Neste capítulo tem-se como objectivo agrupar o trabalho desenvolvido nesses capítulos, no sentido da definição de um método genérico de classificação em c+2 classes. No resto do capítulo dão-se alguns exemplos de aplicação deste método que se julga contribuírem para observar as suas boas propriedades.

6.1 Método genérico

Uma vez que o objectivo final deste método consiste em integrá-lo na classificação em c+2 classes conforme referido na Secção 2.2 do Capítulo 2 apresenta-se nesta secção o método genérico de classificação. Há, no entanto, que ter em atenção que a RRO terá de ser aplicada independentemente a cada classe. E mais ainda, esta regra pode e deve também ser aplicada antes da classificação propriamente dita. Quer-se com isto salientar que a utilização da RRO pode também servir como processo de "melhoramento" da amostra de treino (sem *outliers*) permitindo a construção de uma regra de decisão mais precisa.

Método genérico de classificação

1. Para cada classe, "limpeza" dos *outliers* da amostra de treino com base na regra de rejeição de *outliers*, RRO.
2. Construção da regra de decisão $\Rightarrow$ Definição das classes G_1, G_2, ..., G_c, I.

3. Classificação de um novo objecto de origem desconhecida:

3.1. Verificar se a nova observação é *outlier* relativamente à amostra de treino (para todas as classes) da seguinte forma:

3.1.1 Aplicar a RRO a cada classe$\Rightarrow$ O_1, O_2, ..., O_c.

3.1.2 $O = \bigcap_{i=1}^{c} O_i$ ou seja, a observação é classificada como *outlier* se e só se for *outlier* para todas as classes.

3.2 Se em 3.1 a conclusão é:

sim $\Rightarrow$ a observação é rejeitada por ser um *outlier* (não pertencer a nenhuma das classes), é classificada na classe *O*, e o processo pára.

se não $\Rightarrow$ passo 4.

4. Com base na estimação das densidades, ou equivalentemente, na estimação das probabilidades *a posteriori*, verificar se a observação pertence a alguma das *c* classes.

Se sim $\Rightarrow$ a observação é classificada na classe correspondente G_i, i=1,2,...,*c*

Se não $\Rightarrow$ a observação é rejeitada pela existência de indecisão i.e. é classificada na classe *I*.

Notas

- A amostra de treino "limpa" fornece os dados $\hat{\mu}_{ij}, \hat{\Sigma}_{ij}, n_{ij}$ *e* c_{ij} *i*=1,..., *c* e *j*=1,..., k_i onde k_i representa o número de nuvens consideradas na *i*-ésima classe. Esta informação permite posteriormente calcular as distâncias de Mahalanobis e compará-las por forma a decidir se uma dada observação deve ou não ser considerada *outlier*.
- O passo 1 da RRO deve passar a ser lido como " segmentar as n_i observações de cada classe (i=1,2,...,*c*), em k_i sub-nuvens, através da utilização de um método de partição *clustering*".
- Sugere-se que o valor de α adoptado para a "limpeza" da amostra de treino seja superior ao utilizado para a classificação. Se no passo da detecção de *outliers* houver um "forte" grau de exigência para uma "limpeza" eficiente da amostra, na classificação essa exigência poderá ser mais branda.

Acerca da estimação

Para que o método possa ser aplicado há necessidade de se ter em conta aspectos tais como as probabilidades *a priori*, o custo de rejeição *t* (definido no Capítulo 3) ou ainda os custos mais gerais referidos em (3.3) e a estimação das densidades (ou das probabilidades *a posteriori*) para cada classe.

Quanto à estimação das probabilidades *a priori* já aqui se fez bastante referência ao processo habitual de estimação, baseado no cálculo da frequência relativa com que cada classe ocorre na amostra de treino.

Para o custo de rejeição por indecisão, *t*, podem ser experimentados vários valores. Relembre-se que para o caso particular de duas variáveis $0 \leq t \leq 0.5$ pelo que se $t=0.5$ cai-se no caso usual de inexistência de rejeição.

As densidades condicionais às classes (ou equivalentemente das probabilidades *a posteriori*) podem ser estimadas por vários processos: discriminante linear, quadrática ou logística- como foi exemplificado no Capítulo 3- ou através de redes neuronais, árvores de classificação ou outros métodos descritos no Capítulo 2. Não se quer no entanto deixar de destacar uma proposta alternativa, que irá aqui ser exemplificada nos exemplos de aplicação do método genérico, e que consiste na estimação da densidade $f_i(\mathbf{x})$, através de um modelo de mistura de normais com k_i componentes correspondentes às nuvens fixadas na *i*-ésima classe, da seguinte forma:

$$\widehat{f}_i(\mathbf{x}) = \sum_{j=1}^{k_i} \frac{n_{ij}}{n_i} f_{N(\hat{\mu}_{ij}, \hat{\Sigma}_{ij})}(\mathbf{x}), \ \forall i = 1, 2, ..., c. \tag{6.1}$$

A utilização de modelos mistura para estimação das densidades condicionais às classes é hoje em dia uma prática comum (ver por exemplo Fraley e Raftery, 2002). Jain *et al*. (2000), acerca das fronteiras do r.p. referem "*Another example is mixture modeling, which was proposed in 1977, and which is now very popular for density estimation and clustering, due to computing power available today*".

6.2 Tratamento de exemplos

Nesta secção examina-se a *performance* do método proposto em alguns exemplos usados na literatura[1]. No primeiro exemplo começa-se apenas por considerar a avaliação da RRO num conjunto de dados que constitui uma referência em grande parte da literatura sobre a detecção de *outliers*. Seguem-se dois exemplos de aplicação do método genérico para o caso bivariado, tratados em Santos-Pereira e Pires (2004a), dos quais o primeiro se refere a um conjunto de dados reais que constitui um exemplo clássico na análise discriminante, e o outro a um conjunto de dados simulados. Para terminar esta secção apresenta-se um exemplo característico do tema do r.p. onde se exemplifica a utilização do método genérico para uma amostra de maiores dimensões e para um número mais elevado de variáveis.

Exemplo 6.1 (Dados HBK)

Este primeiro exemplo trata de um conjunto de dados construídos por **H**awkins, **B**radu e **K**ass (1984), referidos como HBK, e consiste numa amostra de 75 observações na qual se observaram 3 variáveis.

Na Secção 4.3 descreveram-se duas situações ilustradas com dois exemplos, uma em que a detecção dos *outliers* falhava com os estimadores clássicos mas já funcionava com estimadores robustos (MCD) e outra onde falhava em ambos os casos. Quando se propõe a RRO é óbvio que se pretende que a regra funcione nem situações complexas mas que não deixe de funcionar nas situações tradicionais. É esta a razão que está por detrás da escolha deste conjunto de dados. Como referem Rocke e Woodruff (1996) estes dados contêm 14 *outliers* e quer com o método apresentado por estes autores, quer com o MCD, estas observações são detectadas. No entanto como afirmam Rosseeuw e Leroy (1987) e facilmente se verifica (ver Figura 6.1), com a distância de Mahalanobis clássica apenas se detecta um *outlier* (observação 14).

[1] Todos os conjuntos de dados reais utilizados, à excepção do último (*pendigits*) por ser de bastante maior dimensão, encontram-se listados no Apêndice F.

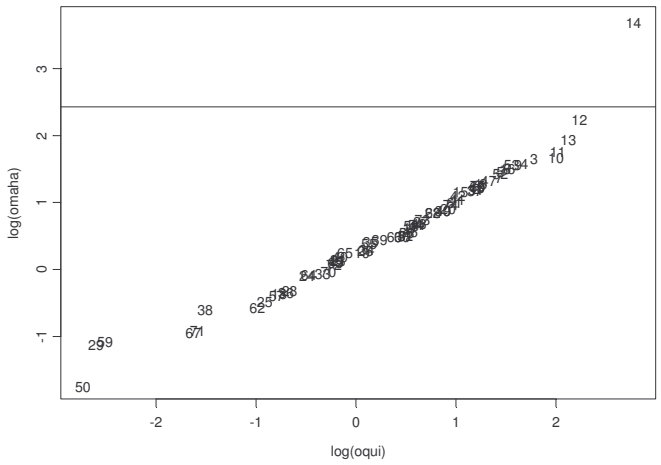


Figura 6.1- Quadrado das distâncias de Mahalanobis versus estatísticas de ordem do χ^2_3 e linha horizontal correspondente ao $\chi^2_{3,0.99}$.

Experimentou-se a RRO com os três métodos de *clustering* utilizados no estudo de simulação e para os estimadores clássicos e MCD nas situações com 3, 4 e 5 nuvens e para o caso onde não são consideradas nuvens. À excepção do método *mclust*, quer o *k-means* quer o *pam*, detectaram os 14 *outliers* para qualquer número de nuvens considerado. O mesmo se verificou na utilização do MCD sem nuvens embora com estimadores clássicos apenas se conseguisse detectar um único *outlier*. Conclui-se então que os resultados obtidos com a RRO são similares aos reportados na literatura para outros métodos de detecção de *outliers* indicando que o método proposto também apresenta um bom comportamento para este conjunto de dados.

Exemplo 6.2 (Mulheres portadoras de Hemofilia: dados HEMO)

Sendo este exemplo um dos clássicos da análise discriminante, surge a curiosidade de analisar o comportamento do método proposto.

A hemofilia é uma doença hereditária ligada aos cromossomas sexuais. Como refere Pires (1995, Cap. 1, pág. 7) as mulheres não são geralmente doentes (a probabilidade de o serem é muito pequena e normalmente não sobrevivem) mas podem ser portadoras do gene recessivo que poderá provocar hemofilia nos seus descendentes do sexo

masculino, ou seja, aparentemente "normais" mas geneticamente "defeituosas". Também segundo Pires (1995, Cap. 1, pág. 7), estudos realizados nos anos 70 revelaram que duas variáveis obtidas a análises ao sangue e relacionadas com a capacidade de coagulação, tomam valores significativamente diferentes nas mulheres "normais ("não portadoras") e nas "portadoras": FactorVIII- actividade (x_1) e FactorVIII- antigene (x_2). Por outro lado, a introdução de custos é de extrema importância neste exemplo. O dar a uma mulher a indicação de que é portadora quando de facto não o é (o que poderá ter como consequência o facto de que não queira ter filhos) não é igual a dizer a uma portadora que não o é (o que poderá levá-la a ter filhos com elevada probabilidade de serem hemofílicos).

Os dados originais são os apresentados por Hermans e Habbema (1975) e referem-se a 30 mulheres "normais" e 22 "portadoras". No entanto, tendo-se disponível em Johnson e Wichern (1992, Cap. 11, pág. 570) também 30 observações de mulheres "normais" diferentes das restantes decidiu-se juntar os dois conjuntos dando origem a uma amostra de 60 mulheres "normais". Por outro lado, em relação às mulheres portadoras, estão disponíveis não apenas as 22 observações mas sim um conjunto de 45 pelo que se optou por considerá-las todas. Tem-se assim uma amostra de 105 observações, 60 "normais" (classe G_1) e 45 "portadoras" (classe G_2) que se encontram no Apêndice F (após uma transformação logarítmica). Na Figura 6.2 representam-se as duas classes no plano das variáveis transformadas ($\log_{10}x_1$, $\log_{10}x_2$).

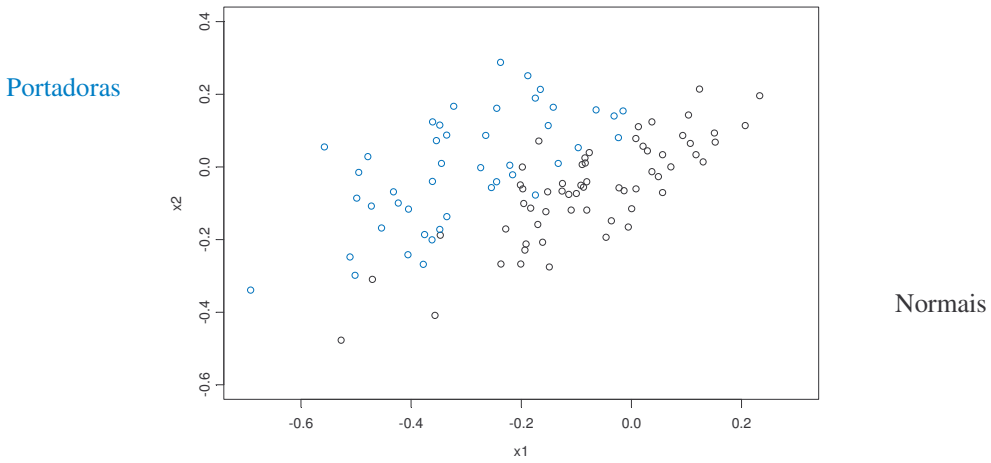


Figura 6.2- Dados do Exemplo 6.3.

Enveredando pelo método genérico proposto e com o propósito de obtenção da regra de classificação foram feitas as seguintes escolhas:

- Três métodos de *clustering*: *k-means*, *pam* e *mclust*, cada um deles com k=3, 4 e 5 nuvens.
- Três estimadores de localização-dispersão: clássicos com limites de detecção assintóticos; RMCD25 com limites de detecção determinados previamente através de uma simulação de 1000 conjuntos de dados normais; $OGK_{(2)}(0.9)$ com limites de detecção determinados previamente através de uma simulação de 1000 conjuntos de dados normais.
- Nível de significância global: α=0.1.
- Classificar as nuvens desligadas (no passo 4) da RRO) como *outliers*.
- Estimação das densidades através da mistura de normais sugerida na secção anterior.
- Critério de selecção para a escolha da melhor combinação na "limpeza" dos *outliers* (método de *clustering*/ estimador/ k): AIC.

Obtidos os resultados da RRO, para cada uma das classes, a utilização do critério AIC conduziu à escolha da "melhor" combinação que se passa a apresentar juntamente com o número de *outliers* detectados.

- G_1: *k-means*/ estimador RMCD25/ k=4$\Rightarrow$ "limpeza" de 3 *outliers*;
- G_2: *k-means*/ estimador clássico/ k=4$\Rightarrow$ "limpeza" de 3 *outliers*.

Nas Figuras 6.3 e 6.4 apresenta-se, para cada classe, o conjunto de dados "limpo" com os respectivos contornos. Note-se que apenas aparecem três contornos porque, em ambos os casos, a quarta nuvem tinha apenas três observações (classificadas como *outliers* e representadas a cores diferentes).

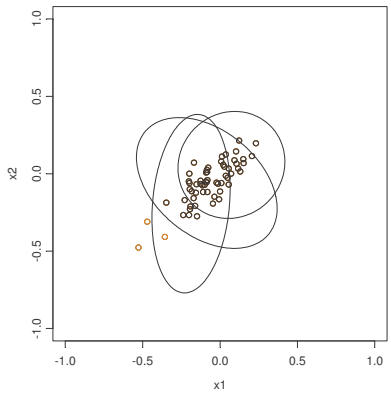


Figura 6.3- Dados do Exemplo 6.2 com contornos para a classe G_1.

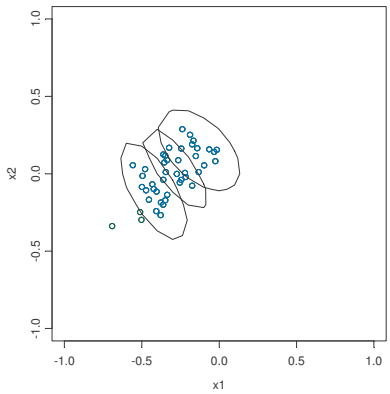


Figura 6.4- Dados do Exemplo 6.2 com contornos para a classe G_2.

Feita a "limpeza" dos *outliers* da amostra de treino, para cada classe, e tomando $C(i|i)=0$, $C(i|j)=1$ e $C(I|j)=t$ para $i,j=1,2,...,c$, procedeu-se à construção da regra de decisão. Escolhidos dois custos de indecisão diferentes, $t=0.1$ e $t=0.3$, procedeu-se à representação gráfica das regiões de classificação com indecisão nas Figuras 6.5 e 6.6, respectivamente.

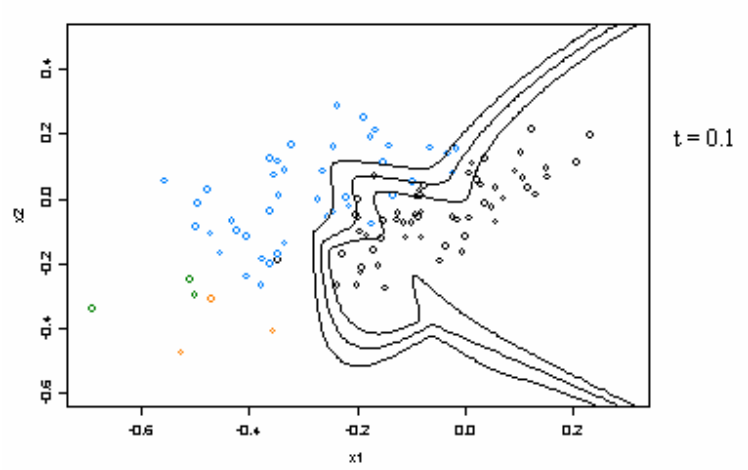


Figura 6.5- Regiões de classificação com custo de indecisão $t=0.1$ para o Exemplo 6.2.

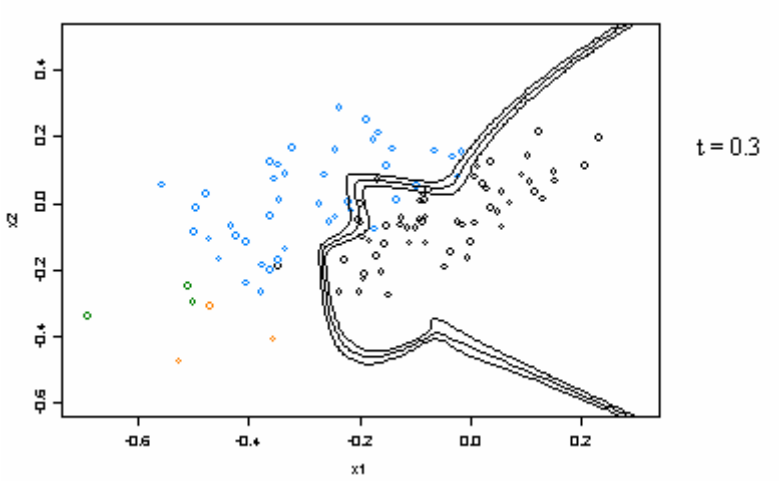


Figura 6.6- Regiões de classificação com custo de indecisão t=0.3 para o Exemplo 6.2.

Outros custos poderiam ainda ter sido experimentados tendo apenas reflexo na escolha dos limiares de decisão e por consequência no número de observações correctamente classificadas e rejeitadas.

Exemplo 6.3

Para este exemplo geraram-se um total de 320 observações bivariadas (n_1=120 e n_2=200) correspondentes a duas classes da seguinte forma:

- $G1$- 50 observações $t_{2,3}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$ e 50 observações $t_{2,3}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)$ com $\boldsymbol{\mu}_1=(0,12)^T$, $\boldsymbol{\Sigma}_1$=diag(1,0.3), $\boldsymbol{\mu}_2=(1.5,6)^T$ e $\boldsymbol{\Sigma}_2$=diag(0.2,9) e 20 observações *outliers* $t_{2,3}((-2,6)^T,0.01\mathbf{I})$.
- $G2$- 100 observações N_2 $((7,5)^T,0.5\mathbf{I})$ e 100 observações $N_2((2,3)^T, \boldsymbol{\Sigma}_1)$ com $\boldsymbol{\Sigma}_1$=diag(0.2,9).

As observações obtidas encontram-se representadas graficamente na Figura 6.7.

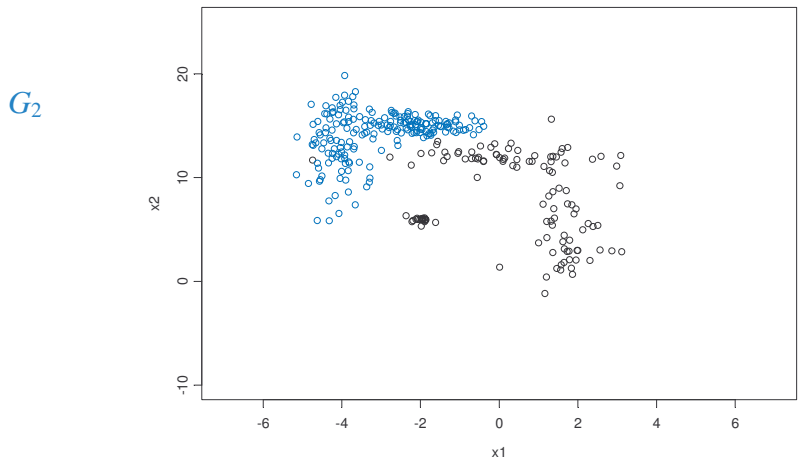


Figura 6.7- Dados do Exemplo 6.3.

Enveredando pelo método genérico e realizando as mesmas escolhas que no exemplo anterior, a utilização do critério AIC conduziu à escolha da "melhor" combinação, para cada classe, que se passa a apresentar juntamente com o número de *outliers* detectados.

- G_1: *mclust*/ estimador clássico/ k=3⇒ "limpeza" de 21 *outliers*;
- G_2: *mclust*/ estimador clássico/ k=2⇒ não foram detectados *outliers*.

Feita a "limpeza" dos *outliers* da amostra de treino (observações a rosa), para cada classe, procedeu-se à construção da regra de decisão tendo-se obtido, para um custo de indecisão t=0.1, as regiões de classificação que se passam a apresentar na Figura 6.8.

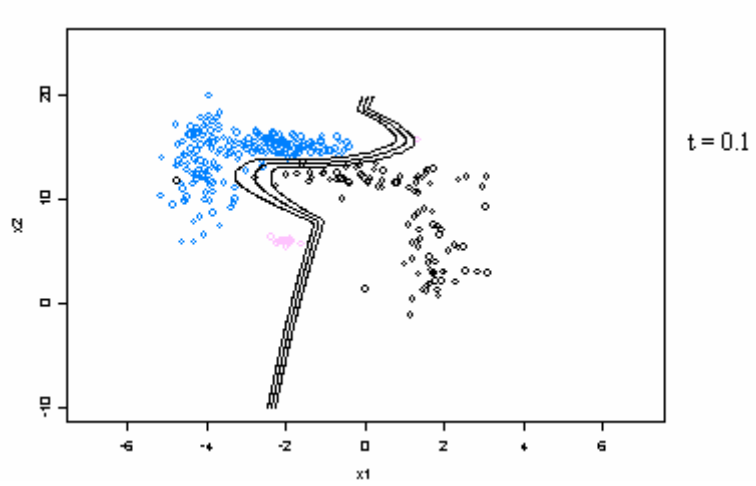


Figura 6.8- Regiões de classificação com custo de indecisão t=0.1 para o Exemplo 6.3.

Exemplo 6.4 (Dados *pendigits*)

No reconhecimento de caracteres impressos, qualquer imagem de um caracter é uma possível de um número definido de estilos que habitualmente são designados por fontes. Por essa razão, o reconhecimento de caracteres impressos, contrasta com o reconhecimento de caracteres escritos à mão onde há praticamente infinitas formas de os escrever (Leondes, 1998, pág. 63). É este segundo caso que nos interessa estudar na prática e sobre o qual incide este exemplo.

Os dados analisados designam-se por "pen-based recognition of handwritten digits" e encontram-se disponíveis em *http://www.ics.uci.edu/~mlearn/MLSummary.html*. A escrita foi realizada numa tábua WACOM PL-100V que envia as coordenadas que constituem os elementos da amostra, i.e., uma sequência de posições para cada caracter escrito.

Como se refere em Santos-Pereira e Pires (2002), uma característica interessante destes dados é a facilidade com que os caracteres podem ser representados tornando fácil julgar a classificação de uma observação como sendo ou não *outlier*. Esta possibilidade pode ser conveniente para analisar a competitividade dos vários métodos sendo, no entanto, muito tediosa se o fizermos para a totalidade das observações. Além do mais o objectivo final será o da classificação automática dos caracteres.

Para este conjunto de dados a amostra completa perfaz um total de 10992 observações (somando o número de observações de treino e teste) que se distribuem por 10 algarismos, num conjunto de 16 variáveis correspondentes às várias posições (oito coordenadas). Como se trata apenas de exemplificar o método foram seleccionadas duas classes correspondentes aos algarismos 0 e 1, tendo-se disponível uma amostra de 1143 observações para cada uma das classes. Para o algarismo 0 utilizaram-se 780 observações da amostra de treino mais 363 da amostra de teste e para o algarismo 1, 779 mais 364 observações, respectivamente.

Uma vez que não se dispunha dos valores das constantes c(p, n, α_n) para p=16 (sendo o seu cálculo bastante moroso) e dada a grande dimensão dos dados, optou-se então por utilizar os estimadores clássicos e os métodos de clustering *k-means* e *pam* já que o *mclust* não é adequado para dados destas dimensões, com número de nuvens[2] k=3,4 e 5. Em relação ao nível de significância global inicialmente optou-se por utilizar

[2] Neste caso, uma vez que a amostra é de grande dimensão, poderiam ter sido experimentados outros valores de k. No entanto tal não foi possível já que os programas só foram construídos por forma a contemplar no máximo 5 nuvens e o estudo já ia demasiado longo para se proceder a alterações.

α=0.10, mantendo o que foi feito anteriormente. No entanto, conforme se poderá verificar nas Tabelas 6.2 e 6.3 (consoante se trata do algarismo 0 ou 1) eram detectados muitas observações como sendo *outliers* sem na realidade o serem, pelo menos de uma forma evidente, e principalmente tendo apenas o objectivo de distinção entre os algarismos 0 e 1. Se o objectivo fosse o de classificar nas 10 classes iniciais talvez já se tivesse de ter em conta estas observações como *outliers* nomeadamente porque algumas delas poderiam ser confundidas com o algarismo 2. Observe-se por exemplo a Figura 6.9 onde se representa a observação número 75 associada ao algarismo 1.

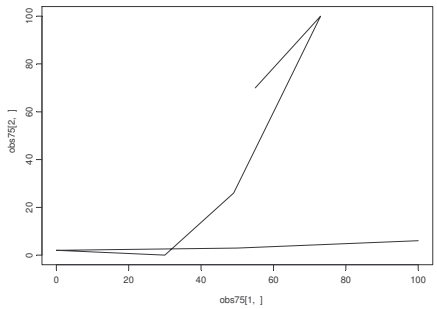


Figura 6.9- Observação 75 do algarismo 1.

Esta observação é classificada como *outlier* tomando α=0.10. No entanto, reduzindo esta valor para 0.01 e desactivando a opção 4 da RRO pelas razões apontadas na Secção 4.5, isto já não acontece; esta observação não é detectada por nenhum dos métodos de *clustering* nem nenhum dos valores de *k* escolhidos. Esta última opção foi aquela que se considerou para a análise que se seguiu. Nas Tabelas 6.1 e 6.2 as observações detectadas como *outliers*, tomando-se α=0.10, são apresentados entre parêntesis.

	Clássicos com nuvens	**Clássicos sem nuvens**
k-means	3- 65 (90) 4- 86 (95) 5- 111 (140)	79 (106)
pam	3- 139 (158) 4- 92 (113) 5- 74 (96)	

Tabela 6.1- Número de *outliers* detectados para o algarismo 0 e α=0.01, 0.10.

	Clássicos com nuvens	**Clássicos sem nuvens**
k-means	3- 174 (610) 4- 147 (446) 5- 83 (112)	83 (249)
pam	3- 145 (598) 4- 133 (443) 5- 152 (440)	

Tabela 6.2- Número de *outliers* detectados para o algarismo 1 e α=0.01, 0.10.

Mesmo tendo-se optado por diminuir o α e desactivar a opção 4 da RRO, verifica-se a existência de um grande número de *outliers*, número esse que varia num intervalo de amplitude 74 para o algarismo 0 e 91 para o algarismo 1. Coloca-se então agora a questão de como se proceder para decidir quais as observações que serão realmente consideradas *outliers*. Serão *outliers* todas as observações detectadas por alguma das situações experimentadas ou pelo comum a todas as situações experimentadas? será que basta considerar apenas as observações consideradas por um dos métodos de *clustering*? E será que basta considerar a situação em que se utiliza a distância de Mahalanobis usual ou haverá vantagens em se considerar nuvens? É o que se irá analisar de seguida fazendo a distinção entre os resultados para o algarismo 0 e o algarismo 1.

Algarismo 0: *outliers*

Para as opções com nuvens o número de *outliers* varia entre 65 e 139. O número de *outliers* comum às três situações do *k-means* é de 54. Mas será que a intersecção dos *outliers* detectados por todas as situações experimentadas com este método é suficiente? Será que não há observações detectadas por uma ou duas das situações que são na verdade *outliers*? Há algumas observações detectadas apenas por uma ou por duas que são suspeitas de serem *outliers* embora não muito evidentes pensando-se apenas na distinção entre os algarismos 0 e 1. É por exemplo o caso das observações 123 ou 201 que se apresentam na Figura 6.10.

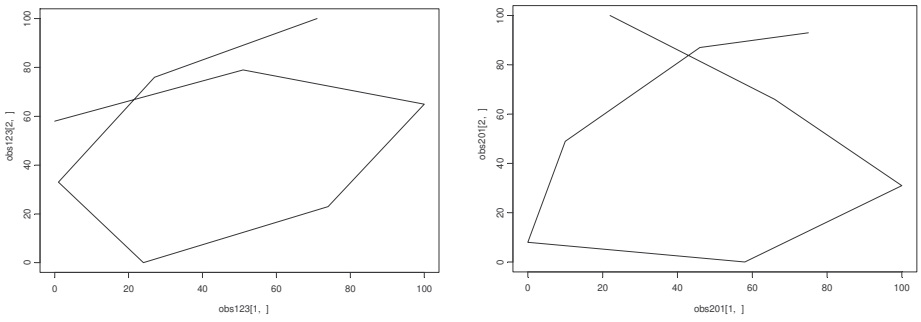


Figura 6.10- Observações 123 e 201 do algarismo 0.

Analisando-se agora o método *pam* verifica-se que a intersecção das três situações detecta 60 *outliers* dos quais 11 não são detectados pelo comum do *k-means* (mas todos eles detectados por uma ou duas das situações). Por outro lado há 5 observações que são detectadas pela intersecção dos *k-means* e não pela intersecção dos *pam*. Isto conduz à reunião do comum dos resultados de cada um dos métodos de *clustering*. Sendo assim

obtém-se um total de 54+11=65 *outliers* para o algarismo 0. Falta ainda, contudo, verificar o que se passa relativamente à situação sem nuvens. Neste caso são detectadas 79 observações. Destas, apenas 49 são comuns à situação considerada (reunião das intersecções dos dois métodos de *clustering*). Das restantes 30, com excepção da observação 166, representada na Figura 6.14, não há grande evidência para classificar as observações como *outliers*. Veja-se por exemplo na Figura 6.11 as observações 73 e 1065.

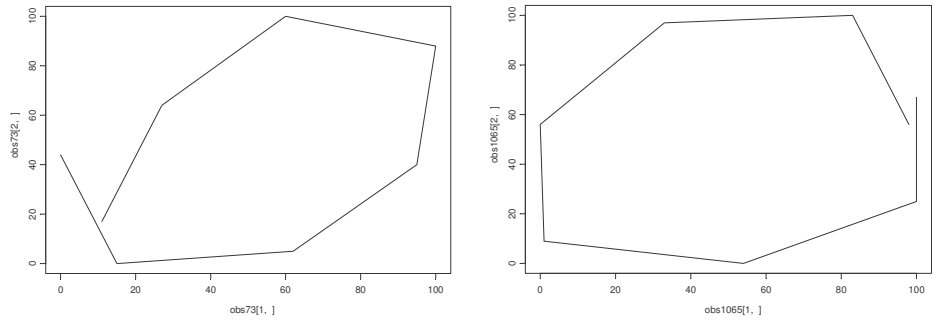


Figura 6.11- Observações 73 e 1065 do algarismo 0.

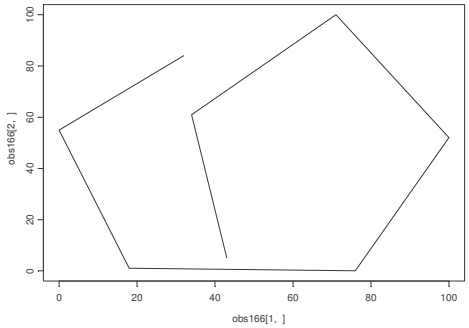


Figura 6.12- Observação 166 do algarismo 0.

Há, no entanto, 16 observações que são *outliers* e que não são detectadas pelo método sem nuvens. É por exemplo o caso das observações representadas na Figura 6.13.

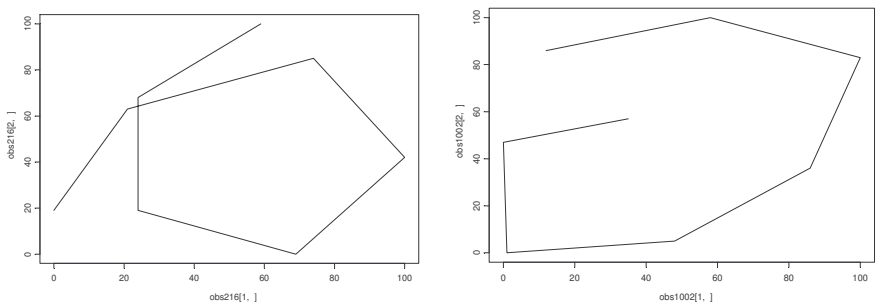


Figura 6.13- Observações 216 e 1002 do algarismo 0.

Na Figura 6.14 apresentam-se duas de um conjunto de observações completamente aberrantes detectadas como *outliers* por todas as variantes, com e sem nuvens.

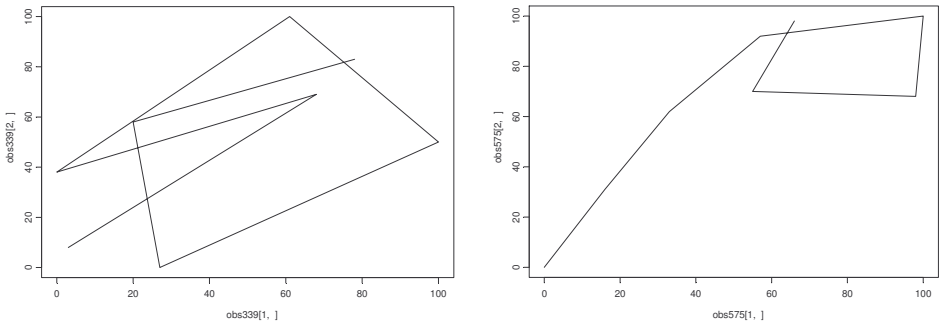


Figura 6.14- Observações 339 e 575 do algarismo 0.

Algarismo 1: *outliers*

Para as situações com nuvens o número de *outliers* detectados varia entre 83 e 174 para o *k-means* e entre 133 e 152 para o *pam*. Há 80 observações *outliers* que são comuns aos três valores de *k* com o método *k-means*. Por outro lado, no caso do método *pam*, esse número é de 106 das quais 29 não pertencem ao comum obtido com o método *k-means* assim como 5 não pertencem ao comum do *pam* (no entanto, todas elas eram detectadas por uma ou duas das situações consideradas). Analisadas essas observações (as 29 mais as 5) conclui-se que todas elas aparentam ser *outliers*.

Sendo assim, tal como foi feito para o algarismo 0, optou-se por considerar *outliers* todas as observações que pertenciam ao comum do *k-means* e ao comum do *pam* tendo-se obtido um total de 109 observações *outliers*. Destas observações há 50 que não são detectadas pelo método sem nuvens! E estas observações são, no geral, completamente "aberrantes". É o caso, por exemplo das observações representadas na Figura 6.15.

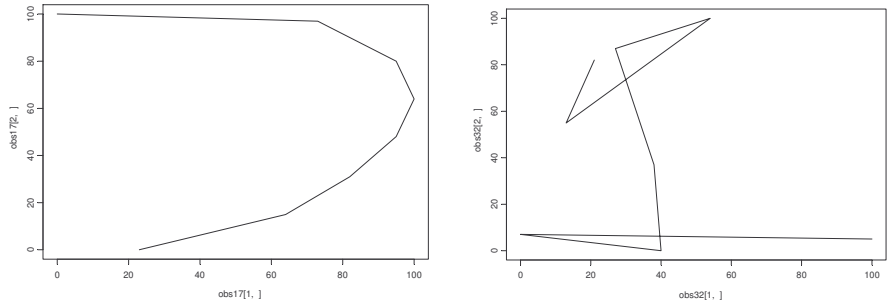


Figura 6.15- Observações 17 e 32 do algarismo 1.

Neste caso é notória a superioridade do método com a utilização de nuvens. A utilização do método sem nuvens detecta um total de 83 *outliers* das quais 24 não pertencem à reunião do comum dos métodos *k-means* e *pam*. Analisando essas observações verifica-se que com excepção da 4 e da 592, todas poderão ser confundidas com o algarismo 2; veja-se o exemplo de duas dessas observações na Figura 6.16. As outras duas observações embora possam ser consideradas *outliers* se se pensar apenas no algarismo 1, tendo em conta que o objectivo final é o da discriminação entre as classes dos algarismos 0 e 1 essas observações, tais como as que se confundem com o algarismo 2, não serão consideradas *outliers*.

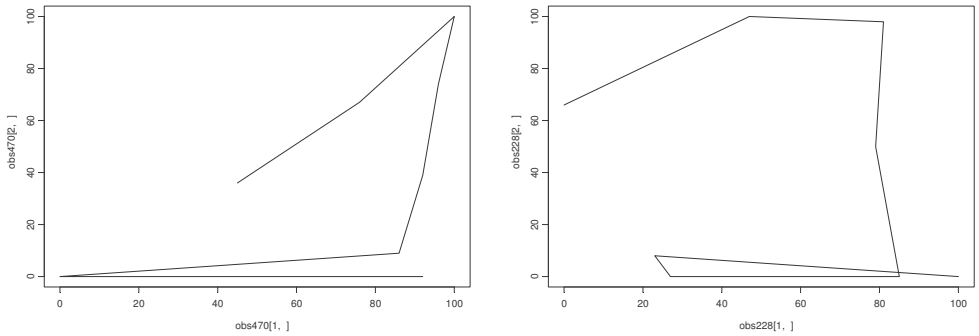


Figura 6.16- Observações 228 e 470 do algarismo 1.

As duas observações, representadas com os números 4 e 592, que constituem a excepção à possível confusão entre os algarismos 1 e 2 são apresentadas na Figura 6.17.

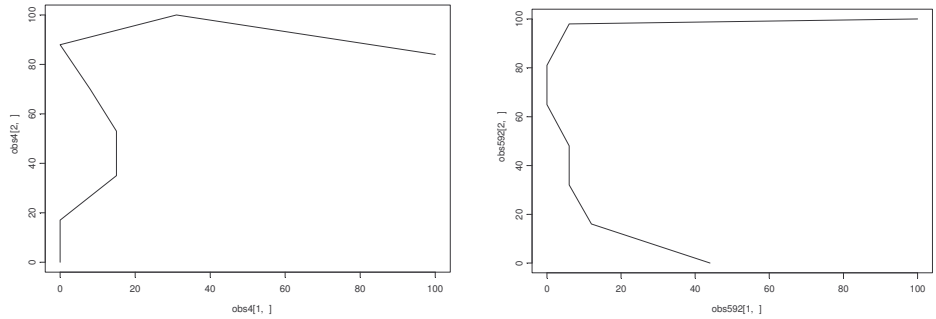


Figura 6.17- Observações 4 e 592 do algarismo 1.

Feitas todas estas considerações e antes de se partir para a utilização do método genérico de classificação considera-se que o exemplo acabado de tratar permite destacar a seguinte importante conclusão:

- Há observações consideradas "aberrantes" que só são detectadas considerando-se a RRO com nuvens. E embora a distância de Mahalanobis tradicional consiga detectar algumas observações não detectadas com a utilização de métodos de *clustering*, na generalidade essas observações não são realmente *outliers*.
- Foram detectadas observações *outliers* por todos os procedimentos incluindo o sem nuvens. No entanto, tendo em conta o objectivo da classificação em duas classes, as observações detectadas como *outliers* pelos procedimentos que incluem métodos de *clustering* parecem mais razoáveis.
- A análise dos resultados em termos da "junção" dos *outliers* detectados pelos dois métodos de *clustering* torna-se oportuna para grandes conjuntos de dados.

Aplicação do método genérico

Pretende-se agora avaliar o impacto da retirada destas observações na precisão do processo de classificação no conjunto de dados completo testando-se a funcionalidade do método genérico.

Dada a necessidade de avançar com a escolha de uma combinação método de *clustering*/estimador/k, para cada classe, optou-se mais uma vez por recorrer à utilização do critério AIC havendo para isso necessidade de reajustar o critério associado à escolha das observações que irão ser consideradas *outliers*. O AIC indicou como "melhor" combinação o *k-means*/clássico/k=3 para a classe G_1 associada ao algarismo 0 e o *k-means*/clássico/k=5 para a classe G_2 à qual se associa o algarismo 1. No entanto, de acordo com o critério anteriormente adoptado uma observação era considerada *outlier* se fosse detectada para todos os valores de k experimentados com *k-means* ou para todos os valores de k experimentados com o *pam*. Daí resulta a não inclusão de 6 observações detectadas pelo *k-means*/clássico/k=3 e de 2 observações detectadas pelo *k-means*/clássico/k=5. Como neste momento vão ser utilizadas estas duas combinações para que se possa proceder à classificação de acordo com o método genérico proposto, faz então sentido que se juntem mais estas 8 observações aquelas que anteriormente foram consideradas *outliers*. Sendo assim passa-se a trabalhar com 1072 observações na classe G_1 e 1032 na classe G_2.

Com base nas "melhores" combinações encontradas define-se a regra que permite classificar uma nova observação como sendo ou não *outlier*. Nesta situação particular a regra é composta pela conjunção de oito condições do tipo:

$$(\mathbf{x}-\hat{\mu}_{ij})^{\mathrm{T}}\,\hat{\mathbf{\Sigma}}_{ij}(\mathbf{x}-\hat{\mu}_{ij})>\mathrm{c}_{ij},\ i=1,2,\ j=1,...,k_i,$$

onde k_i representa o número de nuvens em cada classe. As estimativas dos parâmetros $(\widehat{\mu}_{ij},\hat{\mathbf{\Sigma}}_{ij})$ e das constantes designadas por c_{ij} são fornecidas pela amostra de treino "limpa". Caso uma nova observação não seja considerada *outlier*, o método prossegue com a classificação em *c*+1 classes. Tendo em conta a função de custos (3.2) definida no Capítulo 3 e tomando um custo de rejeição (indecisão) *t*=0.1, obtém-se a seguinte regra de classificação:

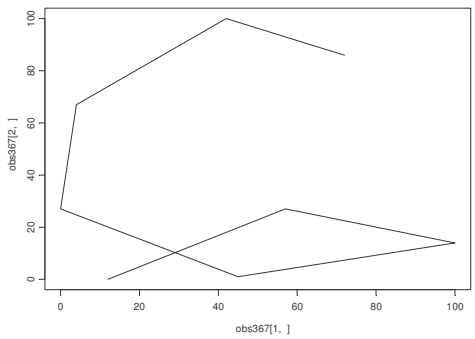
$$\mathbf{x}\in G_1 \text{ se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\geq 8.64$$

$$\mathbf{x}\in G_2 \text{ se} \qquad \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\leq 0.29$$

$$\mathbf{x}\in I \text{ se} \qquad 0.29<\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}<8.64\,.$$

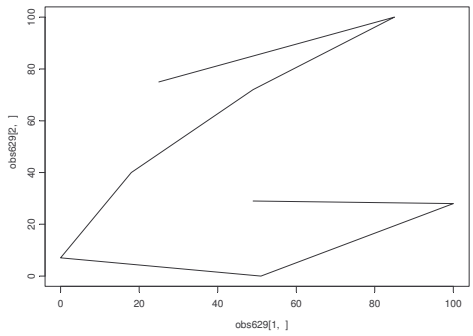
Para estimar as densidades $f_1(\mathbf{x})$ e $f_2(\mathbf{x})$ recorreu-se novamente ao modelo mistura de normais de acordo com a expressão (6.1).

O modelo foi testado em várias observações das quais se escolheu as que se passam a apresentar, juntamente com a respectiva classificação:



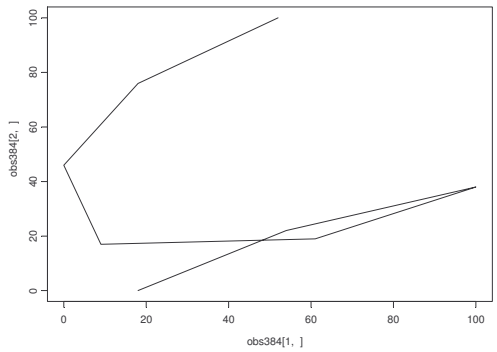
$\Rightarrow$ Classificada em *O*.

Figura 6.18 Observação 367 do algarismo 6.



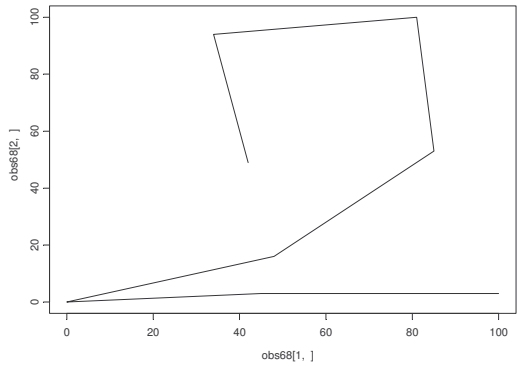
⇒ Classificada em *O*.

Figura 6.19- Observação 629 do algarismo 6.



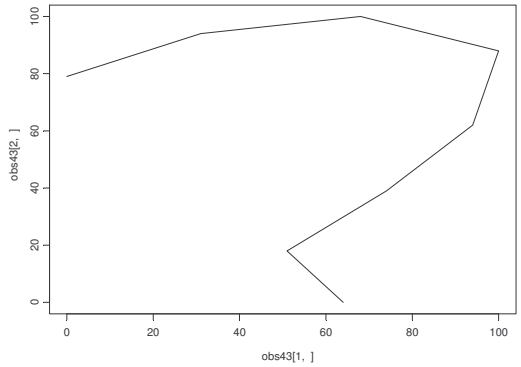
⇒ Classificada em G_1.

Figura 6.20- Observação 384 do algarismo 6.



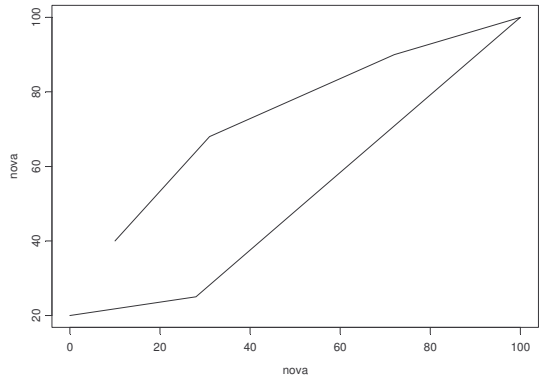
⇒ Classificada em *O*.

Figura 6.21- Observação 68 do algarismo 2.



$\Rightarrow$ Classificada em G_2.

Figura 6.22- Observação 43 do algarismo 2.



$\Rightarrow$ Classificada em I.

Figura 6.23- Nova observação.

Embora se pudesse certamente fazer uma análise mais profunda dos dados, em face dos resultados apresentados só por si, mais uma vez se conclui que o método de classificação proposto revela um bom desempenho.

Capítulo 7

Conclusões e trabalho futuro

Ao longo deste trabalho abordou-se o tema do r.p. de uma forma genérica e em particular a classificação supervisionada. Este capítulo final contempla duas partes. Na primeira começa-se por fazer um breve resumo do trabalho desenvolvido, apresentando-se os resultados e conclusões mais relevantes. De seguida avança-se com algumas linhas de orientação para modificações e novos desafios para futuros trabalhos de investigação, que permitam a continuação do trabalho aqui iniciado e que resultam inevitavelmente da percepção da existência de outros aspectos interessantes que foram surgindo ao longo desta longa caminhada.

Quando se iniciou este trabalho e se começou por fazer uma pesquisa na *internet* sobre o tema do r.p., o número de páginas encontradas era bastante satisfatório, embora em português praticamente não existisse nada. Começavam também a aparecer as grandes referências bibliográficas nesta área, conforme se pode perceber pela análise das datas enunciadas. Entretanto, já bastante adiantado o primeiro capítulo, surge no final de 1999 o livro de Marques (1999) intitulado "Reconhecimento de Padrões: Métodos Estatísticos e Neuronais". Hoje em dia efectuando-se novamente uma pesquisa na *internet*, verifica-se a existência de várias páginas em português onde inclusivamente se deduz da existência de cadeiras, ligadas inteiramente a esse tema, em algumas das nossas licenciaturas. Esta vasta e rápida expansão tem como principal causa, não só o numeroso conjunto de problemas por resolver e o fascínio que deles resulta- que aliás esteve presente ao longo da elaboração deste trabalho- mas principalmente o aparecimento de processadores rápidos, da *internet* e da possibilidade de aumentar o armazenamento em memória acrescido de um menor custo. O avanço da tecnologia computacional tornou possível a implementação de algoritmos de optimização e aprendizagem complexos, impossíveis há algumas décadas atrás. De destacar, como grande candidato de aplicação do r.p, o *data mining*.

Watanabe (1972) escreveu no prefácio do seu livro intitulado "*Frontiers of Pattern Recognition*"

> "*Pattern recognition is a fast-moving and proliferating discipline. It is not easy to form a well-balanced and well-informed summary view of the newest developments in this field. It is still harder to have a vision of its future progress*."

Face ao exposto considera-se esta afirmação perfeitamente actual. Assumindo-se que o primeiro objectivo desta tese (contemplado nos Capítulos 1 e 2) consistia em dar uma visão global sobre o tema do r.p e em particular sobre a classificação supervisonada, com o intuito de introduzir os principais conceitos e discutir com clareza os principais aspectos, na altura pouco contemplados na literatura, por forma a servir de base aos restantes capítulos, considera-se que o objectivo foi cumprido na sua totalidade.

A linha de trabalho seguida a partir do Capítulo 3 teve como propósito a prossecução do segundo objectivo desta tese. Recorde-se,

- desenvolver uma metodologia de classificação de um objecto (padrão) alternativa à clássica com introdução de uma classe de rejeição por indecisão e de uma classe de rejeição por existência de *outliers*, e que seja competitiva em termos de *performance* com as regras de decisão tradicionais.

É em relação a este segundo objectivo que são realmente encontrados os principais desenvolvimentos e resultados originais (Capítulo 3 e seguintes).

Quando se procede a uma classificação de um objecto numa das c classes tradicionalmente consideradas surgem naturalmente várias dúvidas, não só na definição do problema, como também aquando da tomada da decisão. É com frequência questionável a consequência de uma má classificação ou a existência de *outliers* no conjunto de dados. De uma forma superficial, o objectivo da construção de uma regra de decisão é o da classificação correcta de tantos objectos quanto possível. No entanto, este objectivo aparente esconde de facto outra questão com ele relacionada. Será que o objecto pertence necessariamente a uma das c classes postuladas? Esta questão foi a que serviu de ideia base para a concepção deste trabalho.

Dissecando agora a questão da classificação em mais de c classes surge então a nova possibilidade a que se apelidou de "rejeição" e as duas vertentes possíveis: rejeição por indecisão e rejeição por *outliers*. Assim, no terceiro capítulo, generalizou-se a regra de decisão óptima de Chow, considerando-se uma função genérica de custos de classificação correcta e má classificação condicionais às classes, permitindo assim a criação de uma classificação em c+1 classes.

Dos exemplos tratados no Capítulo 3, onde se procede ao cálculo das taxas de erro e rejeição, no contexto da classificação com e sem rejeição, pode-se concluir que:

- os valores da taxa de erro para a situação em que não se inclui opção de rejeição são sempre superiores aqueles que se obtêm quando se introduzem custos de rejeição.

- a construção da curva de compromisso erro/rejeição, em função do limiar de rejeição t, é um procedimento que se revela útil na tomada de decisão.

Relativamente à última secção apresentada no terceiro capítulo, onde se compara a classificação com rejeição, do ponto de vista da generalização da abordagem de Chow e do ponto de vista da abordagem de Tortorella (2000) baseada em curvas ROC, cabe registar as seguintes conclusões:

- teoricamente, as duas abordagens são equivalentes.

- o método baseado na abordagem de Chow é mais genérico, uma vez que tem aplicabilidade independente do número de classes.

- para as combinações de custos adoptadas no exemplo apresentado, a comparação dos resultados com e sem rejeição evidencia que o risco com rejeição é geralmente inferior ao risco sem rejeição.

- a comparação assente nos classificadores linear, logístico e MLP, para as combinações de custos adoptadas, permite concluir que no exemplo apresentado o método baseado na generalização da abordagem de Chow permite atingir menores riscos do que o método de rejeição baseado nas curvas ROC.

No Capítulo 4, onde é tratada a questão da rejeição por *outliers*, propõe-se uma regra de detecção de *outliers* multivariados. É neste capítulo que se dissecam todos os aspectos que servem de suporte à construção e implementação do método. A partir da ideia base é necessário, por um lado, ter em conta vários aspectos como: opção de distância a utilizar, definição formal de *outlier*, utilização de estimadores robustos e partição dos dados em nuvens. Por outro lado, associados aos vários aspectos considerados há que proceder a determinadas escolhas, que foram alvo de análise detalhada e que são responsáveis pelo surgimento de outras questões, tais como: que estimador(es) robusto(s) utilizar, que métodos de partição *clustering* escolher, quantas nuvens devem ser tomadas e que constantes devem ser usadas para funcionar como limites de detecção dos *outliers*. Relativamente à questão dos estimadores robustos salienta-se, como aspecto importante do trabalho, a referência (e utilização no quinto capítulo) de um estimador recente, OGK, devido a Maronna e Zamar (2002).

A regra de detecção de *outliers*, RRO, proposta no quarto capítulo foi testada no Capítulo 5 através da realização de um extenso estudo de simulação realizado com o intuito de comparar o método tradicional, baseado no cálculo das distâncias de Mahalanobis, com o novo método proposto baseado na partição dos dados em nuvens (com a utilização maioritária dos métodos, *k-means*, *pam* e *mclust*) e em estimadores robustos (RMCD25 e OGK) e clássicos. A RRO foi posta à prova numa série de configurações normais e não normais com diferentes dimensões e diferentes percentagens de contaminação. Os resultados obtidos foram medidos em termos dos indicadores de simulação sugeridos por Kosinski (1999) e de um novo indicador, DIF, proposto também neste capítulo, e que resulta de uma combinação dos anteriores. Para a generalidade das configurações e no que diz respeito às opções efectuadas, salientam-se as seguintes conclusões:

- os estimadores clássicos com nuvens têm melhor desempenho que os estimadores robustos com nuvens.
- em termos de estimadores robustos o desempenho do estimador OGK é, em geral, superior ao do estimador RMCD25.
- dos métodos de *clustering* adoptados o *mclust* é o que apresenta um melhor desempenho na maior parte das situações.

Embora não tenha sido possível eleger um único estimador como sendo o melhor, julga-se que este estudo de simulação foi bastante elucidativo. A partir dos resultados obtidos para comparação do método usual de detecção de *outliers* e do novo método proposto, com base nas simulações efectuados no Capítulo 5 e nos exemplos tratados no Capítulo 6, destacam-se as seguintes conclusões gerais:

- o método proposto funciona nas mesmas situações em que o método usual funciona.
- há muitas situações, onde é violada a hipótese dos dados serem elipticamente simétricos, onde o método proposto funciona e o método usual não funciona.

Uma vez apresentadas as conclusões, que abonam a favor do método proposto, cabe também aqui registar a seguinte limitação que se julga ser comum a qualquer método de detecção de *outliers*:

> *" Qualquer que seja o método de detecção de outliers que se crie, será sempre possível inventar um conjunto de dados onde ele falhe".*

Importa também fazer aqui referência ao facto de, apesar do maior esforço desenvolvido se ter centrado na construção rigorosa do método, ter sido também desenvolvido um grande esforço na construção dos programas de implementação do mesmo. No entanto, não foi objecto de análise profunda, a optimização destes mesmos programas. Pretende-se com isto frisar que não houve grande preocupação na minimização das linhas de código já que este ponto não fazia parte dos objectivos deste estudo. Este é um dos aspectos que poderia ser melhorado com vista à minimização dos tempos de cálculo.

Com base nos exemplos tratados no Capítulo 6 e relativamente ao desempenho do método genérico de classificação com rejeição de observações, onde se utilizou o modelo mistura de normais para estimação das densidades condicionais às classes, regista-se a seguinte conclusão final:

- o método de classificação proposto revela-se promissor: é competitivo relativamente aos métodos tradicionais e é aplicável qualquer que seja o conjunto de dados, o método de estimação de densidades e o método de *clustering* adoptado.

De todo o estudo realizado, conclui-se que, com a metodologia proposta e aplicada, os dois principais objectivos propostos foram atingidos.

No contexto do desenvolvimento do método de classificação supervisionada com rejeição por indecisão e observações *outliers*, surgiram naturalmente vários itens que não foram aqui abordados ou que não foram tratados com todo o detalhe e que podem constituir linhas de orientação para futuros trabalhos de investigação, nomeadamente:

- a utilização de outros estimadores robustos tais como os estimadores-S.
- a experimentação de maiores números de nuvens.
- a construção de outros indicadores de simulação ou a utilização de outros critérios já existentes tais como o critério do "tamanho do maior *outlier* não identificável" ("*size of the largest nonidentifiable outlier*") devido a Becker e Gather (2001).
- o estudo do ponto de rotura do método genérico de classificação.[1]
- a utilização de outras percentagens de contaminação e da sua relação com o ponto de rotura.
- o aperfeiçoamento dos programas computacionais com vista à minimização do tempo de cálculo.
- a utilização de diferentes métodos de *clustering* que não obriguem a que os objectos sejam classificados numa classe particular.

Relativamente ao último item pode pensar-se por exemplo na classificação imprecisa (*fuzzy-clustering*), também sugerido por Becker como se pode ver na Secção 5.8. Optando por esse procedimento não há necessidade de classificar um objecto numa nuvem específica ou por adaptação do método numa classe específica. A cada objecto é associada a probabilidade de pertencer a cada uma das nuvens, aquilo a que se apelida de "*membership values*". Na situação em que se está na presença de um *outlier* é provável que este valor seja pequeno para qualquer nuvem considerada. Uma aplicação em que esta ideia tem com certeza interesse é na medicina. Um conjunto de referências que se inserem nesta linha são devidas a Ha (1995, 1996a, 1996b, 1996c).

[1] Note-se que do método proposto apenas se conhece o ponto de rotura do estimador MCD.

Um outro ponto bastante interessante diz respeito ao estudo das propriedades teóricas do método de classificação. No entanto, o facto da *performance* estar bastante dependente da aplicação, torna este estudo bastante complexo.

No contexto da estimação robusta podem ainda ser realizadas outras tentativas tais como a utilização da função de influência.

Para terminar deixa-se um último desafio em relação aos dados: dados com variáveis de vários tipos e dados omissos.

Espera-se que o trabalho desenvolvido nesta tese tenha continuidade em termos de investigação futura mas que seja essencialmente uma contribuição para um melhor entendimento do tema do reconhecimento de padrões e das diversas áreas que este contempla, onde os problemas por solucionar constituem verdadeiros desafios.

"O que nós chamamos resultados são apenas começos".

Ralph Waldo Emerson

Apêndice A

Designações dos métodos estatísticos e neuronais

LDA- *linear discriminant analysis*

QDA- *quadratic discriminant analysis*

Piecewise discriminant analysis

Logistic discriminant analysis / logistic regression

RDA- *regularised discriminant analysis*

MDA- *mixture discriminant analysis*

FDA- *flexible discriminant analysis*

PDA- *penalized discriminant analysis*

SIMCA- *soft independent modelling of class analogy*

DASCO- *discriminant analysis with shrunken covariances*

MLP- *multi-layer perceptron*

RBF- *radial basis function*

MARS- *multivariate adaptative regression splines*

PP- *projection pursuit*

PPR- *projection pursuit regression*

ARVORES / CART- *classification and regression tree s*/ **ID3 / C4.5**

KDA - *kernel discriminant analysis*

RKDA- *reduced kernel discriminant analysis*

K-NN- *K- nearest neigbors*

LVQ- *learning vector quantatization*

Naive Bayes

ACE- *alternating conditional expectation*

DIPOL92 / SMART / ALOCC80 / CASTLE / Kohonen / AC^2 / NEWID

LLR- *local linear regression*

CLAFIC- *class-featuring information compression*

ALSM- *average learning subspace method*

PNN- *probabilistic neural network*

PCR- *principal components regression*

PLS- *partial least squares*

SLS- *shortest least squares*

RR- *ridge regression*

Apêndice B

Aspectos computacionais, programas e tabelas relativas ao Capítulo 3

ASPECTOS COMPUTACIONAIS

Descrevem-se, espera-se que de uma forma sucinta, os programas apresentados neste Apêndice e que serviram de suporte à obtenção dos resultados apresentados na Secção 3.3.3, nomeadamente no que diz respeito à obtenção dos coeficientes estimados, para cada uma das discriminantes, e aos valores das estimativas das diversas taxas de erro e rejeição. São ainda salientados alguns aspectos importantes do ponto de vista da utilização destes programas.

Para todas as discriminantes, fixou-se o limite de t a variar entre 0 e 0.5 (ver justificação na Secção 3.1.3) com incremento de 0.05. Tanto no Exemplo 3.1, como no Exemplo 3.2, foram utilizadas probabilidades *a priori* diferentes, estimadas através da expressão apresentada na Secção 2.5.2 (onde se lê taxa de erro aparente). (Dados CHORON: $\hat{\pi}_1 = 30/113$, Dados PIMA: $\hat{\pi}_1 = 132/200$).

Discriminante Linear (LDA)

Os programas agora apresentados permitem a obtenção dos coeficientes não normalizados da discriminante linear (COEFLDA), os valores dos totais e percentagens de observações mal classificadas em cada uma das classes e da percentagem de observações rejeitadas e mal classificadas, para as amostra de treino e teste, e ainda a obtenção dos valores que funcionam como limiares, t_1 e t_2 para auxilio no cálculo das taxa de erro e rejeição "teóricas". Note-se que embora em algumas situações se opte por trabalhar com os coeficientes normalizados, neste caso, para que fossem utilizadas as expressões apresentadas na secção anterior, houve necessidade de não o fazer.

Um outro aspecto que tem de ser salientado prende-se com cálculo dos valores de t_1 e t_2 apresentados na Secção 3.2.1. A forma como os coeficientes da discriminante linear são aqui calculados, já inclui o logaritmo do quociente das probabilidades *a priori* (no parâmetro que foi designado por "alfa0") pelo que no programa que permite obter o

número e as percentagens de observações mal classificadas e rejeitadas nas amostras de treino e de teste (PREDLDACR e PREDLDACRTE, respectivamente) os valores que tem essa designações correspondem a uma versão "adaptada" (os parâmetros t1 e t2 que dão entrada no programa acabam por ser valores simétricos i.e., t1=- t2).

Programa	***Input***	***Output***
COEFLDA- estimativa dos coeficientes não normalizados da discriminante linear.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1	Coeficientes estimados π_1 Δ (ver Secção 3.3.2)
PREDLDACR- Número e percentagem de observações mal classificadas em cada uma das classes e percentagem de observações rejeitadas, para a <u>amostra de treino</u>.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1 **p**- número de variáveis **t1**- $\ln\left[\frac{1-t}{t}\right]$ **t2**- $\ln\left[\frac{t}{1-t}\right]$	<u>Amostra de treino</u> Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas
PREDLDACRTE- Número e percentagem de observações mal classificadas em cada uma das classes e percentagem de observações rejeitadas, para a <u>amostra de teste</u>.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1 **p**- número de variáveis **dat1**- dados da amostra de teste **nk**- dimensão da amostra de teste **nk1**- número de observações, da amostra de teste, na classe G_1 **nk2**- número de observações, da amostra de teste, na classe G_2 **t1**- $\ln\left[\frac{1-t}{t}\right]$ **t2**- $\ln\left[\frac{t}{1-t}\right]$	<u>Amostra de teste</u> Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas
TISERTEORICO- Valores de t_1 e t_2 que permitem o cálculo das taxas de erro e rejeição "teóricas".	**prior1**- π_1 **dd**- t (custo de rejeição)	t_1 t_2

Discriminante Quadrática (QDA)

Os programas que se passa a apresentar permitem a obtenção dos coeficientes associados à discriminante quadrática (COEFQDA), o cálculo do número e percentagem de observações mal classificadas em cada classe e a percentagem de observações rejeitadas e mal classificadas, quer para a amostra de treino (PREDQDACR) quer para a amostra de teste (PREDQDACRTE).

Programa	***Input***	***Output***
COEFQDA- estimativa dos coeficientes da discriminante quadrática.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1	Coeficientes estimados
PREDQDACR- Número e percentagem de observações mal classificadas, em cada uma das classes e percentagem de observações rejeitadas na amostra de treino.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1 **p**- número de variáveis **t1**- $2\ln\left[\frac{1-t}{t}\right]$ **t2**- $2\ln\left[\frac{t}{1-t}\right]$	Amostra de treino Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas
PREDQDACRTE- Número e percentagem de observações mal classificadas, em cada uma das classes e percentagem de observações rejeitadas na amostra de teste.	**dat**- dados da amostra de treino **k**- vector que contém as dimensões das classes **prior1**- π_1 **p**- número de variáveis **dat1**- dados da amostra de teste **nk**- dimensão da amostra de teste **nk1**- número de observações, da amostra de teste, na classe G_1 **nk2**- número de observações, da amostra de teste, na classe G_2 **t1**- $2\ln\left[\frac{1-t}{t}\right]$ **t2**- $2\ln\left[\frac{t}{1-t}\right]$	Amostra de teste Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas

Em relação aos coeficientes estimados julga-se conveniente detalhar o seu significado e a ordem pela qual aparecem. O resultado retornado pelo programa (ver Apêndice B) indica os valores que são designados por "limiar", "tindep", "q_1" e "q_2" que correspondem, respectivamente, ao limiar de decisão (sem consideração da rejeição

i.e., feitas as contas, $2\ln\frac{\pi_2}{\pi_1}-\ln\frac{|\Sigma_2|}{|\Sigma_1|}$), ao termo independente, aos coeficientes dos termos x_1^2, x_1x_2 e x_2^2 (designação "q_1") e aos coeficientes dos termos x_1 e x_2 (designação "q_2"). Tal como na discriminante linear, uma vez que os coeficientes já contemplam o termo independente, os parâmetros designados agora por t1 e t2 correspondem a valores diferentes dos que aparecem nas fórmulas apresentadas anteriormente.

Discriminante Logística

Para obtenção dos coeficientes da discriminante utilizou-se a função GLM incluída no S-Plus. De destacar que os coeficientes vem com os sinais contrários à forma como foram escritas as expressões ao longo desta dissertação (ver, por exemplo, Secção 3.2.1). Quer-se com isto salientar que nesta situação particular e tomando o caso mais simples em que não se inclui a rejeição e em que as probabilidades *a priori* são iguais, os objectos são classificados em G_1 se $d(\mathbf{x})\leq 0$. Um outro aspecto que é importante referir é que esta função do S-Plus já inclui, no cálculo dos coeficientes discriminantes, os valores das probabilidades *a priori* proporcionais aos tamanhos das amostras i.e., calculados da forma que já tem vindo a ser utilizada. Os coeficientes que esta função retorna não são normalizados o que é precisamente o que se pretende da forma como o problema foi definido.

São agora apresentados os programas que permitem calcular o número e a percentagem de observações mal classificadas em cada classe e a percentagem de observações rejeitadas e mal classificadas, quer para a amostra de treino (PREDLOGCR) quer para a amostra de teste (PREDLOGCRTE). Neste caso, o cálculo dos coeficientes discriminantes antecede a utilização destes programas.

Programa	***Input***	***Output***
PREDLOGCR- Número e percentagem de observações mal classificadas em cada uma das classes e percentagem de observações rejeitadas, para a amostra de treino.	**res**- *output* da função GLM para obtenção dos coeficientes discriminantes com a amostra de treino **dat-** dados da amostra de treino **p**- número de variáveis **k**- vector que contém as dimensões das classes **t1**- $\ln\left[\frac{1-t}{t}\right]$ **t2**- $\ln\left[\frac{t}{1-t}\right]$	Amostra de treino Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas
PREDLOGCRTE- Número e percentagem de observações mal classificadas em cada uma das classes e percentagem de observações rejeitadas, para a amostra de teste.	**res**- *output* da função GLM para obtenção dos coeficientes discriminantes com a amostra de treino **dat1-** dados da amostra de teste **p**- número de variáveis **nk**- dimensão da amostra de teste **nk1**- número de observações, da amostra de teste, na classe G_1 **nk2**- número de observações, da amostra de teste, na classe G_2 **t1**- $\ln\left[\frac{1-t}{t}\right]$ **t2**- $\ln\left[\frac{t}{1-t}\right]$	Amostra de teste Número de observações mal classificadas em G_1 Número de observações mal classificadas em G_2 Percentagem de observações mal classificadas em G_1 Percentagem de observações mal classificadas em G_2 Percentagem de observações mal classificadas Percentagem de observações rejeitadas

PROGRAMAS

COEFLDA- Estimativa dos coeficientes não normalizados da discriminante linear.

```
function(dat,k,prior1)
{
        dat<-as.matrix(dat)
        m<-ncol(dat)
        n1<-k[1]
        n2<-k[2]
        n<-n1+n2
        if (n!=nrow(dat)) stop("matriz dos dados não coincide com vector de
        dimensoes")

                medg1 <- apply(dat[1:n1,],2,mean)
                covg1 <- var(dat[1:n1,])
                medg2 <- apply(dat[(n1+1):(n),] ,2,mean)
                covg2 <- var(dat[(n1+1):(n),])
                poolcov<-((n1-1)*covg1+(n2-1)*covg2)/(n-2)
                poolinv<-solve(poolcov)
                difmed<-(medg1-medg2)
                somed<-(medg1+medg2)
                alfa<-poolinv%*%difmed
                dm<-t(difmed)%*%poolinv%*%difmed
                alfa0<-as.numeric((t(alfa)%*%somed))
                alfa0<-((-0.5)*alfa0-log((1-prior1)/prior1))
                alfa<-c(alfa0,alfa,prior1,dm)
                return(alfa)
        }
```

PREDLDACR- Número e percentagem de observações mal classificadas, pela discriminante linear, em cada uma das classes e percentagem de observações rejeitadas na amostra de treino.

```
function(dat,k,prior1,p,t1,t2)
{
        res<-lda(dat,k,prior1)
        res<-as.matrix(res)
        # tirar o primeiro elemento do resultado da discriminante linear (termo
        independente)
        res1<-res[1:p+1,]
        res1<-as.matrix(res1)
        # vector para ser somado aos scores
        n1<-k[1]
        n2<-k[2]
        n<-n1+n2
        const<-rep(res[1,1],n)
        const<-as.matrix(const)
        score<-dat%*%res1+const
        scg1<-score[1:n1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(n1+1):(n),]
        scg2<-as.matrix(scg2)
        counts1<-0
        for(i in 1:n1) {if (scg1[i,1]<t2) counts1<-counts1+1}
        counts2<-0
        for(i in 1:n2) {if (scg2[i,1]>t1) counts2<-counts2+1}
        counts3<-0
        for(i in 1:n) {if (score[i,1]>t2 & score[i,1]<t1) counts3<-counts3+1}
        perrej<-counts3/n
        perg1<-counts1/n1
        perg2<-counts2/n2
        percentagem<-(counts1+counts2)/n
        solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

PREDLDACRTE- Número e percentagem de observações mal classificadas, pela discriminante linear, em cada uma das classes e percentagem de observações rejeitadas na amostra de teste.

```
function(dat,k,prior1,p,dat1,nk,nk1,nk2,t1,t2)
{
        res<-lda(dat,k,prior1)
        res<-as.matrix(res)
        # tirar o primeiro elemento do resultado da discriminante linear (termo
        independente)
        res1<-res[1:p+1,]
        res1<-as.matrix(res1)
        # vector para ser somado aos scores
        const<-rep(res[1,1],nk)
        const<-as.matrix(const)
        score<-dat1%*%res1+const
        scg1<-score[1:nk1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(nk1+1):(nk),]
        scg2<-as.matrix(scg2)
        counts1<-0
        for(i in 1:nk1) {if (scg1[i,1]<t2) counts1<-counts1+1}
        counts2<-0
        for(i in 1:nk2) {if (scg2[i,1]>t1) counts2<-counts2+1}
        counts3<-0
        for(i in 1:nk) {if (score[i,1]>t2 & score[i,1]<t1) counts3<-counts3+1}
    perrej<-counts3/nk
        perg1<-counts1/nk1
    perg2<-counts2/nk2
        percentagem<-(counts1+counts2)/nk
    solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

TISERTEORICO- Valores de t_1 e t_2 que permitem o cálculo das taxas de erro e rejeição "teóricas".

```
function(prior1,dd)
{
        t1<-0
        t2<-0
        t1<-log(((1-prior1)*(1-dd))/(prior1*dd))
        t2<-log(((1-prior1)*dd)/(prior1*(1-dd)))
        resultado<-c(t1,t2)
        return(resultado)
}
```

COEFQDA- Estimativa dos coeficientes da discriminante quadrática.

```
function(dat,k,prior1)
{
        dat<-as.matrix(dat)
        m<-ncol(dat)
        n1<-k[1]
        n2<-k[2]
        n<-n1+n2
        if (n!=nrow(dat)) stop("matriz dos dados não coincide com vector de dimensões
        dos grupos")
        medg1 <- apply(dat[1:n1,],2,mean)
        covg1 <- var(dat[1:n1,])
        covg1inv<-solve(covg1)
        medg2 <- apply(dat[(n1+1):(n),] ,2,mean)
        covg2 <- var(dat[(n1+1):(n),])
        covg2inv<-solve(covg2)
        difcovinv<-(covg1inv-covg2inv)
        cm1<-covg1inv%*%medg1
        cm2<-covg2inv%*%medg2
        q1<--difcovinv
        q2<-2*(cm1-cm2)
        q3<-t(cm1)%*%medg1
        q4<-t(cm2)%*%medg2
        tindep<--q3+q4
        library(Matrix)
        detg1<-det.Matrix(covg1,logarithm=F)
        detg2<-det.Matrix(covg2,logarithm=F)
        logaritmo<-log(detg2$modulus/detg1$modulus)
        ppriori<--2*log((1-prior1)/prior1)
        # limiar k com os sinais referentes ao primeiro membro
        limiar<-ppriori+logaritmo
        limiar<-as.matrix(limiar)
        coeficientes<-c(limiar,tindep,q1,q2)
        return(coeficientes)

}
```

PREQDACR- Número e percentagem de observações mal classificadas, pela discriminante quadrática, em cada uma das classes e percentagem de observações rejeitadas na amostra de treino.

```
function(dat,k,prior1,p,t1,t2)
{
        res<-qda(dat,k,prior1)
        res<-as.matrix(res)
        n1<-k[1]
        n2<-k[2]
        n<-n1+n2
        # tirar limiar e repetir
        limiar<-0
        limiar<-rep(res[1,1],n)
        limiar<-as.matrix(limiar)
        # tirar o termo independente e repetir
        tindep<-0
        tindep<-rep(res[2,1],n)
        tindep<-as.matrix(tindep)
        # tirar q2 (coeficientes de xi) e multiplicar
        coefx<-res[3:(p+2),1]
        coefx<-as.matrix(coefx)
        rescoefx<-dat%*%coefx
        #transformação da matrix dat para coeficientes de xixj
        matrix1<-as.matrix(dat)
        p2<-p^2
        matrix2<-matrix(n,p2)
        matrix2<-matrix(matrix2,n,p2)
        for(i in 1:n)
        {
        for(j in 1:p)
                {
                for(k in 1:p)
                        {
                        matrix2[i,(p*(j-1))+k]<-matrix1[i,j]*matrix1[i,k]
                        }
                }

        }
        matrix2<-as.matrix(matrix2)
        coefxixj<-res[(p+3):(2+p+p2),1]
        rescoefxixj<-matrix2%*%coefxixj
        score<-tindep+rescoefx+rescoefxixj+limiar

        scg1<-score[1:n1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(n1+1):(n),]
        scg2<-as.matrix(scg2)
```

```
        counts1<-0
        for(i in 1:n1) {if (scg1[i,1]<2*t2) counts1<-counts1+1}
        counts2<-0
        for(i in 1:n2) {if (scg2[i,1]>2*t1) counts2<-counts2+1}
        counts3<-0
        for(i in 1:n) {if (score[i,1]>2*t2 & score[i,1]<2*t1) counts3<-counts3+1}
        perrej<-counts3/n
        perg1<-counts1/n1
        perg2<-counts2/n2
        percentagem<-(counts1+counts2)/n
        solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

PREQDACRTE- Número e percentagem de observações mal classificadas, pela discriminante quadrática, em cada uma das classes, e percentagem de observações rejeitadas na amostra de teste.

```
function(dat,k,prior1,dat1,nk1,nk2,nk,p,t1,t2)
{
        res<-qda(dat,k,prior1)
        res<-as.matrix(res)
        # tirar limiar e repetir
        limiar<-0
        limiar<-rep(res[1,1],nk)
        limiar<-as.matrix(limiar)
        # tirar o termo independente e repetir
        tindep<-0
        tindep<-rep(res[2,1],nk)
        tindep<-as.matrix(tindep)
        # tirar q2 (coeficientes de xi) e multiplicar
        coefx<-res[3:(p+2),1]
        coefx<-as.matrix(coefx)
        rescoefx<-dat1%*%coefx
        #transformação da matrix dat para coeficientes de xixj
        matrix1<-as.matrix(dat1)
        p2<-p^2
        matrix2<-matrix(nk,p2)
        matrix2<-matrix(matrix2,nk,p2)
        for(i in 1:nk)
        {
        for(j in 1:p)
                {
                for(k in 1:p)
                        {
                        matrix2[i,(p*(j-1))+k]<-matrix1[i,j]*matrix1[i,k]
                        }
                }
```

```
        }
        matrix2<-as.matrix(matrix2)
        coefxixj<-res[(p+3):(2+p+p2),1]
        rescoefxixj<-matrix2%*%coefxixj
        score<-tindep+rescoefx+rescoefxixj+limiar
        scg1<-score[1:nk1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(nk1+1):(nk),]
        scg2<-as.matrix(scg2)
        counts1<-0
        for(i in 1:nk1) {if (scg1[i,1]<2*t2) counts1<-counts1+1}
        counts2<-0
        for(i in 1:nk2) {if (scg2[i,1]>2*t1) counts2<-counts2+1}
        counts3<-0
        for(i in 1:nk) {if (score[i,1]>2*t2 & score[i,1]<2*t1) counts3<-counts3+1}
        perrej<-counts3/nk
        perg1<-counts1/nk1
        perg2<-counts2/nk2
        percentagem<-(counts1+counts2)/nk
       solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

Coeficientes da discriminante logística

```
# colocar a variável que indica a classe na última coluna do ficheiro de dados
attach(dat)
log.dat<-glm(classe~V1+V2+...+Vp, family=binomial)
res<-summary(log.dat)
```

PREDLOGCR- Número e percentagem de observações mal classificadas, pela discriminante logística, em cada uma das classes e percentagem de observações rejeitadas na amostra de treino.

```
function(res,dat,p,k,t1,t2)
{

        res.log<-log.dat$coefficients
        res.log<-as.matrix(res.log)
        n1<-k[1]
        n2<-k[2]
        n<-n1+n2
        # tirar o primeiro elemento do resultado da discriminante logística(termo
        independente)
        res1.log<-res.log[1:p+1,]
        # vector para ser somado aos scores
        const<-rep(res.log[1,1],n)
        const<-as.matrix(const)
        score<-dat%*%res1.log+const
        scg1<-score[1:n1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(n1+1):(n),]
        scg2<-as.matrix(scg2)
        counts1<-0
        for(i in 1:n1) {if (scg1[i,1]>t1) counts1<-counts1+1}
        counts2<-0
        for(i in 1:n2) {if (scg2[i,1]<t2) counts2<-counts2+1}
        counts3<-0
        for(i in 1:n) {if (score[i,1]>t2 & score[i,1]<t1) counts3<-counts3+1}
        perrej<-counts3/n
        perg1<-counts1/n1
        perg2<-counts2/n2
        percentagem<-(counts1+counts2)/n
        solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

PREDLOGCRTE- Número e percentagem de observações mal classificadas, pela discriminante logística, em cada uma das classes e percentagem de observações rejeitadas na amostra de teste.

```
function(res,dat1,p,nk,nk1,nk2,t1,t2)
{

        res.log<-log.dat$coefficients
        res.log<-as.matrix(res.log)
        # tirar o primeiro elemento do resultado da discriminante logística(termo
        independente)
        res1.log<-res.log[1:p+1,]
        # vector para ser somado aos scores
        const<-rep(res.log[1,1],nk)
        const<-as.matrix(const)
        score<-dat1%*%res1.log+const
        scg1<-score[1:nk1,]
        scg1<-as.matrix(scg1)
        scg2<-score[(nk1+1):(nk),]
        scg2<-as.matrix(scg2)
        counts1<-0
        for(i in 1:nk1) {if (scg1[i,1]>t1) counts1<-counts1+1}
        counts2<-0
        for(i in 1:nk2) {if (scg2[i,1]<t2) counts2<-counts2+1}
        counts3<-0
        for(i in 1:nk) {if (score[i,1]>t2 & score[i,1]<t1) counts3<-counts3+1}
        perrej<-counts3/nk
        perg1<-counts1/nk1
        perg2<-counts2/nk2
        percentagem<-(counts1+counts2)/nk
        solucao<-c(counts1,counts2,perg1,perg2,percentagem,perrej)
        return(solucao)
}
```

TABELAS

t	t1	t2	E	Taxa de erro aparente Amostra de treino	R	Taxa de rejeição aparente Amostra de treino
0	+∞	-∞	0	0	1	1
0.05	3.962081	-1.926796	0.007494	(0+1) 0.008850	0.603216	0.557522
0.10	3.214866	-1.179583	0.018420	(2+1) 0.026549	0.452864	0.415929
0.15	2.752243	-0.716958	0.030365	(6+1) 0.061947	0.356211	0.274336
0.20	2.403936	-0.0368652	0.043045	(9+1) 0.088496	0.283354	0.168142
0.25	2.116254	-0.080970	0.056447	(9+1) 0.088496	0.223606	0.150442
0.30	1.864940	0.170344	0.070651	(10+1) 0.097345	0.171858	0.115044
0.35	1.636681	0.398602	0.085789	(10+1) 0.097345	0.125228	0.088496
0.40	1.423107	0.816971	0.102038	(11+2) 0.115044	0.080047	0.061947
0.45	1.218313	0.612177	0.119625	(13+3) 0.141593	0.040474	0.026549
0.50	1.017642	1.017642	0.138848	**(13+3)** **0.141593**	0	0

Tabela B.1: Taxas de erro e rejeição "teóricas" e estimadas (amostra de treino) com os dados CHORON para a discriminante linear.

t	t1	Taxa de erro aparente Amostra de treino	Taxa de erro Amostra simulada	Taxa de rejeição aparente Amostra de treino	Taxa de rejeição Amostra simulada
0	+∞	0	0	1	1
0.05	2.94444	(1+2) 0.026549	(40+12) 0.026	0.451327	0.427
0.10	2.197225	(1+2) 0.026549	(67+12) 0.0395	0.380531	0.3590
0.15	1.734601	(2+2) 0.035398	(96+18) 0.0570	0.283186	0.2955
0.20	1.386294	(4+2) 0.053097	(145+22) 0.0835	0.203540	0.2175
0.25	1.098612	(5+2) 0.061947	(165+25) 0.095	0.176991	0.1820
0.30	0.847298	(5+3) 0.070796	(183+30) 0.1065	0.150442	0.1485
0.35	0.619039	(5+3) 0.070796	(195+41) 0.118	0.141593	0.1180
0.40	0.405465	(6+5) 0.097345	(212+43) 0.1275	0.088496	0.0935
0.45	0.200671	(8+5) 0.115044	(239+51) 0.145	0.070796	0.058
0.50	0	(10+8) **0.159292**	(266+73) **0.1695**	0	0

Tabela B.2: Taxas de erro e rejeição estimadas (amostra de treino e amostra "simulada") com os dados CHORON para a discriminante quadrática.

t	t1	Taxa de erro aparente Amostra de treino	E	Taxa de rejeição aparente Amostra de treino	R
0	+∞	0	0	1	1
0.05	2.944444	(0+1) 0.008885	0.009289	0.619469	0.610944
0.10	2.197225	(1+1) 0.017699	0.021453	0.486726	0.460759
0.15	1.734601	(2+1) 0.026549	0.034328	0.398230	0.363301
0.20	1.386294	(3+1) 0.035398	0.047771	0.327434	0.289429
0.25	1.098612	(5+1) 0.053097	0.061836	0.238938	0.228628
0.30	0.847298	(7+2) 0.079646	0.076637	0.141593	0.175839
0.35	0.619039	(9+3) 0.106195	0.092328	0.088496	0.128189
0.40	0.405465	(10+3) 0.115044	0.109100	0.053097	0.083832
0.45	0.200671	(10+3) 0.115044	0.127192	0.017699	0.041452
0.50	0	(11+3) **0.123894**	**0.146908**	0	0

Tabela B.3: Taxas de erro e rejeição "teóricas" e estimadas (amostra de treino) com os dados CHORON para a discriminante logística.

t	t1	t2	E	Taxa de erro Amostra de teste	R	Taxa de rejeição Amostra de teste
0	+∞	-∞	0	0	1	1
0.05	2.281145	-3.607742	0.004438	(3+1) 0.012048	0.827609	0.786145
0.10	1.533930	-2.860520	0.015378	(3+2) 0.015060	0.676328	0.620482
0.15	1.071307	-2.397896	0.030099	(5+9) 0.042169	0.555888	0.478916
0.20	0.723000	-2.049589	0.047612	(7+11) 0.054217	0.454342	0.367470
0.25	0.435318	-1.761907	0.067459	(9+18) 0.081325	0.365104	0.295181
0.30	0.184004	-1.510592	0.089555	(11+24) 0.105422	0.284133	0.228916
0.35	-0.044255	-1.282333	0.113902	(13+28) 0.123494	0.208820	0.177711
0.40	-0.257829	-1.068759	0.140635	(16+33) 0.147590	0.137289	0.114458
0.45	-0.462623	-0.863965	0.169992	(21+37) 0.174699	0.068088	0.054217
0.50	-0.663294	-0.663294	**0.202316** **Amostra treino:** **0.235 (18+29)**	**(25+42)** **0.201807**	0	0

Tabela B.4: Taxas de erro e rejeição "teóricas" e estimadas (amostra de teste) com os dados PIMA para a discriminante linear.

t	t1	Taxa de erro Amostra de treino	Taxa de erro Amostra de teste	Taxa de erro Amostra simulada	Taxa de rejeição Amostra de treino	Taxa de rejeição Amostra de teste	Taxa de rejeição Amostra simulada
0	+∞	0	0	0	1	1	1
0.05	2.94444	(3+1) 0.02	(7+6) 0.039157	(3+12) 0.0075	0.66	0.629518	0.6980
0.10	2.197225	(5+4) 0.045	(8+14) 0.066265	(5+46) 0.0255	0.465	0.445783	0.5595
0.15	1.734601	(8+7) 0.075	(11+20) 0.093373	(10+72) 0.0410	0.35	0.340361	0.4625
0.20	1.386294	(9+9) 0.09	(16+25) 0.123494	(14+105) 0.0595	0.29	0.262048	0.373
0.25	1.098612	(10+17) 0.135	(18+29) 0.141566	(25+140) 0.0825	0.195	0.210843	0.294
0.30	0.847298	(12+18) 0.15	(22+33) 0.165663	(37+176) 0.1065	0.15	0.132530	0.226
0.35	0.619039	(13+20) 0.165	(23+35) 0.174699	(54+211) 0.1325	0.115	0.093373	0.1695
0.40	0.405465	(16+21) 0.185	(28+43) 0.213855	(75+240) 0.1575	0.07	0.042169	0.1145
0.45	0.200671	(17+22) 0.195	(28+45) 0.219880	(91+274) 0.1825	0.03	0.033133	0.0535
0.50	0	(17+25) **0.21**	(31+48) **0.237952**	(102+317) **0.2095**	0	0	0

Tabela B.5: Taxas de erro e rejeição estimadas (amostra de treino, de teste e amostra "simulada") com os dados PIMA para a discriminante quadrática.

t	t1	Taxa de erro Amostra de teste	Taxa de erro aparente Amostra de treino	E	Taxa de rejeição Amostra de teste	Taxa de rejeição aparente Amostra de treino	R
0	+∞	0	0	0	1	1	1
0.05	2.944444	(3+1) 0.012048	(0+0) 0	0.003766	0.828313	0.9	0.871985
0.10	2.197225	(3+1) 0.012048	(0+3) 0.015	0.013600	0.680723	0.69	0.721942
0.15	1.734601	(3+7) 0.030120	(1+3) 0.02	0.027245	0.539157	0.565	0.594722
0.20	1.386294	(4+9) 0.039157	(2+7) 0.045	0.043842	0.424699	0.45	0.485552
0.25	1.098612	(9+16) 0.075301	(3+10) 0.065	0.063069	0.322289	0.365	0.389317
0.30	0.847298	(12+20) 0.096386	(4+15) 0.095	0.084857	0.250000	0.27	0.302274
0.35	0.619039	(13+27) 0.120482	(7+19) 0.13	0.109287	0.183735	0.195	0.221600
0.40	0.405465	(16+29) 0.135542	(12+23) 0.175	0.136555	0.132530	0.12	0.145565
0.45	0.200671	(21+35) 0.168675	(13+26) 0.195	0.166969	0.054217	0.06	0.063066
0.50	0	(24+41) 0.195783	(15+29) **0.22**	0.200953	0	0	0

Tabela B.6: Taxas de erro e rejeição "teóricas" e estimadas (amostra de treino e de teste) com os dados PIMA para a discriminante logística.

Apêndice C

Programas relativos ao Capítulo 4

OGK- Estimativas do vector de médias e da matriz de covariâncias associadas ao estimador OGK e correspondentes pesos.

Programa principal: final (estimativas finais, com duas iterações e reponderação- aqui é realizada a iteração 2.

```
final<-function(dat,beta)
 {
      # Vai buscar o Z da iteracção 1
      ajudametodo<-metodo(dat)
      ZZ1<-ajudametodo$ZZ
      # Aplica o método a Z em vez de X
      iteracao2<-metodo(ZZ1)
      ajudaVdex2<-iteracao2$Vdex
      ajudatdex2<-iteracao2$tdex
      ajudaAA<-ajudametodo$AA
      Vdex2<-ajudaAA%*%ajudaVdex2%*%t(ajudaAA)
      tdex2<-ajudaAA%*%ajudatdex2
      # Executa a reponderação
      p<-ncol(dat)
      dist2<-mahalanobis(dat,tdex2,Vdex2)
      dis<-dist2*qchisq(0.5,p)/median(dist2)
      wZZ<-ifelse(dis<qchisq(beta,p),1,0)
      resultado<-cov.wt(dat,wt=wZZ)
      vw<-resultado$cov
      tw<-resultado$center
       list(vw=vw,tw=tw,OGK.wt=wZZ)
 }
```

Subrotina: metodo (iteracção 1).

```
metodo<-function(dat)
{
     dat<-as.matrix(dat)
     n<-nrow(dat)
     p<-ncol(dat)
     # Obtenção da matriz D
     resultado<-escapos(dat,3,4.5)
     sigma<-resultado$sigma
```

```
    sigma<-as.vector(sigma)
    DD<-diag(sigma)
    invDD<-solve(DD)
    # Obtenção da matriz Y
    YY<-matrix(0,n,p)
    YY<-as.matrix(YY)
    YY<-dat%*%invDD
    # Obtenção da matriz U
    UU1<-matrix(0,p,p)
    UU2<-matrix(0,p,p)
    UU<-matrix(1,p,p)
    i<-1
    while (i<=p-1)
    {
        j<-i+1
        while(j<=p)
        {
            resultado1<-escapos(YY[,i]+YY[,j],3,4.5)
            UU1[i,j]<-resultado1$sigma
            resultado2<-escapos(YY[,i]-YY[,j],3,4.5)
            UU2[i,j]<-resultado2$sigma
            UU[i,j]<-(UU1[i,j]^2-UU2[i,j]^2)/4
            UU[j,i]<-UU[i,j]
            j<-j+1
        }
        i<-i+1
    }
    # Cálculo dos valores e vectores próprios de U
    VPROP<-eigen(UU)
    # Obtenção da matriz dos vectores próprios de U: E
    EE<-VPROP$vectors
    # Obtenção da matriz A
    AA<-DD%*%EE
    # Obtenção da matriz Z
    ZZ<-YY%*%EE
    # Obtenção da matriz tau
    restau<-escapos(ZZ,3,4.5)
    sigma<-restau$sigma
    sigma<-as.vector(sigma)
    tau<-diag(sigma)
    tau<-tau%*%tau
    # Cálculo de v torto
    vtorto<-restau$mu
    vtorto<-t(vtorto)
    # Obtenção do estimador da matriz de covariâncias
    Vdex<-AA%*%tau%*%t(AA)
    # Obtenção do estimador do vector de médias
    tdex<-AA%*%vtorto
    list (Vdex=Vdex,tdex=tdex,ZZ=ZZ,AA=AA)
}
```

***Subrotina*:** escapos (calcula as estimativas iniciais de μ e σ).

```
escapos<-function(x,kesca,kposi)
{
    x<-as.matrix(x)
    n<-nrow(x)
    p<-ncol(x)
    sigma0<-matrix(0,1,p)
    resi<-matrix(0,n,p)
    pesoposi<-matrix(0,n,p)
    mu<-matrix(0,1,p)
    sigma<-matrix(0,1,p)
    for (i in 1:p)
    {
      sigma0[1,i]<-median(abs(x[1:n,i]-median(x[1:n,i])))
      resi[,i]<-abs(x[1:n,i]-median(x[1:n,i]))/sigma0[1,i]
      pesoposi[,i]<-(1-(resi[,i]/kposi)**2)**2*(resi[,i] <= kposi)
      mu[1,i]<-sum(pesoposi[,i]*t(x[,i]))/sum(pesoposi[,i])
      sigma[1,i]<-
      mean(resi[,i]**2*(resi[,i]**2<=kesca**2)+kesca**2*(resi[,i]**2>kesca**2))
      sigma[1,i]<-sigma0[1,i]*sqrt(sigma[1,i])
    }
    list(mu=mu,sigma=sigma)
}
```

Apêndice D

Tabelas relativas ao Capítulo 5

Este apêndice contém as tabelas seguintes relativas aos resultados provenientes do estudo de simulação (100 simulações para cada configuração):

Tabela D.1: Resultados da simulação para a configuração Normal
Tabela D.2: Resultados da simulação para a configuração LUAN3
Tabela D.3: Resultados da simulação para a configuração CRUZU
Tabela D.4: Resultados da simulação para a configuração CRUZU com 10 *outliers*
Tabela D.5: Resultados da simulação para a configuração CRUZU com 30 *outliers*
Tabela D.6: Resultados da simulação para a configuração CRUZN
Tabela D.7: Resultados da simulação para a configuração LUAT3
Tabela D.8: Resultados da simulação para a configuração LUAT3 com 10 *outliers*
Tabela D.9: Resultados da simulação para a configuração LUAT3 com 30 *outliers*
Tabela D.10: Resultados da simulação para a configuração ESFERA
Tabela D.11: Resultados da simulação para a configuração ESFERA com o dobro dos dados e 2v.
Tabela D.12: Resultados da simulação para a configuração DONUT
Tabela D.13: Resultados da simulação para a configuração T1
Tabela D.14: Resultados da simulação para a configuração T2

Legenda das tabelas

nd- indicador **n**ão **d**efinido
"melhor" combinação
"pior" combinação

Tabela D.1: Resultados da simulação para a configuração Normal

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**
	k-means	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.002	0.78	**0.0**
N	"	**4**	nd	nd	0.005	0.53	**0.0**	nd	nd	0.004	0.58	**0.0**	nd	nd	0.001	0.87	**0.0**
SO	"	**5**	nd	nd	0.003	0.67	**0.0**	nd	nd	0.004	0.73	**0.0**	nd	nd	0.002	0.85	**0.0**
2v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.002	0.78	**0.0**
	"	**4**	nd	nd	0.005	0.58	**0.0**	nd	nd	0.005	0.5	**0.0**	nd	nd	0.002	0.78	**0.0**
	"	**5**	nd	nd	0.004	0.65	**0.0**	nd	nd	0.003	0.64	**0.0**	nd	nd	0.001	0.81	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.001	0.93	**0.0**
	"	**4**	nd	nd	0.003	0.75	**0.0**	nd	nd	0.003	0.73	**0.0**	nd	nd	0	0.95	**0.0**
	"	**5**	nd	nd	0.002	0.78	**0.0**	nd	nd	0.002	0.8	**0.0**	nd	nd	0.001	0.92	**0.0**
	s/ nuvens	**1**	nd	nd	0.001	0.92	**0.0**	nd	nd	0	0.94	**0.0**	nd	nd	0.001	0.91	**0.0**
	k-means	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.002	0.76	**0.0**
N	"	**4**	1	0	0.006	0.51	**0.0**	1	0	0.005	0.56	**0.0**	1	0	0.002	0.69	**0.0**
CO	"	**5**	1	0	0.004	0.62	**0.0**	1	0	0.006	0.52	**0.0**	1	0	0.001	0.85	**0.0**
2v	***pam***	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.002	0.73	**0.0**
	"	**4**	1	0	0.005	0.54	**0.0**	1	0	0.005	0.58	**0.0**	1	0	0.002	0.75	**0.0**
	"	**5**	1	0	0.003	0.66	**0.0**	1	0	0.005	0.57	**0.0**	1	0	0.002	0.75	**0.0**
	mclust	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.001	0.85	**0.0**
	"	**4**	1	0	0.003	0.72	**0.0**	1	0	0.003	0.71	**0.0**	1	0	0.001	0.82	**0.0**
	"	**5**	1	0	0.003	0.72	**0.0**	1	0	0.003	0.73	**0.0**	1	0	0.001	0.93	**0.0**
	s/ nuvens	**1**	1	0	0.001	0.88	**0.0**	1	0	0.001	0.87	**0.0**	0.84	0.16	0.001	0.93	**5.5**
	k-means	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.001	0.91	**0.0**
N	"	**4**	nd	nd	0.001	0.94	**0.0**	nd	nd	0.001	0.87	**0.0**	nd	nd	0	0.95	**0.0**
SO	"	**5**	nd	nd	0	0.99	**0.0**	nd	nd	0.002	0.95	**0.0**	nd	nd	0	0.99	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.001	0.91	**0.0**
	"	**4**	nd	nd	0.001	0.93	**0.0**	nd	nd	0.001	0.89	**0.0**	nd	nd	0.001	0.93	**0.0**
	"	**5**	nd	nd	0	0.95	**0.0**	nd	nd	0.001	0.91	**0.0**	nd	nd	0	0.99	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	0.98	**0.0**
	"	**4**	nd	nd	0.001	0.91	**0.0**	nd	nd	0.001	0.85	**0.0**	nd	nd	0	1	**0.0**
	"	**5**	nd	nd	0.001	0.94	**0.0**	nd	nd	0.002	0.92	**0.0**	nd	nd	0.001	0.98	**0.0**
	s/ nuvens	**1**	nd	nd	0	0.95	**0.0**	nd	nd	0.001	0.9	**0.0**	nd	nd	0.001	0.91	**0.0**
	k-means	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.003	0.67	**0.0**
N	"	**4**	1	0	0.007	0.44	**0.0**	1	0	0.005	0.54	**0.0**	1	0	0.003	0.69	**0.0**
CO	"	**5**	1	0	0.005	0.57	**0.0**	1	0	0.003	0.74	**0.0**	1	0	0.001	0.82	**0.0**
4v	***pam***	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.002	0.76	**0.0**
	"	**4**	1	0	0.005	0.52	**0.0**	1	0	0.005	0.54	**0.0**	1	0	0.002	0.81	**0.0**
	"	**5**	1	0	0.004	0.65	**0.0**	1	0	0.005	0.52	**0.0**	1	0	0.001	0.87	**0.0**
	mclust	**3**	1	0	0	1	**0.0**	1	0	0	1	**0.0**	1	0	0.002	0.8	**0.0**
	"	**4**	1	0	0.005	0.64	**0.0**	0.99	0.01	0.004	0.68	**0.0**	1	0	0.001	0.87	**0.0**
	"	**5**	1	0	0.003	0.71	**0.0**	1	0	0.002	0.82	**0.0**	1	0	0.001	0.89	**0.0**
	s/ nuvens	**1**	1	0	0.001	0.84	**0.0**	1	0	0.001	0.91	**0.0**	0.6	0.4	0.004	0.94	**17.5**

Tabela D.2: Resultados da simulação para a configuração LUAN3

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**
	k-means	**3**	nd	nd	0.199	0.14	**14.9**	nd	nd	0.163	0.31	**11.3**	nd	nd	0.063	0.6	**1.3**
NN	"	**4**	nd	nd	0.017	0.77	**0.0**	nd	nd	0.016	0.72	**0.0**	nd	nd	0.037	0.72	**0.0**
SO	"	**5**	nd	nd	0.005	0.82	**0.0**	nd	nd	0.007	0.72	**0.0**	nd	nd	0.022	0.77	**0.0**
2v	***pam***	**3**	nd	nd	0.193	0.16	**14.3**	nd	nd	0.124	0.47	**7.4**	nd	nd	0.029	0.74	**0.0**
LUAN3	"	**4**	nd	nd	0.012	0.73	**0.0**	nd	nd	0.016	0.69	**0.0**	nd	nd	0.03	0.63	**0.0**
	"	**5**	nd	nd	0.006	0.7	**0.0**	nd	nd	0.009	0.77	**0.0**	nd	nd	0.036	0.73	**0.0**
	mclust	**3**	nd	nd	0.046	0.81	**0.0**	nd	nd	0.039	0.83	**0.0**	nd	nd	0.041	0.73	**0.0**
	"	**4**	nd	nd	0.018	0.71	**0.0**	nd	nd	0.003	0.89	**0.0**	nd	nd	0.044	0.75	**0.0**
	"	**5**	nd	nd	0.001	0.83	**0.0**	nd	nd	0.001	0.85	**0.0**	nd	nd	0.024	0.84	**0.0**
	s/ nuvens	**1**	nd	nd	0.001	0.9	**0.0**	nd	nd	0	0.98	**0.0**	nd	nd	0	0.99	**0.0**
	k-means	**3**	0.94	0.06	0.213	0.06	**8.7**	0.82	0.18	0.188	0.18	**13.4**	0.14	0.86	0.035	0.81	**40.5**
NN	"	**4**	0.38	0.62	0.104	0.52	**31.2**	0.4	0.59	0.036	0.56	**27.0**	0	1	0.067	0.61	**48.4**
CO	"	**5**	0.5	0.5	0.023	0.75	**22.5**	0.45	0.55	0.009	0.75	**25.0**	0.51	0.49	0.039	0.72	**22.0**
2v	***pam***	**3**	0.89	0.11	0.207	0.11	**10.9**	0.61	0.39	0.146	0.39	**21.8**	0.16	0.84	0.042	0.71	**39.5**
LUAN3	"	**4**	0.84	0.16	0.169	0.23	**11.5**	0.88	0.12	0.124	0.34	**7.2**	0.78	0.22	0.051	0.69	**8.6**
	"	**5**	0.83	0.17	0.008	0.72	**6.0**	0.76	0.24	0.013	0.73	**9.5**	1	0.01	0.029	0.72	**0.0**
	mclust	**3**	0.21	0.79	0.054	0.78	**37.2**	0.08	0.92	0.026	0.9	**43.5**	0.17	0.83	0.047	0.75	**39.0**
	"	**4**	0.99	0.01	0.026	0.8	**0.0**	1	0	0.029	0.78	**0.0**	0.99	0.01	0.043	0.73	**0.0**
	"	**5**	0.94	0.06	0.009	0.8	**0.5**	0.89	0.11	0.014	0.93	**3.0**	1	0	0.014	0.85	**0.0**
	s/ nuvens	**1**	0	1	0	0.96	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	k-means	**3**	nd	nd	0.181	0.25	**13.1**	nd	nd	0.051	0.78	**0.1**	nd	nd	0.018	0.78	**0.0**
NN	"	**4**	nd	nd	0.028	0.71	**0.0**	nd	nd	0.001	0.85	**0.0**	nd	nd	0.026	0.82	**0.0**
SO	"	**5**	nd	nd	0.008	0.85	**0.0**	nd	nd	0.004	0.8	**0.0**	nd	nd	0.04	0.76	**0.0**
4v	***pam***	**3**	nd	nd	0.124	0.47	**7.4**	nd	nd	0.051	0.79	**0.1**	nd	nd	0.016	0.8	**0.0**
LUAN3	"	**4**	nd	nd	0.007	0.82	**0.0**	nd	nd	0.002	0.76	**0.0**	nd	nd	0.037	0.81	**0.0**
	"	**5**	nd	nd	0.005	0.89	**0.0**	nd	nd	0.003	0.77	**0.0**	nd	nd	0.033	0.76	**0.0**
	mclust	**3**	nd	nd	0.034	0.86	**0.0**	nd	nd	0.007	0.97	**0.0**	nd	nd	0.006	0.9	**0.0**
	"	**4**	nd	nd	0.007	0.88	**0.0**	nd	nd	0.004	0.83	**0.0**	nd	nd	0.012	0.85	**0.0**
	"	**5**	nd	nd	0.002	0.91	**0.0**	nd	nd	0.003	0.93	**0.0**	nd	nd	0.013	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0.001	0.9	**0.0**	nd	nd	0	0.97	**0.0**	nd	nd	0	0.97	**0.0**
	k-means	**3**	0.76	0.24	0.205	0.21	**17.3**	0.24	0.76	0.058	0.76	**35.9**	0.07	0.93	0.019	0.82	**44.0**
NN	"	**4**	0.13	0.87	0.045	0.57	**41.0**	0.03	0.97	0.017	0.71	**46.0**	0.01	0.99	0.074	0.75	**48.2**
CO	"	**5**	0.16	0.84	0.015	0.69	**39.5**	0.09	0.91	0.012	0.8	**43.0**	0.26	0.74	0.04	0.79	**34.5**
4v	***pam***	**3**	0.66	0.34	0.168	0.33	**20.4**	0.18	0.82	0.043	0.82	**38.5**	0.04	0.96	0.011	0.85	**45.5**
LUAN3	"	**4**	0.88	0.12	0.131	0.36	**7.6**	0.89	0.11	0.032	0.77	**3.0**	0.9	0.1	0.011	0.86	**2.5**
	"	**5**	0.74	0.26	0.013	0.79	**10.5**	0.53	0.47	0.001	0.82	**21.0**	0.98	0.02	0.018	0.85	**0.0**
	mclust	**3**	0.23	0.77	0.291	0.53	**48.1**	0.07	0.93	0.167	0.72	**49.9**	0.18	0.82	0.171	0.64	**44.6**
	"	**4**	1	0	0.013	0.84	**0.0**	0.96	0.04	0.006	0.88	**0.0**	0.99	0.01	0.009	0.88	**0.0**
	"	**5**	0.94	0.06	0.007	0.92	**0.5**	0.77	0.23	0.001	0.93	**9.0**	0.98	0.02	0.015	0.9	**0.0**
	s/ nuvens	**1**	0	1	0.001	0.9	**47.5**	0	1	0	0.95	**47.5**	0	1	0	0.96	**47.5**

Tabela D.3: Resultados da simulação para a configuração CRUZU

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.004	0.97	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.47	**0.0**
NN	"	**4**	nd	nd	0.039	0.21	**0.0**	nd	nd	0.035	0.17	**0.0**	nd	nd	0.009	0.77	**0.0**
SO	"	**5**	nd	nd	0.013	0.65	**0.0**	nd	nd	0.006	0.54	**0.0**	nd	nd	0.005	0.9	**0.0**
2v	***pam***	**3**	nd	nd	0.006	0.96	**0.0**	nd	nd	0.003	0.98	**0.0**	nd	nd	0.015	0.32	**0.0**
CRUZU	"	**4**	nd	nd	0.038	0.09	**0.0**	nd	nd	0.031	0.13	**0.0**	nd	nd	0.10	0.69	**5.0**
	"	**5**	nd	nd	0.010	0.44	**0.0**	nd	nd	0.006	0.55	**0.0**	nd	nd	0.004	0.92	**0.0**
	mclust	**3**	nd	nd	0.001	0.99	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.10	0.76	**5.0**
	"	**4**	nd	nd	0.018	0.25	**0.0**	nd	nd	0.014	0.33	**0.0**	nd	nd	0.002	0.71	**0.0**
	"	**5**	nd	nd	0.009	0.48	**0.0**	nd	nd	0.006	0.57	**0.0**	nd	nd	0.002	0.88	**0.0**
	s/ nuvens	**1**	nd	nd	0.219	0	**16.9**	nd	nd	0.067	0	**1.7**	nd	nd	0.006	0.48	**0.0**
	k-means	**3**	0.31	0.273	0.064	0.33	**11.9**	0.21	0.322	0.019	0.43	**13.6**	0.17	0.16	0.008	0.44	**5.5**
NN	"	**4**	0.57	0.033	0.146	0.01	**4.8**	0.51	0.065	0.062	0.08	**1.4**	0.49	0.06	0.009	0.44	**0.5**
CO	"	**5**	0.61	0.027	0.124	0.01	**3.7**	0.67	0.022	0.058	0.05	**0.4**	0.41	0.049	0.011	0.46	**0 0**
2v	***pam***	**3**	0.12	0.462	0.022	0.66	**20.6**	0.07	0.433	0.02	0.7	**19.2**	0.03	0.362	0.024	0.71	**15.6**
CRUZU	"	**4**	0.24	0.194	0.084	0.15	**8.9**	0.23	0.22	0.042	0.3	**8.5**	0.1	0.338	0.003	0.75	**14.4**
	"	**5**	0.26	0.178	0.064	0.18	**7.1**	0.23	0.251	0.032	0.37	**10.1**	0.07	0.245	0.004	0.75	**9.8**
	mclust	**3**	0.12	0.722	0.001	0.92	**33.6**	0.16	0.664	0.001	0.92	**30.7**	0.17	0.699	0.001	0.93	**32.5**
	"	**4**	0.2	0.679	0.001	0.92	**31.5**	0.16	0.675	0.002	0.91	**31.3**	0.13	0.602	0.001	0.9	**27.6**
	"	**5**	0.14	0.642	0.002	0.84	**29.6**	0.24	0.589	0.002	0.83	**27.0**	0.28	0.424	0.005	0.69	**18.7**
	s/ nuvens	**1**	0.79	0.012	0.227	0	**8.9**	0.53	0.032	0.071	0.06	**1.1**	0.26	0.061	0.007	0.38	**0.6**
	k-means	**3**	nd	nd	0.003	0.96	**0.0**	nd	nd	0.001	0.99	**0.0**	nd	nd	0.003	0.77	**0.0**
NN	"	**4**	nd	nd	0.017	0.49	**0.0**	nd	nd	0.009	0.59	**0.0**	nd	nd	0.002	0.95	**0.0**
SO	"	**5**	nd	nd	0.006	0.86	**0.0**	nd	nd	0.005	0.77	**0.0**	nd	nd	0.006	0.87	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.53	**0.0**
CRUZU	"	**4**	nd	nd	0.011	0.43	**0.0**	nd	nd	0.007	0.44	**0.0**	nd	nd	0.001	0.83	**0.0**
	"	**5**	nd	nd	0.001	0.85	**0.0**	nd	nd	0.002	0.82	**0.0**	nd	nd	0.003	0.95	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.003	0.89	**0.0**
	"	**4**	nd	nd	0.007	0.57	**0.0**	nd	nd	0.004	0.66	**0.0**	nd	nd	0.004	0.84	**0.0**
	"	**5**	nd	nd	0.006	0.74	**0.0**	nd	nd	0.005	0.75	**0.0**	nd	nd	0.005	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0.126	0.03	**7.6**	nd	nd	0.036	0.11	**0.0**	nd	nd	0.003	0.63	**0.0**
	k-means	**3**	0.84	0.012	0.031	0.14	**0.0**	0.84	0.008	0.012	0.33	**0.0**	0.72	0.016	0.005	0.55	**0.0**
NN	"	**4**	0.9	0.005	0.041	0.13	**0.0**	0.91	0.005	0.022	0.2	**0.0**	0.75	0.015	0.003	0.77	**0.0**
CO	"	**5**	0.89	0.006	0.055	0.07	**0.3**	0.86	0.008	0.017	0.27	**0.0**	0.77	0.012	0.009	0.61	**0.0**
4v	***pam***	**3**	0.73	0.067	0.036	0.13	**0.9**	0.87	0.023	0.02	0.19	**0.0**	0.53	0.1	0.004	0.63	**2.5**
CRUZU	"	**4**	0.68	0.021	0.046	0.06	**0.0**	0.85	0.009	0.025	0.14	**0.0**	0.32	0.145	0.006	0.76	**4.8**
	"	**5**	0.68	0.028	0.037	0.12	**0.0**	0.89	0.006	0.021	0.2	**0.0**	0.2	0.131	0.002	0.78	**4.1**
	mclust	**3**	0.42	0.486	0	0.97	**21.8**	0.38	0.494	0.001	0.92	**22.2**	0.36	0.434	0	0.93	**19.2**
	"	**4**	0.19	0.581	0	0.97	**26.6**	0.33	0.465	0.001	0.94	**20.8**	0.4	0.384	0.002	0.75	**16.7**
	"	**5**	0.3	0.529	0	0.99	**24.0**	0.33	0.464	0.001	0.89	**20.7**	0.51	0.209	0.004	0.63	**8.0**
	s/ nuvens	**1**	0.95	0.003	0.118	0.03	**3.4**	0.92	0.005	0.033	0.09	**0.0**	0.78	0.012	0.003	0.65	**0.0**

Tabela D.4: Resultados da simulação para a configuração CRUZU com 10 *outliers*

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.004	0.97	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.47	**0.0**
NN	"	**4**	nd	nd	0.039	0.21	**0.0**	nd	nd	0.035	0.17	**0.0**	nd	nd	0.009	0.77	**0.0**
SO	"	**5**	nd	nd	0.013	0.65	**0.0**	nd	nd	0.006	0.54	**0.0**	nd	nd	0.005	0.9	**0.0**
2v	***pam***	**3**	nd	nd	0.006	0.96	**0.0**	nd	nd	0.003	0.98	**0.0**	nd	nd	0.015	0.32	**0.0**
CRUZU	"	**4**	nd	nd	0.038	0.09	**0.0**	nd	nd	0.031	0.13	**0.0**	nd	nd	0.10	0.69	**5.0**
10	"	**5**	nd	nd	0.010	0.44	**0.0**	nd	nd	0.006	0.55	**0.0**	nd	nd	0.004	0.92	**0.0**
	mclust	**3**	nd	nd	0.001	0.99	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.10	0.76	**5.0**
	"	**4**	nd	nd	0.018	0.25	**0.0**	nd	nd	0.014	0.33	**0.0**	nd	nd	0.002	0.71	**0.0**
	"	**5**	nd	nd	0.009	0.48	**0.0**	nd	nd	0.006	0.57	**0.0**	nd	nd	0.002	0.88	**0.0**
	s/ nuvens	**1**	nd	nd	0.219	0	**16.9**	nd	nd	0.067	0	**1.7**	nd	nd	0.006	0.48	**0.0**
	k-means	**3**	0.53	0.054	0.086	0.04	**2.0**	0.69	0.043	0.027	0.12	**0.0**	0.56	0.066	0.009	0.35	**20.2**
NN	"	**4**	0.86	0.014	0.118	0	**3.4**	0.73	0.32	0.045	0.05	**13.5**	0.64	0.041	0.009	0.49	**20.6**
CO	"	**5**	0.73	0.033	0.1	0	**2.5**	0.72	0.031	0.035	0.08	**0.0**	0.66	0.043	0.009	0.44	**2.5**
2v	***pam***	**3**	0.32	0.561	0.034	0.55	**25.6**	0.37	0.461	0.032	0.48	**20.6**	0.39	0.236	0.012	0.52	**21.3**
CRUZU	"	**4**	0.79	0.023	0.056	0.03	**0.3**	0.69	0.035	0.046	0.07	**0.0**	0.45	0.193	0.014	0.7	**28.2**
10	"	**5**	0.7	0.035	0.031	0.16	**0.0**	0.69	0.04	0.024	0.15	**0.0**	0.38	0.174	0.001	0.83	**26.0**
	mclust	**3**	0.22	0.664	0.002	0.92	**30.7**	0.22	0.718	0.002	0.91	**33.4**	0.26	0.651	0	0.95	**32.5**
	"	**4**	0.32	0.534	0.004	0.85	**24.2**	0.32	0.458	0.004	0.74	**20.4**	0.35	0.472	0.002	0.86	**32.5**
	"	**5**	0.32	0.51	0.002	0.82	**23.0**	0.39	0.425	0.003	0.78	**18.8**	0.4	0.399	0.005	0.68	**20.8**
	s/ nuvens	**1**	0.83	0.019	0.229	0	**9.0**	0.76	0.027	0.081	0	**1.6**	0.5	0.069	0.006	0.44	**0.3**
	k-means	**3**	nd	nd	0.003	0.96	**0.0**	nd	nd	0.001	0.99	**0.0**	nd	nd	0.003	0.77	**0.0**
NN	"	**4**	nd	nd	0.017	0.49	**0.0**	nd	nd	0.009	0.59	**0.0**	nd	nd	0.002	0.95	**0.0**
SO	"	**5**	nd	nd	0.006	0.86	**0.0**	nd	nd	0.005	0.77	**0.0**	nd	nd	0.006	0.87	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.53	**0.0**
CRUZU	"	**4**	nd	nd	0.011	0.43	**0.0**	nd	nd	0.007	0.44	**0.0**	nd	nd	0.001	0.83	**0.0**
10	"	**5**	nd	nd	0.001	0.85	**0.0**	nd	nd	0.002	0.82	**0.0**	nd	nd	0.003	0.95	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.003	0.89	**0.0**
	"	**4**	nd	nd	0.007	0.57	**0.0**	nd	nd	0.004	0.66	**0.0**	nd	nd	0.004	0.84	**0.0**
	"	**5**	nd	nd	0.006	0.74	**0.0**	nd	nd	0.005	0.75	**0.0**	nd	nd	0.005	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0.126	0.03	**7.6**	nd	nd	0.036	0.11	**0.0**	nd	nd	0.003	0.63	**0.0**
	k-means	**3**	0.92	0.009	0.034	0.13	**0.0**	0.93	0.01	0.015	0.2	**0.0**	0.91	0.011	0.003	0.71	**1.3**
NN	"	**4**	0.9	0.011	0.046	0.1	**0.0**	0.94	0.007	0.016	0.22	**0.0**	0.94	0.006	0.005	0.64	**0.0**
CO	"	**5**	0.91	0.01	0.058	0.08	**0.4**	0.91	0.009	0.02	0.23	**0.0**	0.91	0.013	0.013	0.52	**0.0**
4v	***pam***	**3**	0.5	0.437	0.03	0.43	**19.4**	0.52	0.453	0.018	0.47	**20.2**	0.65	0.109	0.003	0.62	**5.3**
CRUZU	"	**4**	0.78	0.034	0.03	0.21	**0.0**	0.9	0.011	0.025	0.15	**0.0**	0.32	0.238	0.004	0.78	**3.3**
10	"	**5**	0.71	0.04	0.018	0.37	**0.0**	0.95	0.005	0.01	0.44	**0.0**	0.33	0.207	0.002	0.8	**3.4**
	mclust	**3**	0.5	0.464	0.001	0.92	**20.7**	0.57	0.367	0.002	0.9	**15.9**	0.58	0.402	0.001	0.97	**25.5**
	"	**4**	0.57	0.366	0	0.88	**15.8**	0.63	0.33	0.002	0.74	**14.0**	0.62	0.308	0.001	0.88	**16.7**
	"	**5**	0.39	0.492	0.002	0.86	**22.1**	0.58	0.335	0.001	0.9	**14.3**	0.74	0.198	0.003	0.75	**13.4**
	s/ nuvens	**1**	0.98	0.002	0.114	0.05	**3.2**	0.96	0.004	0.028	0.01	**0.0**	0.96	0.004	0.003	0.62	**0.0**

Tabela D.5: Resultados da simulação para a configuração CRUZU com 30 *outliers*

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.004	0.97	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.47	**0.0**
NN	"	**4**	nd	nd	0.039	0.21	**0.0**	nd	nd	0.035	0.17	**0.0**	nd	nd	0.009	0.77	**0.0**
SO	"	**5**	nd	nd	0.013	0.65	**0.0**	nd	nd	0.006	0.54	**0.0**	nd	nd	0.005	0.9	**0.0**
2v	***pam***	**3**	nd	nd	0.006	0.96	**0.0**	nd	nd	0.003	0.98	**0.0**	nd	nd	0.015	0.32	**0.0**
CRUZU	"	**4**	nd	nd	0.038	0.09	**0.0**	nd	nd	0.031	0.13	**0.0**	nd	nd	0.10	0.69	**5.0**
30	"	**5**	nd	nd	0.010	0.44	**0.0**	nd	nd	0.006	0.55	**0.0**	nd	nd	0.004	0.92	**0.0**
	mclust	**3**	nd	nd	0.001	0.99	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.10	0.76	**5.0**
	"	**4**	nd	nd	0.018	0.25	**0.0**	nd	nd	0.014	0.33	**0.0**	nd	nd	0.002	0.71	**0.0**
	"	**5**	nd	nd	0.009	0.48	**0.0**	nd	nd	0.006	0.57	**0.0**	nd	nd	0.002	0.88	**0.0**
	s/ nuvens	**1**	nd	nd	0.219	0	**16.9**	nd	nd	0.067	0	**1.7**	nd	nd	0.006	0.48	**0.0**
	k-means	**3**	0.08	0.711	0.018	0.8	**33.1**	0	0.817	0.002	0.92	**38.4**	0.02	0.453	0.002	0.81	**20.2**
NN	"	**4**	0.3	0.451	0.104	0.23	**22.8**	0.18	0.508	0.038	0.39	**22.9**	0.14	0.462	0.016	0.66	**20.6**
CO	"	**5**	0.47	0.124	0.141	0.06	**8.3**	0.39	0.151	0.048	0.14	**5.1**	0.19	0.099	0.011	0.39	**2.5**
2v	***pam***	**3**	0	0.789	0.001	0.94	**37.0**	0	0.794	0.001	0.94	**37.2**	0.02	0.476	0.021	0.91	**21.3**
CRUZU	"	**4**	0.06	0.601	0.109	0.23	**30.5**	0.05	0.722	0.023	0.68	**33.6**	0.01	0.614	0	0.95	**28.2**
30	"	**5**	0.03	0.604	0.062	0.33	**28.3**	0.02	0.662	0.023	0.53	**30.6**	0	0.569	0.001	0.87	**26.0**
	mclust	**3**	0.07	0.753	0	0.96	**35.2**	0.06	0.734	0.001	0.93	**34.2**	0.13	0.699	0.002	0.89	**32.5**
	"	**4**	0.14	0.674	0.001	0.91	**31.2**	0.14	0.642	0.002	0.86	**29.6**	0.11	0.7	0.001	0.9	**32.5**
	"	**5**	0.08	0.655	0.002	0.82	**30.3**	0.09	0.646	0.001	0.83	**29.8**	0.25	0.466	0.003	0.72	**20.8**
	s/ nuvens	**1**	0.6	0.018	0.231	0	**9.1**	0.39	0.03	0.074	0.01	**1.2**	0.21	0.056	0.005	0.51	**0.3**
	k-means	**3**	nd	nd	0.003	0.96	**0.0**	nd	nd	0.001	0.99	**0.0**	nd	nd	0.003	0.77	**0.0**
NN	"	**4**	nd	nd	0.017	0.49	**0.0**	nd	nd	0.009	0.59	**0.0**	nd	nd	0.002	0.95	**0.0**
SO	"	**5**	nd	nd	0.006	0.86	**0.0**	nd	nd	0.005	0.77	**0.0**	nd	nd	0.006	0.87	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.007	0.53	**0.0**
CRUZU	"	**4**	nd	nd	0.011	0.43	**0.0**	nd	nd	0.007	0.44	**0.0**	nd	nd	0.001	0.83	**0.0**
30	"	**5**	nd	nd	0.001	0.85	**0.0**	nd	nd	0.002	0.82	**0.0**	nd	nd	0.003	0.95	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.003	0.89	**0.0**
	"	**4**	nd	nd	0.007	0.57	**0.0**	nd	nd	0.004	0.66	**0.0**	nd	nd	0.004	0.84	**0.0**
	"	**5**	nd	nd	0.006	0.74	**0.0**	nd	nd	0.005	0.75	**0.0**	nd	nd	0.005	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0.126	0.03	**7.6**	nd	nd	0.036	0.11	**0.0**	nd	nd	0.003	0.63	**0.0**
	k-means	**3**	0.67	0.14	0.022	0.38	**4.5**	0.6	0.17	0.009	0.44	**6.0**	0.63	0.076	0.004	0.61	**1.3**
NN	"	**4**	0.87	0.004	0.053	0.01	**0.2**	0.91	0.003	0.021	0.2	**0.0**	0.64	0.014	0.006	0.61	**0.0**
CO	"	**5**	0.74	0.01	0.042	0.14	**0.0**	0.81	0.006	0.022	0.22	**0.0**	0.69	0.015	0.015	0.64	**0.0**
4v	***pam***	**3**	0.41	0.22	0.019	0.45	**8.5**	0.35	0.205	0.007	0.51	**7.8**	0.29	0.156	0.004	0.64	**5.3**
CRUZU	"	**4**	0.6	0.04	0.061	0.07	**0.6**	0.55	0.124	0.027	0.23	**3.7**	0.37	0.115	0.003	0.74	**3.3**
30	"	**5**	0.64	0.039	0.042	0.11	**0.0**	0.68	0.044	0.019	0.26	**0.0**	0.25	0.118	0.002	0.78	**3.4**
	mclust	**3**	0.22	0.599	0	0.96	**27.5**	0.21	0.572	0.001	0.94	**26.1**	0.25	0.56	0	0.92	**25.5**
	"	**4**	0.18	0.619	0	0.97	**28.5**	0.31	0.501	0	0.94	**22.6**	0.35	0.384	0	0.8	**16.7**
	"	**5**	0.27	0.537	0	0.94	**24.4**	0.28	0.519	0.001	0.89	**23.5**	0.33	0.318	0.003	0.78	**13.4**
	s/ nuvens	**1**	0.85	0.005	0.117	0.02	**3.4**	0.88	0.004	0.035	0.13	**0.0**	0.82	0.006	0.004	0.62	**0.0**

Tabela D.6: Resultados da simulação para a configuração CRUZN

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.004	0.97	**0.0**	0	0	0	1	**0.0**	nd	nd	0.007	0.47	**0.0**
NN	"	**4**	nd	nd	0.039	0.21	**0.0**	0	0	0.019	0.36	**0.0**	nd	nd	0.009	0.77	**0.0**
SO	"	**5**	nd	nd	0.013	0.65	**0.0**	0	0	0.002	0.88	**0.0**	nd	nd	0.005	0.9	**0.0**
2v	***pam***	**3**	nd	nd	0.006	0.96	**0.0**	0	0	0	1	**0.0**	nd	nd	0.015	0.32	**0.0**
CRUZN	"	**4**	nd	nd	0.038	0.09	**0.0**	0	0	0.022	0.11	**0.0**	nd	nd	0.10	0.69	**5.0**
	"	**5**	nd	nd	0.010	0.44	**0.0**	0	0	0.003	0.87	**0.0**	nd	nd	0.004	0.92	**0.0**
	mclust	**3**	nd	nd	0.001	0.99	**0.0**	0	0	0	1	**0.0**	nd	nd	0.10	0.76	**5.0**
	"	**4**	nd	nd	0.018	0.25	**0.0**	0	0	0.008	0.53	**0.0**	nd	nd	0.002	0.71	**0.0**
	"	**5**	nd	nd	0.009	0.48	**0.0**	0	0	0.002	0.83	**0.0**	nd	nd	0.002	0.88	**0.0**
	s/ nuvens	**1**	nd	nd	0.219	0	**16.9**	0	0	0.059	0.03	**0.9**	nd	nd	0.006	0.48	**0.0**
	k-means	**3**	0.01	0.99	0.005	0.96	**47.0**	0	1	0.004	0.96	**47.5**	0	1	0.008	0.42	**47.5**
NN	"	**4**	0.04	0.95	0.003	0.05	**45.0**	0.09	0.909	0.074	0.07	**44.2**	0.02	0.98	0.008	0.62	**46.5**
CO	"	**5**	0.18	0.82	0.062	0.08	**39.1**	0.17	0.829	0.047	0.09	**39.0**	0.08	0.92	0.009	0.58	**43.5**
2v	***pam***	**3**	0	1	0.011	0.93	**47.5**	0	1	0	1	**47.5**	0	1	0.003	0.69	**47.5**
CRUZN	"	**4**	0.07	0.93	0.089	0.03	**46.0**	0.07	0.929	0.065	0.02	**44.7**	0.02	0.98	0.004	0.83	**46.5**
	"	**5**	0.3	0.7	0.032	0.26	**32.5**	0.15	0.849	0.026	0.12	**40.0**	0.1	0.9	0.009	0.59	**42.5**
	mclust	**3**	0.99	0.01	0	1	**0.0**	0.91	0.09	0	1	**2.0**	0.77	0.23	0.002	0.83	**9.0**
	"	**4**	0.89	0.11	0.018	0.49	**3.0**	0.91	0.09	0.01	0.51	**2.0**	0.74	0.26	0.003	0.84	**10.5**
	"	**5**	0.87	0.13	0.025	0.23	**4.0**	0.94	0.06	0.016	0.33	**0.5**	0.84	0.16	0.003	0.85	**5.5**
	s/ nuvens	**1**	0.12	0.88	0.042	0.25	**41.5**	0	1	0.016	0.23	**47.5**	0	1	0.004	0.54	**47.5**
	k-means	**3**	nd	nd	0.003	0.96	**0.0**	0	0	0.001	0.99	**0.0**	nd	nd	0.003	0.77	**0.0**
NN	"	**4**	nd	nd	0.017	0.49	**0.0**	0	0	0.013	0.5	**0.0**	nd	nd	0.002	0.95	**0.0**
SO	"	**5**	nd	nd	0.006	0.86	**0.0**	0	0	0.004	0.84	**0.0**	nd	nd	0.006	0.87	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	0	0	0	1	**0.0**	nd	nd	0.007	0.53	**0.0**
CRUZN	"	**4**	nd	nd	0.011	0.43	**0.0**	0	0	0.014	0.3	**0.0**	nd	nd	0.001	0.83	**0.0**
	"	**5**	nd	nd	0.001	0.85	**0.0**	0	0	0.003	0.84	**0.0**	nd	nd	0.003	0.95	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	0	0	0	1	**0.0**	nd	nd	0.003	0.89	**0.0**
	"	**4**	nd	nd	0.007	0.57	**0.0**	0	0	0.009	0.51	**0.0**	nd	nd	0.004	0.84	**0.0**
	"	**5**	nd	nd	0.006	0.74	**0.0**	0	0	0.008	0.71	**0.0**	nd	nd	0.005	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0.126	0.03	**7.6**	0	0	0.060	0.05	**1.0**	nd	nd	0.003	0.63	**0.0**
	k-means	**3**	0	1	0	0.99	**47.5**	0	1	0	1	**47.5**	0	1	0.004	0.62	**47.5**
NN	"	**4**	0	1	0.048	0.12	**47.5**	0	1	0.041	0.2	**47.5**	0.01	0.99	0.005	0.72	**47.0**
CO	"	**5**	0.03	0.97	0.016	0.41	**46.0**	0.04	0.959	0.03	0.12	**45.5**	0.07	0.93	0.006	0.71	**44.0**
4v	***pam***	**3**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0.002	0.79	**47.5**
CRUZN	"	**4**	0.03	0.97	0.036	0.12	**46.0**	0.02	0.98	0.03	0.12	**46.5**	0.01	0.99	0.001	0.82	**47.0**
	"	**5**	0.03	0.97	0.009	0.6	**46.0**	0.06	0.94	0.007	0.54	**44.5**	0.07	0.93	0.003	0.64	**44.0**
	mclust	**3**	0.95	0.05	0	1	**0.0**	0.89	0.11	0	1	**3.0**	0.83	0.17	0.001	0.94	**6.0**
	"	**4**	0.8	0.2	0.004	0.75	**7.5**	0.88	0.12	0.003	0.74	**3.5**	0.95	0.05	0.004	0.96	**0.0**
	"	**5**	0.57	0.43	0.005	0.64	**19.0**	0.87	0.13	0.004	0.59	**4.0**	0.95	0.05	0	0.94	**0.0**
	s/ nuvens	**1**	0	1	0.011	0.23	**47.5**	0	1	0.008	0.34	**47.5**	0	1	0.003	0.63	**47.5**

Tabela D.7: Resultados da simulação para a configuração LUAT3

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	DIF	p_1	p_2	p_3	p_4	DIF	p_1	p_2	p_3	p_4	DIF
	k-means	**3**	nd	nd	0.134	0.25	**8.4**	nd	nd	0.129	0.28	**7.9**	nd	nd	0.029	0.03	**0.0**
NN	"	**4**	nd	nd	0.071	0.01	**2.1**	nd	nd	0.055	0	**0.5**	nd	nd	0.036	0.01	**0.0**
SO	"	**5**	nd	nd	0.059	0	**0.9**	nd	nd	0.055	0.03	**0.5**	nd	nd	0.039	0.03	**0.0**
2v	***pam***	**3**	nd	nd	0.045	0.8	**0.0**	nd	nd	0.066	0.69	**1.6**	nd	nd	0.039	0	**0.0**
LUAT3	"	**4**	nd	nd	0.047	0	**0.0**	nd	nd	0.044	0	**0.0**	nd	nd	0.059	0.02	**0.9**
	"	**5**	nd	nd	0.038	0	**0.0**	nd	nd	0.048	0	**0.0**	nd	nd	0.037	0.03	**0.0**
	mclust	**3**	nd	nd	0.008	0.95	**0.0**	nd	nd	0.011	0.94	**0.0**	nd	nd	0.023	0.17	**0.0**
	"	**4**	nd	nd	0.021	0.26	**0.0**	nd	nd	0.026	0.19	**0.0**	nd	nd	0.018	0.56	**0.0**
	"	**5**	nd	nd	0.013	0.378	**0.0**	nd	nd	0.021	0.28	**0.0**	nd	nd	0.010	0.68	**0.0**
	s/ nuvens	**1**	nd	nd	0.019	0.067	**0.0**	nd	nd	0.014	0.12	**0.0**	nd	nd	0.013	0.14	**0.0**
	k-means	**3**	0.5	0.495	0.594	0.04	**49.5**	0.52	0.48	0.409	0.11	**39.5**	0.01	0.99	0.038	0.01	**47.0**
NN	"	**4**	0.39	0.599	0.259	0.01	**37.9**	0.29	0.701	0.211	0.01	**40.6**	0.06	0.94	0.074	0.02	**45.7**
CO	"	**5**	0.29	0.701	0.083	0.01	**34.2**	0.3	0.694	0.086	0.01	**34.0**	0.4	0.599	0.051	0.04	**27.5**
2v	***pam***	**3**	0.38	0.62	0.657	0.03	**58.9**	0.46	0.54	0.425	0.24	**43.3**	0.04	0.96	0.056	0.01	**45.8**
LUAT3	"	**4**	0.77	0.222	0.095	0	**10.9**	0.74	0.255	0.1	0	**12.8**	0.75	0.249	0.029	0.03	**10.0**
	"	**5**	0.4	0.592	0.039	0	**27.1**	0.45	0.542	0.045	0	**24.6**	0.62	0.38	0.054	0.03	**16.7**
	mclust	**3**	0.11	0.871	0.107	0.81	**43.9**	0.04	0.958	0.108	0.81	**48.3**	0	0.999	0.088	0.06	**49.4**
	"	**4**	0.92	0.052	0.053	0.01	**0.3**	0.87	0.081	0.049	0.04	**1.6**	0.53	0.451	0.022	0.15	**20.1**
	"	**5**	0.21	0.783	0.020	0.26	**36.7**	0.3	0.695	0.023	0.21	**32.3**	0.36	0.611	0.006	0.57	**28.1**
	s/ nuvens	**1**	0	1	0.011	0.29	**47.5**	0	1	0.008	0.27	**47.5**	0	1	0.008	0.26	**47.5**
	k-means	**3**	nd	nd	0.061	0.15	**1.1**	nd	nd	0.074	0.18	**2.4**	nd	nd	0.045	0	**0.0**
NN	"	**4**	nd	nd	0.077	0	**2.7**	nd	nd	0.084	0	**3.4**	nd	nd	0.061	0.01	**1.1**
SO	"	**5**	nd	nd	0.005	0	**0.0**	nd	nd	0.101	0	**5.1**	nd	nd	0.102	0	**5.2**
4v	***pam***	**3**	nd	nd	0.02	0.91	**0.0**	nd	nd	0.018	0.92	**0.0**	nd	nd	0.044	0.01	**0.0**
LUAT3	"	**4**	nd	nd	0.054	0.01	**0.4**	nd	nd	0.077	0	**2.7**	nd	nd	0.043	0.02	**0.0**
	"	**5**	nd	nd	0.051	0.01	**0.1**	nd	nd	0.073	0	**2.3**	nd	nd	0.048	0.04	**0.0**
	mclust	**3**	nd	nd	0.004	0.92	**0.0**	nd	nd	0.006	0.94	**0.0**	nd	nd	0.022	0.20	**0.0**
	"	**4**	nd	nd	0.028	0.18	**0.0**	nd	nd	0.032	0.06	**0.0**	nd	nd	0.014	0.5	**0.0**
	"	**5**	nd	nd	0.017	0.39	**0.0**	nd	nd	0.026	0.15	**0.0**	nd	nd	0.012	0.59	**0.0**
	s/ nuvens	**1**	nd	nd	0.064	0	**1.4**	nd	nd	0.059	0	**0.9**	nd	nd	0.037	0	**0.0**
	k-means	**3**	0.2	0.8	0.259	0.11	**48.0**	0.15	0.85	0.454	0.12	**60.2**	0.01	0.99	0.086	0	**48.8**
NN	"	**4**	0.11	0.887	0.207	0	**49.7**	0.21	0.772	0.332	0	**50.2**	0.08	0.92	0.082	0.01	**45.1**
CO	"	**5**	0.04	0.958	0.132	0.01	**49.5**	0.06	0.926	0.196	0	**51.1**	0.21	0.79	0.080	0.02	**38.5**
4v	***pam***	**3**	0.16	0.84	0.340	0.43	**54.0**	0.2	0.8	0.549	0.21	**62.5**	0	1	0.11	0.01	**50.5**
LUAT3	"	**4**	0.42	0.58	0.071	0	**27.6**	0.74	0.242	0.105	0	**12.4**	0.72	0.28	0.037	0	**11.5**
	"	**5**	0.13	0.869	0.049	0	**41.0**	0.31	0.675	0.066	0	**32.1**	0.63	0.361	0.059	0	**16.0**
	mclust	**3**	0.02	0.98	0.042	0.92	**46.5**	0	0.999	0.068	0.88	**48.4**	0.01	0.99	0.110	0.01	**50.0**
	"	**4**	0.49	0.498	0.054	0.09	**22.6**	0.65	0.329	0.076	0.03	**15.3**	0.25	0.73	0.023	0.29	**34.0**
	"	**5**	0.03	0.97	0.021	0.23	**46.0**	0.1	0.895	0.032	0.2	**42.3**	0.16	0.84	0.012	0.51	**39.5**
	s/ nuvens	**1**	0	1	0.093	0	**49.7**	0	1	0.085	0	**49.3**	0	1	0.040	0.01	**47.5**

Tabela D.8: Resultados da simulação para a configuração LUAT3 com 10 *outliers*

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.134	0.25	**8.4**	nd	nd	0.129	0.28	**7.9**	nd	nd	0.029	0.03	**0.0**
NN	"	**4**	nd	nd	0.071	0.01	**2.1**	nd	nd	0.055	0	**0.5**	nd	nd	0.036	0.01	**0.0**
SO	"	**5**	nd	nd	0.059	0	**0.9**	nd	nd	0.055	0.03	**0.5**	nd	nd	0.039	0.03	**0.0**
2v	***pam***	**3**	nd	nd	0.045	0.8	**0.0**	nd	nd	0.066	0.69	**1.6**	nd	nd	0.039	0	**0.0**
LUAT3	"	**4**	nd	nd	0.047	0	**0.0**	nd	nd	0.044	0	**0.0**	nd	nd	0.059	0.02	**0.9**
10	"	**5**	nd	nd	0.038	0	**0.0**	nd	nd	0.048	0	**0.0**	nd	nd	0.037	0.03	**0.0**
	mclust	**3**	nd	nd	0.008	0.95	**0.0**	nd	nd	0.011	0.94	**0.0**	nd	nd	0.023	0.17	**0.0**
	"	**4**	nd	nd	0.021	0.26	**0.0**	nd	nd	0.026	0.19	**0.0**	nd	nd	0.018	0.56	**0.0**
	"	**5**	nd	nd	0.013	0.378	**0.0**	nd	nd	0.021	0.28	**0.0**	nd	nd	0.010	0.68	**0.0**
	s/ nuvens	**1**	nd	nd	0.019	0.067	**0.0**	nd	nd	0.014	0.12	**0.0**	nd	nd	0.013	0.14	**0.0**
	k-means	**3**	0.47	0.485	0.116	0.03	**25.1**	0.43	0.535	0.092	0.23	**26.4**	0.02	0.98	0.027	0.04	**46.5**
NN	"	**4**	0.74	0.235	0.075	0	**10.5**	0.74	0.215	0.059	0	**8.7**	0.02	0.98	0.036	0.05	**46.5**
CO	"	**5**	0.7	0.261	0.049	0.02	**10.6**	0.59	0.35	0.05	0.1	**15.0**	0.15	0.85	0.046	0.03	**40.0**
2v	***pam***	**3**	0.32	0.68	0.086	0.66	**33.3**	0.33	0.67	0.074	0.67	**32.2**	0.02	0.98	0.032	0.01	**46.5**
LUAT3	"	**4**	0.66	0.331	0.090	0	**16.1**	0.69	0.297	0.104	0	**15.1**	0.7	0.3	0.029	0	**12.5**
10	"	**5**	0.4	0.6	0.043	0	**27.5**	0.47	0.529	0.042	0.01	**24.0**	0.64	0.36	0.042	0.02	**15.5**
	mclust	**3**	0.02	0.98	0.10	0.96	**49.0**	0.03	0.97	0.009	0.94	**46.0**	0.02	0.98	0.019	0.2	**46.5**
	"	**4**	0.65	0.331	0.040	0.08	**14.1**	0.81	0.183	0.039	0.05	**6.7**	0.51	0.481	0.016	0.24	**21.6**
	"	**5**	0.33	0.67	0.021	0.32	**31.0**	0.36	0.63	0.021	0.28	**29.0**	0.27	0.721	0.013	0.54	**33.6**
	s/ nuvens	**1**	0	1	0.013	0.14	**47.5**	0	1	0.10	0.22	**50.0**	0	1	0.010	0.28	**47.5**
	k-means	**3**	nd	nd	0.061	0.15	**1.1**	nd	nd	0.074	0.18	**2.4**	nd	nd	0.045	0	**0.0**
NN	"	**4**	nd	nd	0.077	0	**2.7**	nd	nd	0.084	0	**3.4**	nd	nd	0.061	0.01	**1.1**
SO	"	**5**	nd	nd	0.005	0	**0.0**	nd	nd	0.101	0	**5.1**	nd	nd	0.102	0	**5.2**
4v	***pam***	**3**	nd	nd	0.02	0.91	**0.0**	nd	nd	0.018	0.92	**0.0**	nd	nd	0.044	0.01	**0.0**
LUAT3	"	**4**	nd	nd	0.054	0.01	**0.4**	nd	nd	0.077	0	**2.7**	nd	nd	0.043	0.02	**0.0**
10	"	**5**	nd	nd	0.051	0.01	**0.1**	nd	nd	0.073	0	**2.3**	nd	nd	0.048	0.04	**0.0**
	mclust	**3**	nd	nd	0.004	0.92	**0.0**	nd	nd	0.006	0.94	**0.0**	nd	nd	0.022	0.20	**0.0**
	"	**4**	nd	nd	0.028	0.18	**0.0**	nd	nd	0.032	0.06	**0.0**	nd	nd	0.014	0.5	**0.0**
	"	**5**	nd	nd	0.017	0.39	**0.0**	nd	nd	0.026	0.15	**0.0**	nd	nd	0.012	0.59	**0.0**
	s/ nuvens	**1**	nd	nd	0.064	0	**1.4**	nd	nd	0.059	0	**0.9**	nd	nd	0.037	0	**0.0**
	k-means	**3**	0.15	0.839	0.083	0.13	**41.1**	0.1	0.891	0.082	0.18	**43.7**	0	1	0.045	0	**47.5**
NN	"	**4**	0.08	0.92	0.086	0	**45.3**	0.07	0.927	0.099	0	**46.3**	0.04	0.96	0.070	0	**46.5**
CO	"	**5**	0.09	0.904	0.098	0	**45.1**	0.07	0.929	0.111	0	**47.0**	0.06	0.94	0.109	0	**47.5**
4v	***pam***	**3**	0.07	0.93	0.024	0.92	**44.0**	0.15	0.85	0.042	0.83	**40.0**	0	1	0.042	0	**47.5**
LUAT3	"	**4**	0.46	0.54	0.60	0	**52.0**	0.71	0.29	0.089	0	**14.0**	0.7	0.291	0.039	0	**12.1**
10	"	**5**	0.09	0.91	0.049	0.02	**43.0**	0.27	0.73	0.069	0	**35.0**	0.63	0.37	0.048	0.01	**16.0**
	mclust	**3**	0.02	0.953	0.011	0.86	**45.2**	0	0.982	0.075	0.85	**47.9**	0.01	0.936	0.073	0.16	**45.5**
	"	**4**	0.62	0.362	0.059	0.03	**16.1**	0.63	0.352	0.068	0.06	**16.0**	0.39	0.547	0.30	0.12	**37.4**
	"	**5**	0.07	0.825	0.019	0.03	**38.8**	0.24	0.715	0.03	0.23	**33.3**	0.18	0.75	0.011	0.47	**35.0**
	s/ nuvens	**1**	0	1	0.071	0	**48.6**	0	1	0.068	0	**48.4**	0	1	0.037	0.01	**47.5**

Tabela D.9: Resultados da simulação para a configuração LUAT3 com 30 *outliers*

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.134	0.25	**8.4**	nd	nd	0.129	0.28	**7.9**	nd	nd	0.029	0.03	**0.0**
NN	"	**4**	nd	nd	0.071	0.01	**2.1**	nd	nd	0.055	0	**0.5**	nd	nd	0.036	0.01	**0.0**
SO	"	**5**	nd	nd	0.059	0	**0.9**	nd	nd	0.055	0.03	**0.5**	nd	nd	0.039	0.03	**0.0**
2v	***pam***	**3**	nd	nd	0.045	0.8	**0.0**	nd	nd	0.066	0.69	**1.6**	nd	nd	0.039	0	**0.0**
LUAT3	"	**4**	nd	nd	0.047	0	**0.0**	nd	nd	0.044	0	**0.0**	nd	nd	0.059	0.02	**0.9**
30	"	**5**	nd	nd	0.038	0	**0.0**	nd	nd	0.048	0	**0.0**	nd	nd	0.037	0.03	**0.0**
	mclust	**3**	nd	nd	0.008	0.95	**0.0**	nd	nd	0.011	0.94	**0.0**	nd	nd	0.023	0.17	**0.0**
	"	**4**	nd	nd	0.021	0.26	**0.0**	nd	nd	0.026	0.19	**0.0**	nd	nd	0.018	0.56	**0.0**
	"	**5**	nd	nd	0.013	0.378	**0.0**	nd	nd	0.021	0.28	**0.0**	nd	nd	0.010	0.68	**0.0**
	s/ nuvens	**1**	nd	nd	0.019	0.067	**0.0**	nd	nd	0.014	0.12	**0.0**	nd	nd	0.013	0.14	**0.0**
	k-means	**3**	0.49	0.51	0.846	0	**62.8**	0.56	0.44	0.895	0.01	**61.8**	0.03	0.97	0.076	0.05	**47.3**
NN	"	**4**	0.5	0.492	0.414	0	**40.3**	0.56	0.423	0.402	0	**36.3**	0.2	0.8	0.077	0.07	**38.9**
CO	"	**5**	0.32	0.665	0.163	0.01	**36.4**	0.52	0.455	0.182	0	**26.9**	0.44	0.56	0.056	0.02	**25.8**
2v	***pam***	**3**	0.5	0.5	0.853	0	**62.7**	0.46	0.54	0.898	0	**66.9**	0	1	0.118	0.02	**50.9**
LUAT3	"	**4**	0.76	0.23	0.108	0	**11.9**	0.67	0.323	0.131	0	**17.7**	0.81	0.18	0.029	0.01	**6.5**
30	"	**5**	0.42	0.567	0.044	0.01	**25.9**	0.38	0.601	0.045	0	**27.6**	0.64	0.359	0.038	0	**15.5**
	mclust	**3**	0.07	0.921	0.085	0.87	**45.3**	0.02	0.979	0.173	0.73	**52.6**	0	0.999	0.124	0.1	**51.2**
	"	**4**	0.83	0.14	0.066	0.01	**5.3**	0.95	0.049	0.057	0.03	**0.4**	0.55	0.43	0.017	0.22	**19.0**
	"	**5**	0.26	0.714	0.018	0.25	**33.2**	0.26	0.693	0.022	0.22	**32.2**	0.24	0.75	0.007	0.6	**35.0**
	s/ nuvens	**1**	0	1	0.009	0.25	**47.5**	0	1	0.007	0.37	**47.5**	0	1	0.009	0.22	**47.5**
	k-means	**3**	nd	nd	0.061	0.15	**1.1**	nd	nd	0.074	0.18	**2.4**	nd	nd	0.045	0	**0.0**
NN	"	**4**	nd	nd	0.077	0	**2.7**	nd	nd	0.084	0	**3.4**	nd	nd	0.061	0.01	**1.1**
SO	"	**5**	nd	nd	0.005	0	**0.0**	nd	nd	0.101	0	**5.1**	nd	nd	0.102	0	**5.2**
4v	***pam***	**3**	nd	nd	0.02	0.91	**0.0**	nd	nd	0.018	0.92	**0.0**	nd	nd	0.044	0.01	**0.0**
LUAT3	"	**4**	nd	nd	0.054	0.01	**0.4**	nd	nd	0.077	0	**2.7**	nd	nd	0.043	0.02	**0.0**
30	"	**5**	nd	nd	0.051	0.01	**0.1**	nd	nd	0.073	0	**2.3**	nd	nd	0.048	0.04	**0.0**
	mclust	**3**	nd	nd	0.004	0.92	**0.0**	nd	nd	0.006	0.94	**0.0**	nd	nd	0.022	0.20	**0.0**
	"	**4**	nd	nd	0.028	0.18	**0.0**	nd	nd	0.032	0.06	**0.0**	nd	nd	0.014	0.5	**0.0**
	"	**5**	nd	nd	0.017	0.39	**0.0**	nd	nd	0.026	0.15	**0.0**	nd	nd	0.012	0.59	**0.0**
	s/ nuvens	**1**	nd	nd	0.064	0	**1.4**	nd	nd	0.059	0	**0.9**	nd	nd	0.037	0	**0.0**
	k-means	**3**	0.16	0.83	0.756	0.02	**74.3**	0.15	0.849	0.817	0	**78.3**	0	1	0.18	0	**54.0**
NN	"	**4**	0.16	0.825	0.266	0	**49.6**	0.26	0.697	0.376	0	**48.7**	0.13	0.869	0.117	0	**44.3**
CO	"	**5**	0.04	0.939	0.177	0	**50.8**	0.23	0.734	0.197	0	**41.6**	0.39	0.607	0.091	0	**29.9**
4v	***pam***	**3**	0.13	0.87	0.745	0.06	**75.8**	0.21	0.789	0.865	0	**77.7**	0	1	0.208	0	**55.4**
LUAT3	"	**4**	0.33	0.664	0.074	0.01	**31.9**	0.63	0.331	0.103	0	**16.7**	0.74	0.259	0.038	0	**10.5**
30	"	**5**	0.86	0.13	0.149	0	**9.0**	0.34	0.621	0.076	0	**29.9**	0.54	0.456	0.053	0	**20.5**
	mclust	**3**	0.02	0.97	0.053	0.91	**46.2**	0	0.999	0.079	0.87	**48.9**	0	0.998	0.091	0.01	**49.5**
	"	**4**	0.53	0.468	0.057	0.05	**21.3**	0.62	0.361	0.066	0.07	**16.4**	0.27	0.719	0.021	0.2	**33.5**
	"	**5**	0.1	0.879	0.021	0.28	**41.5**	0.17	0.824	0.029	0.17	**38.7**	0.07	0.918	0.009	0.67	**43.4**
	s/ nuvens	**1**	0	1	0.105	0	**50.3**	0	1	0.102	0	**50.1**	0	1	0.045	0	**47.5**

Tabela D.10: Resultados da simulação para a configuração ESFERA

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.519	0.02	**46.9**	nd	nd	0.485	0.13	**43.5**	nd	nd	0.022	0.95	**0.0**
NN	"	**4**	nd	nd	0.873	0	**82.3**	nd	nd	0.816	0	**76.6**	nd	nd	0.125	0.58	**7.5**
SO	"	**5**	nd	nd	0.881	0	**83.1**	-	-	-	-	**-**	nd	nd	0.225	0.36	**17.5**
2v	***pam***	**3**	nd	nd	0.55	0.01	**50.0**	nd	nd	0.547	0.07	**49.7**	nd	nd	0.017	0.95	**0.0**
ESFERA	"	**4**	nd	nd	0.925	0	**87.5**	nd	nd	0.878	0	**82.8**	nd	nd	0.084	0.71	**3.4**
	"	**5**	nd	nd	0.883	0	**83.3**	-	-	-	-	**-**	nd	nd	0.204	0.37	**15.4**
	mclust	**3**	nd	nd	0.518	0.02	**46.8**	nd	nd	0.485	0.13	**43.5**	nd	nd	0.116	0.72	**6.6**
	"	**4**	nd	nd	0.878	0	**82.8**	nd	nd	0.812	0	**76.2**	nd	nd	0.275	0.33	**22.5**
	"	**5**	nd	nd	0.885	0	**83.5**	nd	nd	0.789	0	**73.9**	nd	nd	0.432	0.07	**38.2**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0.48	0.52	0.363	0.23	**39.2**	0.61	0.39	0.337	0.25	**31.4**	0.04	0.96	0.013	0.96	**45.5**
NN	"	**4**	1	0	0.919	0	**43.5**	0.99	0.01	0.856	0.01	**40.3**	0.13	0.87	0.054	0.81	**41.2**
CO	"	**5**	1	0	0.869	0	**41.0**	-	-	-	-	**-**	0.22	0.78	0.101	0.55	**39.1**
2v	***pam***	**3**	0.02	0.548	0.42	0.01	**43.4**	0.1	0.53	0.437	0.13	**43.4**	0	1	0	1	**47.5**
ESFERA	"	**4**	1	0	0.94	0	**44.5**	-	-	-	-	**-**	0	1	0	1	**47.5**
	"	**5**	1	0	0.891	0	**42.1**	-	-	-	-	**-**	0.55	0.45	0.057	0.79	**20.4**
	mclust	**3**	0.55	0.45	0.341	0.25	**34.6**	0.7	0.3	0.353	0.22	**27.7**	0.2	0.8	0.088	0.79	**39.4**
	"	**4**	0.98	0.02	0.91	0.01	**43.0**	0.88	0.12	0.73	0.01	**37.5**	0.45	0.55	0.169	0.55	**31.0**
	"	**5**	1	0	0.921	0	**43.6**	0.99	0.01	0.765	0	**35.8**	0.92	0.08	0.294	0.34	**13.7**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	k-means	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
NN	"	**4**	nd	nd	0.001	0.91	**0.0**	nd	nd	0	0.99	**0.0**	nd	nd	0	1	**0.0**
SO	"	**5**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
4v	***pam***	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
ESFERA	"	**4**	nd	nd	0	0.94	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	"	**5**	nd	nd	0	0.97	**0.0**	nd	nd	0	0.99	**0.0**	nd	nd	0	1	**0.0**
	mclust	**3**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	"	**4**	nd	nd	0	0.99	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	"	**5**	nd	nd	0	0.95	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0.001	0.99	**0.0**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
NN	"	**4**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
CO	"	**5**	0	1	0	0.98	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
4v	***pam***	**3**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
ESFERA	"	**4**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	"	**5**	0	1	0	0.97	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	mclust	**3**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	"	**4**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	"	**5**	0	1	0	0.97	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**

Tabela D.11: Resultados da simulação para a configuração ESFERA com o dobro dos dados e 2v.

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**
	k-means	**3**	nd	nd	0.464	0.03	**41.4**	nd	nd	0.499	0.14	**44.9**	nd	nd	0	1	**0.0**
NN	"	**4**	nd	nd	0.974	0	**92.4**	nd	nd	0.895	0	**84.5**	nd	nd	0.006	0.98	**0.0**
SO	"	**5**	nd	nd	0.957	0	**90.7**	nd	nd	0.868	0	**81.8**	nd	nd	0.011	0.95	**0.0**
2v	***pam***	**3**	nd	nd	0.507	0	**45.7**	nd	nd	0.576	0.05	**52.6**	nd	nd	0	1	**0.0**
ESFERA	"	**4**	nd	nd	0.988	0	**93.8**	nd	nd	0.938	0	**88.8**	nd	nd	0	1	**0.0**
DOBRO	"	**5**	nd	nd	0.973	0	**92.3**	nd	nd	0.893	0	**84.3**	nd	nd	0.017	0.92	**0.0**
	mclust	**3**	nd	nd	0.398	0	**34.8**	nd	nd	0.467	0.13	**41.7**	nd	nd	0.004	0.99	**0.0**
	"	**4**	nd	nd	0.976	0	**92.6**	nd	nd	0.871	0	**82.1**	nd	nd	0.022	0.94	**0.0**
	"	**5**	nd	nd	0.991	0	**94.1**	nd	nd	0.859	0	**80.9**	nd	nd	0.072	0.71	**2.2**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0.58	0.42	0.446	0.09	**38.3**	0.46	0.54	0.212	0.47	**32.6**	0	1	0	1	**47.5**
NN	"	**4**	1	0	0.979	0	**46.5**	0.99	0.01	0.867	0.02	**40.9**	0	1	0	1	**47.5**
CO	"	**5**	1	0	0.956	0	**45.3**	0.98	0.02	0.844	0.02	**39.7**	0.01	0.99	0.007	0.97	**47.0**
2v	***pam***	**3**	0.05	0.54	0.449	0.06	**44.5**	0.02	0.68	0.305	0.3	**44.3**	0	1	0	1	**47.5**
ESFERA	"	**4**	1	0	0.988	0	**46.9**	1	0	0.926	0	**43.8**	0.38	0.579	0	1	**26.5**
DOBRO	"	**5**	1	0	0.977	0	**46.4**	1	0	0.902	0	**42.6**	0.47	0.514	0.005	0.97	**23.2**
	mclust	**3**	0.48	0.52	0.373	0.19	**39.7**	0.56	0.44	0.249	0.4	**29.5**	0.01	0.99	0.005	0.99	**47.0**
	"	**4**	1	0	0.983	0	**46.7**	0.88	0.12	0.751	0.02	**38.6**	0.04	0.96	0.019	0.94	**45.5**
	"	**5**	1	0	0.982	0	**46.6**	0.99	0.01	0.833	0.02	**39.2**	0.09	0.91	0.035	0.88	**43.0**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**

Tabela D.12: Resultados da simulação para a configuração DONUT

			MCD					OGK					Clássicos				
		k	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**	**p_1**	**p_2**	**p_3**	**p_4**	**DIF**
	k-means	**3**	nd	nd	0.316	0.2	**26.6**	nd	nd	0.241	0.39	**19.1**	nd	nd	0.017	0.95	**0.0**
NN	"	**4**	nd	nd	0.141	0.36	**9.1**	nd	nd	0.063	0.61	**1.3**	nd	nd	0.022	0.91	**0.0**
SO	"	**5**	nd	nd	0.029	0.72	**0.0**	nd	nd	0.007	0.89	**0.0**	nd	nd	0.022	0.9	**0.0**
2v	***pam***	**3**	nd	nd	0.332	0.11	**28.2**	nd	nd	0.239	0.42	**18.9**	nd	nd	0.007	0.98	**0.0**
DONUT	"	**4**	nd	nd	0.193	0.21	**14.3**	nd	nd	0.059	0.57	**0.9**	nd	nd	0.022	0.91	**0.0**
	"	**5**	nd	nd	0.021	0.71	**0.0**	nd	nd	0.008	0.92	**0.0**	nd	nd	0.021	0.9	**0.0**
	mclust	**3**	nd	nd	0.227	0.32	**17.7**	nd	nd	0.157	0.54	**10.7**	nd	nd	0.027	0.93	**0.0**
	"	**4**	nd	nd	0.129	0.3	**7.9**	nd	nd	0.054	0.63	**0.4**	nd	nd	0.069	0.76	**1.9**
	"	**5**	nd	nd	0.035	0.64	**0.0**	nd	nd	0.01	0.89	**0.0**	nd	nd	0.045	0.79	**0.0**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0.35	0.65	0.17	0.54	**36.0**	0.35	0.65	0.13	0.69	**34.0**	0.19	0.81	0.002	0.99	**38.0**
NN	"	**4**	0.66	0.34	0.361	0.09	**30.1**	0.57	0.42	0.221	0.3	**27.1**	0.61	0.39	0.011	0.97	**17.0**
CO	"	**5**	0.86	0.14	0.139	0.28	**9.0**	0.83	0.17	0.045	0.68	**6.0**	0.99	0.01	0.019	0.93	**0.0**
2v	***pam***	**3**	0.47	0.53	0.249	0.43	**34.0**	0.43	0.57	0.177	0.59	**32.4**	0.28	0.72	0.003	0.99	**33.5**
DONUT	"	**4**	0.8	0.2	0.408	0.02	**25.4**	0.86	0.14	0.252	0.21	**14.6**	0.95	0.05	0.01	0.97	**0.0**
	"	**5**	0.9	0.1	0.161	0.16	**8.1**	0.83	0.17	0.068	0.49	**6.9**	1	0	0.018	0.94	**0.0**
	mclust	**3**	0.59	0.41	0.108	0.79	**20.9**	0.41	0.59	0.117	0.81	**30.4**	0.54	0.46	0	1	**20.5**
	"	**4**	0.76	0.24	0.326	0.18	**23.3**	0.67	0.33	0.161	0.47	**19.6**	0.84	0.16	0.021	0.95	**5.5**
	"	**5**	0.89	0.11	0.132	0.28	**7.1**	0.79	0.21	0.035	0.76	**8.0**	0.98	0.02	0.023	0.92	**0.0**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	k-means	**3**	nd	nd	0.276	0.28	**22.6**	nd	nd	0.102	0.73	**5.2**	nd	nd	0.014	0.96	**0.0**
NN	"	**4**	nd	nd	0.085	0.41	**3.5**	nd	nd	0.017	0.89	**0.0**	nd	nd	0.018	0.93	**0.0**
SO	"	**5**	nd	nd	0.01	0.88	**0.0**	nd	nd	0.002	0.94	**0.0**	nd	nd	0.017	0.93	**0.0**
3v	***pam***	**3**	nd	nd	0.265	0.32	**21.5**	nd	nd	0.062	0.82	**1.2**	nd	nd	0.004	0.99	**0.0**
DONUT	"	**4**	nd	nd	0.101	0.38	**5.1**	nd	nd	0.014	0.89	**0.0**	nd	nd	0.009	0.96	**0.0**
	"	**5**	nd	nd	0.014	0.85	**0.0**	nd	nd	0	0.99	**0.0**	nd	nd	0.008	0.96	**0.0**
	mclust	**3**	nd	nd	0.174	0.52	**12.4**	nd	nd	0.045	0.87	**0.0**	nd	nd	0.033	0.92	**0.0**
	"	**4**	nd	nd	0.064	0.49	**1.4**	nd	nd	0.011	0.89	**0.0**	nd	nd	0.031	0.89	**0.0**
	"	**5**	nd	nd	0.019	0.73	**0.0**	nd	nd	0.081	0.97	**3.1**	nd	nd	0.027	0.85	**0.0**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0.36	0.64	0.141	0.62	**34.1**	0.16	0.84	0.011	0.96	**39.5**	0.22	0.78	0	1	**36.5**
NN	"	**4**	0.66	0.34	0.297	0.13	**26.9**	0.45	0.55	0.066	0.76	**25.8**	0.49	0.51	0.004	0.98	**23.0**
CO	"	**5**	0.67	0.33	0.067	0.52	**14.9**	0.59	0.41	0.02	0.9	**18.0**	0.98	0.02	0.004	0.98	**0.0**
3v	***pam***	**3**	0.37	0.63	0.146	0.6	**33.8**	0.23	0.77	0.037	0.9	**36.0**	0.28	0.72	0.003	0.99	**33.5**
DONUT	"	**4**	0.77	0.23	0.272	0.14	**20.1**	0.67	0.33	0.065	0.66	**14.8**	0.84	0.16	0	1	**5.5**
	"	**5**	0.76	0.24	0.077	0.49	**10.9**	0.71	0.29	0.017	0.84	**12.0**	1	0	0	1	**0.0**
	mclust	**3**	0.36	0.64	0.058	0.9	**29.9**	0.29	0.71	0.041	0.93	**33.0**	0.36	0.64	0.006	0.99	**29.5**
	"	**4**	0.64	0.36	0.224	0.23	**24.2**	0.61	0.39	0.038	0.83	**17.0**	0.75	0.25	0.009	0.98	**10.0**
	"	**5**	0.78	0.22	0.06	0.5	**9.0**	0.69	0.31	0.026	0.87	**13.0**	0.92	0.08	0.026	0.9	**1.5**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**
	k-means	**3**	nd	nd	0.142	0.6	**9.2**	nd	nd	0.02	0.95	**0.0**	nd	nd	0.007	0.98	**0.0**
NN	"	**4**	nd	nd	0.041	0.67	**0.0**	nd	nd	0.001	0.94	**0.0**	nd	nd	0.016	0.94	**0.0**
SO	"	**5**	nd	nd	0.001	0.96	**0.0**	nd	nd	0.002	0.96	**0.0**	nd	nd	0.013	0.94	**0.0**
4v	***pam***	**3**	nd	nd	0.151	0.57	**10.1**	nd	nd	0.003	0.92	**0.0**	nd	nd	0.004	0.99	**0.0**
DONUT	"	**4**	nd	nd	0.023	0.72	**0.0**	nd	nd	0.008	0.91	**0.0**	nd	nd	0.002	0.99	**0.0**
	"	**5**	nd	nd	0.001	0.9	**0.0**	nd	nd	0.005	0.97	**0.0**	nd	nd	0.009	0.96	**0.0**
	mclust	**3**	nd	nd	0.076	0.74	**2.6**	nd	nd	0.005	0.98	**0.0**	nd	nd	0	1	**0.0**
	"	**4**	nd	nd	0.045	0.6	**0.0**	nd	nd	0.005	0.92	**0.0**	nd	nd	0.027	0.89	**0.0**
	"	**5**	nd	nd	0.004	0.86	**0.0**	nd	nd	0.005	0.95	**0.0**	nd	nd	0.021	0.89	**0.0**
	s/ nuvens	**1**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**	nd	nd	0	1	**0.0**
	k-means	**3**	0.28	0.72	0.053	0.85	**33.7**	0.07	0.93	0.011	0.97	**44.0**	0.24	0.76	0	1	**35.5**
NN	"	**4**	0.46	0.54	0.142	0.37	**29.1**	0.44	0.56	0.025	0.82	**25.5**	0.4	0.6	0.003	0.99	**27.5**
CO	"	**5**	0.51	0.49	0.025	0.69	**22.0**	0.62	0.38	0.006	0.95	**16.5**	0.98	0.02	0.007	0.97	**0.0**
4v	***pam***	**3**	0.26	0.74	0.117	0.67	**37.9**	0.17	0.83	0.018	0.94	**39.0**	0.22	0.78	0	1	**36.5**
DONUT	"	**4**	0.66	0.34	0.179	0.26	**21.0**	0.6	0.4	0.03	0.88	**17.5**	0.83	0.17	0	1	**6.0**
	"	**5**	0.55	0.45	0.045	0.65	**20.0**	0.58	0.42	0.006	0.97	**18.5**	0.99	0.01	0.005	0.98	**0.0**
	mclust	**3**	0.35	0.65	0.044	0.92	**30.0**	0.16	0.84	0.006	0.99	**39.5**	0.34	0.66	0	1	**30.5**
	"	**4**	0.57	0.43	0.132	0.37	**23.1**	0.56	0.44	0.019	0.91	**19.5**	0.71	0.29	0.011	0.97	**12.0**
	"	**5**	0.54	0.46	0.011	0.74	**20.5**	0.66	0.34	0.004	0.94	**14.5**	0.89	0.11	0.021	0.91	**3.0**
	s/ nuvens	**1**	0	1	0	1	**47.5**	0	1	0	1	**47.5**	0	1	0	1	**47.5**

Tabela D.13: Resultados da simulação para a configuração T1

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**
	k-means	**3**	nd	nd	0.035	0.44	**0.0**	nd	nd	0.037	0.35	**0.0**	nd	nd	0.0025	0.15	**0.0**
NN	"	**4**	nd	nd	0.057	0.09	**0.7**	nd	nd	0.057	0.02	**0.7**	nd	nd	0.031	0.14	**0.0**
SO	"	**5**	nd	nd	0.051	0.07	**0.1**	nd	nd	0.055	0.02	**0.5**	nd	nd	0.045	0.11	**0.0**
2v	***pam***	**3**	nd	nd	0.002	0.97	**0.0**	nd	nd	0.002	0.96	**0.0**	nd	nd	0.018	0.07	**0.0**
T1	"	**4**	nd	nd	0.05	0	**0.0**	nd	nd	0.046	0.02	**0.0**	nd	nd	0.019	0.19	**0.0**
	"	**5**	nd	nd	0.035	0.06	**0.0**	nd	nd	0.037	0.03	**0.0**	nd	nd	0.02	0.22	**0.0**
	mclust	**3**	nd	nd	0.001	0.97	**0.0**	nd	nd	0	0.99	**0.0**	nd	nd	0.005	0.57	**0.0**
	"	**4**	nd	nd	0.015	0.35	**0.0**	nd	nd	0.017	0.33	**0.0**	nd	nd	0.003	0.76	**0.0**
	"	**5**	nd	nd	0.008	0.54	**0.0**	nd	nd	0.012	0.37	**0.0**	nd	nd	0.004	0.82	**0.0**
	s/ nuvens	**1**	nd	nd	0.057	0	**0.7**	nd	nd	0.056	0	**0.6**	nd	nd	0.033	0.02	**0.0**
	k-means	**3**	0.92	0.08	0.016	0.34	**1.5**	0.92	0.08	0.014	0.43	**1.5**	1	0	0.033	0.02	**0.0**
NN	"	**4**	0.76	0.234	0.06	0.01	**9.7**	0.85	0.149	0.059	0.02	**5.4**	0.9	0.1	0.024	0.1	**2.5**
CO	"	**5**	0.66	0.334	0.059	0.03	**14.7**	0.79	0.207	0.054	0.02	**8.1**	0.97	0.03	0.032	0.11	**0.0**
2v	***pam***	**3**	0.98	0.02	0.005	0.42	**0.0**	0.99	0.01	0.005	0.47	**0.0**	0.98	0.02	0.035	0	**0.0**
T1	"	**4**	0.93	0.069	0.056	0	**1.3**	0.93	0.068	0.056	0	**1.2**	0.91	0.089	0.019	0.05	**2.0**
	"	**5**	0.68	0.316	0.048	0.02	**13.3**	0.79	0.203	0.053	0.01	**7.8**	0.95	0.05	0.019	0.08	**0.0**
	mclust	**3**	0.67	0.265	0	0.98	**10.8**	0.69	0.198	0.002	0.91	**7.4**	0.75	0.25	0.007	0.46	**10.0**
	"	**4**	0.74	0.259	0.025	0.26	**10.5**	0.74	0.259	0.02	0.31	**10.5**	0.67	0.33	0.005	0.58	**14.0**
	"	**5**	0.55	0.449	0.014	0.29	**20.0**	0.66	0.33	0.012	0.49	**14.0**	0.59	0.41	0.003	0.76	**18.0**
	s/ nuvens	**1**	1	0	0.055	0	**0.3**	1	0	0.06	0	**0.5**	0	1	0.016	0.13	**47.5**

Tabela D.14: Resultados da simulação para a configuração T2

			MCD					OGK					Clássicos				
		k	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**	p_1	p_2	p_3	p_4	**DIF**
	k-means	**3**	nd	nd	0.041	0.3	**0.0**	nd	nd	0.043	0.34	**0.0**	nd	nd	0.030	0.02	**0.0**
NN	"	**4**	nd	nd	0.056	0	**0.6**	nd	nd	0.057	0	**0.7**	nd	nd	0.038	0.03	**0.0**
SO	"	**5**	nd	nd	0.057	0.02	**0.7**	nd	nd	0.06	0.03	**1.0**	nd	nd	0.05	0	**0.0**
2v	***pam***	**3**	nd	nd	0.009	0.95	**0.0**	nd	nd	0.015	0.91	**0.0**	nd	nd	0.027	0.02	**0.0**
T2	"	**4**	nd	nd	0.047	0	**0.0**	nd	nd	0.047	0	**0.0**	nd	nd	0.028	0.04	**0.0**
	"	**5**	nd	nd	0.046	0.01	**0.0**	nd	nd	0.049	0	**0.0**	nd	nd	0.028	0.09	**0.0**
	mclust	**3**	nd	nd	0.004	0.97	**0.0**	nd	nd	0	0.99	**0.0**	nd	nd	0.014	0.46	**0.0**
	"	**4**	nd	nd	0.022	0.29	**0.0**	nd	nd	0.027	0.19	**0.0**	nd	nd	0.007	0.59	**0.0**
	"	**5**	nd	nd	0.014	0.27	**0.0**	nd	nd	0.018	0.3	**0.0**	nd	nd	0.010	0.74	**0.0**
	s/ nuvens	**1**	nd	nd	0.054	0.01	**0.4**	nd	nd	0.039	0	**0.0**	nd	nd	0.024	0.03	**0.0**
	k-means	**3**	0.18	0.801	0.115	0.27	**40.8**	0.26	0.705	0.033	0.46	**32.8**	0	1	0.038	0.02	**47.5**
NN	"	**4**	0.29	0.694	0.079	0.02	**33.7**	0.32	0.644	0.072	0.02	**30.8**	0.09	0.909	0.034	0.05	**43.0**
CO	"	**5**	0.34	0.652	0.063	0.01	**30.8**	0.38	0.597	0.057	0.01	**27.7**	0.56	0.44	0.039	0.04	**19.5**
2v	***pam***	**3**	0.54	0.46	0.277	0.19	**31.9**	0.46	0.54	0.222	0.31	**33.1**	0.55	0.45	0.043	0	**20.0**
T2	"	**4**	0.51	0.485	0.047	0	**21.8**	0.59	0.401	0.053	0	**17.7**	0.63	0.37	0.026	0.03	**16.0**
	"	**5**	0.24	0.75	0.044	0	**35.0**	0.31	0.674	0.043	0.01	**31.2**	0.63	0.36	0.028	0.02	**15.5**
	mclust	**3**	0.87	0.03	0.037	0.9	**0.0**	0.78	0.074	0.041	0.84	**1.2**	0.74	0.242	0.041	0.01	**9.6**
	"	**4**	0.47	0.519	0.037	0.12	**23.5**	0.6	0.386	0.039	0.1	**16.8**	0.25	0.741	0.007	0.58	**34.6**
	"	**5**	0.23	0.727	0.019	0.25	**33.9**	0.25	0.746	0.019	0.3	**34.8**	0.23	0.77	0.004	0.69	**36.0**
	s/ nuvens	**1**	0.09	0.851	0.035	0	**40.1**	0.03	0.959	0.024	0.04	**45.5**	0	1	0.013	0.16	**47.5**

Apêndice E

Programas relativos ao Capítulo 5

c(p,n_j,α_j)- Constantes (Becker e Gather, 2001) que funcionam como limites de detecção dos *outliers*.

```
constantes<-function(n,p)
{
cons1<-matrix(0,7,1)
cons2<-matrix(0,7,1)
cons3<-matrix(0,7,1)
res<-rep(0,10000)
for (i in 1:10000){
dados<-rmvnorm(n,d=p)
covmcd1<-cov.mcd(dados,quan=floor(0.75*n))
mat<-covmcd1$cov
mu<-covmcd1$center
maha<-mahalanobis(dados,mu,mat,inverted=F)
res[i]<-max(maha)
}
for (k in 1:7){
per1<-0.9^(1/k)
per2<-0.85^(1/k)
per3<-0.8^(1/k)
cons1[k,1]<-quantile(res,probs=c(per1))
cons2[k,1]<-quantile(res,probs=c(per2))
cons3[k,1]<-quantile(res,probs=c(per3))
}
list(cons0.9=cons1,cons0.85=cons2,cons0.8=cons3)
}
```

AIC_EXEMPLO- Valor médio do AIC para 100 simulações com utilização do método *k-means* e estimador robusto MCD.

```
akaikefinal<-0
for (i in 1:100)
{

dens<-function(x,k,nt,mu,sigma){
 fest<-0
 n<-sum(nt)
 p<-nrow(mu)
# print(p)
 for (j in 1:k){
  muj<-mu[,j]
#  print(j)
# print( (((j-1)*p+1):(j*p)) )
  sigmaj<- sigma[(((j-1)*p+1):(j*p)),]
# print(sigmaj)
  fest<- fest+nt[j]/n*dmvnorm(x, mean=muj, cov=sigmaj)
 }
 fest
}

akaike<- function(dados,k,nt,mu,sigma){
  n<-sum(nt)
  p<- nrow(mu)
  lnL<-0
  for (i in 1:n){
    lnL<-lnL+log(dens(dados[i,],k,nt,mu,sigma))
  }
   ak<- -2*lnL+2*k*(p+p*(p+1)/2)
  ak
}

 dart<-
   rbind(rmvnorm(50,c(0,12),diag(c(1,0.3))),rmvnorm(50,c(0,0),diag(c(1,0.3))),
   rmvnorm(50,c(1.5,6),diag(c(0.2,9))),rmvnorm(10,c(-2,6),diag(c(0.01,0.01))))

clusdn2<-kmeans(dart,2)
clusdn3<-kmeans(dart,3)
clusdn4<-kmeans(dart,4)
clusdn5<-kmeans(dart,5)
clusdn6<-kmeans(dart,6)
clusdn7<-kmeans(dart,7)
clusdn8<-kmeans(dart,8)


ig1<- clusdn8$cluster==1
ig2<- clusdn8$cluster==2
ig3<- clusdn8$cluster==3
ig4<- clusdn8$cluster==4
ig5<- clusdn8$cluster==5
ig6<- clusdn8$cluster==6
ig7<- clusdn8$cluster==7
ig8<- clusdn8$cluster==8


g1<-dart[ig1,]
g2<-dart[ig2,]
g3<-dart[ig3,]
g4<-dart[ig4,]
g5<-dart[ig5,]
g6<-dart[ig6,]
g7<-dart[ig7,]
g8<-dart[ig8,]
```

```
tnuvem1<-nrow(g1)
tnuvem2<-nrow(g2)
tnuvem3<-nrow(g3)
tnuvem4<-nrow(g4)
tnuvem5<-nrow(g5)
tnuvem6<-nrow(g6)
tnuvem7<-nrow(g7)
tnuvem8<-nrow(g8)

m1<-cov.mcd(g1,quan=floor(0.75*tnuvem1))
m2<-cov.mcd(g2,quan=floor(0.75*tnuvem2))
m3<-cov.mcd(g3,quan=floor(0.75*tnuvem3))
m4<-cov.mcd(g4,quan=floor(0.75*tnuvem4))
m5<-cov.mcd(g5,quan=floor(0.75*tnuvem5))
m6<-cov.mcd(g6,quan=floor(0.75*tnuvem6))
m7<-cov.mcd(g7,quan=floor(0.75*tnuvem7))
m8<-cov.mcd(g8,quan=floor(0.75*tnuvem8))


mu8<-
    cbind(m1$center,m2$center,m3$center,m4$center,m5$center,m6$center,m7$center
    ,m8$center)

mu8
sigma8<-rbind(m1$cov,m2$cov,m3$cov,m4$cov,m5$cov,m6$cov,m7$cov,m8$cov)

akaikefinal<-akaikefinal+akaike(dart,8,nt=c(sum(ig1),sum(ig2),sum(ig3),
    sum(ig4),sum(ig5),sum(ig6),sum(ig7),sum(ig8)),mu8,sigma8)

}


akaikefinal<-akaikefinal/100
akaikefinal
```

RRO- programa que implementa a RRO

```
RRO<-function(idat, imetodo, iestimador, itipconstantes, iconstantes,
    inumnuvens, ialfa)
{
  numero<-1
  n<-nrow(idat)
  indicn<-c(1:n)
  final<-rep(0,n)
  niteracoes<-1
  #
  while(numero>0)
  {
    datnovo<-idat[indicn,]
    #
    if(inumnuvens==1)
    {
      result<-rroprogsn(datnovo, iestimador, itipconstantes, iconstantes,
   ialfa)
    }
    else
    {
      result<-rroprog(datnovo, imetodo, iestimador, itipconstantes,
   iconstantes, inumnuvens, ialfa)
    }
    #
    numero<-result$numero
    final<-especial(final, result$novadat)
    indicn<-order(final)[1:(n-sum(final))]
    niteracoes<-niteracoes+1
  }
  #
  if(inumnuvens>1)
  {
    wfinal<-rrooutfinal(n, inumnuvens, final, result$mind, result$novoqui,
   result$cluster1)
    final<-wfinal$final
    numerofinal<-sum(wfinal$final)
    indi<-wfinal$indicn
  }
  else
  {
    numerofinal<-sum(final)
    indi<-indicn
  }
  #
list(fim=final,num=numerofinal,indi=indi)
}
```

```
rroprogsn<-function(idat, iestimador, itipconstantes, iconstantes, ialfa)
{
  n<-nrow(idat)
  variaveis<-ncol(idat)
  #
  if (iestimador==1)
  {
    alfan<-(1-ialfa)^(1/n)
    quiquadrado<-qchisq(alfan,variaveis)
    media1<-apply(idat,2,mean)
    variancia1<-var(idat)
    mh1<-mahalanobis(idat,media1,variancia1,inverted=F)
  }
  if (iestimador==2)
  {
    nk<-round(n/5)
    if(variaveis==2) p<-1
    if(variaveis==3) p<-2
    if(variaveis==4) p<-3
    if(ialfa==0.1) a<-1
    if(ialfa==0.15)  a<-2
    if(ialfa==0.2)  a<-3
    #
    if (nk<=50)
    {
      quiquadrado<-iconstantes[nk,p,a]
    }
    else
    {
      alfan<-(1-ialfa)^(1/n)
      quiquadrado<-qchisq(alfan,variaveis)
    }
    #
    covmcd1<-cov.mcd(idat,quan=floor(0.75*n))
    covmcd1<-as.matrix(covmcd1)
    center1<-covmcd1$center
    cov1<-covmcd1$cov
    mh1<-mahalanobis(idat,center1,cov1,inverted=F)
  }
  #
  if (iestimador==3)
  {
    nk<-round(n/5)
    if(variaveis==2) p<-1
    if(variaveis==3) p<-2
    if(variaveis==4) p<-3
    #
    if (nk<=50)
    {
      quiquadrado<-iconstantes[nk,p]
    }
    else
    {
      alfan<-(1-ialfa)^(1/n)
      quiquadrado<-qchisq(alfan,variaveis)
    }
    #
    ogk1<-final(idat,0.9)
    center1<-ogk1$tw
    cov1<-ogk1$vw
    mh1<-mahalanobis(idat,center1,cov1,inverted=F)
  }
  #
  outl<-matrix(0,n,1)
  indics<-matrix(0,n,1)
  indicn<-matrix(0,n,1)
  #
  contan<-0
```

```
  contas<-0
  #
  for (i in 1:n)
  {
    conta<-0
    if(mh1[i]>quiquadrado)
    {
      outl[i]<-1
      contas<-contas+1
      indics[contas]<-i
    }
    else
    {
      outl[i]<-0
      contan<-contan+1
      indicn[contan]<-i
    }
  }
#
list(indics=indics,indicn=indicn,numero=contas,novadat=outl)
}
```

```
rroprog<-function(idat, imetodo, iestimador, itipconstantes, iconstantes,
    inumnuvens, ialfa)
{
  n<-nrow(idat)
  variaveis<-ncol(idat)
  #
  wnuvens<-rronuvens(idat, imetodo, itipconstantes, iconstantes, inumnuvens,
   ialfa, variaveis)
  #
  westimador<-rroestimador(idat, iestimador, inumnuvens, wnuvens$nuvem1,
   wnuvens$nuvem2, wnuvens$nuvem3, wnuvens$nuvem4, wnuvens$nuvem5,
   wnuvens$tamanho, wnuvens$cluster1, ialfa, variaveis)
  #
  woutliers<-rrooutliers(imetodo, inumnuvens, ialfa, variaveis,
   wnuvens$cluster1, westimador$mh1, westimador$mh2, westimador$mh3,
   westimador$mh4, westimador$mh5, westimador$contateste, wnuvens$quiquadrado,
   wnuvens$tamanho, n)
  #
list(indics=woutliers$indics,indicn=woutliers$indicn,numero=woutliers$numero,n
    ovadat=woutliers$novadat, mind=westimador$mind, novoqui=woutliers$novoqui,
    cluster1=wnuvens$cluster1)
}
```

```
rroestimador<-function(idat, iestimador, inumnuvens, inuvem1, inuvem2,
    inuvem3, inuvem4, inuvem5, itamanho, icluster1, ialfa, ivariaveis)
{
  n<-nrow(idat)
  contateste<-0
  mh1<-0
  mh2<-0
  mh3<-0
  mh4<-0
  mh5<-0
  #
  maxd<-matrix(0,inumnuvens,inumnuvens)
  mind<-matrix(0,inumnuvens,inumnuvens)
  novoqui<-matrix(0,inumnuvens,1)
  #
  if(iestimador==1)
  {
    if(itamanho[1]>2*ivariaveis+1)
    {
      tnuvem1<-nrow(inuvem1)
      media1<-apply(inuvem1,2,mean)
      variancia1<-var(inuvem1)
      invariancia1<-ginverse(variancia1)
      mh1<-mahalanobis(idat,media1,invariancia1,inverted=T)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[1,i1]<-max(mh1*(icluster1==i1))
        mind[1,i1]<-sort(mh1*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
    #
    if(itamanho[2]>2*ivariaveis+1)
    {
      tnuvem2<-nrow(inuvem2)
      media2<-apply(inuvem2,2,mean)
      variancia2<-var(inuvem2)
      invariancia2<-ginverse(variancia2)
      mh2<-mahalanobis(idat,media2,invariancia2,inverted=T)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[2,i1]<-max(mh2*(icluster1==i1))
        mind[2,i1]<-sort(mh2*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
    #
    if(inumnuvens>2)
    {
      if(itamanho[3]>2*ivariaveis+1)
      {
        tnuvem3<-nrow(inuvem3)
        media3<-apply(inuvem3,2,mean)
        variancia3<-var(inuvem3)
        invariancia3<-ginverse(variancia3)
        mh3<-mahalanobis(idat,media3,invariancia3,inverted=T)
        contateste<-contateste+1
        #
        for(i1 in 1:inumnuvens)
        {
          maxd[3,i1]<-max(mh3*(icluster1==i1))
          mind[3,i1]<-sort(mh3*(icluster1==i1))[n-itamanho[i1]+1]
        }
      }
    }
    #
```

```
  if(inumnuvens>3)
  {
    if(itamanho[4]>2*ivariaveis+1)
    {
      tnuvem4<-nrow(inuvem4)
      media4<-apply(inuvem4,2,mean)
      variancia4<-var(inuvem4)
      invariancia4<-ginverse(variancia4)
      mh4<-mahalanobis(idat,media4,invariancia4,inverted=T)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[4,i1]<-max(mh4*(icluster1==i1))
        mind[4,i1]<-sort(mh4*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
  #
  if(inumnuvens>4)
  {
    if(itamanho[5]>2*ivariaveis+1)
    {
      tnuvem5<-nrow(inuvem5)
      media5<-apply(inuvem5,2,mean)
      variancia5<-var(inuvem5)
      invariancia5<-ginverse(variancia5)
      mh5<-mahalanobis(idat,media5,invariancia5,inverted=T)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[5,i1]<-max(mh5*(icluster1==i1))
        mind[5,i1]<-sort(mh5*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
}
#
if(iestimador==2)
{
  if(itamanho[1]>2*ivariaveis+1)
  {
    tnuvem1<-nrow(inuvem1)
    covmcd1<-cov.mcd(inuvem1,quan=floor(0.75*tnuvem1))
    covmcd1<-as.matrix(covmcd1)
    center1<-covmcd1$center
    cov1<-covmcd1$cov
    mh1<-mahalanobis(idat,center1,cov1,inverted=F)
    contateste<-contateste+1
    #
    for(i1 in 1:inumnuvens)
    {
      maxd[1,i1]<-max(mh1*(icluster1==i1))
      mind[1,i1]<-sort(mh1*(icluster1==i1))[n-itamanho[i1]+1]
    }
  }
  #
  if(itamanho[2]>2*ivariaveis+1)
  {
    tnuvem2<-nrow(inuvem2)
    covmcd2<-cov.mcd(inuvem2,quan=floor(0.75*tnuvem2))
    covmcd2<-as.matrix(covmcd2)
    center2<-covmcd2$center
    cov2<-covmcd2$cov
    mh2<-mahalanobis(idat,center2,cov2,inverted=F)
    contateste<-contateste+1
    #
```

```
    for(i1 in 1:inumnuvens)
    {
      maxd[2,i1]<-max(mh2*(icluster1==i1))
      mind[2,i1]<-sort(mh2*(icluster1==i1))[n-itamanho[i1]+1]
    }
  }
  #
  if(inumnuvens>2)
  {
    if(itamanho[3]>2*ivariaveis+1)
    {
      tnuvem3<-nrow(inuvem3)
      covmcd3<-cov.mcd(inuvem3,quan=floor(0.75*tnuvem3))
      covmcd3<-as.matrix(covmcd3)
      center3<-covmcd3$center
      cov3<-covmcd3$cov
      mh3<-mahalanobis(idat,center3,cov3,inverted=F)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[3,i1]<-max(mh3*(icluster1==i1))
        mind[3,i1]<-sort(mh3*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
  #
  if(inumnuvens>3)
  {
    if(itamanho[4]>2*ivariaveis+1)
    {
      tnuvem4<-nrow(inuvem4)
      covmcd4<-cov.mcd(inuvem4,quan=floor(0.75*tnuvem4))
      covmcd4<-as.matrix(covmcd4)
      center4<-covmcd4$center
      cov4<-covmcd4$cov
      mh4<-mahalanobis(idat,center4,cov4,inverted=F)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[4,i1]<-max(mh4*(icluster1==i1))
        mind[4,i1]<-sort(mh4*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
  #
  if(inumnuvens>4)
  {
    if(itamanho[5]>2*ivariaveis+1)
    {
      tnuvem5<-nrow(inuvem5)
      covmcd5<-cov.mcd(inuvem5,quan=floor(0.75*tnuvem5))
      covmcd5<-as.matrix(covmcd5)
      center5<-covmcd5$center
      cov5<-covmcd5$cov
      mh5<-mahalanobis(idat,center5,cov5,inverted=F)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[5,i1]<-max(mh5*(icluster1==i1))
        mind[5,i1]<-sort(mh5*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
}
#
```

```
if(iestimador==3)
{
  if(itamanho[1]>2*ivariaveis+1)
  {
    tnuvem1<-nrow(inuvem1)
    ogk1<-final(inuvem1,0.9)
    center1<-ogk1$tw
    cov1<-ogk1$vw
    mh1<-mahalanobis(idat,center1,cov1,inverted=F)
    contateste<-contateste+1
    #
    for(i1 in 1:inumnuvens)
    {
      maxd[1,i1]<-max(mh1*(icluster1==i1))
      mind[1,i1]<-sort(mh1*(icluster1==i1))[n-itamanho[i1]+1]
    }
  }
  #
  if(itamanho[2]>2*ivariaveis+1)
  {
    tnuvem2<-nrow(inuvem2)
    ogk2<-final(inuvem2,0.9)
    center2<-ogk2$tw
    cov2<-ogk2$vw
    mh2<-mahalanobis(idat,center2,cov2,inverted=F)
    contateste<-contateste+1
    #
    for(i1 in 1:inumnuvens)
    {
      maxd[2,i1]<-max(mh2*(icluster1==i1))
      mind[2,i1]<-sort(mh2*(icluster1==i1))[n-itamanho[i1]+1]
    }
  }
  #
  if(inumnuvens>2)
  {
    if(itamanho[3]>2*ivariaveis+1)
    {
      tnuvem3<-nrow(inuvem3)
      ogk3<-final(inuvem3,0.9)
      center3<-ogk3$tw
      cov3<-ogk3$vw
      mh3<-mahalanobis(idat,center3,cov3,inverted=F)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[3,i1]<-max(mh3*(icluster1==i1))
        mind[3,i1]<-sort(mh3*(icluster1==i1))[n-itamanho[i1]+1]
      }
    }
  }
  #
  if(inumnuvens>3)
  {
    if(itamanho[4]>2*ivariaveis+1)
    {
      tnuvem4<-nrow(inuvem4)
      ogk4<-final(inuvem4,0.9)
      center4<-ogk4$tw
      cov4<-ogk4$vw
      mh4<-mahalanobis(idat,center4,cov4,inverted=F)
      contateste<-contateste+1
      #
      for(i1 in 1:inumnuvens)
      {
        maxd[4,i1]<-max(mh4*(icluster1==i1))
        mind[4,i1]<-sort(mh4*(icluster1==i1))[n-itamanho[i1]+1]
```

```
        }
      }
    }
    #
    if(inumnuvens>4)
    {
      if(itamanho[5]>2*ivariaveis+1)
      {
        tnuvem5<-nrow(inuvem5)
        ogk5<-final(inuvem5,0.9)
        center5<-ogk5$tw
        cov5<-ogk5$vw
        mh5<-mahalanobis(idat,center5,cov5,inverted=F)
        contateste<-contateste+1
        #
        for(i1 in 1:inumnuvens)
        {
          maxd[5,i1]<-max(mh5*(icluster1==i1))
          mind[5,i1]<-sort(mh5*(icluster1==i1))[n-itamanho[i1]+1]
        }
      }
    }
  }
  #
list(mh1=mh1, mh2=mh2, mh3=mh3, mh4=mh4, mh5=mh5, contateste=contateste,
   mind=mind)
}
```

```
rronuvens<-function(idat, imetodo, itipconstantes, iconstantes, inumnuvens,
   ialfa, ivariaveis)
{
  n<-nrow(idat)
  #
  if (imetodo==1)
  {
    clus<-kmeans(idat, inumnuvens)
    cluster1<-clus$cluster
  }
  #
  if (imetodo==2)
  {
    clus<-pam(idat, inumnuvens)
    cluster1<-clus$cluster
  }
  #
  if (imetodo==3)
  {
    clusmc<-mclust(idat)
    clus<-mclass(clusmc,inumnuvens)
    cluster1<-category(clus$classification)
  }
  #
  nuvens<-matrix(0,n,ivariaveis+1)
  #
  for (i in 1:n)
  {
    nuvens[i,1]<-cluster1[i]
    #
    for (j in 1:ivariaveis)
    {
      nuvens[i,j+1]<-idat[i,j]
    }
  }
  #
  if (imetodo==1)
  {
    tamanho<-clus$size
  }
  else
  {
    tamanho<-matrix(0,inumnuvens)
    for (i in 1:n)
    {
      tamanho[cluster1[i]]<-tamanho[cluster1[i]]+1
    }
  }
  #
  quiquadrado<-matrix(0,n,1)
  #
  #
  for (i in 1:inumnuvens)
  {
    nuvem<-matrix(0,tamanho[i],ivariaveis)
    cont<-1
    #
    for (j in 1:n)
    {
      if(nuvens[j,1]==i)
      {
        #
        if (itipconstantes==1)
        {
          alfan<-(1-ialfa)^(1/n)
          quiquadrado[j]<-qchisq(alfan,ivariaveis)
        }
        else
```

```
        {
          nk<-round(tamanho[i]/5)
          if(ivariaveis==2) p<-1
          if(ivariaveis==3) p<-2
          if(ivariaveis==4) p<-3
          #
          k<-inumnuvens-1
          if(ialfa==0.1) a<-1
          if(ialfa==0.15)  a<-2
          if(ialfa==0.20)  a<-3
          #
          if (nk<=50)
          {
            if (itipconstantes==2) quiquadrado[j]<-iconstantes[nk,p,k,a]
            if (itipconstantes==3) quiquadrado[j]<-iconstantes[nk,p,k]

          }
          else
          {
            alfan<-(1-ialfa)^(1/n)
            quiquadrado[j]<-qchisq(alfan,ivariaveis)
          }
        }
        #
        for (k in 1:ivariaveis)
        {
          nuvem[cont,k]<-nuvens[j,k+1]
        }
        cont<-cont+1
      }
    }
    #
    if(i==1)
    {
      nuvem1<-matrix(0,tamanho[i],2)
      nuvem1<-nuvem
    }
    #
    if(i==2)
    {
      nuvem2<-matrix(0,tamanho[i],2)
      nuvem2<-nuvem
    }
    #
    if(i==3)
    {
      nuvem3<-matrix(0,tamanho[i],2)
      nuvem3<-nuvem
    }
    #
    if(i==4)
    {
      nuvem4<-matrix(0,tamanho[i],2)
      nuvem4<-nuvem
    }
    #
    if(i==5)
    {
      nuvem5<-matrix(0,tamanho[i],2)
      nuvem5<-nuvem
    }
  }
  #
list(nuvem1=nuvem1, nuvem2=nuvem2, nuvem3=nuvem3, nuvem4=nuvem4,
   nuvem5=nuvem5, tamanho=tamanho, cluster1=cluster1, quiquadrado=quiquadrado)
}
```

```
rrooutfinal<-function(indata, inumnuvens, ifinal, imind, inovoqui, icluster1)
{
  n<-indata
  final<-ifinal
  #
  xcl<-matrix(0,inumnuvens,inumnuvens)
  #
  for (i in 1:inumnuvens)
  {
    for (j in 1:inumnuvens)
    {
      xcl[i,j]<-(imind[i,j] > inovoqui[i])
    }
  }
  #
  antefinal<-matrix(0,length(icluster1),1)
  #
  for (i in 1:inumnuvens)
  {
    if (sum(xcl[,i])==inumnuvens-1) antefinal<-antefinal+(icluster1==i)
  }
  #
  final<-especial(final,antefinal)
  indicn<-c(1:(n-sum(final)))
  indicn<-order(final)[1:(n-sum(final))]
  #
list(final=final,indicn=indicn)
}
```

```
rrooutliers<-function(imetodo, inumnuvens, ialfa, ivariaveis, icluster1, imh1,
    imh2, imh3, imh4, imh5, icontateste, iquiquadrado, itamanho, nin)
{
  n<-nin
  outl<-matrix(0,n,1)
  indics<-matrix(0,n,1)
  indicn<-matrix(0,n,1)
  contan<-0
  contas1<-0
  contas2<-0
  novoqui<-rep(0,inumnuvens)
  #
  for (i in 1:n)
  {
  conta1<-0
  conta2<-0
  conta3<-0
  conta4<-0
  conta5<-0
    #
    if(itamanho[1]>2*ivariaveis+1)
    {
      if(imh1[i]>iquiquadrado[i]) conta1<-conta1+1
    }
    if(itamanho[2]>2*ivariaveis+1)
    {
      if(imh2[i]>iquiquadrado[i]) conta2<-conta2+1
    }
    if(inumnuvens>2)
    {
      if(itamanho[3]>2*ivariaveis+1)
      {
        if(imh3[i]>iquiquadrado[i]) conta3<-conta3+1
      }
    }
    if(inumnuvens>3)
    {
      if(itamanho[4]>2*ivariaveis+1)
      {
        if(imh4[i]>iquiquadrado[i]) conta4<-conta4+1
      }
    }
    if(inumnuvens>4)
    {
      if(itamanho[5]>2*ivariaveis+1)
      {
        if(imh5[i]>iquiquadrado[i]) conta5<-conta5+1
      }
    }
    #
    conta<-conta1+conta2+conta3+conta4+conta5
    #
    if(conta==icontateste)
    {
      outl[i]<-1
      contas1<-contas1+1
      indics[contas1]<-i
    }
    else
    {
      if (itamanho [icluster1[i]]<=2*ivariaveis+1)
      {
        outl[i]<-1
        contas2<-contas2+1
        indics[contas2]<-i
      }
      else
      {
```

```
        outl[i]<-0
        contan<-contan+1
        indicn[contan]<-i
      }
    }
  }
  #
  contas<-contas1+contas2
  #
  if(ivariaveis==2) p<-1
  if(ivariaveis==3) p<-2
  if(ivariaveis==4) p<-3
  #
  k<-inumnuvens-1
  if(ialfa==0.1) a<-1
  if(ialfa==0.15)  a<-2
  if(ialfa==0.20)  a<-3
  #
  for (i in 1:inumnuvens)
  {
    nk<-round(itamanho[i]/5)
    if (nk<=50)
    {
      if (itipconstantes==1)
      {
        alfan<-(1-ialfa)^(1/n)
        novoqui[i]<-qchisq(alfan,ivariaveis)
       }
      #
      if (itipconstantes==2) novoqui[i]<-iconstantes[nk,p,k,a]
      if (itipconstantes==3) novoqui[i]<-iconstantes[nk,p,k]
    }
    else
    {
      alfan<-(1-ialfa)^(1/n)
      novoqui[i]<-qchisq(alfan,ivariaveis)
    }
  }
#
list(numero=contas,novadat=outl,novoqui=novoqui, indics=indics, indicn=indicn)
}
```

```
especial<-function(vec1,vec2)
{
  nant<-0
  n<-length(vec1)
  final<-rep(0,n)
  for (i in 1:n){
  j<-i-nant
  if (vec1[i] == 0 & vec2[j] == 0) final[i] <-0
  if (vec1[i] == 1 )
  {
    final[i]<-1
    nant <- nant+1}
    if (vec1[i] == 0 & vec2[j] == 1) final[i] <-1
  }
  return(final)
}
```

RROSIM- programa que implementa a RRO para dados simulados

```
RROSIM<-function(imetodo, iestimador, itipconstantes, iconstantes, inumnuvens,
    ialfa, inrepeticoes, iopcao)
{

nroutliers<-matrix(0,inrepeticoes)
mi<-matrix(0,inrepeticoes)
si<-matrix(0,inrepeticoes)
idemi<-matrix(0,inrepeticoes)
idesi<-matrix(0,inrepeticoes)

numero<-1

while(numero<=inrepeticoes)
{

  if(iopcao==1)  resdat<-datsim1()
  if(iopcao==2)  resdat<-datsim2()
  if(iopcao==3)  resdat<-datsim3()
  if(iopcao==4)  resdat<-datsim4()
  if(iopcao==5)  resdat<-datsim5()
  if(iopcao==6)  resdat<-datsim6()
  if(iopcao==7)  resdat<-datsim7()
  if(iopcao==8)  resdat<-datsim8()
  if(iopcao==9)  resdat<-datsim9()
  if(iopcao==10) resdat<-datsim10()
  if(iopcao==11) resdat<-datsim11()
  if(iopcao==12) resdat<-datsim12()
  if(iopcao==13) resdat<-datsim13()
  if(iopcao==14) resdat<-datsim14()
  if(iopcao==15) resdat<-datsim15()
  if(iopcao==16) resdat<-datsim16()
  if(iopcao==17) resdat<-datsim17()
  if(iopcao==18) resdat<-datsim18()
  if(iopcao==19) resdat<-datsim19()
  if(iopcao==20) resdat<-datsim20()
  if(iopcao==21) resdat<-datsim21()
  if(iopcao==22) resdat<-datsim22()
  if(iopcao==23) resdat<-datsim23()
  if(iopcao==24) resdat<-datsim24()
  if(iopcao==25) resdat<-datsim25()
  if(iopcao==26) resdat<-datsim26()
  if(iopcao==27) resdat<-datsim27()
  if(iopcao==28) resdat<-datsim28()
  if(iopcao==29) resdat<-datsim29()
  if(iopcao==30) resdat<-datsim30()
  if(iopcao==31) resdat<-datsim31()
  if(iopcao==32) resdat<-datsim32()

  inicio<-resdat$inicio
  n<-resdat$n
  g<-resdat$g
  b<-resdat$b

  resout<-RRO(resdat$dat, imetodo, iestimador, itipconstantes, iconstantes,
   inumnuvens, ialfa)

  nroutliers[numero]<-resout$num

  resinifim<-ressimu(numero,inicio,resout$fim,n)

  mi[numero]<-resinifim$imi
  si[numero]<-resinifim$isi

  if(resinifim$imi==0) idemi[numero]<-1
```

```
  if(resinifim$isi==0) idesi[numero]<-1

  numero<-numero + 1

}

# ATENCAO
# se b=0, p1=0 e p2=0
#
if(b!=0)
{
p1<-(1/inrepeticoes)*sum(idemi)
p2<-(1/(inrepeticoes*b))*sum(mi)
p3<-(1/(inrepeticoes*g))*sum(si)
p4<-(1/inrepeticoes)*sum(idesi)
}
else
{
p1<-0
p2<-0
p3<-(1/(inrepeticoes*g))*sum(si)
p4<-(1/inrepeticoes)*sum(idesi)
}

list(op1=p1,op2=p2,op3=p3,op4=p4)
}
```

```
ressimu<-function(inumero,iinicio,ifim,ene)
{
   imi<-0
   isi<-0

   for (i in 1:ene)
   {
      if(ifim[i]==0 & iinicio[i]==1) imi<-imi+1
      if(ifim[i]==1 & iinicio[i]==0) isi<-isi+1

   }

   list(imi=imi,isi=isi)

}
```

Apêndice F

Conjuntos de dados

F.1 Dados do Exemplo 3.1

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	classe
52	1	0	1	0	0	0	4	0	0
64	1	0	1	0	0	0	0	1	0
45	0	0	1	0	0	0	2	0	0
57	1	0	1	0	0	0	0	0	0
60	1	0	0	0	1	0	0	0	0
53	1	1	0	0	0	0	-5	0	0
36	1	0	1	0	0	0	3	1	0
50	1	0	1	0	0	1	4	1	0
55	1	0	1	0	0	1	0	0	0
51	0	0	1	0	0	1	3	1	0
49	0	1	1	0	0	0	0	1	0
38	1	1	0	0	0	0	1	0	0
32	0	0	1	2	0	0	3	0	0
52	0	0	1	1	0	1	-3	1	0
45	0	0	0	0	0	1	0	0	0
48	0	0	0	0	0	1	4	1	0
40	0	0	1	3	0	1	0	0	0
40	0	0	0	0	0	0	2	0	0
47	0	0	1	2	0	0	-4	1	0
42	0	0	0	2	0	1	-2	0	0
50	1	1	1	1	0	0	0	0	0
49	1	1	1	0	0	0	0	1	0
62	1	1	0	0	0	0	-3	0	0
51	0	0	0	0	0	0	1	0	0
50	0	0	1	0	0	1	3	0	0
55	1	0	0	0	0	0	3	1	0
60	0	0	1	0	0	1	3	1	0
50	0	1	1	0	0	0	0	0	0
48	1	0	1	0	0	0	-3	1	0
55	1	1	0	0	0	0	6	0	0
63	1	1	1	0	0	0	-2	0	1
57	1	0	1	0	1	1	0	1	1
59	1	0	0	0	1	1	0	1	1
48	1	1	1	0	0	0	2	0	1
61	1	1	0	0	0	1	0	0	1
60	1	1	1	0	0	1	3	1	1
41	1	0	1	2	0	1	3	1	1
55	1	1	1	0	0	0	3	1	1
69	1	0	1	0	0	1	2	1	1
48	1	1	1	1	0	1	2	1	1

(continua)

46	1	1	0	0	0	1	3	0	1
48	1	1	0	1	1	0	0	0	1
56	1	0	1	0	0	1	4	1	1
63	1	1	1	0	0	0	-2	1	1
44	1	1	1	3	0	0	4	1	1
53	1	0	1	0	0	1	-2	0	1
45	1	1	1	0	0	1	0	1	1
59	1	1	1	0	1	0	0	0	1
55	1	1	1	0	0	1	5	1	1
42	1	1	1	0	0	0	-3	1	1
60	1	1	1	0	0	0	0	1	1
51	0	0	0	0	0	0	3	0	1
70	1	1	1	0	0	0	0	1	1
56	1	0	1	0	0	1	5	1	1
61	0	1	0	0	1	0	0	0	1
53	1	1	1	0	0	1	-3	1	1
56	1	0	1	0	1	1	0	0	1
41	1	1	1	0	1	0	4	1	1
53	1	1	0	0	1	0	-1	1	1
35	1	1	1	0	0	0	0	1	1
63	1	0	1	2	1	1	0	1	1
35	1	1	1	0	1	1	-5	1	1
42	1	0	1	0	0	1	1	1	1
51	1	0	0	0	1	1	1	0	1
50	1	0	1	0	0	0	1	1	1
55	0	0	0	2	1	0	-1	1	1
54	1	0	1	1	0	0	-1	1	1
49	1	0	1	1	0	1	0	1	1
53	1	0	0	0	0	1	-2	1	1
53	1	1	1	0	0	0	0	0	1
39	1	1	1	0	0	1	2	0	1
55	1	0	1	0	0	0	2	0	1
65	1	0	1	0	0	0	0	1	1
57	1	0	1	2	0	0	0	1	1
56	1	0	1	1	0	0	-1	1	1
52	1	0	1	1	0	1	3	1	1
62	1	1	1	0	1	0	4	1	1
51	1	1	1	0	1	1	3	1	1
62	1	0	1	0	0	0	1	1	1
56	1	0	1	0	0	1	0	0	1
60	1	0	1	1	1	1	0	1	1
43	1	0	1	0	0	1	8	0	1
64	1	0	1	0	1	0	0	1	1
53	0	0	1	2	1	1	3	0	1
57	1	0	1	2	0	1	4	1	1
60	1	0	0	2	1	1	2	0	1
64	1	1	1	1	1	0	0	1	1
57	1	1	0	0	0	1	0	0	1
57	1	0	1	0	1	1	0	1	1
51	1	0	0	0	0	0	-5	0	1
69	1	0	1	1	0	0	0	1	1

(continuação)

(continua)

53	1	1	0	2	1	0	0	0	1
53	1	0	1	0	0	0	0	1	1
62	1	1	1	0	0	0	2	0	1
52	1	1	1	0	0	1	-3	1	1
49	1	1	0	0	1	0	-3	0	1
67	0	1	1	0	0	1	0	1	1
53	1	1	1	1	0	0	2	1	1
40	1	1	1	0	0	0	3	1	1
45	1	0	1	2	0	0	2	1	1
53	1	1	1	0	0	1	0	1	1
53	1	0	1	2	0	0	0	1	1
51	1	1	1	0	0	1	2	1	1
55	1	0	1	0	0	1	3	1	1
60	1	1	0	0	1	1	2	0	1
64	1	0	1	0	0	1	0	1	1
56	1	1	0	0	0	0	-2	0	1
57	1	0	1	0	1	1	0	0	1
59	1	0	0	0	0	1	0	0	1
55	1	1	1	2	0	1	-3	1	1
47	1	1	0	0	0	1	0	0	1
54	1	1	0	0	1	1	-7	1	1
52	1	1	1	2	0	0	-5	0	1

(continuação)

x_1: idade (anos)
x_2: sexo (0-feminino/ 1-masculino)
x_3: factor de risco (0-ausente/ 1-presente)
x_4: dor precordial (0-espontânea/ 1-esforço)
x_5: pressão arterial (0-sem alteração/ 1- aumento da diastólica/ 2- decréscimo da sitólica/ 3-ambas
x_6: aparecimento de arritmias (0-não/ 1-sim)
x_7: depressão do ST (0-não/ 1-sim)
x_8: alteração na onda R (reais)
x_9: dor precordial perante o teste (0-não/ 1-sim)

F.2a Dados do Exemplo 3.2: amostra de treino

npreg	glu	bmi p	ed	age	classe
5	86	30.2	0.364	24	0
7	195	25.1	0.163	55	1
5	77	35.8	0.156	35	0
0	165	47.9	0.259	26	0
0	107	26.4	0.133	23	0
5	97	35.6	0.378	52	1
3	83	34.3	0.336	25	0
1	193	25.9	0.655	24	0
3	142	32.4	0.200	63	0

(continua)

2	128	43.3	1.224	31	1
0	137	43.1	2.288	33	1
9	154	30.9	0.164	45	0
1	189	30.1	0.398	59	1
12	92	27.6	0.926	44	1
1	86	41.3	0.917	29	0
4	99	23.2	0.223	21	0
1	109	25.4	0.947	21	0
11	143	36.6	0.254	51	1
1	149	29.3	0.349	42	1
0	139	22.1	0.207	21	0
2	99	20.4	0.235	27	0
1	100	32.0	0.444	42	0
4	83	29.3	0.317	34	0
0	101	21.0	0.252	21	0
1	87	37.6	0.401	24	0
9	164	30.8	0.831	32	1
1	99	25.4	0.551	21	0
0	140	42.6	0.431	24	1
5	108	36.1	0.263	33	0
2	110	32.4	0.698	27	0
1	79	43.5	0.678	23	0
3	148	32.5	0.256	22	0
0	121	34.3	0.203	33	1
3	158	31.2	0.295	24	0
2	105	33.7	0.711	29	1
13	145	22.2	0.245	57	0
1	79	25.4	0.583	22	0
1	71	20.4	0.323	22	0
0	102	29.3	0.695	27	0
0	119	38.8	0.259	22	0
8	176	33.7	0.467	58	1
1	97	27.2	1.095	22	0
4	129	27.5	0.527	31	0
1	97	18.2	0.299	21	0
0	86	35.8	0.238	25	0
2	125	33.8	0.088	31	0
5	123	34.1	0.269	28	0
2	92	24.2	1.698	28	0
3	171	33.3	0.199	24	1
1	199	42.9	1.394	22	1
3	116	26.3	0.107	24	0
2	83	32.2	0.497	22	0
8	154	32.4	0.443	45	1
1	114	38.1	0.289	21	0
1	106	34.2	0.142	22	0
4	127	34.5	0.598	28	0
1	124	27.8	0.100	30	0
1	109	23.1	0.407	26	0
2	123	42.1	0.520	26	0
8	167	37.6	0.165	43	1

(continuação)

(continua)

7	184	35.5	0.355	41	1
1	96	33.2	0.289	21	0
10	129	35.9	0.280	39	0
6	92	32.0	0.085	46	0
6	109	25.0	0.206	27	0
5	139	31.6	0.361	25	1
6	134	35.4	0.542	29	1
3	106	30.9	0.292	24	0
0	131	34.3	0.196	22	1
0	135	40.6	0.284	26	0
5	158	39.4	0.395	29	1
3	112	31.6	0.197	25	1
8	181	30.1	0.615	60	1
2	121	39.1	0.886	23	0
1	168	35.0	0.905	52	1
1	144	46.1	0.335	46	1
2	101	24.2	0.614	23	0
2	96	21.1	0.647	26	0
3	107	22.9	0.678	23	1
12	121	26.5	0.259	62	0
2	100	29.7	0.368	21	0
4	154	31.3	0.338	37	0
6	125	27.6	0.565	49	1
10	125	31.1	0.205	41	1
2	122	35.9	0.483	26	0
2	114	28.7	0.092	25	0
1	115	34.6	0.529	32	1
7	114	23.8	0.466	31	0
2	115	30.8	0.421	21	0
1	130	28.6	0.692	21	0
1	79	32.0	0.396	22	0
4	112	39.4	0.236	38	0
7	150	35.2	0.692	54	1
1	91	25.2	0.234	23	0
1	100	25.3	0.658	28	0
12	140	39.2	0.528	58	1
4	110	28.4	0.118	27	0
2	94	31.6	0.649	23	0
2	84	30.4	0.968	21	0
10	148	37.6	1.001	51	1
3	61	34.4	0.243	46	0
4	117	29.7	0.380	30	1
3	99	19.3	0.284	30	0
3	80	34.2	1.292	27	1
4	154	32.8	0.237	23	0
6	103	37.7	0.324	55	0
6	111	34.2	0.260	24	0
0	124	27.4	0.254	36	1
1	143	26.2	0.256	21	0
1	81	46.3	1.096	32	0
4	189	28.5	0.680	37	0

(continuação)

(continua)

4	116	22.1	0.463	37	0
7	103	39.1	0.344	31	1
8	124	28.7	0.687	52	1
1	71	33.2	0.422	21	0
0	137	27.3	0.231	59	0
9	112	34.2	0.260	36	1
4	148	30.9	0.150	29	1
1	136	37.4	0.399	24	0
9	145	37.9	0.637	40	1
1	93	22.5	0.417	22	0
1	107	30.8	0.821	24	0
12	151	41.8	0.742	38	1
1	97	38.1	0.218	30	0
5	144	32.0	0.452	58	1
2	112	38.4	0.246	28	0
2	99	24.6	0.637	21	0
1	109	25.2	0.833	23	0
1	120	38.9	1.162	41	0
7	187	37.7	0.254	41	1
3	129	36.4	0.968	32	1
7	179	34.2	0.164	60	0
6	80	26.2	0.313	41	0
2	105	34.9	0.225	25	0
3	191	30.9	0.299	34	0
0	95	36.5	0.330	26	0
4	99	25.6	0.294	28	0
0	137	24.8	0.143	21	0
1	97	18.2	0.147	21	0
0	100	46.8	0.962	31	0
1	167	23.4	0.447	33	1
0	180	36.5	0.314	35	1
2	122	36.8	0.340	27	0
1	90	27.2	0.580	24	0
3	120	42.9	0.452	30	0
6	154	46.1	0.571	27	0
2	56	24.2	0.332	22	0
0	177	34.6	1.072	21	1
3	124	33.2	0.305	26	0
8	85	24.4	0.136	42	0
12	88	35.3	0.378	48	0
9	152	34.2	0.893	33	1
0	198	41.3	0.502	28	1
0	188	32.0	0.682	22	1
5	139	28.6	0.411	26	0
7	168	38.2	0.787	40	1
2	197	34.7	0.575	62	1
2	142	24.7	0.761	21	0
8	126	25.9	0.162	39	0
3	158	31.6	0.851	28	1
3	130	28.4	0.323	34	1
2	100	37.8	0.498	24	0

(continuação)

(continua)

1	164	32.8	0.341	50	0
4	95	35.4	0.284	28	0
2	122	36.2	0.816	28	0
4	85	27.8	0.306	28	0
0	151	42.1	0.371	21	1
6	144	33.9	0.255	40	0
3	111	28.4	0.495	29	0
1	107	26.5	0.165	24	0
6	115	33.7	0.245	40	1
5	105	36.9	0.159	28	0
7	194	35.9	0.745	41	1
4	184	37.0	0.264	31	1
0	95	37.4	0.247	24	1
7	124	25.5	0.161	37	0
1	111	24.0	0.138	23	0
7	137	32.0	0.391	39	0
9	57	32.8	0.096	41	0
2	157	39.4	0.134	30	0
2	95	26.1	0.748	22	0
12	140	37.4	0.244	41	0
0	117	30.8	0.493	22	0
8	100	39.4	0.661	43	1
9	123	33.1	0.374	40	0
0	138	34.6	0.534	21	1
14	100	36.6	0.412	46	1
14	175	33.6	0.212	38	1
0	74	27.8	0.269	22	0
1	133	32.8	0.234	45	1
0	119	34.9	0.725	23	0
5	155	38.7	0.619	34	0
1	128	40.5	0.613	24	1
2	112	34.1	0.315	26	0
1	140	24.1	0.828	23	0
2	141	25.4	0.699	24	0
7	129	38.5	0.439	43	1
0	106	39.4	0.605	22	0
1	118	33.3	0.261	23	0
8	155	34.0	0.543	46	1

(continuação)

F.2b Dados do Exemplo 3.2: amostra de teste

npreg	glu	bmi p	ed	age	classe
1	85	26.6	0.351	31	0
1	89	28.1	0.167	21	0
1	103	43.3	0.183	33	0
3	126	39.3	0.704	27	0

(continua)

1	97	23.2	0.487	22	0
5	109	36.0	0.546	60	0
3	88	24.8	0.267	22	0
10	122	27.6	0.512	45	0
4	103	24.0	0.966	33	0
3	180	34.0	0.271	26	0
7	106	22.7	0.235	48	0
2	71	28.0	0.586	22	0
1	103	19.4	0.491	22	0
1	101	24.2	0.526	26	0
5	88	24.4	0.342	30	0
7	150	34.7	0.718	42	0
1	73	23.0	0.248	21	0
0	105	41.5	0.173	22	0
5	99	29.0	0.203	32	0
1	95	19.6	0.334	25	0
4	146	28.9	0.189	27	0
4	129	35.1	0.231	23	0
5	95	37.7	0.370	27	0
2	112	25.0	0.307	24	0
3	113	22.4	0.140	22	0
7	83	29.3	0.767	36	0
0	101	24.6	0.237	22	0
13	106	36.6	0.178	45	0
2	100	38.5	0.324	26	0
4	123	32.0	0.443	34	0
7	81	46.7	0.261	42	0
2	92	31.6	0.130	24	0
6	93	28.7	0.356	23	0
1	81	26.6	0.283	24	0
1	126	28.7	0.801	21	0
4	144	29.5	0.287	37	0
1	89	31.2	0.192	23	0
4	97	28.2	0.443	22	0
2	107	33.6	0.404	23	0
8	84	38.3	0.457	39	0
0	100	30.8	0.597	21	0
0	93	28.7	0.532	22	0
5	106	39.5	0.286	38	0
2	108	32.5	0.318	22	0
2	106	30.5	1.400	34	0
2	90	27.3	0.085	22	0
1	153	40.6	0.687	23	0
2	88	29.0	0.229	22	0
4	151	29.7	0.294	36	0
7	102	37.2	0.204	45	0
0	114	44.2	0.167	27	0
2	75	29.7	0.370	33	0
0	113	31.0	0.874	21	0
0	108	27.3	0.787	32	0
5	111	23.9	0.407	27	0

(continuação)

(continua)

2	81	27.7	0.290	25	0
0	147	42.8	0.375	24	0
6	125	30.0	0.464	32	0
7	142	28.8	0.687	61	0
1	100	23.6	0.666	26	0
1	87	34.6	0.101	22	0
4	197	36.7	2.329	31	0
0	117	45.2	0.089	24	0
3	74	29.7	0.293	23	0
1	91	29.2	0.192	21	0
4	91	33.1	0.446	22	0
2	146	38.2	0.329	29	0
0	165	52.3	0.427	23	0
9	124	35.4	0.282	34	0
1	111	30.1	0.143	23	0
2	90	24.4	0.249	24	0
3	111	30.1	0.557	30	0
4	95	32.1	0.612	24	0
5	96	33.6	0.997	43	0
2	128	40.0	1.101	24	0
2	108	25.2	0.128	21	0
2	100	40.5	0.677	25	0
0	104	27.8	0.454	23	0
2	108	25.3	0.881	22	0
7	133	32.4	0.262	37	0
7	136	26.0	0.647	51	0
4	96	20.8	0.340	26	0
0	78	36.9	0.434	21	0
6	151	35.5	0.692	28	0
0	126	30.7	0.520	24	0
2	120	39.7	0.215	29	0
3	113	29.5	0.626	25	0
3	115	38.1	0.150	28	0
2	112	35.7	0.148	21	0
1	157	25.6	0.123	24	0
6	105	30.8	0.122	37	0
8	118	23.1	1.476	46	0
2	87	32.7	0.166	25	0
1	95	23.9	0.260	22	0
1	130	25.9	0.472	22	0
1	95	25.9	0.673	36	0
8	126	38.5	0.349	49	0
1	139	28.7	0.654	22	0
3	99	21.8	0.279	26	0
5	103	39.2	0.305	65	0
4	147	34.9	0.385	30	0
5	99	34.0	0.499	30	0
3	81	27.5	0.306	22	0
0	84	35.8	0.545	21	0
0	98	25.2	0.299	22	0
1	87	37.2	0.509	22	0

(continuação)

(continua)

0	93	43.4	1.021	35	0
0	105	20.0	1.021	22	0
1	90	25.1	1.268	25	0
1	125	24.3	0.221	25	0
1	119	22.3	0.205	24	0
3	100	31.6	0.949	28	0
1	131	23.7	0.389	21	0
2	127	27.7	1.600	25	0
3	96	24.7	0.944	39	0
9	72	31.6	0.280	38	0
6	102	35.7	0.674	28	0
1	112	34.4	0.528	25	0
1	143	42.4	1.076	22	0
1	119	45.3	0.507	26	0
2	94	26.0	0.561	21	0
0	102	40.6	0.496	21	0
3	89	30.4	0.551	38	0
1	80	30.0	0.527	22	0
1	90	24.5	1.138	36	0
4	117	33.2	0.230	24	0
0	120	30.5	0.285	26	0
1	82	21.2	0.415	23	0
0	91	39.9	0.381	25	0
9	134	25.9	0.460	81	0
9	120	20.8	0.733	48	0
8	74	35.3	0.705	39	0
5	88	27.6	0.258	37	0
0	124	21.8	0.452	21	0
0	97	36.8	0.600	25	0
1	144	41.3	0.607	28	0
0	137	33.2	0.170	22	0
4	132	28.0	0.419	63	0
0	123	35.2	0.197	29	0
0	84	38.2	0.233	23	0
1	139	40.7	0.536	21	0
0	173	46.5	1159.000	58	0
2	83	36.8	0.629	24	0
2	89	33.5	0.292	42	0
4	99	32.8	0.145	33	0
2	81	30.1	0.547	25	0
6	154	29.3	0.839	39	0
2	117	25.2	0.313	21	0
3	84	37.2	0.267	28	0
7	94	33.3	0.738	41	0
3	96	37.3	0.238	40	0
3	99	25.6	0.154	24	0
6	129	19.6	0.582	60	0
2	68	25.0	0.187	25	0
3	87	21.8	0.444	21	0
2	122	29.8	0.717	22	0
1	77	33.3	1.251	24	0

(continuação)

(continua)

0	127	36.3	0.804	23	0
4	84	39.5	0.159	25	0
1	88	32.0	0.365	29	0
4	131	33.1	0.160	28	0
1	116	27.4	0.204	21	0
3	84	31.9	0.591	25	0
1	88	29.9	0.422	23	0
1	84	36.9	0.471	28	0
11	103	46.2	0.126	42	0
6	99	26.9	0.497	32	0
1	99	38.6	0.412	21	0
3	111	29.5	0.430	22	0
2	98	34.7	0.198	22	0
1	143	30.1	0.892	23	0
1	119	35.5	0.280	25	0
6	108	24.0	0.813	35	0
2	112	39.4	0.175	24	0
2	82	28.5	1.699	25	0
6	123	33.6	0.733	34	0
1	89	27.8	0.559	21	0
1	108	27.1	0.400	24	0
1	124	35.8	0.514	21	0
1	92	19.5	0.482	25	0
0	152	41.5	0.270	27	0
6	105	32.5	0.878	26	0
2	68	20.1	0.257	23	0
0	94	43.5	0.347	21	0
4	90	37.7	0.362	29	0
4	94	24.7	0.148	21	0
0	102	34.5	0.238	24	0
1	128	27.5	0.115	22	0
1	100	19.5	0.149	28	0
3	103	27.6	0.730	27	0
1	117	33.8	0.466	27	0
2	101	21.8	0.155	22	0
1	112	34.8	0.217	24	0
6	98	34.0	0.430	43	0
6	165	33.6	0.631	49	0
10	68	35.5	0.285	47	0
3	123	57.3	0.880	22	0
0	95	44.6	0.366	22	0
2	129	33.2	0.591	25	0
1	107	28.3	0.181	29	0
6	80	39.8	0.177	28	0
2	127	34.4	0.176	22	0
5	126	29.6	0.439	40	0
0	134	26.4	0.352	21	0
10	94	23.1	0.595	56	0
1	108	35.5	0.415	24	0
5	117	39.1	0.251	42	0
1	116	36.1	0.496	25	0

(continuação)

(continua)

(continuação)

0	141	32.4	0.433	22	0
2	106	29.0	0.426	22	0
0	126	27.4	0.515	21	0
8	65	32.0	0.600	42	0
2	99	36.6	0.453	21	0
3	102	30.8	0.400	26	0
1	109	28.5	0.219	22	0
13	153	40.6	1.174	39	0
12	100	30.0	0.488	46	0
1	121	39.0	0.261	28	0
3	108	26.0	0.223	25	0
2	88	28.4	0.766	22	0
10	101	32.9	0.171	63	0
5	121	26.2	0.245	30	0
1	93	30.4	0.315	23	0
6	148	33.6	0.627	50	1
3	78	31.0	0.248	26	1
2	197	30.5	0.158	53	1
5	166	25.8	0.587	51	1
0	118	45.8	0.551	31	1
9	119	29.0	0.263	29	1
9	102	32.9	0.665	46	1
2	90	38.2	0.503	27	1
4	111	37.1	1.390	56	1
9	171	45.4	0.721	54	1
0	180	42.0	1.893	25	1
0	109	32.5	0.855	38	1
2	100	32.9	0.867	28	1
15	136	37.1	0.153	43	1
1	122	49.7	0.325	31	1
7	160	30.5	0.588	39	1
0	162	53.2	0.759	25	1
1	88	55.0	0.496	26	1
1	117	34.5	0.403	40	1
4	173	29.7	0.361	33	1
3	170	34.5	0.356	30	1
9	156	34.3	1.189	42	1
7	152	50.0	0.337	36	1
17	163	40.9	0.817	47	1
6	104	29.9	0.722	41	1
8	179	32.7	0.719	36	1
0	129	67.1	0.319	26	1
1	128	32.0	1.321	33	1
8	109	27.9	0.640	31	1
4	109	34.8	0.905	26	1
8	196	37.5	0.605	57	1
5	109	35.8	0.514	25	1
5	85	29.0	1224.000	32	1
3	162	37.2	0.652	24	1
6	134	46.2	0.238	46	1
7	181	35.9	0.586	51	1

(continua)

0	179	44.1	0.686	23	1
6	119	27.1	1.318	33	1
9	184	30.0	1.213	49	1
1	113	33.6	0.543	21	1
11	155	33.3	1.353	51	1
10	101	45.6	1.136	38	1
7	106	26.5	0.296	29	1
1	119	45.6	0.808	29	1
0	107	36.6	0.757	25	1
2	146	28.0	0.337	29	1
2	144	31.6	0.422	25	1
10	161	25.5	0.326	47	1
0	128	30.5	1.391	25	1
2	124	32.9	0.875	30	1
2	155	26.6	0.433	27	1
7	109	35.9	1.127	43	1
13	152	26.8	0.731	43	1
1	122	35.1	0.692	30	1
2	102	45.5	0.127	23	1
1	125	33.3	0.962	28	1
1	196	36.5	0.875	29	1
5	189	31.2	0.583	29	1
3	173	38.4	2.137	25	1
5	116	32.3	0.660	35	1
8	105	43.3	0.239	45	1
3	193	34.9	0.241	25	1
5	136	35.0	0.286	35	1
1	172	42.4	0.702	28	1
3	173	35.7	0.258	22	1
4	144	38.5	0.554	37	1
3	129	26.4	0.219	28	1
8	151	42.9	0.516	36	1
1	181	34.1	0.328	38	1
1	95	35.0	0.233	43	1
0	189	34.3	0.435	41	1
0	180	59.4	2.420	25	1
0	104	33.6	0.510	22	1
3	158	35.5	0.344	35	1
0	135	42.3	0.365	24	1
4	125	28.9	1.144	45	1
12	84	29.7	0.297	46	1
3	163	31.6	0.268	28	1
9	145	30.3	0.771	53	1
3	128	32.4	0.549	27	1
10	90	34.9	0.825	56	1
8	186	34.5	0.423	37	1
5	187	43.6	1.034	53	1
3	176	33.3	1.154	52	1
11	111	46.8	0.925	45	1
1	181	40.0	1.258	22	1
3	174	32.9	0.593	36	1

(continuação)

(continua)

11	138	36.1	0.557	50	1
9	112	28.2	1.282	50	1
7	97	40.9	0.871	32	1
0	179	37.8	0.455	22	1
11	136	28.3	0.260	42	1
2	155	38.7	0.240	25	1
4	145	32.5	0.235	70	1
10	111	27.5	0.141	40	1
0	162	49.6	0.364	26	1
7	142	30.4	0.128	43	1
3	169	29.9	0.268	31	1
2	93	38.0	0.674	23	1
10	129	41.2	0.441	38	1
7	187	33.9	0.826	34	1
3	173	33.8	0.970	31	1
2	174	44.5	0.646	24	1
11	120	42.3	0.785	48	1
1	147	49.3	0.358	27	1
3	187	36.4	0.408	36	1
0	181	43.3	0.222	26	1
1	128	36.5	1.057	37	1
9	170	44.0	0.403	43	1

(continuação)

npreg: número de vezes que esteve grávida
glu: concentração de glucose no sangue
bmi p: concentração de insulina
ed: índice de massa muscular
age: idade.

F.3 Dados do Exemplo 6.1: HBK

x_1	x_2	x_3
10.1	19.6	28.3
9.5	20.5	28.9
10.7	20.2	31
9.9	21.5	31.7
10.3	21.1	31.1
10.8	20.4	29.2
10.5	20.9	29.1
9.9	19.6	28.8
9.7	20.7	31.0
9.3	19.7	30.3
11.0	24.0	35.0
12.0	23.0	37.0

(continua)

12.0	26.0	34.0
11.0	34.0	34.0
3.4	2.9	2.1
3.1	2.2	0.3
0.0	1.6	0.2
2.3	1.6	2.0
0.8	2.9	1.6
3.1	3.4	2.2
2.6	2.2	1.9
0.4	3.2	1.9
2.0	2.3	0.8
1.3	2.3	0.5
1.0	0.0	0.4
0.9	3.3	2.5
3.3	2.5	2.9
1.8	0.8	2.0
1.2	0.9	0.8
1.2	0.7	3.4
3.1	1.4	1.0
0.5	2.4	0.3
1.5	3.1	1.5
0.4	0.0	0.7
3.1	2.4	3.0
1.1	2.2	2.7
0.1	3.0	2.6
1.5	1.2	0.2
2.1	0.0	1.2
0.5	2.0	1.2
3.4	1.6	2.9
0.3	1.0	2.7
0.1	3.3	0.9
1.8	0.5	3.2
1.9	0.1	0.6
1.8	0.5	3.0
3.0	0.1	0.8
3.1	1.6	3.0
3.1	2.5	1.9
2.1	2.8	2.9
2.3	1.5	0.4
3.3	0.6	1.2
0.3	0.4	3.3
1.1	3.0	0.3
0.5	2.4	0.9
1.8	3.2	0.9
1.8	0.7	0.7
2.4	3.4	1.5
1.6	2.1	3.0
0.3	1.5	3.3
0.4	3.4	3.0
0.9	0.1	0.3
1.1	2.7	0.2

(continuação)

(continua)

(continuação)

2.8	3.0	2.9
2.0	0.7	2.7
0.2	1.8	0.8
1.6	2.0	1.2
0.1	0.0	1.1
2.0	0.6	0.3
1.0	2.2	2.9
2.2	2.5	2.3
0.6	2.0	1.5
0.3	1.7	2.2
0.0	2.2	1.6
0.3	0.4	2.6

F.4 Dados do Exemplo 6.2: HEMO

Normais

$\log_{10} x_1$	$\log_{10} x_2$
-0.0056	-0.1657
-0.1698	-0.1585
-0.3469	-0.1879
-0.0894	0.0064
-0.1679	0.0713
-0.0836	0.0106
-0.1979	-0.0005
-0.0762	0.0392
-0.1913	-0.2123
-0.1092	-0.119
-0.5268	-0.4773
-0.0842	0.0248
-0.0225	-0.058
0.0084	0.0782
-0.1827	-0.1138
0.1237	0.214
-0.4702	-0.3099
-0.1519	-0.0686
0.0006	-0.1153
-0.2015	-0.0498
-0.1932	-0.2293
0.1507	0.0933
-0.1259	-0.0669
-0.1551	-0.1232
-0.1952	-0.1007
0.0291	0.0442
-0.228	-0.171
-0.0997	-0.0733
-0.1972	-0.0607
-0.0867	-0.056
0.0492	-0.0268
0.0569	0.0334
-0.0132	-0.0655

(continua)

-0.1135	-0.0757
0.1303	0.0139
0.1038	0.1430
0.0718	0.0000
-0.1487	-0.2757
0.0569	-0.0705
0.1072	0.0644
0.2068	0.1139
0.1172	0.0334
-0.1249	-0.0457
-0.0457	-0.1938
-0.0915	-0.0506
-0.0809	-0.1191
0.0211	0.0569
-0.0809	-0.0409
0.1522	0.0681
0.0086	-0.0604
-0.2365	-0.2676
-0.0362	-0.1487
-0.3565	-0.4089
-0.2006	-0.2676
-0.1611	-0.2076
0.0374	-0.0132
0.0934	0.0863
0.2329	0.1959
0.0374	0.1238
0.0128	0.1105

(continuação)

Portadoras

$\log_{10} x_1$	$\log_{10} x_2$
-0.3478	0.1151
-0.3618	-0.2008
-0.4986	-0.086
-0.5015	-0.2984
-0.1326	0.0097
-0.6911	-0.339
-0.3608	0.1237
-0.4535	-0.1682
-0.3479	-0.1721
-0.3539	0.0722
-0.4719	-0.1079
-0.361	-0.0399
-0.3226	0.167
-0.4319	-0.0687
-0.2734	-0.002
-0.5573	0.0548
-0.3755	-0.1865
-0.495	-0.0153
-0.5107	-0.2483
-0.1652	0.2132

(continua)

(continuação)

-0.2447	-0.0407
-0.4232	-0.0998
-0.2375	0.2876
-0.2205	0.0046
-0.2154	-0.0219
-0.3447	0.0097
-0.254	-0.0573
-0.3778	-0.2682
-0.4046	-0.1162
-0.0639	0.1569
-0.3351	-0.1368
-0.0149	0.1539
-0.0312	0.14
-0.174	-0.0776
-0.1416	0.1642
-0.1508	0.1137
-0.0964	0.0531
-0.2642	0.0867
-0.0234	0.0804
-0.3352	0.0875
-0.1878	0.251
-0.1744	0.1892
-0.4055	-0.2418
-0.2444	0.1614
-0.4784	0.0282

F.5 Dados do Exemplo 6.4: *pendigits* (*pen-based recognition of handwritten digits*)

Disponível em *http://www.ics.uci.edu/~mlearn/MLSummary.html*

Referências bibliográficas

Alder, M. (1997). An introduction to pattern recognition: statistical, neural net and syntactic methods of getting robots to see and hear. Disponível em: *http://ciips.ee.uwa.edu.au/~mike/PatRec/*
ftp://ciips.ee.uwa.edu.au/pub/syntactic/book/

Anderson, J. (1972). A simple neural network generating an interactive memory. *Mathematical Biosciences*, **14**, 197-220.

Atkinson, A. (1994). Fast very robust methods for the detection of multiple outliers. *Journal of the American Statistical Association,* **89**, 1329-1339.

Banfield, J. e Raftery, A. (1992). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, **49**, 803-822.

Barnett, V. e Lewis, T. (1994). *Outliers in Statistical Data.* Wiley, Nova Iorque.

Batchelor, B. G. (1974). *Pratical Approach to Pattern Classification.* Plenum Press, Londres e Nova Iorque.

Becker, C. (2000). The size of the largest non identifiable outlier as a performance criterion for multivariate outlier identification: the case of high-dimensional data. *Proceedings in Computacional Statistics*. Bethlehem, J. e Van der Heijden, P. (editores). Physica-Verlag, Amesterdão, pp. 211-216.

Becker, C. e Gather, U. (1999). The masking breakdown point of multivariate outlier identification rules. *Journal of the American Statistical Association*, **94**, 947-955.

Becker, C., e Gather, U. (2001). The largest nonidentifiable outlier: a comparison of multivariate simultaneous outlier identification rules. *Computational Statistics and Data Analysis*, **36**, 119-127.

Beckman, R. e Cook, R. (1983). Outlier…s. *Technometrics*, **25**, 119-163.

Bezdek, J. C. (1992). *Fuzzy Models for Pattern Recognition- Methods That Search for Stuctures in Data*. IEEE Press, Piscataway.

Billor, N., Hadi, A. e Velleman, P. (2000). BACON: blocked adaptive computationally efficient outlier nominators. *Computational Statistics and Data Analysis*, **34**, 279-298.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford.

Bishop, C. (1998). *Neural Networks and Machine Learning.* Springer-Verlag, Berlim.

Bonferroni, C. (1936). Teoria statistica delle classi e calcolo delle probabilità, *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze*, **8**, 3-62.

Box, G. (1996). Scientific statistics, teaching, learning and the computer. *Proceedings in Computacional Statistics.* Prat, A. (editor). Physica-Verlag, Heidelberg, pp.3-10.

Braga, A. (2001). *Curvas ROC: Aspectos Funcionais e Aplicações*. Tese de Doutoramento. Universidade do Minho, Braga.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, **26**, 123-140.

Campbell, N. (1978). The influence function as an aid in outlier detection in multivariate analysis. *Applied Statistics*, **27**, 251-258.

Campbell, N. (1982). Robust procedures in multivariate analysis II: robust canonical variate analysis. *Applied Statistics*, **31**, 1-8.

Cheng, B. e Titterington, D. (1994). Neural networks: a review from a statistical perpective. *Statistical Science,* **9**, 2-54.

Chow, C. K. (1957). An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, **6**, 247-254.

Chow, C. K. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, **16**, 41-46.

Croux, C. e Haesbroeck, G. (1999). Influence function and efficiency of the minimum covariance determinant scatter matrix estimator. *Journal of Multivariate Analysis*, **71**, 161-190.

Davies, P. (1987). Asymptotic behaviour of S-estimates of multivariate location parameters and dispersion matrices. *Annals of Statistics*, **15**, 1269-1292.

Davies, P. (1992). The assimptotics of Rousseuw´s minimum volume ellipsoid. *Annals of Statistics*, **20**, 1828-1843.

Davies, L. e Gather, U. (1993). The identification of multiple outliers. *Journal of the American Statistical Association*, **88**, 782-801.

Devijver, P. A. e Kittler J. (1982). *Pattern Recognition: a Statistical Approach.* Prentice-Hall International, Londres.

Devroye, L., Györfi, L. e Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition.* Springer-Verlag, Nova Iorque.

Donoho, D. (1982). Breakdown properties of multivariate location estimators, Ph.D. Qualifying Paper, Harvard University, Boston.

Duda, R. O. e Hart, P. E. (1973). *Pattern Classification and Scene Analysis*. Wiley, Nova Iorque.

Duda, R., Hart, P. e Stork, D. (2001). *Pattern Classification.* Wiley, Nova Iorque.

Duin, B. (2000). *Pattern recognition research area.*
Disponível em: *http://www.ph.tn.tudelft.nl/PRInfo/prarea.html*.

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, **7**, 1-26.

Efron, B. (1982). *The Jackknife, the Bootstrap and other Resampling Plans*. SIAM, Filadélfia.

Efron, B. (1983). Estimating the error rate of a predition rule: improvement on cross-validation. *Journal of the American Statistical Association*, **78**, 316-331.

Fisher, R. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, **7**, 179-188.

Fogia, P., Sansone, C., Tortorella, F. e Vento, M. (1999). Multiclassification: reject criteria for the Bayesian combiner. *Journal of the Pattern Recognition Society*, **32**, 1435-1447.

Forgy, E. (1965). Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. *Biometrics*, **21**, 768.

Fraley, C e Raftery, A. (1998). How many clusters? Which clustering method? - answers via model-based cluster analysis**.** *Computer Journal*, **41**, 578-588.

Fraley, C e Raftery, A. (1999). MCLUST: software for model-based clustering and discriminant analysis. *Journal of Classification*, **16**, 297-306.

Fraley, C e Raftery, A. (2002). Software for model-based clustering, discriminant analysis and density estimation. *Technical Report* 415. Department of Statistics, University of Washington.

Freund, Y. (1995). Boosting a weak learning algorithm by majority. *Information and Computation*, **121**, 256-285.

Freund, Y. e Schapire, R. (1997). A decision-theoretic generalization of online learning and an application of boosting. *Journal of Computer and System Sciences*, **55**, 119-139.

Friedman, J (1997). Data mining and statistics: what´s the connection? *Technical Report.* Disponível em: *http://www-stat.stanford.edu/~jhf*.

Friedman, M. e Kandel, A. (1999). *Introduction to Pattern Recognition: Statistical, Structural Neural and Fuzzy Logic Approaches*. Imperial College Press. Londres.

Fukunaga, K. (1972). *Introduction to Statistical Pattern Recognition.* Academic Press, Nova Iorque e Londres.

Gather, U. e Becker, C. (1997). Outlier identification and robust methods. *Handbook of Statistics*, **15**, *Robust Inference*, 123-143. Maddala, G. e Rao, C. (editores.). Elsevier, Amesterdão, pp. 123-143

Gentle, J. (1998). *Random Number Generation and Monte Carlo Methods*. Spinger-Verlag, Nova Iorque.

Gnanadesikan, R. e Kettenring, J. (1972). Robust estimates, residuals and outlier detection with multiresponse data. *Biometrics*, **28**, 81-124.

Golfarelli, M., Maio, D. e Maltoni, D. (1997). On the error-reject tradeoff in biometric verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 786-796.

Ha, T. (1995). An optimum decision rule for pattern recognition. *Technical Report*, IAM-95-009.

Ha, T. (1996a). On functional relation between recognition error and class-selective reject. *Technical Report*, IAM-96-007.

Ha, T. (1996b). An experimental study of the optimal class-selective rejection rule. *Technical Report*, IAM-96-008.

Ha, T. (1996c). Application of the optimal class-selective rejection rule to the detection of abnormalities in OCR databases. *Technical Report*, IAM-96-009.

Hadi, A. (1992). Identifying multiple outliers in multivariate data. *Journal of the Royal Statistical Society*. Series (B), **54**, 761-771.

Hadi, A. (1994). A modification of a method for the detection of outliers in multivariate samples. *Journal of the Royal Statistical Society*. Series (B), **56**, 393- 396.

Halpern, E., Albert, M., Krieger, A., Metz, C. e Maidment, A. (1996). Comparison of receiver operating characteristic curve on the basis of optimal operating points. *Statistics for Radiologists*, **3**, 245-253.

Hampel, F. (2000). Robust inference. *Research Report,* **93**. ETH, Zurique.

Hampel, F. (2001). Robust statistics: a brief introduction and overview. *Research Report,* **94**. ETH, Zurique.

Hampel, F. (2002). Some thoughts about classification. *Research Report,* **102**. ETH, Zurique.

Hampel, F., Ronchetti, E., Rousseeuw, P. e Stahel, W. (1986). *Robust Statistics: The Approach Based on Influence Functions*. Wiley, Nova Iorque.

Hand, D. J. (1996). Classification and computers: shifting the focus. *Proceedings in Computational Statistics*. Prat, A. (editor). Physica-Verlag, Heidelberg. pp.77-88.

Hand, D. J. (1997). *Construction and Assessment of Classification Rules*. Wiley, Nova Iorque.

Hand, D. J. (2000). Methodological issues in data mining. *COMPSTAT 2000. Proceedings in Computacional Statistics*. Bethlehem, J. e van der Heijden, P. (editores). Physica-Verlag. Amesterdão, pp.77-85.

Hardin, J. e Rocke, D. (2004). Outlier detection in multiple cluster setting using the minimum covariance determinant estimator. *Computational Statistics and Data Analysis*, **44**, 625-638.

Hardin, J. e Rocke, D. (1999). The distribution of robust distances. Disponível em: *http://handel.cipic.ucdavis.edu/~dmrocke/Robdist5.pdf*.

Hardin, J., Rocke, D. e Woodruff, D. (2000). Robust model-based clustering of genes in microarray data: are there gene clusters? Disponível em: *http://www.camda.duke.edu/CAMDA00/abstracts.asp*.

Hastie, T., Buja, A. e Tibshirani, R. (1995). Penalized discriminant analysis. *The Annals of Statistics*, **23**, 73-102.

Hastie, T., Tibshirani, R. e Buja, A. (1994). Flexible discriminant analysis by optimal scoring. *Journal of the American Statistical Association*, **89**, 1255-1270.

Hastie, T., Tibshirani, R. e Friedman, J. (2001). *The Elements of Statistical Learning: Data Mining, Inference and Prediction.* Springer-Verlag, Nova Iorque.

Hawkins, D. (1980). *Identification of Outliers*. Chapman and Hall, Londres.

Hawkins, D., Bradu, D. e Kass, G. (1984). Location of several outliers in multiple regression data using elemental sets. *Technometrics*, **26**, 197-208.

Hebb, D. (1949). *The Organization of Behavior*. Wiley, Nova Iorque.

Hébrail, G. (2000). Pratical data mining in a large utility company. *Proceedings in Computacional Statistics*. Bethlehem, J. e van der Heijden, P. (editores). Physica-Verlag. Heidelberg, pp.87-95.

Hermans, J. e Habbema, J. D. (1975). Comparison of five methods to estimate posterior probabilities. *EDV in Medizin und Biologie*, **6**, 14-19.

Jacobs, R., Jordan, A., Nowlan, I. e Hinton, G. (1991). Adaptive mixtures of local experts. *Neural Computation*, **3**, 79-87

Jain, A. e Chandrasekaran, B. (1982). Dimensionality and sample size considerations in pattern recognition practice. *Handbook of Statistics*, Krishnainh, P. e Kanal, L. (editores), pp. 835-855.

Jain, A., Duin, R. e Mao, J. (2000). Statistical Pattern Recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 4-37.

James, W. (1890). *Principles of Psychology*. Henry Holt & Co, Nova Iorque.

Jiang, M., Tseng, S. e Su, C. (2001). Two-phase clustering process for outliers detection. *Pattern Recognition Letters*, **22**, 691-700.

Johnson, R. e Wichern, D. (1992). *Applied Multivariate Statistical Analysis*. 3ª Edição. Prentice Hall. Englewood Cliffs.

Jordan, M. e Jacobs, R. (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, **6**, 181-214.

Kaufman, L. e Rousseuw, P. (1990). *Finding Groups in Data: an Introduction to Cluster Analysis.* Wiley, Nova Iorque.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, **43**, 59-69.

Kosinski, A. (1999). A procedure for the detection of multivariate outliers. *Computational Statistics and Data Analysis*, **29**, 145-161.

Kovesi, B., Boucher, J. e Saoudi, S. (2001). Stochastic K-means algorithm for vector quantization. *Pattern Recognition Letters*, **22**, 603-610.

Leondes, C. T. (1998). *Image Processing and Pattern Recognition.* Academic Press, Nova Iorque.

Lowe, D. (1995). Radial basis function networks. *The Handbook of Brain Theory and Neural Networks*. Arbib (editor). MIT Press, Cambridge, pp. 779- 782.

Macieira-Coelho, E., Oliveira, M. e Amaral-Turkman, M. (1990). Diagnóstico de cardiopatia isquémica no doente ambulatório: análise multivariada de dados clínicos e electrocardiográficos. *Acta Médica Portuguesa*, **3**, 277-282.

Maronna, R. (1976). Robust M-estimators of multivariate location and scatter. *The Annals of Statistics*, **4**, 51-56.

Maronna, R. e Yohai, V. (1995). The behavior of the Stahel-Donoho robust multivariate estimator. *Journal of the American Statistical Association*, **90**, 330-341.

Maronna, R. e Zamar, R. (2002). Robust estimates of location and dispersion for high-dimensional data sets. *Technometrics*, **44**, 307-317.

Marques, J. S. (1999). *Reconhecimento de Padrões- Métodos Estatísticos e Neuronais*. IST Press, Lisboa.

McCulloch, W. e Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, **5**, 115-133.

McLachlan, G. J. (1992). *Discriminant Analysis and Statistical Pattern Recognition.* Wiley, Nova Iorque.

McLachlan, G. J. e Basford (1988). *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, Nova Iorque.

McLachlan, G. J. e Peel, D. (2000). *Finite Mixture Models.* Wiley, Nova Iorque.

Michie, D., Spiegelhalter, D. e Taylor, C. (1994). *Machine Learning, Neural and Statistical Classification.* Ellis Horwood Limited, Nova Iorque.

Minsky, M. e Papert (1969). *Perceptrons*. MIT Press, Cambridge.

Neter, J., Kutner, M., Nachtsheim, C. e Wasserman, W. (1996). *Applied Linear Statistical Models*. McGraw-Hill, U.S.A.

Olmsted, D. (1998). History and principles of neural networks. Disponível em: *http://neurocomputing.org/history.htm*.

Pagnotta, S. (2003). An improvement of the FAST-MCD algorithm. (submetido a publicação).

Pavel, M. (1993). *Fundamentals of Pattern Recognition.* Marcel Dekker, Nova Iorque.

Pavlidis, T. (1977). *Structural Pattern Recognition.* Springer-Verlag, Berlim.

Peirce, B. (1852). Criterion for the rejection of doubtful observations. *Astronomy Journal,* **2**, 161-163.

Peña, D. e Prieto, J. (2001). Multivariate outlier detection and robust covariance matrix estimation. *Technometrics*, **43**, 286-303.

Pires, A. M. (1995). *Análise Discriminante. Novos Métodos Robustos de Estimação.* Tese de Doutoramento. Instituto Superior Técnico, Universidade Técnica de Lisboa, Lisboa.

Pissarra, M. e Reis, A. (1996). *Rumos de Filosofia-Lógica e Argumentação*. Volume 1. Edições Rumo, Lisboa.

Powell, M. (1987). Radial basis functions for multivariate interpolation: a review. Mason e Cox (editores). *Algorithms for Approximation*, 143-167. Oxford: Clarendon Presss.

Raudys, S. (1998). Structures of the covariance matrices in the classifier design. *Advances in Pattern Recognition*, *SSPR 1998 e SPR 1998. Lectures Notes in Computer Science,* Vol. 1451. Amin, A., Dori, D., Pavel, P. E Frieeman, H.(editores). Springer-Verlag, Heidelberg. pp. 583-92.

Raudys, S. (2001). *Statistical and Neural Classifiers: an Integrated Approach to Design*. Springer-Verlag, Londres.

Rencher, A. (1998). *Multivariate Statistical Inference and Applications*. Wiley, Nova Iorque.

Ripley, B.D. (1996). *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge.

Rocke, D. (1998). A perspective on statistical tools for data mining applications. *Proceedings of the Second International Conference on Practical Application on Knowledge Discovery and Data Mining*. pp. 313-318.

Rocke, D. e Woodruff, D. (1996). Identification of outliers in multivariate data. *Journal of the American Statistical Society*, **91**, 1047-1061.

Rocke, D. e Woodruff, D. (1998). Some statistical tools for data mining applications. Disponível em:
http://handel.cipic.ucdavis.edu/~dmrocke/preprints.html

Rocke, D. e Woodruff, D. (1999). A synthesis of outlier detection and cluster identification. Disponível em: *http://handel.cipic.ucdavis.edu/~dmrocke/Synth5.pdf*

Rosado, F. (1984). *Existência e Detecção de "Outliers". Uma Abordagem Metodológica*. Tese de Doutoramento. Faculdade de Ciências, Universidade de Lisboa, Lisboa.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, **65**, 386-408.

Rousseeuw, P. (1985). Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, Volume B, W. Grossman, G. Pflug, I. Vincze and Werz, W. (editores). Reidel, Dordrecht, pp. 283-297.

Rousseeuw, P. e Leroy, A. (1987). *Robust Regression and Outlier Detection*. Wiley, Nova Iorque.

Rousseeuw, P., Kaufman, L. e Trauwaert, E. (1996). Fuzzy clustering using scatter matrices. *Computational Statistics and Data analysis*, **23**, 135-151.

Rousseeuw, P. e van Driessen, K. (1999). A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, **41**, 212-223

Rousseeuw, P. e van Zomeren, B. (1990). Unmasking multivariate outliers and leverage points. *Journal of the American Statistical Association*, **85**, 633-639.

Rumelhart, D., Durbin, R., Golden, R. e Chauvin, Y. (1995). Backpropagation: the basic theory. Chuvin e Rumelhart (editores), *Backpropagation: Theory, Architectures and Applications*, pp. 1-34. Hillsdale, NJ: Lawrence Erlbaum.

Sain, S., Woodward, W. e Fisk, M. (1999). Outlier detection from a mixture distribution when training data are unlabeled. *Bulletin of the Seismological Society of America*, **89**, 294-304.

Sakamoto, Y., Ishiguro, M., and Kitagawa G. (1986). *Akaike Information Criterion Statistics.* D. Reidel, Tóquio.

Santos-Pereira, C. e Pires, A. M. (2001). Classificação supervisionada com dúvidas: compromisso erro/rejeição. *A Estatística em Movimento: Actas do VIII Congresso Anual da Sociedade Portuguesa de Estatística*. M. Neves *et al.* (editores). Edições SPE, Lisboa, pp. 313-322.

Santos-Pereira, C. e Pires, A. M. (2002). Detection of outliers in multivariate data: a method based on clustering and robust estimators. *Proceedings in Computational Statistics*. Hardle, W. e Ronz, B.(editores). Phisica-Verlag, Heidelberg, pp.291-296.

Santos-Pereira, C. e Pires, A. M. (2004a). Método de classificação com rejeição por indecisão e observações atípicas. (aceite para publicação em *Actas do XI Congresso Anual da Sociedade Portuguesa de Estatística*).

Santos-Pereira, C. e Pires, A. M. (2004b). On optimal reject rules and ROC curves. (condicionalmente aceite para publicação em *Pattern Recognition Letters*).

Schalkoff, R. (1992). *Pattern Recognition: Statistical, Structural and Neural Approaches.* Wiley, Nova Iorque.

Schapire, R. (1990). The strength of weak learnability. *Machine Learning*, **5**, 197-227.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, **6**, 461-464.

Shürmann, J. (1996). *Pattern Classification: a Unified View of Statistical and Neural Approaches.* Wiley, Nova Iorque.

Skurichina, M. e Duin, R. (2000). The role of combining rules in bagging and boosting. *Advances in Pattern Recognition.* Vol. 1876. Ferri, F., Iñesta, J., Amin, A., e Pudil, P.(editores). Springer-Verlag, Heidelberg, pp. 631-640.

Silva, A. e Campilho, A. (2000). Análise experimental de métodos de combinação de classificadores. *V Ibero-American Symposium on Pattern Recognition*, Lisboa.

Silva, M. (2000). *Estimação da Propensão à Compra de um Produto Financeiro.* Trabalho Final de Curso. INESC, DM-ISL, Universidade Técnica de Lisboa, Lisboa.

Smith, L. (1996). An introduction to neural networks. Disponível em: *http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html*

Soares, R. (1997). *Métodos não paramétricos em Análise Discriminante.* Tese de Mestrado. Universidade Técnica de Lisboa. ISEG.

Stahel, W. (1981). Breakdown of covariance estimators. *Research Report* **31**, ETH, Zurique.

Suen, C., Nadal, C., Legault, R., Mai, T. E Lam, L. (1992). Computer recognition of unconstrained handwritten numerals. *Proceedings of the IEEE*, **80**, 1162-1180.

Tibshirani, R. e Knight, K. (1997). Model search and inference by bootstrap "bumping". *Technical Papers & Reports on WWW/FTP sites.* Disponível em: *http://zernike.uwinnipeg.ca/techrep.html*

Tibshirani, R., Walther, G. e Hastie, T. (2000). Estimating the number of clusters in a dataset via the Gap statistic. Disponível em: *http://www-stat.stanford.edu/~tibs/ftp/gap.ps*

Tortorella, F. (2000). An optimal reject rule for binary classifiers. *Advances in Pattern Recognition: Joint IAPR International Workshops, SSPR 2000 and SPR 2000. Lectures Notes in Computer Science,* Vol. 1876. Ferri, F., Iñesta, J., Amin, A., e Pudil, P.(editores). Springer-Verlag, Heidelberg, pp. 611-620.

Turney, P. (2000). Types of cost in inductive concept learning. *Paper presented on ICML-2000 Workshop on COST-SENSITIVE LEARNING*. Disponível em: *http://www.cs.orst.edu/~margindr/Workshops/CSL-ICML2k/worknotes.html*

Van Trees, H. (1968). *Detection, Estimation, and Modulation Theory*. Wiley, Nova Iorque.

Venables, W. e Ripley, B. (1999). *Modern Applied Statistics with S-Plus*. Springer-Verlag, Nova Iorque.

Wan, E. (1990). Neural network classification: a Bayesian interpretation. *IEEE Transactions on Neural Networks,***1**, 303-305.

Wang, S., Woodward, W., Wiechecki, S. e Sain, S. ((1997). A new test for outlier detection from a multivariate mixture distribution. *Journal of Computational and Graphical Statistics*, **6**, 285-299.

Watanabe, S. (1972). *Frontiers of Pattern Recognition*. Academic Press, Nova Iorque.

Webb, A. (1999). *Statistical Pattern Recognition*. Arnold Publishers, Londres.

Widrow, B. e Hoff, M. (1960). Adaptive switching circuits. IRE WESCON *Convention Record*, **4**, 96-104.

Wilks, S. (1963). Multivariate statistical outliers. *Sankhya*, Serie A, 25, 407-426.

Wolpert, D. (1992). Stacked generalization. *Neural Networks*, **5**, 241-259.

Woods, K., Kegelmeyer, W. e Bowyer, K. (1997). Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence,***19**, 405-410.

Yohai, V. e Zamar, R. (1988). High breakdown point and high efficiency estimates of regression by means of the minimization of an efficient scale. *Journal of the American Statistical Association*, **83**, 406-413.

"A essência da determinação consiste em persistir por mais um momento quando o objectivo parece inalcançável."

Thomas Edison