Modelos de Previsão de Fenómenos Raros

Rita Paula Almeida Ribeiro

Tese submetida à Faculdade de Economia da Universidade do Porto para obtenção do grau de Mestre em Inteligência Artificial e Computação

Orientador: Professor Doutor Luís Torgo

Faculdade de Economia da Universidade do Porto
2003

Dedicado à minha família e ao Jorge.

Nota Biográfica

Rita Paula Almeida Ribeiro, nasceu a 30 de Setembro de 1977 na freguesia de Santo Ildefonso, na cidade do Porto.

Concluiu, em Maio de 2001, a licenciatura em Ciência de Computadores - Ramo Educacional na Faculdade de Ciências da Universidade do Porto. No âmbito do estágio pedagógico leccionou disciplinas de Introdução às Tecnologias de Informação na Escola Secundária José Régio, em Vila do Conde.

O interesse pelas noções de Inteligência Artificial apreendidas no curso, levaram-na a deixar o ensino e a ingressar no Mestrado de Inteligência Artificial e Computação em Setembro de 2001, altura em que integrou o Núcleo de Inteligência Artificial e Análise de Dados (NIAAD) da Universidade do Porto.

O trabalho desenvolvido neste grupo proporcionou-lhe um aprofundamento de conhecimentos no âmbito da Inteligência Artificial e um maior interesse pela área de *data mining*. Foi com base nesse crescente envolvimento que optou pelo estudo e desenvolvimento do tema desta tese. Decorrente da sua execução teve oportunidade de participar na 14^{th} *European Conference on Machine Learning* - 7^{th} *European Conference on Principles and Practice of Knowledge Discovery in Databases* (ECML/PKDD'03), em Cavtat-Dubrovnik, Croácia de 22 a 26 de Setembro de 2003, bem como publicar os seguintes artigos em conferências internacionais com revisão:

- Torgo,L., e Ribeiro,R.: **Predicting Outliers**.

  Proceedings of Principles of Data Mining and Knowledge Discovery (PKDD'03). Lavrac,N. et al. (eds.). LNAI 2838, (c) Springer-Verlag.

- Ribeiro,R. e Torgo,L.: **Predicting Harmful Algae Blooms**.

  A aparecer em *Proceedings of Portuguese Artificial Intelligence Conference (EPIA'03). (c) Springer-Verlag.*

Agradecimentos

Várias foram as pessoas que, a diferentes níveis, contribuíram para a produção desta tese.

Em primeiro lugar, gostaria de deixar aqui um agradecimento especial ao meu orientador Luís Torgo, não só pelos comentários e sugestões valiosas mas também pelo acompanhamento e disponibilidade constante ao longo de todo este trabalho.

Gostaria também de agradecer às pessoas que integram o grupo do NIAAD pelas boas condições de trabalho oferecidas, pelo apoio e interesse demonstrado pelo meu trabalho e pela oportunidade de participação em seminários e conferências internacionais.

Não poderia deixar de expressar o meu agradecimento à minha família e em especial aos meus pais, Mário e Virgínia, pela formação que me deram e continuam a dar como pessoa e pelo incentivo prestado em todas as minhas decisões.

Gostaria de deixar aqui uma palavra a todos os amigos que me souberam dar o apoio moral necessário e compreender a minha menor disponibilidade de tempo, não deixando, nunca, que isso desvanecesse a amizade.

Por fim, não posso deixar de agradecer ao Jorge, por todo o afecto, compreensão, dedicação e apoio prestado nas horas mais difíceis. A ele, o meu mais sincero obrigada.

Resumo

Em muitas das aplicações de *data mining* o interesse principal reside na previsão de situações comuns de modo a suportar o processo de tomada de decisões. Contudo, noutras aplicações o maior interesse reside sobre a previsão de situações cuja ocorrência é rara e, por isso mesmo, importante no sentido de accionar alarmes. Em alguns casos, a situação de "alarme" é identificada por valores raros e extremos. Este é o caso do problema que serviu de motivação à elaboração desta tese: previsão do aparecimento de *blooms* de algas nocivas em rios de onde se extrai a água potável. Estes *blooms* são raros e caracterizam-se por valores extremamente elevados de frequência de ocorrência de comunidades de algas nocivas que degradam a qualidade da água, tornando-a imprópria para consumo.

Nesta tese, descrevemos um método de *data mining* que pretende dar uma resposta mais eficaz a problemas de previsão de fenómenos raros de valor extremo, quando comparado com os métodos tradicionais de aprendizagem. Mais eficaz no sentido de cobrir o maior número possível de situações raras, mesmo que à custa de alguns falsos alarmes, e tentando-se uma maior precisão na previsão do seu valor numérico.

O paradigma base escolhido para a concepção deste novo método foi a árvore de regressão pela sua eficiência, interpretabilidade e flexibilidade. Pretendemos não só prever mas também entender sob que circunstâncias essas situações ocorrem. Propomos um novo critério de *split* para a construção de árvores de regressão, vocacionado para a satisfação dos nossos objectivos e "preenchendo" as medidas de avaliação adequadas a este tipo de aplicações.

Experiências realizadas em várias aplicações reais, permitem-nos constatar que os nossos modelos, quando comparados com aqueles obtidos por árvores de regressão *standard*, como por exemplo o CART (Breiman et al., 1984), são mais eficazes naquilo que é considerado mais relevante neste tipo de aplicações.

Abstract

In many data mining applications, the prediction of common situations as a way to support the process of decision making assumes particular importance. Nevertheless, in other applications the main significance rests on the prediction of rare situations, to allow timely firing of alarms. In some situations, the alarm status is associated to rare and extreme values. This is the case of the problem that motivated the elaboration of this thesis: the prediction of harmful algae blooms in rivers where drinkable water is pulled out. These blooms are rare and are characterized by extremely high frequency values of harmful algae communities that diminish the water quality, making it unsuitable for consumption.

In this thesis, we describe a data mining method that intends to give a more effective response to problems related to the prediction of rare and extreme values, when compared to the traditional learning methods. More effective in the sense of covering the highest number of extreme and rare situations, accepting possible false alarms and trying to be accurate in the prediction of their numeric value.

The base paradigm that was chosen for the conception of this new method was the regression tree, for its efficiency, interpretability and flexibility. We intend not only to predict but also to understand the circumstances under which these situations occur. We propose a new split criterion for regression trees that is suitable for the achievement of our objectives, "filling"the adequate evaluation measures for this kind of applications.

Experiments that were carried out in several real applications allow us to claim that our models, when compared to those that are obtained by standard regression trees, such as CART (Breiman et al., 1984), is more effective in terms of the statistics that are considered relevant in this kind of applications.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

A tarefa de conceber e implementar algoritmos capazes de simular aquele que seria o raciocínio humano perante a necessidade de resolução de um determinado problema, tem vindo a apresentar-se cada vez mais desafiadora.

A Inteligência Artificial assume, exactamente, esse objectivo: dotar máquinas de capacidades próximas às da inteligência humana, em particular aquelas que se referem ao raciocínio lógico, com vista à resolução de problemas. A ideia consiste em construir uma máquina capaz de executar tarefas requerendo o reconhecimento de padrões ou a aprendizagem de determinados conceitos ou até mesmo o entendimento de uma linguagem, entre outras capacidades. No entanto, todas estas capacidades ainda se encontram longe das capacidades da inteligência humana, sendo comparativamente muito mais limitadas.

É recorrendo a diversas técnicas de Inteligência Artificial que tentamos a resolução de problemas que, à partida, se mostram demasiado complexos. Entre um leque variado de problemas, existem alguns cuja resolução pode ser de extrema importância e de grande utilidade para o Homem.

O carácter raro da ocorrência de certos fenómenos torna escassa a informação disponível acerca dos mesmos, colocando dificuldades adicionais à aprendizagem da sua previsão. Neste contexto, a sua previsão pode constituir um problema especialmente interessante do ponto de vista da aquisição de novo conhecimento. Dependendo da natureza do fenómeno, a sua previsão poderá ser, de facto, importante. Em muitos domínios, o carácter raro do fenómeno a prever é assinalado por valores extremos de uma determinada variável.

É precisamente sobre problemas de previsão de fenómenos raros de valor extremo, que incidimos a nossa atenção para a elaboração desta tese.

Neste capítulo, começa-se por referir a motivação geral, do ponto de vista da Inteligência Artificial, subjacente a esta tese. Posto isto é exposto, de uma forma mais concreta, o tipo de problemas que serão aqui alvo de estudo e os objectivos por nós estabelecidos. Por fim, é indicada a forma como está organizada esta tese.

1.1 Motivação

Actualmente, com o avanço tecnológico, facilmente nos deparamos com o registo de dados em larga escala. Registos que decorrem de tarefas rotineiras do dia-a-dia e nas mais variadas áreas. Podemos falar do registo de vendas de um hipermercado, registos clínicos, registos de cotações num mercado bolsista, registos relativos ao funcionamento interno de empresas, etc. Sobre essas bases de dados, podem ser colocados os mais variados problemas que exigem algum processamento para obtenção de informação de extrema utilidade.

Dada a cada vez maior quantidade de dados disponível, torna-se humanamente impossível fazer uma análise a longo prazo dos dados para a tomada de decisões “inteligentes”. As correlações existentes nos dados devem-se aos mais variados factores e facilmente o seu número se torna exponencial. É no contexto da necessidade de resolução de problemas a partir de um grande conjunto de dados que surge o processo de *data mining*.

A tarefa do *data mining* compreende a análise de dados, a descoberta de padrões, a aprendizagem de conceitos, a previsão de valores futuros, etc. A aquisição de um novo conceito é feita sob uma perspectiva interdisciplinar que abrange desde funcionalidades de sistemas de base de dados, até métodos de Inteligência Artificial, em particular, métodos de aprendizagem automática, passando pela análise estatística. A ideia consiste em extrair conhecimento implícito nos dados como uma informação potencialmente útil. Alguns dos padrões encontrados serão banais e comuns; outros poderão ser surpreendentes ou anormais.

Este processo, tal como referido por Witten e Frank (1999), é utilizado em tarefas de caracterização de conceitos, classificação, regressão, aprendizagem de regras de associação, aprendizagem de padrões sequenciais, *clustering*, etc.

1.2 Problema e Objectivos

Os problemas de previsão de fenómenos raros de valor extremo, alvo de estudo nesta tese, enquadram-se no conceito de *data mining*.

Um problema de previsão coloca-se sempre que pretendamos prever o futuro com base em factos passados. No dia-a-dia, deparamo-nos com este tipo de problemas, nas mais variadas áreas. A título de exemplo, podemos falar da já tão banalizada previsão meteorológica, da previsão de sismos, da previsão do melhor plano de *marketing* para a venda de um produto, ou até mesmo do acompanhamento do mercado de acções.

Em grande parte das aplicações, o que se pretende é a previsão de situações comuns no respectivo domínio. Situações cuja previsão pode trazer, de facto, uma mais valia dado o seu carácter frequente. Estas previsões podem vir a traduzir-se num apoio sólido para a tomada de decisões. A necessidade de venda de um produto a um grupo específico de potenciais consumidores é um exemplo disso. É baseado no *feedback* mais frequentemente obtido em campanhas anteriores, que se faz uma previsão no sentido de traçar o "melhor", o mais eficaz, plano de *marketing* para a comercialização de um produto num grupo específico de consumidores.

Contudo, também a resolução de problemas de previsão de situações raras, fora do comum, pode assumir um papel relevante. Esta é uma classe de problemas relativamente pouco explorada, mas de vital importância em muitos domínios concretos. A título de exemplo podemos referir o fenómeno sísmico que, apesar da sua rara ocorrência, tem por vezes consequências catastróficas. A sua previsão será, por isso, de grande utilidade no sentido de permitir o desencadeamento de mecanismos de prevenção.

No nosso estudo, focamo-nos sobre a previsão de fenómenos raros que assumem valores exageradamente altos ou exageradamente baixos em relação aos valores ditos normais para o domínio em questão. Interessa-nos a concepção de modelos, capazes de obter a descrição desses fenómenos e, com base nela, prever a sua ocorrência de forma tão exacta, quanto possível.

1.3 Organização da tese

Nesta tese começamos por definir de uma forma informal, no Capítulo 2, o problema de previsão de fenómenos raros, falando um pouco do tipo de desafios que a resolução deste tipo de problemas nos coloca logo à partida e dos conceitos que esta envolve.

No Capítulo 3 é feito um sumário de algumas das abordagens traçadas para a resolução de problemas, de alguma forma, relacionados com o problema de previsão de fenómenos raros.

É no Capítulo 4 que expomos a abordagem por nós seguida na resolução de problemas de previsão de fenómenos raros de valor extremo, mencionando o problema concreto que motivou a concepção de um novo método. Falamos dos pressupostos de que partimos, dos critérios de avaliação utilizados e da descrição do método concebido.

No Capítulo 5 é feita uma avaliação de resultados obtidos pelo nosso método, quando aplicado sobre um variado número de domínios. São levantadas questões quanto aos resultados e sensibilidade do método face a alguns parâmetros.

O último Capítulo contém as conclusões do nosso trabalho e possibilidades para o prosseguimento deste estudo.

Capítulo 2

O Problema de Previsão de Fenómenos Raros

Os problemas de previsão constituem uma classe de problemas alvo de estudo na Inteligência Artificial. A sua resolução é de reconhecida importância qualquer que seja a natureza do fenómeno a prever. Existem, contudo, fenómenos cuja previsão assume um papel fulcral seja na economia, na saúde, na sociedade, na política, etc. Estamos a falar de fenómenos que embora raros, de modo algum podem ser negligenciados.

Neste capítulo, começa-se por dar uma definição um pouco abstracta do problema de previsão de fenómenos raros. A previsão como problema e o conceito de fenómeno raro são dois aspectos abordados. Menciona-se também os desafios que a resolução deste tipo de problemas nos coloca. Posto isto, numa segunda secção, são dados como exemplo alguns casos práticos cujo domínio evidencia a importância atribuída a fenómenos raros, porém relevantes. São ainda expostas aplicações concretas nas quais a resolução deste tipo de problemas pode ser de extrema importância.

2.1 Definição do problema

Normalmente, a previsão surge associada a problemas de vital importância, motivada pela necessidade e pelos efeitos benéficos que nos pode trazer. Por exemplo, a capacidade de previsão meteorológica, impossível há alguns anos atrás, permite-nos adoptar medidas de defesa e de prevenção capazes de evitar tragédias e antecipar soluções.

Em muitos dos casos é conhecida a descrição completa do fenómeno que pretendemos prever. Nesses casos, a previsão é governada por regras absolutamente determinísticas usadas para simulações do fenómeno.

As marés constituem um fenómeno resultante da atracção gravitacional exercida pela Lua e pelo Sol sobre a Terra. O seu carácter sistemático torna-o num fenómeno aparentemente fácil de prever, sobretudo para pessoas que vivam na proximidade do mar. Mesmo desconhecendo as razões científicas para este fenómeno, sabe-se que durante aproximadamente um dia o nível das águas atinge, por duas vezes, uma altura máxima na preia-mar e, por outras duas vezes, uma altura mínima na baixa-mar. Para além disso, é também conhecida alguma relação existente entre as marés e as diferentes fases da Lua.

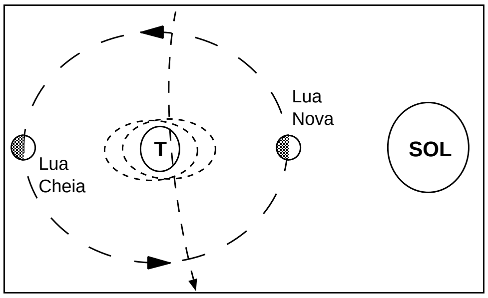
Através de Isaac Newton (1643 - 1727) as marés tornaram-se num fenómeno cuja descrição é perfeitamente conhecida e explicada pela física.

Tanto o Sol como a Lua exercem uma força de atracção gravitacional sobre a Terra. Dada a maior distância a que nos encontramos do Sol, a intensidade da força de atracção gravitacional exercida por este é menor do que aquela exercida entre a Lua e a Terra. A acção gravitacional da Lua é duas vezes e meia mais intensa que a do Sol porque, apesar de possuir massa muito menor, ela está muitíssimo mais próxima. São estas forças que produzem as marés.

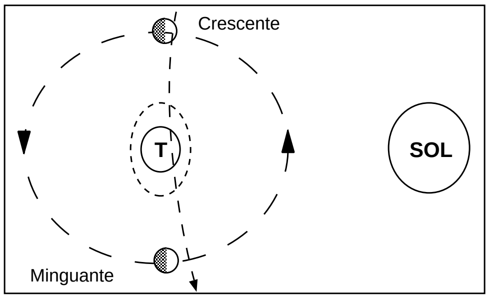
A força de maré é directamente proporcional à massa do corpo que provoca a maré e inversamente proporcional ao cubo da distância entre o corpo que provoca a maré e o que sofre a maré.

Pela atracção gravitacional exercida, a água irá concentrar-se no ponto mais próximo da Lua. Para além da Lua girar em torno da Terra, ambas giram em torno de um centro de massa comum, criando uma força centrífuga que age exactamente no lado oposto da Terra, mas na mesma direcção da Lua. É nesses dois pontos que o nível das águas sobe. À medida que esses pontos se afastam da Lua, as águas descem. Este fenómeno de subida e descida periódicas da água é denominado de maré. As Figuras 2.1 (a) e (b) esquematizam o efeito das fases da Lua sobre as marés.

Como sabemos, a Terra realiza uma volta em torno de si mesma a cada 24 horas. Ao longo de um dia solar a Lua também se move. Por isso, para que um ponto do planeta volte à sua posição inicial relativamente à Lua são necessárias, aproximadamente, 24 horas e 50



(a) marés vivas



(b) marés mortas

Figura 2.1: Efeito do Sol e da Lua sobre as marés.

minutos. Este é o tempo correspondente ao ciclo das marés. Se num dado momento um certo ponto da Terra está exactamente debaixo da Lua, ele terá maré alta. Seis horas mais tarde, a rotação da Terra terá levado esse ponto a 90° da Lua, e ele terá maré baixa. Dali a mais seis horas, o mesmo ponto estará a 180° da Lua, e terá maré alta novamente. Por esta razão, a água sobe e desce, aproximadamente, a cada 12 horas, 25 minutos.

Às vezes, a maré alta é mais alta que o normal. Isso acontece sempre que temos Lua Cheia ou Lua Nova. Quando temos Lua Cheia ou Lua Nova (*c.f.* Figura 2.1 (a)), o Sol, a Terra, e a Lua estão mais ou menos "alinhados" e portanto o efeito do Sol soma-se ao da Lua criando as chamadas "marés vivas". Elas caracterizam-se por preia-mares de grande altura e baixa-mares muito baixas.

Da mesmo modo, quando temos a Lua em Quarto Crescente ou Quarto Minguante (*c.f.* Figura 2.1 (b)), as marés baixas tornam-se menos baixas pois os efeitos do Sol não contribuem na mesma direcção em que se encontra a Lua. Nessa altura, temos as chamadas "marés mortas". Elas caracterizam-se por preia-mares de fraca elevação e baixa-mares pouco baixas.

O fenómeno das marés também é observado na parte sólida do planeta, ainda que com menor intensidade. O solo terrestre pode elevar-se até 45 centímetros nas fases de Lua Cheia ou Nova. Mas não nos apercebemos uma vez que tudo à nossa volta sofre essa elevação e deixamos assim de ter uma referência.

Podemos então, com base na descrição deste fenómeno, prever a maré num dado ponto do planeta para um determinado dia e a uma determinada hora.

No entanto, apesar de estarmos perante um fenómeno governado por factores determinísticos, existe um certo grau de indeterminação neste fenómeno devido ao número de factores que nele influem. Estudos revelam que existe, por exemplo, um certo atraso na periodicidade das marés devido à inércia exercida pelo solo do oceano. A água é puxada pela Lua ao mesmo tempo que é puxada pela Terra. Aí interferem as irregularidades do solo oceânico próximo de onde se observam ou se medem as marés.

Mas se o fenómeno das marés é, ainda assim, um fenómeno dito regular, muitos fenómenos existem que, dada a sua grande irregularidade, são denominados de "caóticos". Lorenz (1963) foi o primeiro cientista a perceber a importância desse tipo de fenómenos.

A previsão meteorológica é um fenómeno desse tipo, já que os modelos numéricos não conseguem reproduzir a enorme diversidade de factores que podem influenciar a evolução das condições atmosféricas. O que se sabe nem sempre é o que se pratica e, muitas vezes, a imprevisibilidade de certos acontecimentos introduz situações não contempladas pela descrição conhecida e dada como válida. Mesmo assim, isso não põe em causa a validade da descrição do fenómeno. À componente de imprevisibilidade deste fenómeno estão associados fenómenos como: enchentes, secas, furacões entre outras catástrofes. No entanto, apesar de não podermos fazer uma previsão exacta, o resultado da computação da previsão é sempre um dos comportamentos possíveis da atmosfera.

O mesmo acontece com o mercado bolsista, um sistema extremamente influenciável por todo o tipo de factores sejam eles da conjuntura económica, política, social, etc. Aliás, esta é uma das características que torna esse mercado tão desafiador.

No campo geológico, também a previsão de sismos se apresenta como um fenómeno com uma componente indeterminística. Os sismos ocorrem de forma irregular, variando o local de ocorrência, a sua frequência e a sua magnitude. Os terramotos que tanto aterrorizam as populações, não são mais do que sismos catastróficos.

Os sismos são abalos naturais da crosta terrestre que ocorrem num período de tempo restrito, em determinado local, e que se propagam pelas chamadas ondas sísmicas. Eles são originados por uma libertação brusca de energia num ponto (foco) do interior da terra, que por sua vez causam uma ou mais rupturas nas placas tectónicas que se traduzem por falhas. É a energia bruscamente libertada ao longo destas falhas que origina os sismos. Os deslocamentos dos blocos ao longo do plano de falha, não actuam nem se fazem sentir de maneira contínua e uniforme, mas por impulsos sucessivos. Por cada impulso é originado um sismo: acontecem as chamadas réplicas. Os sismos manifestam-se, geralmente, pelo abalo principal, logo no seu início. Depois da ruptura inicial, verifica-se uma série de rupturas secundárias, as quais correspondem ao reajustamento progressivo das rochas fracturadas. Por vezes, antes do abalo principal observam-se sismos de fraca intensidade denominados por abalos premonitórios.

Os sismos não apresentam uma distribuição aleatória à superfície do planeta Terra: estão repartidos de acordo com a localização das Placas Tectónicas. No entanto, são ainda assim denominados de sistemas caóticos, já que se torna muito difícil prever com exactidão o lugar, a magnitude e a data de ocorrência de um sismo.

Desde a década de 70 que actividade sísmica é monitorizada nas estações sismográficas. Inicialmente, o registo sísmico era apenas utilizado para localizar a origem do sismo e assim poder proceder a operações de salvamento. Foi na década de 90 que essa monitorização foi melhorada no sentido de registar mais parâmetros e com maior exactidão. Desta forma, a previsão tornou-se mais concretizável.

De um estudo realizado pela Universidade de Washington, decorreu o gráfico apresentado na Figura 2.2.

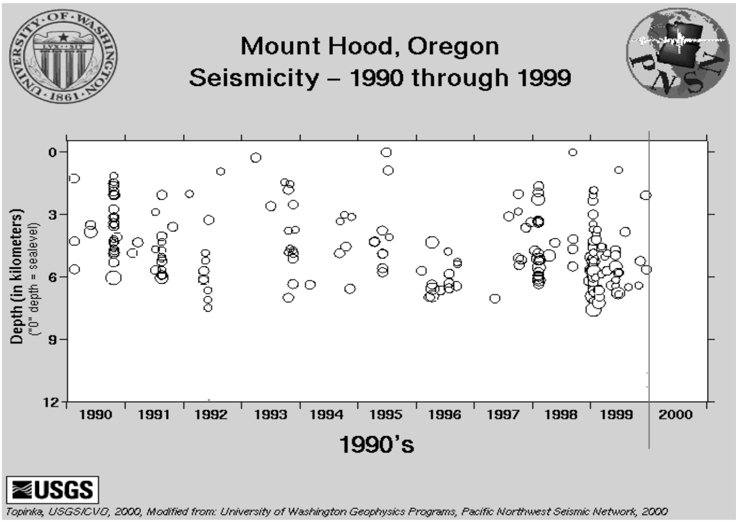


Figura 2.2: Sismos registados no período 1990-1999 na região de Mount Hood (Oregon).

Este gráfico permite visualizar a profundidade a que ocorreram sismos no década de 90 em Mount Hood, Estado de Oregon, região cuja localização favorece o surgimento de sismos.

Esta é uma região montanhosa, que possui um dos maiores vulcões que regista abalos mais ou menos frequentes.

Da observação da Figura 2.2 não se consegue identificar, à partida, um padrão de ocorrência de sismos, pelo menos no que respeita os registos relativos à década de 90.

No início dos anos 90, uma equipa de geólogos da Universidade de Coimbra avançou com um processo científico para a previsão de sismos. Esse processo passa pelo recurso a técnicas de Inteligência Artificial para que, com base nos sismos ocorridos entre 1912 e 1989, na sua magnitude, local e tempo de pausa entre eles, se tente prever com alguma exactidão o local, a magnitude e o intervalo de tempo onde ocorrerá um sismo.

A verdade é que nem sempre dispomos da descrição completa do fenómeno a prever. Por vezes, apenas dispomos do registo histórico da ocorrência do fenómeno e pretendemos, a partir desse registo, fazer uma previsão.

É no contexto da componente de imprevisibilidade associada a certos fenómenos ou da ausência da descrição dos fenómenos a prever, que a Inteligência Artificial assume um papel relevante. A Inteligência Artificial torna-se fundamental na resolução deste tipo de problemas por recorrer a métodos de aprendizagem a fim de descrever o conceito do fenómeno com base nos seus registos históricos e de fazer previsões fundamentadas na descrição obtida tentando, eventualmente, contemplar a sua componente de imprevisibilidade.

As primeiras aplicações de métodos de aprendizagem automática a este tipo de problemas tinham por objectivo a concepção de modelos que fizessem a descrição de um fenómeno assente na descoberta de padrões regulares da sua ocorrência. Isto é, a descoberta de padrões que se aplicassem a pelo menos uma grande parte das observações do fenómeno. Estas primeiras aplicações não tinham por objectivo responder às situações mais raras, mas às mais comuns.

Muitas vezes, o que se pretende com a resolução de problemas de previsão é ajudar no processo de tomada de decisões. A extracção de conhecimento a partir de dados passados leva, em princípio, à concepção de previsões mais fundamentadas que acabam por se reflectir na tomada de decisões de uma forma, talvez, mais "consciente" e "ponderada".

Um problema de previsão já bem conhecido no âmbito da Inteligência Artificial, e cuja resolução presta algum apoio na tomada de decisões, é o problema da concessão de crédito bancário. Apesar da concessão de crédito ser feita com base em factores determinísticos como o ordenado do cliente, a declaração de IRS,etc., a resolução deste problema trouxe uma mais valia à simples automatização de tarefas. A Inteligência Artificial permitiu conceber um modelo que sugere a concessão ou rejeição de crédito bancário com base no sucesso de decisões anteriormente tomadas. A previsão é feita de forma a minimizar o risco para o banco.

Como já foi referido anteriormente, o facto é que em muitos domínios existe uma forte componente de imprevisibilidade: fenómenos que acabam por ocorrer não sendo contemplados pelos modelos que os descrevem, dada a raridade da sua ocorrência. Em muitos dos algoritmos de aprendizagem existentes, é atribuída alguma atenção a estas ocorrências raras (excepções), mas apenas no sentido de conceber modelos mais tolerantes à sua ocorrência. No entanto, muitas vezes o interesse reside precisamente em conceber um modelo que faça a descrição desses fenómenos raros. Fenómenos raros são frequentemente fenómenos interessantes. Em muitos dos domínios eles constituem problemas de previsão de extrema importância. Senão vejamos o caso da previsão de catástrofes ecológicas, da previsão de oscilações significativas no mercado bolsista e do impacto que isso tem na economia de todo o mundo, da detecção de falhas ou ataques em grandes redes de computadores, detecção de fraudes, entre outros problemas. A descrição deste tipo de fenómenos pode, de facto, ser mais interessante do ponto de vista da descoberta de conhecimento. É sobre a resolução deste tipo específico de problemas de previsão que incide o nosso estudo.

Estamos interessados no estudo de métodos vocacionados para a previsão de fenómenos raros, situações fora do comum. Estes métodos deverão ser capazes de conceber modelos que consigam descrever o conceito de fenómeno raro no seu domínio de aplicação. No fundo, pretendemos, dado um conjunto de dados, ser capazes de detectar os pontos de frequência rara e que se distinguem por ser significativamente diferentes do resto dos pontos que constituem o conjunto de dados e aprender a sua descrição. Estes pontos são denominados por *outliers*.

2.1.1 Conceito de *outlier*

Desde que surgiu a preocupação de interpretar dados, o conceito de *outlier* despertou interesse na comunidade científica.

Em 1980, D. M. Hawkins formulou uma definição de *outlier*, ainda hoje considerada como a mais intuitiva: *"an observation which deviates so much from other observations as to arouse suspicious that it was generated by a different mechanism"*. Isto é, se num conjunto de observações conseguimos detectar que existem apenas algumas observações que se distinguem por se desviarem significativamente de todas as restantes, então existem suspeitas de que estas terão sido geradas por um mecanismo diferente.

Uma das questões que sempre se levantou foi o procedimento a adoptar na presença de *outliers*. Será que representam ruído nos dados e deverão por isso ser eliminados ou será que eles contêm, de facto, informação surpreendente e que deve por isso ser alvo de uma análise mais cuidada?

Tomemos como exemplo a Tabela 2.1 referente a um subconjunto do registo das notas obtidas por um conjunto de alunos a uma determinada cadeira. Imaginemos que se pretende fazer avaliação do aproveitamento dos alunos a essa cadeira

Alunos	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Notas	12	15	13	14	75	11	12	16	15	13	12	14	13	5	14	13	12	15

Tabela 2.1: Registo de notas com *outliers*.

Da observação dos dados podemos destacar as notas dos alunos E e N como sendo, aparentemente, *outliers*. Considerando a escala standard de notas $(0 - 20)$, poderemos supor que estes dois *outliers* têm proveniências diferentes. Ao aluno E foi atribuído uma nota errónea: a sua nota foi, por engano, registada em percentagem, dizendo nesse caso respeito ao valor 15. No entanto, nada nos garante que seja essa a razão: poderemos optar pela sua eliminação ou não. Já ao aluno N poderá ter sido atribuída uma nota perfeitamente legítima, ainda que se destaque de todas as notas obtidas pelos outros alunos.

Barnett e Lewis apresentaram em 1978 uma revisão dos estudos sobre *outliers* feitos até então, sob uma perspectiva estatística. Descreveram uma análise da problemática associada à interpretação de *outliers*. As primeiras abordagens entendiam que os *outliers* produziam um efeito nocivo sobre a interpretação de dados e davam origem a erros indesejáveis. Mais tarde surgiram abordagens mais elaboradas onde o reconhecimento e tratamento de *outliers* se tornou mais amplo e variado.

A comunidade científica ligada à estatística, estudou o conceito de *outlier* de uma forma bastante intensiva. Barnett e Lewis (1978) referiram que lidar com *outliers* é um problema inevitável para quem trabalha com a interpretação de dados. Os autores definiram *outlier* como sendo uma observação ou um pequeno subconjunto de observações que parecem estar inconsistentes com o resto do conjunto de dados. Será sempre subjectiva esta noção de inconsistência a que os autores se referem, já que depende do domínio e até de quem está a interpretar os dados. Contudo, na maioria dos trabalhos desenvolvidos nesta área, a definição de D. M. Hawkins, acabou por constituir a definição base de *outlier*.

Os *outliers* poderão ser univariados, baseados nos valores registados para um dado atributo, ou multivariados, constituídos por um conjunto de valores registados para diferentes atributos. Se no primeiro caso algumas técnicas estatísticas podem detectar a sua presença, no segundo caso essa detecção torna-se mais difícil, exigindo métodos mais sofisticados de detecção.

Muitas das técnicas estatísticas são sensíveis à presença de *outliers*. O simples cálculo da média e desvio-padrão de um conjunto de valores pode ser distorcido por *outliers*. O exemplo apresentado na Tabela 2.1 contém outliers univariados que acabam por distorcer a média e desvio-padrão das notas da cadeira. Tal como está o registo das notas, seria obtida uma média de $\simeq 15.5$ com um desvio-padrão de $\simeq 14.8$ (um desvio-padrão bastante elevado, considerando a escala $(0-20)$).

A investigação de *outliers* deve, por isso, ser feita de forma cuidadosa e incluída como uma fase do processo de análise de dados.

Perante a presença de *outliers* nos dados, são vários os procedimentos adoptados face aos objectivos estabelecidos: poderão ser simplesmente eliminados por corresponderem a ruído resultante de uma introdução errada de valores; poderão ser incorporados na concepção do modelo de modo a que este seja robusto à sua presença; ou então poderão ser identificados através de um modelo que descreva exactamente a sua ocorrência no domínio em análise.

Em qualquer um dos casos, a detecção de *outliers* constitui uma tarefa importante: quer para tentar perceber ocorrências excepcionais do fenómeno, quer para evitar que o modelo concebido para a descrição do fenómeno seja enviesado por erros no registo de ocorrências do mesmo. Contudo, distinguir uma ocorrência rara do fenómeno, de simples ruído é uma questão um pouco ambígua. Antes de considerar uma possível eliminação destas observações raras, há que tentar perceber a razão pela qual elas surgiram e se será provável que observações semelhantes do fenómeno voltem a acontecer num futuro próximo.

Em vez de detectar as ocorrências raras de um fenómeno para conceber um modelo tolerante a esse tipo de ocorrências, o nosso objectivo aqui é conceber modelos vocacionados, unicamente, para a detecção desse tipo de ocorrências.

Em suma, qualquer que seja o problema que envolva a análise de dados com possíveis *outliers*, os passos apresentados na Figura 2.3 deverão ser seguidos. Estes passos estão perfeitamente dependentes do domínio do problema e dos objectivos impostos para a sua resolução.

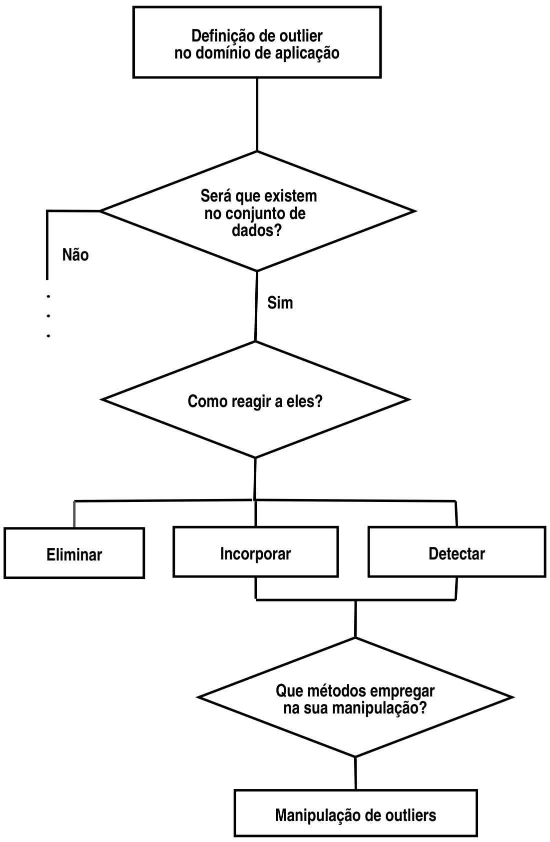


Figura 2.3: Procedimento *standard* no tratamento de *outliers*.

2.1.2 Desafios colocados

O objectivo deste tipo de problemas de previsão de fenómenos raros consiste na concepção de um modelo capaz de identificar uma dada ocorrência do fenómeno em estudo como sendo rara ou comum. Existem, contudo, dois desafios que se colocam quando tentamos proceder à resolução deste tipo de problemas.

O primeiro desafio prende-se com o facto de ser necessário localizar e identificar os *outliers* no conjunto de dados. De acordo com o domínio, esta poderá ser uma tarefa mais ou menos complexa.

O segundo desafio que se coloca é, de facto, aprender a descrição de um fenómeno cujo número de ocorrências é bastante reduzido. São portanto poucas as instâncias em que poderemos basear a nossa aprendizagem.

2.2 Alguns exemplos práticos

São várias as situações que implicam a resolução de problemas que lidam com a presença de *outliers* no conjunto de dados e, em particular, com a previsão de fenómenos raros.

Os registos clínicos são frequentemente usados para a detecção de uma determinada enfermidade. Essa detecção será tão mais valiosa quanto mais rara for a ocorrência dessa enfermidade. Tomemos como exemplo, a disfunção da Tiróide.

A disfunção da Tiróide é determinada através de uma análise ao sangue chamada de TSH. Esta análise regista valores como idade, sexo e uma série valores hormonais. Esta disfunção ocorre sempre que a glândula da Tiróide produz hormonas em quantidades excessivamente superiores ou inferiores aos valores assumidos como normais. As hormonas produzidas por esta glândula controlam o metabolismo do corpo, tendo influência no funcionamento de vários órgãos.

O conjunto de dados[1] aqui utilizado, é constituído por 3163 análises feitas ao sangue com o intuito de identificar essa disfunção. Das 3163 análises feitas ao sangue, é natural que

[1]Fornecido pelo Garavan Institute e J. Ross Quinlan, New South Wales Institute, Sydney, Australia, 1987 e disponível em *http://www.cs.sfu.ca/~wangk/ucidata/dataset/thyroid-disease/*.

a sua maioria indique a ausência desta disfunção, aliás isso é constatável pelo gráfico da Figura 2.4. Apenas 9.26% das análises realizadas indicam a presença da disfunção.

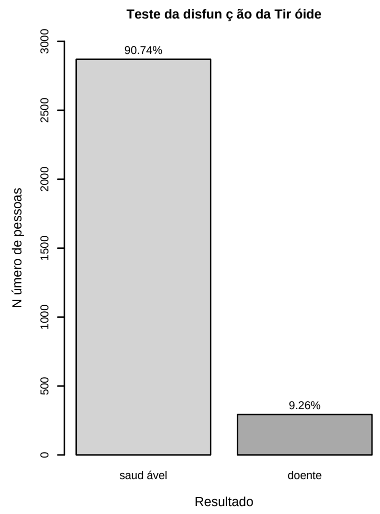
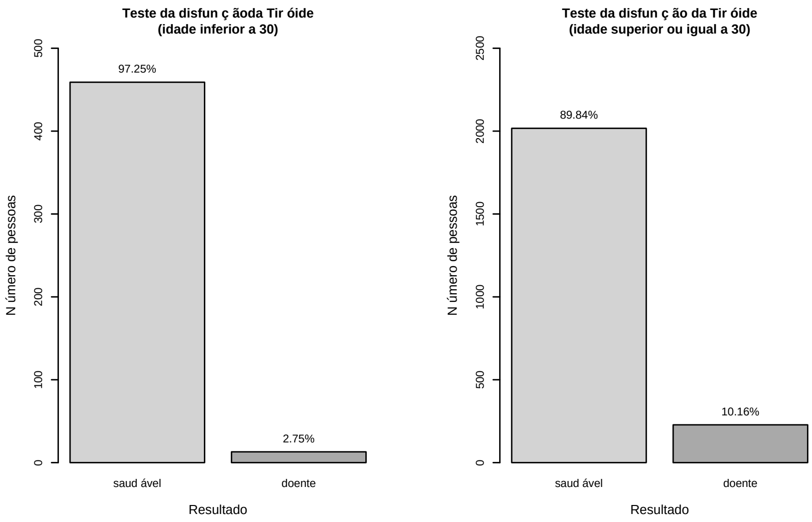


Figura 2.4: Resultados do testes da disfunção da Tiróide.

Existem algumas pessoas com esta disfunção e certamente também existirão algumas pessoas jovens neste conjunto de análises. O que é insólito é a existência de pessoas jovens que possuam esta disfunção. Neste contexto, este grupo de pessoas constituem um conjunto de registos de frequência rara e portanto interessante, sob o ponto de vista médico. Estes registos correspondem a *outliers*. Supondo que tínhamos por objectivo fazer algum estudo relacionado com as causas que levam pessoas jovens a serem portadoras desta disfunção, estes *outliers* não seriam, certamente, desprezáveis.

Nas Figuras 2.5 (a) e (b) são apresentados os gráficos obtidos a partir do conjunto de análises ao sangue de pessoas com idade inferior e superior ou igual a 30. Do conjunto de análises de que dispomos, apenas $\simeq$ 4.2% correspondem a análises a pessoas com idade inferior a 30. Claramente, o número de pessoas jovens dadas como doentes é significativamente inferior ao número de pessoas saudáveis, elas constituem apenas $\simeq$ 2.75% do total de análises efectuadas a pessoas deste grupo etário (*c.f.* Figura 2.5 (a)). Esta percentagem é, de facto, pequena se comparamos com a percentagem de pessoas portadoras desta disfunção com idade superior ou igual a 30. Neste caso, $\simeq$ 10.16% das pessoas possuem esta disfunção (*c.f.* Figura 2.5 (b)).



(a) faixa etária 0-29 anos (b) faixa etária $\geq$ 30 anos

Figura 2.5: Resultados do testes da disfunção da Tiróide a pessoas de duas faixas etárias.

A detecção de *outliers* abrange uma vasta área de aplicações. Entre elas, encontram-se as situações em que se pretende prever quando uma ocorrência fora do normal de um determinado fenómeno irá acontecer. Nesses casos, o estudo incide sobre dados que dizem respeito a registos do fenómeno correlacionados temporalmente. Por vezes, é a quebra de um padrão presente dados que torna um subconjunto de pontos em potenciais *outliers*. O mercado bolsista é uma das áreas onde isso pode acontecer. Nesta área, a previsão de fenómenos raros, ou se quisermos a previsão de mudanças bruscas no mercado de acções, pode vir a revelar-se bastante vantajosa. O interesse reside exactamente em antever possíveis quebras ou subidas significativas nas cotações. As quebras, tal como as subidas, poderão levar ao desencadeamento de ordens de compra e venda importantes.

A Figura 2.6 apresenta a evolução de cotações de um determinado índice, por um período de sensivelmente 300 dias. Da observação do gráfico é possível constatar que por volta do dia 230 deve ter ocorrido algum acontecimento que levou a uma quebra bastante acentuada no valor do índice e que deve precipitar a tomada de certas medidas. O dia 230 constitui de facto um *outlier* no contexto do padrão embebido nos preços de fecho do índice nos 300 dias aqui visualizados.

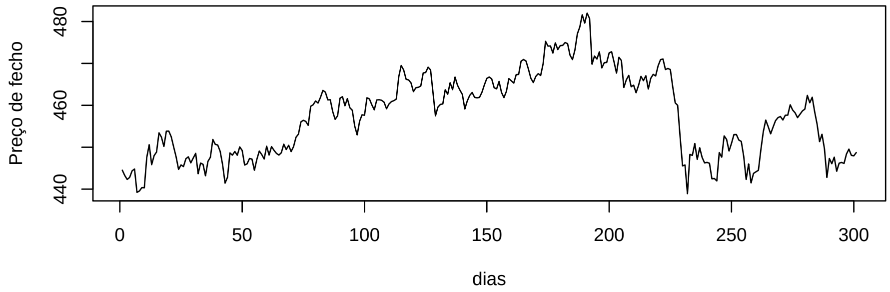


Figura 2.6: Cotações da bolsa durante um período de tempo.

Em qualquer um destes casos os potenciais *outliers*, partindo do princípio que não constituem ruído, revelam conhecimento que poderá ser de extrema importância.

Todos os problemas aqui referidos, apesar de pertencerem a domínios bastante dissemelhantes, possuem um ponto em comum: para todos eles a resolução passa pela aprendizagem de conceitos traduzidos por *outliers*.

Em suma, qualquer que seja a natureza do fenómeno raro a prever, ele será descrito por um conjunto de ocorrências identificadas como *outliers* nos dados.

Capítulo 3

Problemas Relacionados

Na literatura de *data mining* existem vários trabalhos relacionados com problemas de previsão de fenómenos raros.

As suas aplicações vão desde a detecção de situações anormais, passando pela monitorização de actividades para o accionamento de um alarme, pela antecipação de padrões surpreendentes, identificação de subconjuntos de dados particularmente interessantes, entre outras.

A monitorização de actividades é talvez a área de aplicação mais abrangente. Trata-se do registo contínuo de uma actividade e portanto de dados com alguma correlação temporal. Neste contexto, podemos falar do desencadeamento de um alarme em situações de previsão de catástrofes ecológicas, de previsões de falhas ou detecção de intrusão numa rede de computadores, detecção de fraudes telefónicas, de fraudes com cartões de crédito, detecção de oportunidades de investimento na bolsa, eventualmente com base em notícias, etc.

Neste processo estão normalmente envolvidas técnicas standard de aprendizagem automática como a classificação, a regressão ou a análise de séries temporais. A escolha das técnicas a utilizar reside na natureza do problema. Se a variável a prever assumir valores discretos então é natural que a abordagem seguida seja de classificação. Pelo contrário, se a variável a prever assumir valores contínuos, então a abordagem de regressão será a mais provável. Adicionalmente, se os dados possuem alguma correlação temporal, então deve-se recorrer a técnicas específicas para a manipulação destes dados. Contudo estas técnicas não resolvem, por si só, o problema da previsão de fenómenos raros.

Neste capítulo, é feita uma referência a algumas das abordagens adoptadas para a resolução de problemas relacionados com *outliers*. Ao mesmo tempo que se tenta dar a noção da evolução dos trabalhos nesta área, contrastam-se as principais abordagens. Na Secção 3.1 será feito um breve sumário de alguns dos conceitos de *outliers* usados em muitos dos trabalhos existentes vocacionados para a sua detecção.

Na Secção 3.2 são referidas algumas das abordagens de classificação seguidas para a identificação de classes raras. A necessidade de medidas de avaliação para estes métodos conduziu à constituição de medidas que acabaram por ser identificadas como medidas de avaliação base para o tipo de problemas ligados à identificação e descrição de *outliers*.

Em muitas das situações reais a previsão de fenómenos raros está, de facto, ligada à monitorização de actividades em contínuo. Na Subsecção 3.3 são referidos alguns dos trabalhos feitos nesta área, onde está subjacente uma correlação temporal nos dados.

Por último, na Subsecção 3.4 são mencionados alguns trabalhos relacionados com a detecção de conjuntos interessantes de dados. Este trabalhos, ainda que não vocacionados para a previsão de fenómenos raros, estabelecem uma estreita relação com este tipo de problemas. Fenómenos raros serão certamente fenómenos interessantes, embora o inverso não seja necessariamente verdade.

3.1 Detecção de *outliers*

Em muitos domínios ser um *outlier* não se trata de uma propriedade binária da observação. Depende de como esta está isolada relativamente à sua estrutura envolvente.

Para tentar perceber as condições que estão subjacentes à ocorrência de *outliers*, torna-se necessário proceder à sua identificação no respectivo domínio. Se em alguns domínios eles já vêm identificados como tal, em outros é necessário identificá-los. Este último caso é o mais comum em situações reais. De acordo com o domínio, teremos de recorrer a métodos de aprendizagem supervisionada ou não-supervisionada na detecção de *outliers*.

O problema que muitas vezes se coloca é: dado um conjunto com N observações encontrar possíveis *outliers*. Estamos, neste contexto, a referirmo-nos a uma conhecida técnica de *data mining*: *outlier mining*. Os objectivos do *outlier mining* prendem-se, fundamentalmente, com dois pontos: definir o conceito de *outlier* e encontrar um método eficiente para a sua detecção.

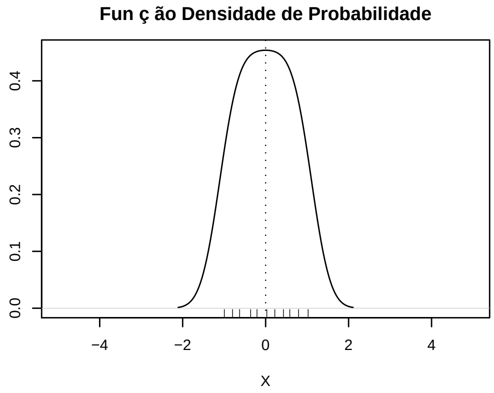
Não existe uma definição formal e universal do conceito de *outlier*. Na maior parte dos casos, a definição de *outlier* está intrínseca ao método adoptado para a sua detecção. São por isso várias as noções existentes de *outliers*. Aqui irão ser referidas algumas delas, utilizadas em métodos de aprendizagem não supervisionada.

Os primeiros trabalhos relacionados com detecção de *outliers* foram feitos no campo da estatística. Uma das noções concebidas é baseada em distribuições.

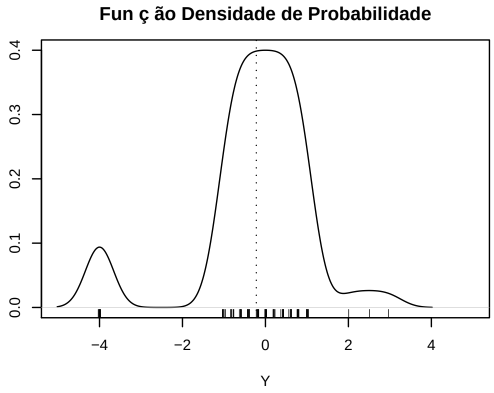
Assumindo que o conjunto de dados possui uma determinada distribuição standard, são considerados *outliers* todos os pontos que satisfazem um teste de discordância para a distribuição assumida, isto é, que significativamente se distanciam do meio da distribuição, em qualquer uma das direcções. Esta abordagem pressupõe que o fenómeno é descrito por uma variável aleatória contínua. Entre as distribuições teóricas para variáveis aleatórias contínuas, uma das mais empregadas é a distribuição Normal. Uma variável com distribuição Normal pode assumir todo e qualquer valor real e é completamente determinada por dois parâmetros: o seu valor esperado (μ), normalmente a média da população, e a sua variância (σ^2). A sua função de densidade de probabilidade é uma curva em forma de sino, simétrica em torno de μ, que recebe o nome de curva normal ou de Gauss. Dada a simetria da curva, se assumirmos μ como sendo a média, a probabilidade de ocorrer um valor maior que a média é igual à probabilidade de ocorrer um valor menor do que a média, isto é, ambas as probabilidades são iguais a 50%. Neste contexto, a média coincide com a mediana. Existem indícios típicos que referem a presença de *outliers*: a assimetria da curva normal em relação à média, caudas demasiado extensas ou demasiado curtas, ou até mesmo falhas.

Nas Figuras 3.1 (a) e (b) são apresentadas as funções de densidade de probabilidade de duas variáveis contínuas: X e Y. Se a variável X segue uma distribuição Normal, segundo as propriedades referidas, o mesmo não se pode dizer da variável Y. A variável Y possui alguns pontos significativamente distantes da sua mediana, o que está na base das caudas demasiado extensas e da assimetria da curva normal em relação à média (*c.f.* Figura 3.1 (b)). Estes indícios sugerem a presença de *outliers* na variável Y.

Esta abordagem coloca, segundo alguns autores (Breunig et al., 1999; Knorr e Ng, 1998), alguns problemas. Um deles é tomar como base distribuições univariadas o que inviabiliza



(a) densidade da variável X



(b) densidade da variável Y

Figura 3.1: Detecção de *outliers* assumindo uma distribuição Normal nos dados.

a detecção de *outliers* num espaço multi-dimensional. Um outro problema prende-se com o facto de pressupor uma distribuição que pode não ser bem ajustada à distribuição real dos dados.

Também relacionada com a detecção de *outliers*, surgiu alguma preocupação no sentido de conceber algoritmos de aprendizagem robustos à sua possível existência. Os algoritmos de *clustering* são um exemplo disso. O *clustering* de dados é um dos problemas de *data mining*, cujo objectivo é descobrir, num conjunto de dados, padrões da distribuição global dos dados, identificando regiões mais e menos populosas. Estas regiões são denominadas por *clusters* e representam um grupo de pontos, descritos por um conjunto de atributos, semelhantes entre si dada uma medida pré-definida de semelhança. Segundo esta medida, os pontos pertencentes a um determinado *cluster* serão mais semelhantes a um qualquer ponto do *cluster* a que pertencem do que a qualquer ponto de um outro *cluster*. Neste contexto, poder-se-á então dizer que os *outliers* são os pontos mais dissemelhantes no conjunto de dados.

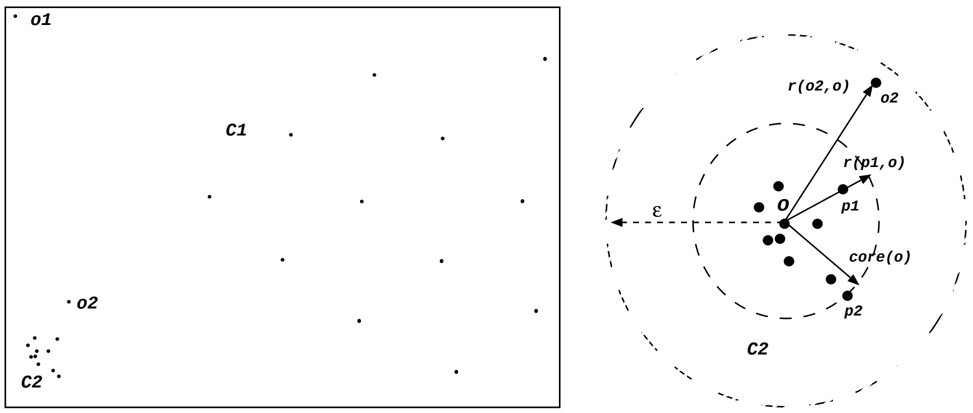
BIRCH (Zhang et al., 1996) foi um dos primeiros algoritmos de *clustering* capaz de lidar com a presença de *outliers* nos dados. No entanto, o seu principal objectivo era o de identificar *clusters* no conjunto de dados. A noção de *outlier* acabava por ser definida, não de uma forma precisa, mas como uma consequência do processo de *clustering*, de acordo com a medida de semelhança estabelecida. Os *outliers* eram encarados como simples ruído nos dados e não como pontos contendo algum tipo de informação valiosa no que respeita ao padrão global de *clustering* dos dados. Segundo esta análise, constituíam potenciais *outliers* todos aqueles pontos incluídos em regiões pouco densas. O algoritmo proposto, identificava, ao fim de algumas iterações, pontos que, de facto, constituíam *outliers*, podendo então ser ou não removidos.

Convém, contudo, não esquecer que este assim como outros algoritmos de *clustering* com alguma sensibilidade a *outliers* foram desenvolvidos para optimizar o *clustering* e não a detecção de *outliers*. A sua preocupação na detecção de *outliers* era unicamente a de evitar que estes interferissem no processo de *clustering*. A detecção acabava por acontecer como um efeito colateral do *clustering*. Por esta razão, este tipo de abordagem ainda que relacionada não se enquadra no âmbito da detecção de *outliers*.

Já com a noção de que o simples propósito da detecção de *outliers*, constitui por si só uma motivação de reconhecida importância, Arning et al. (1996) propuseram uma análise dos dados baseada em desvios. O objectivo imposto era o de encontrar, num determinado conjunto de dados, os pontos que mais se desviassem das características identificadas para os restantes pontos. Com esta abordagem, os autores pretendiam simular o mecanismo utilizado por seres humanos: após a observação de uma série de objectos semelhantes, um objecto que não se enquadre na série anterior é considerado uma excepção, um *outlier*. Para tal, é necessária a definição de uma função que capture quão diferente o novo objecto se apresenta em relação aos restantes. A definição desta função deve estar, contudo, um pouco dependente do domínio.

Num estudo feito por Knorr e Ng (1998) foi introduzida uma nova noção de *outlier* que se baseava em distâncias: *Distance-Based outlier*. Esta nova definição diz que um objecto o num conjunto de dados D é um $DB(f,d) - outlier$ se existe pelo menos uma fracção f de objectos em D que estejam a uma distância superior a d de o. Isto é, sendo n o número de objectos em D, para que o seja considerado um *outlier* não podem existir, numa distância d, mais do que $n(1-f)$ objectos. Se por um lado, os parâmetros f e d, permitem especificar o tipo de *outliers* que pretendemos, de acordo com a sua significância, por outro lado, a especificação do parâmetro d pode ser difícil de determinar no domínio em questão. Para além disso, uma vez identificados os *outliers*, esta definição não permite diferenciar a significância de cada um deles, visto que todos eles são considerados *outliers* da mesma forma.

Contestando as abordagens baseadas em distribuições e em distâncias, Breunig et al. nos seus estudos (Breunig et al., 1999, 2000), argumentaram que a detecção de *outliers* não se deveria basear em parâmetros globais. Esta proposta tem por base fundamentos teóricos associados ao *clustering* baseado em densidades. Segundo esta perspectiva, tal como referido por Zhang et al. (1996), *outliers* serão aqueles pontos que não pertencem a nenhum *cluster*. No entanto, ao contrário do que acontece com as abordagens convencionais de *clustering*, os *outliers* não serão aqui encarados sob uma perspectiva global nem tão dependente dos parâmetros de *clustering* e nem tão pouco como uma propriedade binária. Segundo eles, a identificação de *outliers* deve ser feita com base numa análise local da densidade dos pontos. Defendem que a cada ponto deve ser atribuído um grau, um indicador da significância deste como *outlier*. Este grau é designado por LOF (*Local Outlier Factor*) e diz respeito ao grau em que um ponto é considerado *outlier*, tendo em conta a densidade da sua vizinhança local. Segundo os autores, esta estratégia permite a identificação de outliers que não seriam captados pela abordagem baseada em distâncias referida por Knorr e Ng (1998). Senão, vejamos o exemplo de um conjunto de dados descrito por dois atributos e 24 objectos conforme o apresentado na Figura 3.2 (a).



(a) conjunto de dados

(b) *core-distance; reachability-distance*

Figura 3.2: Detecção de *outliers* baseada em densidades.

De uma primeira observação da distribuição dos pontos, é possível identificar dois *clusters*, isto é, dois grupos de pontos que possuem características semelhantes: C_1 e C_2. De acordo com a definição dada por Hawkins (1980), o_1 e o_2 deveriam ser considerados *outliers*, ao passo que os pontos em C_1 e C_2 não o deveriam ser. Esta é a intenção de Breunig et al., usando a noção de *outlier* local.

Os autores contrastam aquilo que aconteceria num cenário destes fazendo uso da abordagem proposta por Knorr e Ng (1998), onde só a identificação de o_1 seria evidente. Supondo que cada ponto em C_1, possui uma distância ao seu vizinho mais próximo superior à distância de o_2 a C_2, então não é possível, segundo a abordagem anterior, arranjar uma combinação dos parâmetros f e d tal que o_2 seja considerado um *outlier*, não incluindo os pontos de C_1 como *outliers* também. Se d fosse inferior ou igual à distância de o_2 ao seu vizinho mais próximo, então existiria um fracção de $23/24$ pontos mais distantes do que d de o_2. Nestas condições não só o_2 mas também o_1 e todos os pontos em C_1 seriam considerados *outliers*. Se, pelo contrário, o parâmetro d fosse estabelecido com uma distância superior à de o_2 ao seu vizinho mais próximo, a fracção de pontos f com uma distância superior a d a partir de o_2, seria inferior à de qualquer ponto pertencente a C_1. Nestas condições, dependendo do parâmetro f, ou o_1, o_2 e todos os pontos pertencentes a C_1 seriam considerados *outliers*, ou apenas o_1 e C_1 seriam considerados como tal. Este é um exemplo de uma situação em que, seguindo a abordagem proposta por Knorr e Ng (1998), um *outlier* poderia não ser identificado como tal por se encontrar perto de um *cluster* com uma densidade elevada.

Segundo Breunig et al. (1999, 2000), o conceito de *outlier* local é, de facto, importante. Em muitos dos problemas reais a estrutura dos dados pode não ser caracterizada por parâmetros globais de densidade. Diferentes regiões podem apresentar diferentes densidades. Neste contexto, será mais lógico decidir a possibilidade de um ponto constituir um *outlier* com base em outros pontos da sua vizinhança. A ideia base desta abordagem, consiste em identificar como *outliers* todos os pontos que se situem fora de *clusters*, sendo-lhes atribuído um grau: LOF. Este grau resulta de uma computação feita para cada ponto, que envolve alguns conceitos como: *core-object*, *core-distance*, *reachability- -distance* e *local reachability density*. Dada uma distância ϵ e um número mínimo de pontos k, um ponto p diz-se um *core-object* se existem, pelo menos, k objectos num raio de uma vizinhança-ϵ. A *core-distance* é definida como a distância de cada *core-object* ao seu k-ésimo vizinho mais próximo. $N_k(p)$ é definida como a k-vizinhança de p e inclui todos os pontos cuja a distância a p seja menor ou igual à sua *core-distance*. A *reachability- -distance* de um ponto p respeitante a um ponto o ($r(p, o)$), corresponde ao máximo entre a *core-distance* de o ($core(o)$) e a distância entre p e o.

Valores altos de *reachability-distance* entre dois pontos são um indicador de que estes dois pontos, poderão não pertencer ao mesmo *cluster*. A *local reachability-distance* de um ponto p (Breunig et al., 1999) é inversamente proporcional à média da *reachability-distance* da sua k-vizinhança. Segundo a fórmula apresentada por Breunig et al. (1999), a computação de LOF leva à obtenção de valores elevados para pontos cuja *local reachabilty-distance* seja muito inferior comparativamente à dos pontos incluídos na sua k-vizinhança.

Na Figura 3.2 (b), são exemplificados os conceitos de *core-distance* e *reachability-distance*. Segundo esta abordagem, e tomando $k = 10$, ao ponto o_2 seria atribuído um LOF mais elevado em relação aos restantes pontos que constituem o *cluster* C_2 ou C_1.

No entanto, este é um método computacionalmente pesado: exige uma série de computações feitas para cada ponto. Para além disso, a pré-determinação do parâmetro k pode ser problemática. Na realidade, este parâmetro pode levar a marcar como *outliers*, pontos que na verdade formam pequenos *clusters*.

Para reduzir um pouco este peso computacional, Jin et al. (2001) propuseram um algoritmo que, baseado no facto de que à partida serão poucos os pontos com potencial de *outlier*, apenas detectam os n_o *outliers* mais significativos. Desta maneira, é evitada a computação de LOF para a maior parte dos objectos quando $n_o << n$.

Também Ramaswamy et al. (2000), reconheceram algumas falhas na detecção proposta por Knorr e Ng (1998), nomeadamente no que toca à dificuldade de pré-determinar um valor para o parâmetro d e também no facto de não estabelecer uma relação de ordem entre os *outliers* detectados. Muitas vezes, estamos interessados em detectar os n_o *outliers* mais significativos. Propuseram então uma reformulação da definição de *outlier*, igualmente baseada em distâncias, que em vez de se basear numa distância d se baseia na distância do k-ésimo vizinho mais próximo de um ponto, e permite a especificação do número de *outliers* que pretendemos obter. Dado um conjunto de dados com n pontos, parâmetros n_o e k, um ponto p é um $D^k_{n_o}$ outlier se não existem mais do que $n_o - 1$ pontos p' tal que $D^k(p') > D^k(p)$[1]. Normalmente é esperado que n_o assuma um valor pequeno, independentemente do domínio, tornando-o mais fácil de especificar do que o parâmetro d.

Recentemente, num estudo feito por Angiulli e Pizzuti (2002), foi referido o factor densidade como ponto fraco da proposta de Ramaswamy et al. (2000). No caso apresentado na

[1] $D^k(p)$ representa a distância do ponto p ao seu k-ésimo vizinho mais próximo.

Figura 3.3, segundo a definição de Ramaswamy et al. (2000), se $k = 8$ então teríamos que $D^k(q_1) = D^k(q_2)$.

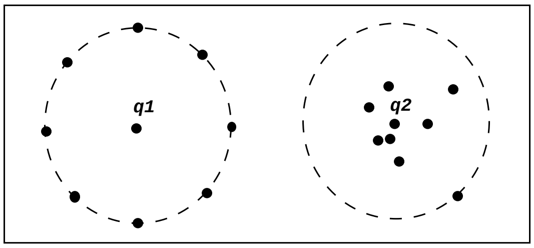


Figura 3.3: Dois pontos com o mesmo valor de D^k.

Na verdade, q_1 e q_2 não deveriam ser considerados *outliers* da mesma forma. Nesse mesmo estudo, Angiulli e Pizzuti (2002) propuseram uma definição de *outlier* também baseada em distâncias, mas que atribui a cada ponto p um peso resultante da soma das distâncias dos seus k vizinhos mais próximos. *Outliers* serão os pontos com maior peso.

O que de facto acontece é que todas estas abordagens partilham um ponto fraco comum: a noção distância surge sempre como um parâmetro, seja de forma explícita ou implícita. Estes tipo de abordagem é mais aplicável num contexto com um número relativamente reduzido de dimensões. Mas, tal como referido por Aggarwal e Yu (2001), na maioria das aplicações de *data mining* é muito provável depararmo-nos com um grande conjunto de dados descrito por um grande conjunto de atributos. Aqui os conceitos de proximidade e distância tornam-se um pouco subjectivos. Em muitos casos o conjunto de dados encontra-se globalmente disperso e, de acordo com as medidas de distância, qualquer ponto poderá constituir um potencial *outlier*. Num espaço multi-dimensional, a detecção de *outliers* torna-se substancialmente menos óbvia e mais complexa.

De facto, em aplicações reais, como aquelas em que poderá ser útil a detecção de *outliers* é com um grande conjunto de dados com que nos deparámos. Para além disso, a noção de *outlier* no domínio de aplicação não terá necessariamente a ver com todos os atributos que descrevem a observação. Por exemplo, a detecção de fraudes de um cartão de crédito está normalmente associada a um pequeno subconjunto de atributos de todos os atributos registados no uso dos cartões: atributos que foram de alguma afectados por uma anormalidade.

É com base nestes argumentos que Aggarwal e Yu (2001), propõem a detecção de outliers num espaço multi-dimensional, através da análise de densidades em projecções do conjunto de dados num menor número de atributos. Dado o elevado número de combinações que constituem sub-espaços possíveis, propõem proceder à procura de sub-espaços localmente dispersos, fazendo uma procura "evolucionária" à semelhança dos algoritmos genéticos.

As técnicas aqui referidas para a detecção de *outliers* estão longe de constituir um estudo completo das abordagens actualmente existentes nesta área. No entanto, não era esse o nosso objectivo. Pretendeu-se apenas dar uma perspectiva mais geral de algumas das abordagens, contrastando algumas das suas diferenças.

O que é importante salientar é que todas estas abordagens pretendem optimizar a detecção de *outliers*. Um factor determinante quando se fala de *outlier mining* é, de facto, a complexidade dos algoritmos utilizados na detecção de *outliers*. A investigação nesta área tem incidido sobretudo na concepção ou melhoramento de algoritmos no sentido de os tornar eficientes, capazes de dar resposta a conjuntos razoavelmente grandes de dados, com que cada vez mais frequentemente nos deparámos no dia-a-dia.

3.2 A classificação de fenómenos raros

Vários estudos foram feitos em relação à previsão de fenómenos raros reformulando o problema de previsão num problema de classificação supervisionada. A ideia consiste em conceber um modelo capaz de classificar um exemplo como uma situação comum ou fora do comum (*outlier*).

Nestes casos, o primeiro problema que se coloca é a evidente desproporção das classes. Partindo do pressuposto de que se trata de um problema de previsão de fenómenos raros, existirá sempre uma classe claramente minoritária em relação a todas as outras: a que pretendemos prever. Mas se o objectivo se prende apenas em detectar a ocorrência rara de fenómeno indicado por uma classe, todas as restantes classes poderão ser incluídas numa única classe, dita maioritária, que descreve as situações mais comuns. Qualquer que seja o domínio, o problema poderá então reduzir-se a um problema de duas classes apenas. O problema dos registos clínicos que indicam a presença ou ausência de uma disfunção da Tiróide em pessoas jovens, apresentado na Secção 2.2 e na Figura 2.5, encontra-se nestas circunstâncias.

Weiss e Provost (2001) fizeram um estudo sobre a influência da distribuição de classes na aprendizagem e de que forma isso afecta a *performance* do classificador gerado.

Através da realização de alguns testes comparativos, foi possível constatar que, tipicamente, os classificadores gerados a partir de um conjunto de dados com uma desproporção evidente de classes possuíam um pior desempenho na classificação da classe minoritária, relativamente àqueles gerados a partir do mesmo conjunto de dados, mas com uma proporção mais equilibrada das classes.

São duas as razões pelas quais isso acontece. A primeira está relacionada com o facto das regras geradas para a classe minoritária serem baseadas em menos exemplos e portanto mais sobre-ajustadas. O classificador tenderá a aprender limites mais rígidos do conceito. Aliás, este comportamento está relacionado com um problema já bem conhecido na área da aprendizagem: *small-disjuncts* (Weiss e Hirsh, 1998b, 2000b). Mais à frente, no Capítulo 4, Secção 4.4, será dada uma noção um pouco mais detalhada deste problema. A segunda razão tem, exactamente, a ver com o facto de, dadas as características do domínio, existirem mais exemplos de teste da classe maioritária do que da classe minoritária. A classe mais frequentemente prevista será a maioritária. Haverá, portanto, uma maior probabilidade de classificar incorrectamente exemplos da classe minoritária.

Numa primeira abordagem, e no sentido de tentar corrigir a natural desproporção das classes, poderíamos pensar em alterar a distribuição das classes na amostra de treino tornando-a mais equilibrada. Existem dois métodos básicos para tornar a distribuição de classes mais equilibrada:

- *under-sampling*: obter uma amostra mais pequena do conjunto de exemplos da classe maioritária;
- *over-sampling*: replicar exemplos da classe minoritária, obtendo uma amostra maior do conjunto de exemplos da classe minoritária.

Estes dois métodos trazem consigo algumas desvantagens. No caso do *under-sampling* podemos estar a desprezar dados potencialmente úteis. No caso do *over-sampling* estamos não só a aumentar o tamanho do conjunto de treino e portanto o tempo de computação, como também a proporcionar um maior sobre-ajustamento aos dados.

Para tentar minimizar estes efeitos nocivos foram desenvolvidas estratégias mais "inteligentes". Uma delas consiste numa técnica de *under-sampling* que apenas elimina exemplos redundantes da classe maioritária (Kubat e Matwin, 1997). Outra consiste em produzir novos exemplos da classe minoritária, interpolando o conjunto de exemplos disponíveis dessa classe que estejam muito próximos uns dos outros evitando, desta forma, a replicação do *over-sampling* tradicional (Chawla et al., 2000) e permitindo o relaxamento dos limites do conceito.

Para além das estratégias de *under-sampling* e *over-sampling* acima referidas existem outras abordagens a este problema. Uma delas, apresentada por Chan e Stolfo (1998), consiste em executar experiências preliminares a fim de determinar a melhor distribuição de classes para a aprendizagem de acordo com uma determinada função de custo. São gerados vários conjuntos de treino e combinados os modelos obtidos. Regra geral, a distribuição encontrada como melhor inclui todos os exemplos da classe minoritária e alguns da classe maioritária.

Weiss e Provost (2001) argumentam que para que o efeito da distribuição natural de classes seja o menos nocivo possível, será necessário proceder, de forma cuidadosa, à partição dos dados. Interessa pois tentar descobrir aquele que será o intervalo de distribuições que contenha a melhor distribuição de classes, a mais próxima da óptima, para maximizar a *performance* do classificador nos exemplos da classe minoritária.

Convém, contudo, ter em atenção que uma distribuição "artificial" das classes fornecida para treino não corresponderá, certamente, àquela encontrada no conjunto de dados de teste que será naturalmente desequilibrada.

Em muitos dos algoritmos de indução de classificadores é assumido que a distribuição de classes é igual, o que de facto nem sempre acontece. Por exemplo, numa árvore de decisão os exemplos são classificados de acordo com a classe mais provável na folha. Uma alteração à distribuição de classes no conjunto de treino resultará num conjunto de probabilidades enviesadas. Esse conjunto de probabilidades é baseada no conjunto de treino e deve, segundo Weiss e Provost (2001), sofrer uma correcção decorrente da diferença de distribuições entre o conjunto de treino e o conjunto de teste.

O algoritmo C4.5 (Quinlan, 1993) foi a ferramenta utilizada para este estudo. O processo de construção da árvore não sofreu qualquer alteração. Apenas foi vedada a possibilidade de *pruning* da árvore gerada para minimizar o erro. Caso contrário, o *pruning* seria feito com base na falsa assunção de que a distribuição de classes no conjunto de teste seria igual à distribuição de classes no conjunto de treino. A correcção das probabilidades é feita nas folhas e numa fase posterior à da concepção da árvore.

Após um conjunto de testes, este factor de correcção acabou por revelar uma melhoria substancial da *performance* das árvores de decisão.

Sempre que são feitas alterações à distribuição "natural" das classes, os classificadores a conceber devem ter essas alterações em linha de conta. Caso contrário, poderiam possuir um sobre-ajustamento a uma distribuição que não corresponde à distribuição verdadeira no conjunto de dados. Só assim poderemos esperar uma melhor *performance* geral do classificador.

3.2.1 Medidas de avaliação

Existe um outro factor neste tipo de problemas que os diferenciam dos problemas de classificação mais banais: o custo dos erros de classificação. O custo associado a classificações erradas não deverá ser igual qualquer que seja a classe. O objectivo principal reside na previsão de fenómenos raros. Assim sendo, a classificação de um exemplo que na realidade corresponde a uma situação rara como uma situação comum deve ser, de facto, mais grave do que a classificação de uma situação como rara sendo, na realidade, comum.

Imaginemos que uma ocorrência fora do comum do fenómeno a prever deve desencadear uma série de acções de precaução ou resolução da situação. Neste contexto, enquadram-se, por exemplo, todos os problemas em que deva ser accionado algum tipo de alarme perante uma ocorrência anormal do fenómeno. A previsão de catástrofes ecológicas é um dos casos mais evidentes em relação às precauções a tomar perante situações de alarme. Será portanto mais grave não prever uma catástrofe do que dar um falso alarme.

Estamos pois perante problemas que para além de possuírem uma distribuição não equilibrada de classes, atribuem custos diferentes aos erros cometidos, de acordo com a classe. Na verdade, neste tipo de domínios, é difícil determinar o custo associado a erros de classificação. O custo de não prever uma catástrofe ecológica, por exemplo, é um pouco relativo, pode depender de várias contingências do momento da ocorrência. A distribuição de classes também pode ir oscilando o que acaba por ter influência no custo dos erros.

Em muitas situações da vida real são frequentemente apresentados problemas cujo custo e distribuição de classes não pode ser determinada de forma precisa e estão sujeitos a mudanças. Nestas condições, torna-se difícil avaliar os classificadores gerados.

Tendo em conta o objectivo de prever a classe minoritária, os exemplos dessa classe dizem-se positivos. Dadas as previsões feitas por um determinado algoritmo, obtemos a matriz de confusão apresentada na Tabela 3.1.

		Previstos	
		Classe Minoritária	Classe Maioritária
Observados	Classe Minoritária	TP	FN
	Classe Maioritária	FP	TN

Tabela 3.1: Matriz de confusão associada à classificação de duas classes com uma distribuição não equilibrada.

Esta matriz fornece-nos o número de exemplos positivos verdadeiros (TP - *true positives*) e falsos, (FP - *false positives*), assim como o número de exemplos negativos verdadeiros (TN - *true negatives*) e falsos (FN - *false negatives*).

Tal como referido por Provost e Fawcett (1997), a tradicional *accuracy*, apresentada na Equação 3.1, tão usada em problemas de classificação, não se mostra adequada como medida de *performance*. Isto porque pressupõe que os custos das classificações erradas seja igual e que a distribuição de classes, no conjunto de teste, é conhecida e imutável.

$$accuracy = \frac{\mathsf{TP} + \mathsf{TN}}{\mathsf{TP} + \mathsf{FP} + \mathsf{TN} + \mathsf{FN}} \tag{3.1}$$

Trata-se de uma medida muito enviesada em favor da classe maioritária. Pelas razões referidas anteriormente, o classificador tenderá a ter um pior desempenho na classificação de exemplos da classe minoritária. Isto acontece porque no que se refere à classificação dos exemplos da classe minoritária presentes no conjunto de teste, a tendência será obter uma taxa de TP baixa e um taxa de FN mais elevada: a maioria dos exemplos serão classificados como pertencentes à classe maioritária.

Torna-se então necessário arranjar um método para a comparação de vários classificadores, que seja robusto a ambientes imprecisos e mutáveis.

ROC

Uma das medidas de avaliação de classificadores, disponíveis para este tipo de problemas são as curvas ROC. A análise ROC (Egan, 1975) surgiu no âmbito da *"Teoria da Detecção de Sinal"*. Esta teoria foi desenvolvida durante a segunda Grande Guerra Mundial para a análise de imagens de radar. A tarefa consistia em descobrir se um dado sinal era indicador de um barco inimigo, aliado ou simplesmente ruído. A detecção de sinal mede a capacidade do *receiver operator* do radar em fazer estas importantes distinções. Mais tarde, foi reconhecida a sua utilidade e extendida para a comparação de exames clínicos para diagnósticos médicos. Interessava avaliar a capacidade discriminatória dos exames médicos, em relação à presença ou ausência da doença.

Esta curva permite a avaliação da *performance* de um classificador, independentemente da distribuição de classes ou custo de erros. O uso da curva ROC é apropriado quando existem dois estados possíveis, neste caso duas classes. Ela estabelece a relação entre a taxa de TP (TP/(TP+FN)) e a taxa de FP (FP/(FP+TN)), variando um determinado *threshold*.

No caso das árvores de decisão, tal como referido no estudo de Weiss e Provost (2001), este *threshold* poderia ser definido em termos das probabilidades associadas às classes presentes nos exemplos de treino contidos em cada uma das folhas da árvore. Neste contexto, um ponto na curva ROC pode corresponder à situação em que as folhas da árvore de decisão apenas serão rotuladas com a classe minoritária, se e só se, a probabilidade de um exemplo (contido numa das folhas) pertencer a essa classe for superior a t. Estabelecendo um conjunto de *thresholds*, obteremos a curva ROC associada, neste caso, à *performance* do algoritmo C4.5 para uma dada distribuição de classes. Foi com base neste processo que Weiss e Provost (2001) fizeram um estudo comparativo de classificadores com diferentes distribuições de classes e da influência que estas exercem sobre a *performance* do classificador.

A curva ROC permite visualizar o compromisso entre a sensibilidade (taxa de TP) e a selectividade ($1-$ taxa de FP) do algoritmo. Idealmente, o algoritmo deveria possuir indicadores máximos de sensibilidade e selectividade, ambos igual a um. Tal significaria que o algoritmo não só seria capaz de identificar todos os exemplos da classe minoritária, assim como seria capaz de discriminar bem exemplos da classe minoritária da classe maioritária.

Na Figura 3.4 são apresentadas, a título de exemplo, quatro hipotéticas curvas ROC obtidas para quatro classificadores, fazendo variar um dado *threshold*: A, B, C e D. Conforme é possível constatar, todos eles possuem *performances* diferentes em termos de selectividade e sensibilidade. O ponto (0,0) diz respeito ao caso de nunca prever um exemplo como sendo da classe minoritária. O ponto (1,1), pelo contrário diz respeito ao caso em que todos os exemplos são classificados como pertencentes à classe minoritária. O ponto ideal é pois, o ponto (0,1): todos os exemplos da classe minoritária e apenas esses são identificados como tal.

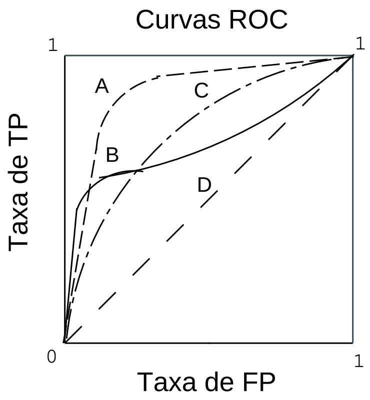


Figura 3.4: Comparação de curvas ROC.

De uma forma informal, um ponto numa curva ROC diz-se melhor do que outro, se este se encontra a noroeste do outro, isto é, se TP é mais elevado e FP é mais baixo. Neste contexto, o classificador A apresenta-se como aquele que possui um melhor desempenho na classificação da classe minoritária, já que grande parte dos pontos que constituem a sua curva se encontram a noroeste dos pontos obtidos pelos outros classificadores. O classificador D apresenta-se como sendo o pior.

Mas se por um lado as curvas ROC constituem uma valiosa técnica de visualização de *performances*, por outro lado são pouco elucidativas para a escolha do melhor classificador. A análise ROC apenas consegue declarar um classificador como sendo o melhor se esse apresenta uma clara evidência sobre os outros, no conjunto de pontos que constituem a sua curva. Observando novamente a Figura 3.4, o classificador B, poderia ser considerado melhor do que o classificador A, ainda que de forma pouco significativa, se apenas estivéssemos interessados em possuir uma taxa de TP de mais ou menos 50%. Passado esse limite, a taxa de TP cresce muito mais lentamente tornando-o pior do que o classificador C.

A fim de fornecer uma comparação um pouco menos empírica da *performance* dos diferentes classificadores, independente da distribuição de classes ou de custos de erros, existem duas técnicas frequentemente utilizadas: AUC (Metz, 1978) e ROCCH (Provost e Fawcett, 1997).

AUC

A AUC (*Area Under Curve*) consiste numa métrica de *performance* do classificador obtida através do cálculo da área por baixo da curva ROC do classificador. Ela assume valores entre 0 e 1 e pode ser interpretada como a probabilidade de um exemplo positivo, escolhido aleatoriamente, ser classificado como tal. No fundo, permite medir a sua *performance* discriminatória. A vantagem desta medida é que permite a comparação entre diferentes conjuntos de dados e com diferentes custos associados e estabelecer a relação de dominância entre os classificadores. Maiores valores de AUC indicam, geralmente, melhores classificadores.

Foi com base na medida AUC que Weiss e Provost (2001) avaliaram a *performance* dos classificadores gerados a partir de versões mais equilibradas de um conjunto de classes naturalmente desequilibrado. Através dessa avaliação puderam concluir que aprender a partir de uma versão equilibrada de um conjunto naturalmente desequilibrado de dados produz, na maior parte dos casos, menos regras de classificação mas mais "certeiras" para a classe minoritária do que para a classe maioritária.

Segundo os autores, isto acontece porque o número de exemplos da classe minoritária é naturalmente menor e formará por isso um conceito mais homogéneo do que a classe maioritária que representará "tudo o resto". Dado um número igual de exemplos, será de esperar que a classe mais homogénea seja melhor aprendida.

Como se torna computacionalmente impraticável testar todas as distribuições de classes possíveis, Weiss e Provost (2001) concentraram-se na procura de um intervalo óptimo de distribuição de classes, isto é, um intervalo que certamente inclua a distribuição óptima.

Com a medida AUC, descobriu-se que a distribuição óptima contém, geralmente, entre 50% a 90% de exemplos da classe minoritária; a distribuição 50:50 fornece, na maior parte dos casos, resultados melhores ou iguais à distribuição natural. Estes resultados podem decorrer do facto de estarem a ser eliminados factores que tendencialmente favorecem a classe maioritária.

ROCCH

A análise ROCCH (*ROC Convex Hull*[2]) permite declarar um subconjunto de classificadores como potencialmente óptimos (Provost e Fawcett, 1997). Incluídos todos os pontos que constituem as curvas ROC dos diferentes classificadores e formada a *convex hull* correspondente é feita uma análise dos pontos que estão sobre a linha de *convex hull*. Se um ponto está sobre a *convex hull*, então existe uma linha (denominada por *iso-performance line*) tangente a esse ponto, tal que não existe mais nenhuma linha com a mesma inclinação que passe por um outro ponto que tenha uma taxa de TP superior. Sendo assim, o classificador representado por esse ponto é óptimo sob a distribuição assumida correspondente a essa inclinação.

A inclinação das linhas *iso-performance* é obtida com base nos custos esperados da classificação. Todos os pontos de uma linha têm o mesmo custo esperado. Quão menor for a inclinação da linha, maior será a taxa de TP, menor será o custo associado ao classificador em questão.

Se um ponto não está sobre a *convex hull* então para o conjunto de linhas *iso-performance* existe um outro ponto numa dessas linhas com a mesma inclinação mas que possui uma taxa de TP mais elevada. Nestas circunstâncias, o classificador representado por esse ponto, não será óptimo, e portanto não representa um classificador óptimo.

Da observação da Figura 3.5, é possível constatar que o classificador B pode ser simplesmente excluído, ele nunca será um classificador óptimo na medida em que nenhum dos pontos da sua curva ROC está sobre a *convex hull*. Os classificadores A e C poderão ser potenciais classificadores óptimos, mesmo assim, poderão ser também excluídos da nossa análise todos os pontos das suas curvas ROC que não estejam sobre a *convex hull*. Os

[2]De uma forma um pouco informal, poderemos dizer que a *Convex Hull*, também conhecida por *envolvente convexa*, de um conjunto de pontos é o menor conjunto convexo de pontos, que contém todos os pontos desse conjunto.

classificadores A e C são óptimos para os *thresholds* respectivos que fizeram com que um ponto da sua curva ROC estivesse sobre a *convex hull*.

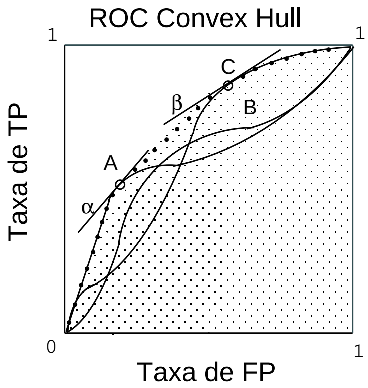


Figura 3.5: ROC *Convex Hull*.

Informação mais detalhada sobre este método poderá ser encontrada nos trabalhos: Provost e Fawcett (1997, 2001).

Precision e Recall

Em ambientes adversos com uma distribuição desequilibrada de classes e custos de erros pouco precisos, a avaliação da *performance* de classificadores decorre, muitas vezes, de duas medidas base: *precision* e *recall*.

Estas medidas são bastante utilizadas para a avaliação em domínios de *Information Retrieval*. Elas estabelecem que o custo de uma classificação errada deve ser dependente do número de classificações erradas nos outros casos.

A *Information Retrieval* (extracção de informação) é uma área de investigação que começou a despertar interesse por volta de 1940. De uma forma sucinta, podemos dizer que ela tem por objectivo a procura de documentos tão relevantes quanto possível, face ao pedido de um utilizador. Esta noção de relevância é, contudo, bastante subjectiva. A relevância de documentos obtidos face a um mesmo pedido pode variar de utilizador para utilizador. Ela poderá ser medida em termos de proximidade da informação contida nos documentos face à informação pedida e também em relação à utilidade dos mesmos para o utilizador. No livro de Meadow et al. (2000) é citada a seguinte opinião de Don Swatson (1988) em relação à noção de relevância: *"A document cannot be considered relevant to an information* *need independently of all other documents that the requester may take into account".* A relevância não se trata portanto de um conceito isolado: ela depende do contexto onde será avaliada. Mas, para que a construção do modelo tenha em conta a noção de relevância, torna-se necessário quantificá-la. A relevância está na base de duas conhecidas medidas: *precision*, e *recall*. As suas definições devem-se a Kent et al. (1955).

A *precision* mede a taxa de documentos relevantes em relação ao conjunto de documentos retornados.

O *recall* mede a taxa de documentos relevantes retornados em relação ao número de documentos relevantes existentes.

Idealmente, gostaríamos de obter todos os documentos relevantes e apenas esses. Pretende-se maximizar o número de documentos relevantes, minimizando o número de documentos não relevantes retornados.

Se transferirmos estes conceitos para o domínio de previsão de fenómenos raros, podemos ver as ocorrências raras de um dado fenómeno como sendo documentos relevantes.

Neste contexto, e de um ponto de vista teórico, a *precision* diz respeito ao número de ocorrências raras de um fenómeno que se conseguiram prever correctamente; o *recall* tem mais a ver com cobertura, isto é, do conjunto de todos fenómenos raros que realmente ocorreram quais foram aqueles que se conseguiram, de facto, prever.

Adoptando uma abordagem de classificação, podemos dizer que não nos interessa tanto o número de vezes que conseguimos fazer classificações correctas no conjunto de todos os dados. Interessa-nos sim dar especial atenção a uma das classes, neste caso a minoritária, ou se quisermos, a classe relevante. Idealmente, queremos tentar prever todos os exemplos desta classe e com precisão.

Formalizando estas noções podemos, com base na matriz de confusão da Tabela 3.1, definir as medidas de *precision* (Equação 3.2) e *recall* (Equação 3.3).

$$precision = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}} \tag{3.2}$$

$$recall = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}} \tag{3.3}$$

No caso concreto do nosso problema, estamos interessados em soluções precisas ($\mathsf{FP} \to 0$), e sobretudo em soluções com um *recall* razoável ($\mathsf{FN} \to 0$).

Existe uma relação típica entre *precision* e *recall* expressa no gráfico da Figura 3.6.

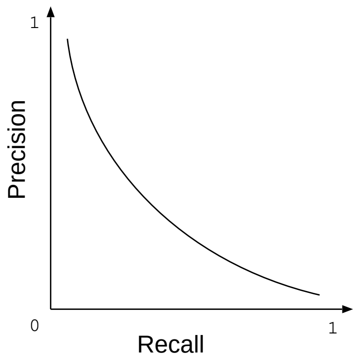


Figura 3.6: Relação entre *recall* e *precision*.

O *recall* pode assumir valores mais elevados através da geração de regras bastante gerais que classificariam mais exemplos como sendo da classe minoritária, mas isso fará decrescer certamente a precisão. Do mesmo modo, poderiam ser geradas regras bastante específicas e precisas que classificariam correctamente alguns exemplos da classe minoritária, mas certamente o valor de *recall* decresceria.

Em certas aplicações, aceitam-se soluções com baixa precisão desde que prevejam a maior parte dos exemplos da classe minoritária (*outliers*), isto é, com alto *recall*.

Segundo vários autores (Joshi et al., 2002; Weiss e Hirsh, 1998a), a melhor medida de *performance* para modelos concebidos para a previsão de fenómenos raros deve estabelecer um certo compromisso entre o *recall* e *precision*.

De acordo com o problema, estas medidas podem ser utilizadas na própria concepção do classificador. Deste modo, poderemos obter modelos mais orientados aos nossos objectivos concretos. Neste caso, o nosso objectivo consiste em obter umas boas medidas de *precision* e *recall* para a classe minoritária.

O *AdaCost* (Fan et al., 1999) surgiu como uma versão mais elaborada do *boosting* que se distinguia por dar alguma ênfase às medidas de *precision* e *recall*, sendo sensível a custos.

No que toca à distribuição desequilibrada de classes, o *boosting* tem-se revelado um bom meta-classificador pela sua adaptabilidade. Esta técnica gera uma sequência de classificadores cada vez mais peritos na classificação dos exemplos mais "difíceis", isto é, aqueles nos quais o classificador mais frequentemente comete erros. A estes exemplos são atribuídos pesos cada vez maiores. Múltiplos modelos são gerados e são derivados os pesos das suas previsões de acordo com as suas *performances*. As previsões feitas por cada um deles, são agregadas tendo em conta o peso de cada modelo. O que se comprovou ultimamente é que esta técnica consegue sempre melhorar a *accuracy*, isto é taxa de acertos, à medida que as iterações vão progredindo. Contudo, não se fez nenhuma análise teórica relacionada com o seu comportamento relativo ao *recall* e à *precision*.

O algoritmo *AdaCost*, baseia-se na técnica de *boosting*. O *boosting* convencional, atribui a cada exemplo mal classificado o mesmo peso, isto é, não usa o custo do erro na actualização dos exemplos. Com o *AdaCost*, a actualização dos pesos dos exemplos mais custosos é feita de um modo diferente em relação aos restantes exemplos. No caso de erro de classificação, o incremento do peso do exemplo é feito de forma mais agressiva. No caso de classificação correcta, o decremento do peso do exemplo é feito de forma mais conservadora. A redução do custo dos erros de classificação é feita de forma mais significativa. Os pesos associados a exemplos mais custosos acabam por ser mais elevados quando comparados com os outros exemplos.

No contexto do nosso problema, os exemplos mais custosos são os pertencentes à classe minoritária. O *AdaCost* acaba por dar um incremento maior ao peso dos FN e um decremento menor ao peso dos TP.

Implicitamente, e ao contrário do *boosting*, a técnica utilizada por este algoritmo, acaba por atribuir menos importância à *accuracy* e dar alguma ênfase quer ao *recall*, quer à *precision*.

Um outro algoritmo sensível às medidas de *recall* e *precision* é o *PNrule*. O *PNrule* é um algoritmo de indução de regras, proposto por Joshi et al. (2002), que aprende um modelo para a classe objectivo em duas fases.

O modelo resultante consiste em dois tipos de regras:

- *P-rules*: regras que prevêem a presença da classe C;
- *N-rules*: regras que prevêem a ausência da classe C.

Neste caso concreto a classe C, refere-se à classe minoritária, o fenómeno raro que pretendemos prever.

A primeira fase destina-se à aprendizagem das *P-rules* através modo de cobertura sequencial. O objectivo é obter um bom *recall* para C, induzindo regras com grande suporte. Nas últimas iterações desta fase, a *precision* é desprezada em favor do suporte. Isto resulta num modelo menos preciso.

Reunidos os exemplos TP e FP cobertos pelas regras geradas na primeira fase, numa segunda fase são aprendidas as *N-rules*. A aprendizagem é feita também através de um modo de cobertura sequencial e de maneira a remover os FP, mas de uma forma colectiva. Aqui o objectivo é melhorar a *precision*, enquanto o *recall* é mantido a um nível dado como aceitável.

Os processos de adição de regras na *P-phase* e crescimento de regras na *N-phase* são conduzidos por dois limites inferiores de *recall*, rp e rn, respectivamente. Estes parâmetros poderão ser ajustados no sentido de obter uma melhor *performance*.

Segundo os autores (Joshi et al., 2002), o *PNrule* acaba por ter uma melhor *performance* que o *AdaCost*. A razão, segundo eles, prende-se com o facto do primeiro ser mais sensível à previsão de classes raras. Se o *AdaCost* vai fazendo uma actualização incremental dos pesos, que acaba por se traduzir na remoção de FP, o *PNrule* está mais preocupado em recolher todos os FP e aprender a sua descrição correcta e completa.

O *AdaCost* não tem, de facto, a capacidade de efectivamente capturar as regras produzidas na fase das *N-rules*.

Resultados apresentados por Joshi et al. (2002) demonstraram que o *AdaCost* falha na habilidade que o *PNrule* possui de recolher apenas os FP relevantes e aprender regras específicas para os excluir.

Concluiu-se que o *PNrule* mostra um melhor desempenho em problemas de classificação de classes raras, quando existe uma correlação forte entre regras de distinção entre classes raras e frequentes.

F-measure

Também no âmbito da *Information Retrieval*, Rijsbergen (1979) desenvolveu uma medida de avaliação para traduzir a eficácia da recolha de documentos relevantes. Esta medida denominada por *effectiveness* e apresentada na Equação 3.4, tem por base as medidas de *precision* e *recall*.

$$E = 1 - \frac{1}{\alpha \cdot \frac{1}{precision} + (1-\alpha) \cdot \frac{1}{recall}} \tag{3.4}$$

O parâmetro α, permite a especificação da diferença relativa da importância do *recall* e da *precision*.

Se tomarmos $\alpha = 1/(\beta^2 + 1)$, a partir dessa medida é obtida a conhecida *F-measure* apresentada na Equação 3.6, com a transformação dada pela Equação 3.5.

$$F = 1 - E \tag{3.5}$$

$$F = \frac{(\beta^2 + 1) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{3.6}$$

Esta medida assume valores entre 0 e 1, tal como a *precision* e o *recall*, e estabelece um compromisso entre *precision* e *recall*, penalizando as situações em que ambas ou uma das medidas assuma um valor baixo. Porque o *recall* e a *precision* são inversamente proporcionais, valores elevados de *precision* que sacrifiquem o *recall* ou vice-versa, conduzem a um baixo valor de F. Deste modo, valores baixos de F são indicadores de uma pior *performance*. Esta medida assume valores mais elevados quando as medidas *recall* e *precision* forem ambas elevadas. O parâmetro β permite especificar o compromisso entre *precision* e *recall*. Ele define quão importante é o *recall* face à *precision*:

- $\beta = 1$, as medidas *precision* e *recall* são igualmente balanceadas;
- $\beta \to 0$ atribui ao *recall* menor importância; em particular, $\beta = 0.5$ atribui ao *recall* metade da importância da *precision* e $\beta = 0$ corresponde simplesmente à *precision*;
- $\beta \to \infty$ atribui ao *recall* uma maior importância; em particular, $\beta = 2$ atribui ao *recall* o dobro da importância da *precision* e $\beta = \infty$ corresponde simplesmente ao *recall*.

Os gráficos da Figura 3.7 apresentam os valores obtidos pela *F-measure* em função da *precision* e *recall*.

Da observação dos gráficos contidos na figura, é possível constatar que a *F-measure* assume diferentes valores de acordo com o parâmetro β especificado.

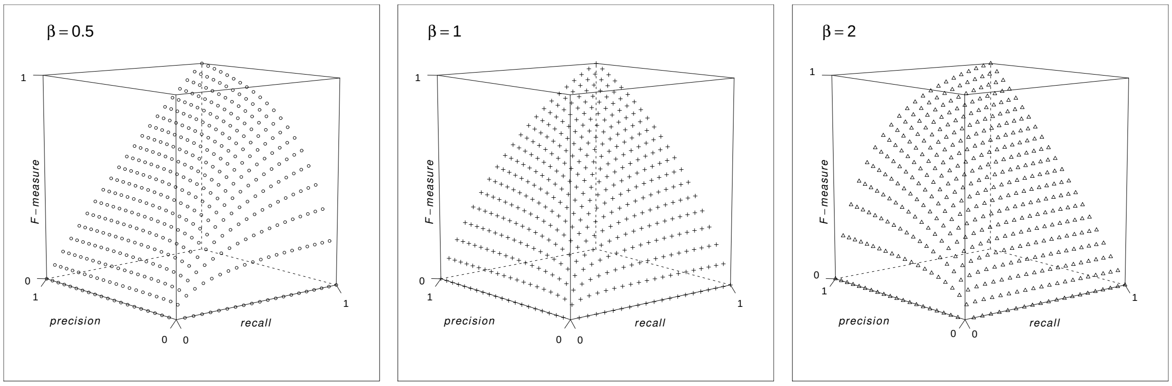


Figura 3.7: *F-measure* em função de *precision* e *recall*.

Valores mais baixos de β ($\beta = 0.5$, por exemplo) decrescem a importância do *recall*. A *precision* domina os valores assumidos por F. Da mesma forma, valores mais elevados de β ($\beta = 2$, por exemplo), aumentam a importância do *recall*, que passa assim a dominar os valores assumidos por F. Com $\beta = 1$, ambas as medidas possuem igual importância nos valores assumidos por F. Os valores da *F-measure* serão tão mais elevados quanto mais elevados forem os valores de *precision* e *recall*.

Para além desta *F-measure*, outras medidas de eficácia foram concebidas, recorrendo às medidas *precision* e *recall*. Meadow et al. (2000) referem algumas delas, propostas por Heine (1973), Meadow (1973) e Vickery (1979). Contudo, estas medidas estabelecem um compromisso pré-determinado entre a *precision* e o *recall*. A *F-measure* possui um carácter mais flexível face a diferentes objectivos de diferente problemas.

Todas as medidas de avaliação referidas nesta subsecção, constituem propostas para problemas que de certa forma estão relacionados com a classificação de *outliers*. Problemas em que a distribuição de classes não é de todo equilibrada e em que o custo de erros de classificação pode ser difícil de determinar.

3.3 A monitorização de actividades

Quando falamos de previsão de fenómenos raros, a grande maioria dos problemas prende-se com a detecção de uma anomalia num registo sequencial de fenómenos. Isto é, regra geral, os dados sobre os quais se coloca este tipo de problemas possuem uma correlação temporal.

Este é o caso por exemplo da detecção de fraudes de cartões de crédito ou de telecomunicações. A detecção de uma fraude telefónica só será possível se for feita uma análise da distribuição de chamadas ao longo do tempo. Uma vez identificada a distribuição típica de chamadas, uma alteração significativa a esta distribuição poderá detectar o roubo do telefone. A análise de chamadas telefónicas como actos isolados não nos fornecerá um indicador relativo à distribuição de chamadas e poderá portanto ser dada como perfeitamente legítima.

O caso da intrusão em redes de computadores é também um exemplo evidente. A cada utilizador é associada uma sequência de comandos que, regra geral, costuma executar. A identificação de uma sequência diferente, pode ser um sinal de que um outro utilizador está a fazer uso do mesmo *login*.

Em ambos os casos, estamos perante a necessidade de identificar padrões sequenciais nos dados a fim detectar alguma violação a esse padrão.

É importante não desprezar a correlação sequencial dos dados. É necessário fazer uso de estratégias que tirem partido dessa informação. Deixamos de ter um conjunto de exemplos, para ter um conjunto de registos de eventos, isto é, observações ocorridas em instantes específicos.

Dietterich (2002) contrasta este tipo de problemas de identificação de padrões sequenciais nos dados, por ele denominados como problemas de aprendizagem sequencial e supervisionada, com outros dois tipos de problemas semelhantes: previsão de séries temporais e classificação de sequências.

Na previsão de séries temporais o objectivo é, dada uma sequência de eventos ocorridos até um instante t, prever os eventos que irão acontecer nos próximos instantes consecutivos: $t+1..t+n$. Normalmente as previsões feitas estão assentes num período fixo anterior ao instante da previsão, onde são conhecidos os valores reais da previsão. Um exemplo bem conhecido de séries temporais poderia ser a previsão de cotações no mercado bolsista. No caso da aprendizagem sequencial e supervisionada, dispõe-se de toda a informação registada até ao momento e não de apenas uma período anterior fixo e se nas séries temporais dispomos dos valores reais associados a esse período, aqui todos esses terão de ser previstos. Por outras palavras, o alvo da previsão não é apenas um valor mas uma sequência de valores de forma a constituir os padrões. Este seria o caso se pretendêssemos, por exemplo, associar uma construção gramatical a uma frase. Para além disso, alguns autores, tal como Keogh et al. (2002) por exemplo, afirmam que as séries temporais implicam um registo regular de eventos apenas descritos por variáveis numéricas. Isto impõe limitações relativas à natureza do fenómeno que se pretende prever.

Na classificação de sequências, a ideia é atribuir uma única classe a uma sequência de *input*. Dada uma sequência de letras escritas à mão poderá, por exemplo, ser interessante determinar quem as escreveu. No entanto, este objectivo não é coincidente com o objectivo a que os problemas de aprendizagem sequencial e supervisionada pretendem dar resposta.

Tipicamente, a detecção de fraudes ou de intrusos numa rede, entre outros problemas, envolve a monitorização dos eventos registados a fim de identificar a ocorrência que torne necessária uma reacção. Fawcett e Provost (1999) enquadram este tipo de problemas na monitorização de actividades para a identificação de comportamentos dados como interessantes pela sua raridade e impacto no respectivo domínio. A monitorização consiste num processo *online* que faz uma análise sobre dados que estão continuamente a ser actualizados. Muitas vezes, o que se pretende é procurar padrões que violem padrões encontrados numa fase anterior.

Muito embora técnicas de classificação, regressão e também de séries temporais possam ser usadas na resolução deste tipo de problemas, elas por si só não constituem uma solução.

O objectivo da monitorização de actividades é dar a indicação de que a sequência está a exibir uma actividade positiva, isto é, a dar um alarme. Mas há que ter em atenção que o alarme deve ser dado de tal modo que haja tempo para tomar as devidas precauções. As previsões devem estar restritas a uma certa janela de tempo.

A ideia não é classificar todos os eventos como indicadores de uma actividade positiva ou negativa mas apenas detectar o início da actividade positiva numa perspectiva temporal.

Voltando ao caso da detecção de chamadas fraudulentas, pretende-se analisar um grande conjunto de chamadas registadas e despoletar um alarme tão cedo quanto possível.

Weiss e Hirsh (1998a, 2000a) formalizaram este tipo de problemas. Interessa aprender um modelo de previsão P que preveja correctamente o evento-objectivo. O evento-objectivo será aquele cuja ocorrência deve desencadear um alarme. P é uma função que mapeia uma sequência de eventos numa previsão booleana.

$$P : E_{t_1}, E_{t_2}, \dot{;} E_{t_x} \rightarrow \{+, -\}$$

Uma previsão positiva significa que o evento-objectivo irá acontecer. Basta um previsão singular positiva para que a previsão do evento-objectivo seja positiva, independentemente de eventuais previsões negativas subsequentes.

O período de previsão está delimitado pelo tempo de *warning* (w) e pelo tempo de *monitoring* (m), ambos dependentes do domínio, conforme argumentam Weiss e Hirsh (1998a, 2000a).

Tomemos como exemplo a detecção de uma falha numa rede de computadores. O tempo de *warning* corresponde ao tempo necessário para que, após prevista a falha, o alarme seja accionado e portanto a previsão se torne verdadeiramente útil. O tempo de *monitoring* define quão distante pode ser a previsão no futuro.

A previsão diz-se correcta se o evento previsto ocorre no período de previsão que lhe é atribuído (*c.f.* Figura 3.8).

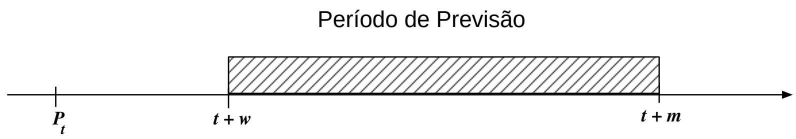


Figura 3.8: Período de previsão para um instante t.

No caso apresentado na Figura 3.9 a previsão P_{t1}, para o evento X_t, seria demasiado distante, ao passo que a previsão P_{t2} seria demasiado próxima, não dando tempo de *warning*.

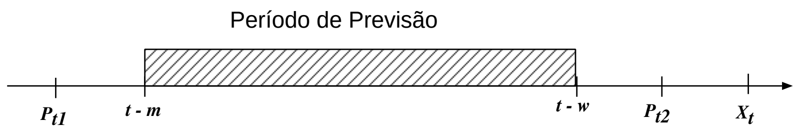


Figura 3.9: Previsões diferentes para um dado evento num instante t.

O problema será tão mais fácil quanto menor for o tempo de *warning* e maior for o tempo de *monitoring*. Um tempo de *warning* reduzido pode melhorar a detecção de situações de alarme, no sentido em que mais informação pode ser usada do período imediatamente anterior ao alarme. Um tempo de *monitoring* maior, aumenta a probabilidade de ser feita uma previsão correcta. No entanto, quão mais distante se tornar a previsão, menos significativa ela poderá ser.

A questão dos custos associados às previsões neste tipo de problemas, foi também alvo de discussão no trabalho de Fawcett e Provost (1999). Os alarmes não possuem sempre o mesmo valor: num mesmo período de previsão, o primeiro alarme é o mais importante; alarmes subsequentes não fornecem nenhuma informação nova e por isso a sua contribuição é quase nula. Um alarme será tão valioso quão mais prematuro for. O custo normalmente associado à monitorização de actividades diz respeito aos acontecimentos anteriores ao primeiro alarme. Por exemplo, no caso da detecção de chamadas fraudulentas, o custo de não ser identificada uma situação de alarme pode ser quantificado pelo número de chamadas fraudulentas efectuadas antes de ter sido accionado o alarme, ou então pela duração dessas mesmas chamadas ou até pelo número de chamadas que, possivelmente, seriam efectuadas se o alarme não tivesse sido accionado. Pelas razões já referidas na Subsecção 3.2.1, também Fawcett e Provost (1999) propõem uma avaliação da monitorização de actividades com base nas curvas ROC, mas adaptada a este este domínio, tendo em conta a noção de unidade de tempo, a taxa de alarmes falsos e o benefício de um dado alarme.

Outros autores, tais como Weiss e Hirsh, recorrem às medidas de *precision* e *recall*, transpondo estes conceitos para um contexto de previsão de eventos. A noção de TP teve de ser alvo de uma reformulação. A contabilização de TP, tal como foi definida na Subsecção 3.2.1, pode levar a valores erróneos no contexto da previsão de eventos já que podem ocorrer múltiplas previsões positivas do mesmo evento. O número de previsões positivas correctas é substituído, pelo número de eventos correctamente previstos. O número de eventos correctamente previstos, diz respeito ao número de eventos diferentes previstos pelos menos uma vez no período de previsão que lhes é atribuído e que, de facto, ocorreram nesse período.

As medidas de *precision* e *recall* passam a embeber o conceito de eventos-objectivo correctamente previstos. Assim o *recall* passa a ser definido pela Equação 3.7.

$$recall = \frac{\#\ eventos\ correctamente\ previstos}{total\ de\ eventos} \tag{3.7}$$

Da mesma forma, a *precision* passa a ser definida pela Equação 3.8, sendo denominada por *normalized precision*.

$$normalized\ precision = \frac{\#\ eventos\ correctamente\ previstos}{\#\ eventos\ correctamente\ previstos + \mathsf{FP}} \tag{3.8}$$

No entanto, a *normalized precision* não tem em conta o facto de n previsões positivas incorrectas próximas umas das outras não terem o mesmo efeito nocivo do que o mesmo número de previsões positivas incorrectas distribuídas por um período maior de tempo. O efeito nocivo está dependente do domínio e das acções necessárias a tomar como reposta. A *reduced precision*, apresentada na Equação 3.9, contempla este aspecto. Uma previsão é considerada "activa" durante o seu período de monitorização, período durante o qual ela deve ocorrer. A *reduced precision* substitui o número de FP pelo número de períodos de *monitoring*, não sobrepostos, associados a uma previsão positiva falsa.

$$reduced\ precision = \frac{\#\ eventos\ correctamente\ previstos}{\#\ eventos\ correctamente\ previstos + \mathsf{FP}\ ''descontado''} \tag{3.9}$$

Foi com a motivação de detectar falhas em redes de telecomunicações que Weiss e Hirsh (1998a) propuseram um algoritmo genético que gera padrões para a previsão de eventos. Neste caso, estes eventos correspondem às falhas: eventos raros e difíceis de prever.

A tarefa consistia em identificar padrões nos *logs* de alarmes da rede que seriam utilizados na previsão de falhas. Nestes *logs* estão registados os alarmes ocorridos de forma sequencial, cada registo corresponde por isso a um evento. Interessa aprender padrões a partir desse conjunto de dados. Contudo, convém não desprezar a correlação temporal dos dados incorporando-a na aprendizagem.

Nestes casos, é frequente recorrer a duas técnicas já bem conhecidas: *sliding-window* ou *recurrent sliding-window* (Dietterich, 2002). Através de um pré-processamento dos dados, estas técnicas permitem reformular o problema num problema de aprendizagem de um conceito, incorporando a noção temporal intrínseca aos dados. A ideia base destas técnicas é construir exemplos representativos de janelas de tempo. Após fixado o tamanho da janela (n), em cada exemplo, são incorporados o registo de n eventos consecutivos. O conjunto de exemplos é concebido pelo arrastamento desta janela de tempo fixa. Se na técnica *sliding-window* é assumida uma relação de independência entre cada janela de tempo que constitui cada um dos exemplos, as *recurrent sliding-windows* não o fazem. Por essa razão, nesta última técnica não só são incluídos os registos de n eventos consecutivos, como também variáveis "explicativas" que descrevam alguma correlação temporal dos dados adicional. Os exemplos são classificados de acordo com os eventos inseridos na janela. Neste caso concreto, a classe poderá indicar a ocorrência do evento que se pretende prever, na janela de tempo correspondente.

Contudo, em qualquer uma destas técnicas é inevitável a perda de alguma informação: em cada momento apenas tiramos partido de um intervalo fixo de tempo. Para além disso, a constituição destas janelas envolve um pré-processamento que pode ser dispendioso a nível computacional.

Assim, Weiss (2002); Weiss e Hirsh (1998a, 2000a), propõem um método básico de aprendizagem que faz uma abordagem directa ao problema, sem necessidade de reformulações e que, ao contrário da maioria dos métodos aplicáveis a séries temporais, consegue lidar com atributos nominais.

O método compreende duas fases:

- identificar padrões de previsão usando o algoritmo genético *TimeWeaver*[3];
- gerar uma lista ordenada de regras de previsão a partir dos melhores padrões identificados.

Este método demonstrou bons resultados em problemas de previsão de fenómenos raros. Segundo os autores, isto acontece porque, quer o *TimeWeaver* na primeira fase, quer a escolha de regras feita na segunda fase, fazem uso dos conceitos de *precision* e *recall*.

O carácter adaptativo dos algoritmos genéticos e a grandeza do espaço de procura dos possíveis padrões, esteve na base do desenvolvimento do *TimeWeaver*. Um padrão de previsão consiste numa sequência ordenada de eventos, ligados por primitivas de ordenação que definem restrições temporais. Não procedendo a uma reformulação do problema, as características deste passam a ter uma forte influência na forma como as soluções são apresentadas.

Em cada iteração, o algoritmo genético constitui uma população definida por um conjunto de indivíduos, que representa o conjunto de soluções tidas como as melhores no momento. Cada indivíduo é um padrão de previsão e é inicializado com apenas um evento com valores aleatórios para metade dos atributos que o descrevem. Os operadores de *crossover* e *mutation* foram definidos de forma a dar origem a padrões mais específicos ou mais gerais e com durações variadas. Estão aqui subjacentes as noções de *precision* e *recall*. É tendo em conta a importância dessas medidas, que o operador de *selecção* usa a *F-measure* apresentada na Equação 3.6. Em cada iteração, β vai assumindo valores entre 0 e 1, de uma

[3] *http://www.cs.rutgers.edu/~gweiss/timeweaver/*

forma cíclica. Desta forma, são atribuídos diferentes valores de importância à *precision* e ao *recall*. A fim de encorajar a diversidade de soluções, a *selecção* dos indivíduos também recompensa aqueles indivíduos que mais se diferenciam do resto da população.

No fim de cada geração (iteração) são eliminados os padrões (indivíduos) que pior desempenho têm em termos de *precision* e *recall*. Com este procedimento, ao longo da execução do algoritmo, os eventos irrelevantes acabarão por "morrer" reduzindo o espaço de procura.

Após a execução do algoritmo genético por um certo número de gerações, obtém-se uma família de padrões de previsão. Torna-se então necessário, identificar o melhor subconjunto de padrões como solução. Para tal, os valores de *precision* e *recall* de cada padrão são retornados pelo *TimeWeaver* e posteriormente ordenados do padrão mais preciso para o menos preciso. O melhor subconjunto é avaliado pela curva *precision/recall*, com base em diferentes valores de importância da *precision* face ao *recall* para prevenir o sobre-ajustamento aos dados. Desta forma é derivada uma lista de regras, com as "melhores" (tipicamente as mais precisas) no topo da lista.

O processo proposto, por Weiss (2002); Weiss e Hirsh (1998a, 2000a), para avaliar o progresso do *TimeWeaver*, é correr este algoritmo de criação de regras de previsão a cada 250 iterações do *TimeWeaver*. A *performance* de um algoritmo genético não se deve basear na qualidade do melhor indivíduo gerado; o indivíduo tem de ser avaliado enquanto parte integrante da população que constitui a solução. Este processo permite que as regras sejam guardadas e avaliadas novamente em conjunto com novas regras, assegurando que as regras de melhor qualidade nunca sejam perdidas.

Quando comparado a outros sistemas de indução de regras como o *C4.5rules* (Quinlan, 1993), RIPPER (Cohen, 1995) e Foil (Quinlan e Cameron-Jones, 1993), a proposta de Weiss e Hirsh (1998a, 2000a) apresentou melhor desempenho na previsão de eventos raros, especialmente no que toca ao *recall*.

Ainda na linha da monitorização de actividades e também tendo como ponto de partida um problema semelhante ao apresentado por Weiss (2002); Weiss e Hirsh (1998a, 2000a), Domeniconi et al. (2002) propuseram, mais recentemente, o método: SVD-SVM. Este método segue uma abordagem de classificação para ajudar a administração de redes, prevendo eventos críticos ou fatais para o seu funcionamento. Estes constituem os eventos-objectivo e são tipicamente eventos de frequência rara. O objectivo é identificar situações que levem à ocorrência de eventos-objectivo. A informação de input consiste numa sequência histórica de eventos capturada por janelas de tempo.

Dado o instante em que nos encontramos e tendo acesso a um determinado intervalo de tempo passado, queremos ser capazes de identificar se esta é uma situação que reúne as propriedades especificadas por um dos padrões encontrados, que levam à ocorrência do evento-objectivo num intervalo de tempo futuro (tempo de *monitoring*), após um tempo de *warning*.

Para cada evento-objectivo é produzido um conjunto de treino, com exemplos positivos e negativos. Um exemplo positivo é criado por cada ocorrência do evento-objectivo na janela de *monitoring* anterior à ocorrência do evento e à sua janela de *warning*. Os exemplos negativos são criados com base em eventos anteriores à janela de *monitoring* do evento-objectivo.

Em contraste com a ideia proposta por Weiss (2002); Weiss e Hirsh (1998a, 2000a), a ideia aqui é explorar um modelo de classificação para resolver o problema de previsão. Os padrões são embebidos em exemplos através da análise de co-ocorrência de eventos nos dados. Com este procedimento deixa de ser necessária a procura no espaço de soluções possíveis. A abordagem de classificação aqui adoptada tira partido das propriedades estatísticas do conjunto de eventos descritos nos dados.

A proposta SVD-SVM consiste em duas fases principais. Numa primeira fase, partindo de um conjunto de atributos que codificam a informação relativa a cada tipo de evento, é feita uma análise das propriedades estatísticas dos dados, a fim de poder agrupar alguns deles segundo um critério de correlação. Este critério tem a ver com a co-ocorrência de eventos numa mesma janela de *monitoring*. O método SVD (*Singular Value Decomposition*), permite embeber essa informação de correlação temporal de evento num número mais reduzido de atributos, que se revelam como os mais relevantes no subsequente processo de classificação. O tamanho da janela de *monitoring* terá de ser dependente do domínio. Ele determina a extensão de tempo pela qual um evento permanece correlacionado com eventos anteriores. Na segunda fase, os vectores de atributos resultantes são então usados para treino por SVM (*Support Vector Machines*), de forma a obter uma boa *accuracy* nas previsões dos eventos-objectivo.

O classificador resultante deste processo baseia-se num número menor de exemplos e num conjunto mais reduzido de atributos, tornando a classificação mais eficiente.

Os SVMs foram desenvolvidos por Cortes e Vapnik (1995) para classificação binária. Este algoritmo procura a separação óptima das duas classes no hiperplano. Quando a separação não é linear, os pontos poderão ser projectados num espaço com mais dimensões onde possam ser de facto linearmente separáveis. Diferentes pesos podem ser associados a diferentes classes. Os pontos encontrados no lado errado da separação, vêm o seu peso decrescido no sentido de reduzir a sua influência. Os SVMs têm sido aplicados com sucesso a conjuntos de dados que apresentam certas propriedades estatísticas. Nomeadamente, espaço de atributos de grande dimensão, com alguma redundância para a determinação da classe e com uma distribuição pouco uniforme.

Experiências realizadas no trabalho de Domeniconi et al. (2002), demonstraram que o algoritmo SVD-SVM acaba por revelar uma melhor *performance* no que toca a eventos raros, comparativamente ao C4.5 e ao SVM (ainda que a diferença com este último, não seja muito significativa).

3.4 A detecção de subconjuntos de dados interessantes

Existem alguns estudos que muito embora não tenham por objectivo a previsão de fenómenos raros, estão de certa forma bastante relacionados com este tipo de problemas. Este é o caso dos problemas de detecção de subconjuntos de dados interessantes. Este tipo de problemas seguem abordagens que poderão de alguma forma fornecer mais algumas pistas para a resolução do problema de previsão de fenómenos raros. Afinal, um fenómeno raro, será certamente um acontecimento interessante, embora o inverso nem sempre se aplique.

Buja e Lee (2001) propuseram um método que em vez de modelar o conjunto de treino de uma forma global, tem por objectivo a modelação de um subconjunto de dados interessantes.

Porque as árvores de classificação ou regressão constituem modelos eficazes e de grande interpretabilidade, foram os métodos escolhidos para o estudo de Buja e Lee (2001). O objectivo dos autores consiste não só em detectar subconjuntos de dados interessantes mas também obter um caracterização tão simples quanto possível destes.

A ideia subjacente às árvores é repartir de forma recursiva o conjunto de dados da melhor maneira tal que a previsão seja a mais acertada possível de acordo com os objectivos estabelecidos.

São os critérios segundo os quais é feita a escolha de testes sobre atributos em cada nó da árvore, que determinam a partição dos dados e portanto a estrutura da árvore.

A procura feita pelas árvores é uma procura *greedy*. Em cada passo interessa escolher o teste, do conjunto de testes possíveis para todos os atributos, que melhor se ajuste aos objectivos e ao subconjunto de dados de que se dispõe para constituir o próximo nó da árvore. Este processo é repetido até que não seja obtido nenhum ganho significativo com a criação de mais um nó.

No caso das árvores binárias, as escolhidas para este estudo, cada teste dá origem a apenas dois ramos. Os testes são expressos em termos de relações de ordem ou de pertença a um determinado subconjunto de valores.

Normalmente, na procura do melhor teste, está envolvida uma medida de impureza, como a taxa de erro, a entropia, etc.. Quão mais impuros forem os dois subconjuntos originados por um dado teste face aos objectivos que se pretende, pior será o valor heurístico do mesmo. Os testes escolhidos em cada nó são aqueles que minimizam o critério de impureza. Habitualmente, o critério de impureza estabelece algum tipo de compromisso com os subconjuntos do ramo esquerdo e direito do nó. Isto acontece porque, na maior parte dos casos, o interesse é obter uma árvore com uma *performance* globalmente boa. Cada sub-árvore constituída deve ter um bom desempenho e mais ou menos equiparado ao das outras sub-árvores.

Um dos critérios mais comuns para o crescimento de árvores é o critério usado pelo CART desenvolvido por Breiman et al. (1984). No caso das árvores de regressão, este critério tem em conta a variância da variável que se pretende prever em cada partição do conjunto dados, ponderada pelo tamanho da partição.

Seja y a variável que pretendemos prever e N_L e N_R o número de exemplos contidos no ramo esquerdo e direito de um nó, respectivamente. O critério utilizado para a escolha do melhor teste é definido pela Equação 3.10, fazendo uso da média e variância na partição esquerda e direita (3.11,3.12). A qualidade de um teste corresponde à diferença entre a impureza do nó e a soma ponderada da impureza de cada uma das partições originadas. Assim sendo, o melhor teste será aquele que maximiza o critério apresentado na Equação 3.10, por ser aquele que mais reduz o erro quadrado médio do nó, isto é, a impureza do nó.

$$crit_{LR} = \frac{1}{N_L + N_R}[N_L \hat{\sigma}_L^2 + N_R \hat{\sigma}_R^2] \tag{3.10}$$

$$\hat{\mu}_L = \frac{1}{N_L}\sum_{n \in L} y_n \qquad \hat{\sigma}_L^2 = \frac{1}{N_L}\sum_{n \in L}(y_n - \hat{\mu}_L)^2 \tag{3.11}$$

$$\hat{\mu}_R = \frac{1}{N_R}\sum_{n \in R} y_n \qquad \hat{\sigma}_R^2 = \frac{1}{N_R}\sum_{n \in R}(y_n - \hat{\mu}_R)^2 \tag{3.12}$$

Este critério é sensível a *outliers* na variável objectivo (y), na medida em que a presença destes exponencia o erro. Por vezes, a distribuição dos dados favorece o aparecimento dos chamados *end-cut splits*, isto é, cortes no subconjunto de dados que isolam uma pequena partição de dados.

Estas situações são dadas como problemáticas no sentido em que inibem o crescimento da árvore de forma global e assim o recurso a atributos que poderão ser mais importantes para a descrição do conceito. Para além disso, conclusões retiradas a partir de um conjunto pequeno prematuramente isolado são consideradas em muitos domínios como dúbias. De facto, na maior parte dos casos onde o objectivo reside em prever a regra e não a excepção, este tipo de cortes acabam por causar um desequilíbrio da árvore e afectar assim a interpretabilidade global do conceito a aprender. Estes cortes poderão ser decorrentes de simples ruído nos dados e não deverão, por isso, ser relevantes. Por esta razão, são evitados sendo a sua pureza penalizada pelo tamanho da partição.

No entanto, no estudo feito por Torgo (2001), experiências revelaram que este tipo de cortes pode melhorar a *accuracy* de uma árvore de regressão e ainda assim não afectar a interpretabilidade do conceito, desde que sejam permitidos em níveis mais baixo das árvores. A ideia é que eles isolarão alguns dos *outliers*, habitualmente encarados como ruído nos dados, evitando o enviesamento das médias contidas nas folhas, diminuindo, assim, erros grandes nas previsões feitas por essas folhas.

Contudo, se o interesse é precisamente detectar as excepções, estes cortes seriam mais valiosos do ponto de vista da interpretabilidade, porém insuficientes. Para ilustrar esse facto, Buja e Lee (2001), imaginaram um cenário de regressão, dando exemplos de duas partições de dados com distribuições semelhantes às apresentadas na Figura 3.10.

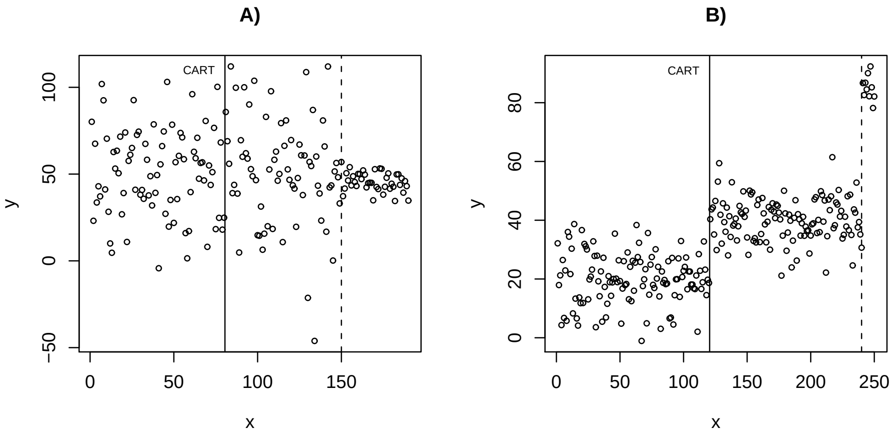


Figura 3.10: Exemplo de *splits* escolhidos pelo CART.

Em ambos os casos, os testes escolhidos pelo CART para as partições, embora enviesados, não conseguiriam isolar o subconjunto de dados mais interessante. No caso A) o teste escolhido pelo CART seria $x = 80.5$. Tal aconteceria porque o critério de impureza tem em conta a variância das duas partições. Isolar o subconjunto mais denso daria origem a dois subconjuntos com variâncias significativamente diferentes. Apesar de inevitavelmente enviesado o teste escolhido tentaria, decerto, equilibrar as variâncias. No caso B), o teste escolhido pelo CART seria $x = 120.5$. Os valores extremos da variável objectivo dão origem a uma média muito enviesada e consequentemente a um erro quadrado médio exponenciado. Sem poder escapar a estes valores extremos e tentando minimizar o erro, o CART inclui apenas alguns dos valores mais baixos, mais comuns na partição. Mas não muitos, apenas os necessários para que a variância e o tamanho da partição não seja significativamente diferente.

O critério do CART não foi concebido para a detecção de subconjuntos de dados interessantes. É por isso natural que não se mostre como o mais adequado face a esse objectivo. Nos exemplos apresentados na Figura 3.10, idealmente seriam escolhidos os testes à volta de $x = 150$ para o caso A) e à volta de $x = 240$ para o caso B). Buja e Lee (2001) propõem então que, para responder a este objectivo, a escolha dos testes para cada nó deve ser feita de modo a que seja obtida a menor impureza possível em apenas uma das duas partições. Interessa maximizar a pureza de uma das partições independentemente da impureza apresentada pela outra partição. Pretende-se identificar subconjuntos de dados interessantes, ainda que pequenos, em detrimento do objectivo de equilibrar a variância em ambos os subconjuntos gerados por um teste, como é feito pelo CART.

Com este procedimento, iremos começar por isolar conceitos, deixando para o fim a descrição dos casos mais comuns, mais gerais e menos interessantes face ao objectivo. Assim iremos obter uma árvore com uma estrutura mais desequilibrada, mas também mais interpretável sob o ponto de vista do objectivo estabelecido. Esta estrutura decorre do facto de conjuntos que minimizam valores do critério de impureza, dificilmente voltarem a ser repartidos. Os casos especiais serão logo identificados no topo da árvore, ponto-chave da interpretabilidade da árvore.

Foi através de observações como as feitas a partir da Figura 3.10, que Buja e Lee (2001) acharam que critérios que minimizassem a variância ou extremassem a média numa das partições, iriam de encontro às suas expectativas. A estas duas estratégias deram o nome de *one-sided purity* e *one-sided extremes*, respectivamente.

Buja e Lee (2001) afirmaram que estratégias equivalentes poderão ser aplicadas a árvores de classificação, isto é, o mesmo conceito pode ser transferido do domínio da regressão para o domínio da classificação. A entropia é um dos critérios de impureza, baseado no ganho de informação, usado para a construção das árvores de classificação. Este critério estabelece que quanto menor for a entropia (aleatoriedade) de um atributo, maior será o seu ganho informativo, mais facilmente discernidas ficarão as classes. Contudo, esta critério tenta também balancear o tamanho das duas partições.

Assumindo que existem apenas duas classes, "interessante" e "não interessante" (1 e 0, respectivamente), poderemos descrever um nó pelo tuplo $(N_L^l, N_L^w, N_R^l, N_R^w)$. N_L^l e N_R^l correspondem ao número de exemplos incorrectamente (*"loosers"*) classificados nas partições esquerda e direita, respectivamente. N_L^w e N_R^w correspondem ao número de exemplos correctamente (*"winners"*) classificados nas partições esquerda e direita, respectivamente. De acordo com o objectivo estabelecido, será preferível o nó (20,30,0,50) ao nó (10,40,10,40). Isto porque o primeiro produz uma partição maior de dados completamente pura.

Podemos estabelecer uma correspondência das estratégias *one-sided purity* e *one-sided extremes* para as árvores de classificação da seguinte forma. Se nas árvores de regressão as estratégias baseavam-se em médias e variâncias, aqui as estratégias basear-se-ão unicamente nas probabilidades. Estaremos aqui perante um problema de classificação com apenas duas classes: "interessante" ou "não-interessante". A cada partição estão associadas as probabilidades dessas duas classes.

Sejam p^0 e p^1 as probabilidades associadas às classe 0 e 1, respectivamente.

A estratégia *one-sided purity* traduz-se na escolha do teste que obtenha o menor valor possível, no conjunto das suas duas partições, para essas probabilidades numa das duas classes. Isto é, o mínimo entre p_L^0, p_L^1, p_R^0 e p_R^1, que é equivalente a encontrar o mínimo entre $p_L^0 p_L^1$ e $p_R^0 p_R^1$. Um resultado equivalente pode ser obtido se em vez de procurar o valor mínimo procurarmos o valor máximo, isto porque se numa das partições o valor é mínimo então na outra o valor é máximo.

A estratégia *one-sided extremes* traduz-se na procura do teste que origine uma partição pura da classe "interessante". Interessa-nos o teste que origine o maior valor de probabilidade para essa classe, isto é,o nó com o maior valor para p_L^1 ou p_R^1. Da mesma forma podemos procurar o teste que origine o menor valor de probabilidade para a classe "não--interessante", isto é, o menor valor para p_L^0 ou p_R^0.

Esta abordagem encontra-se resumida na Tabela 3.2.

		Árvores	
		Regressão	Classificação
Critérios	*one-sided purity*	$\min(\sigma_L^2, \sigma_R^2)$	$\min(p_L^0 p_L^1, p_R^0 p_R^1)$
	one-sided extremes	$\max(\hat{\mu}_L, \hat{\mu}_R)$	$\max(p_L^1, p_R^1)$

Tabela 3.2: Critérios *one-sided purity* e *one-sided extremes*.

Em suma, o trabalho apresentado por Buja e Lee (2001) tem por objectivo a criação de modelos mais interpretáveis para a previsão de dados "interessantes". Questões de *pruning* e de profundidade das árvores, isto é, questões de sobre-ajustamento aos dados, são aqui desprezadas. Os nós em níveis mais profundos da árvore serão simplesmente ignorados. As árvores geradas poderão possuir uma má *performance* global em termos de *accuracy*, mas tal facto será irrelevante no sentido em que apenas interessa obter uma boa *performance* num conjunto reduzido dos dados.

Em alguns conjuntos de dados testados foram obtidas árvores mais interpretáveis, quando comparadas às árvores usando vários critérios convencionais. Ainda que involuntariamente, os métodos propostos por Buja e Lee (2001) acabaram por também isolar *outliers* reconhecidos como dados interessantes. No entanto, não são métodos vocacionados para a sua detecção e por isso não constituem modelos para tal.

Na verdade, os métodos propostos por Buja e Lee (2001), nomeadamente no que toca à regressão, têm em vista a identificação de subconjuntos de dados que serão considerados interessantes por possuírem uma variância mínima ou uma média de valores extrema. De facto, nenhuma destas razões passa pela restrição de raridade intrínseca ao conceito de *outlier*. Por essa razão estes métodos poderão falhar a identificação de alguns *outliers*.

O problema dos *end-cut splits* está mais associado a modelos que descrevem a situação mais comum, no entanto pode, igualmente, ser colocado quando se pretende prever a situação mais fora do comum (*outlier*). As partições criadas devem fazer sentido no domínio de aplicação, por isso não deverá, ser isolado um conjunto de pontos qualquer, ainda que este seja pequeno. Por exemplo, no caso do método de *one-sided purity*, a menor variância que poderemos obter é isolando uma partição com apenas um ponto, ainda que o mesmo não possua um interesse "especial" no conjunto de dados. Foi por esta razão que Buja e Lee (2001) impuseram um tamanho mínimo para as partições obtidas. Este limite mínimo para além de requerer um bom conhecimento do domínio, coloca sempre a possibilidade de falhar a identificação de *outliers* que residam em partições que possuem valores extremos de alguma variável.

Por exemplo, no caso A) apresentado na Figura 3.11, sendo quatro o número de elementos mínimo para constituir uma partição, o *outlier* com o valor de 60 nunca seria identificável, pelo menos não a partir da variável x.

O método *one-sided extremes* pode igualmente falhar no que respeita à detecção de *outliers*. Segundo a definição de Hawkins (1980) referida no Capítulo 2, Subsecção 2.1.1, nada nos é dito quanto à posição que os *outliers* devem ocupar na gama de valores do domínio. Isto é, um *outlier* não terá que assumir um valor extremamente baixo ou extremamente alto, basta que seja significativamente diferente. O método *one-sided extremes* isola partições apenas com valores extremos. Se imaginarmos um domínio onde os valores são, regra geral, extremos, os valores medianos constituirão os *outliers*. A Figura 3.11 apresenta no caso B) uma partição nessas condições. Por este método não seria possível isolar, na partição apresentada, os *outliers*.

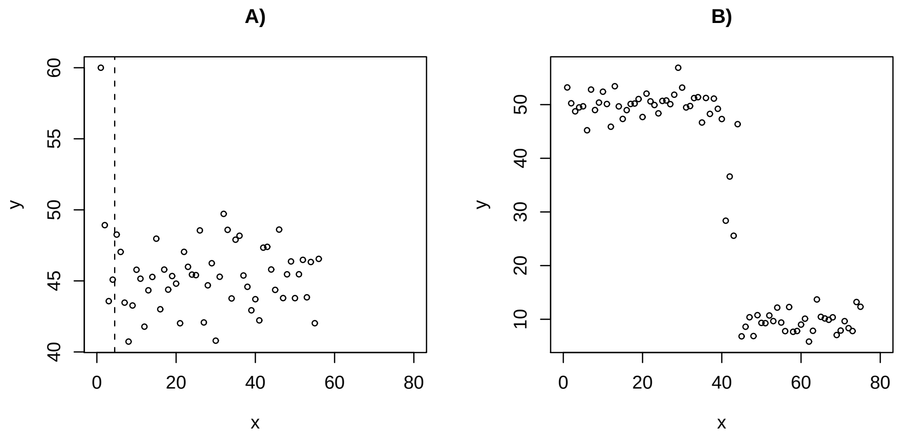


Figura 3.11: Falhas na identificação de *outliers* segundo os critérios propostos por Buja e Lee (2001).

Em suma, a Figura 3.11 fornece dois exemplos que ajudam a diferenciar aquilo que é considerado interessante, sob os parâmetros estabelecidos por Buja e Lee (2001), daquilo em que consiste o conceito de *outlier*.

O problema da detecção de padrões interessantes numa série temporal foi alvo de estudo por Keogh et al. (2002). Definiram como padrão "surpreendente", o padrão cuja frequência na série temporal possa ser significativamente diferente dos restantes padrões anteriormente reconhecidos. Em contraste com as abordagens referidas anteriormente, a ideia aqui consiste em descobrir padrões surpreendentes, sem que haja necessidade de especificar a noção de surpreendente no domínio de aplicação.

Esta abordagem segue uma linha de aprendizagem não supervisionada, mas que procura tal como Fawcett e Provost (1999), Weiss (2002); Weiss e Hirsh (1998a, 2000a), a detecção de padrões sob uma perspectiva temporal.

Contudo, os autores sublinham que o problema abordado nesse estudo, não está directamente relacionado com a detecção de *outliers*. O interesse não é encontrar pontos isolados mas sim padrões isolados cuja frequência seja significativamente superior ou inferior. O interesse não reside apenas nos padrões de frequência rara.

Keogh et al. (2002) propuseram uma noção de surpreendente mais abrangente do que aquela usualmente empregue na análise de séries temporais. Na análise clássica, apenas uma alteração radical na série temporal original seria considerada como algo surpreendente.

Esta definição não tem, contudo, em conta que estas alterações possam ser cíclicas e na realidade não corresponderem a nada de surpreendente. Um exemplo onde este problema se coloca, que é referido por Keogh et al. (2002), são as linhas produzidas por um electrocardiograma. Neste caso, e segundo a noção clássica, o início de cada pulsação seria considerado surpreendente, ao passo que uma paragem cardíaca durante algum instantes adicionais não seria considerado como tal.

A proposta de Keogh et al. (2002), envolve a discretização da série temporal para a aplicação de métodos de processamento sobre *strings* como as *suffix trees* e os modelos de *Markov*. As *suffix trees* constituem um método eficiente de procura de *substrings*. Aqui são usadas para proceder a uma contagem eficiente das frequências dos padrões encontrados. A chamada cadeia de *Markov* é gerada num alfabeto finito e é completamente determinada através das chamadas "probabilidades de transição". O modelo de *Markov* permitirá a previsão da frequência esperada para a ocorrência de padrões ainda não observados. Uma explicação mais detalhada da aplicação destes métodos ao tipo de problemas em análise pode ser encontrada no trabalho de Keogh et al. (2002).

A avaliação do método foi feita com base em duas medidas já referidas na Secção 3.2.1, a propósito das curvas ROC: sensibilidade e selectividade. Também aqui o interesse reside em possuir indicadores máximos destas duas medidas, isto é, uma taxa elevada de TP e uma taxa baixa de FP. Neste contexto, um previsão positiva diz respeito à detecção de um padrão surpreendente na série temporal.

Este tipo de trabalhos vão um pouco mais além do que a clássica detecção de *outliers*. A noção de surpreendente ou interessante é mais geral e independente do domínio. As técnicas e métodos utilizados neste tipo de problemas poderão, contudo, trazer sempre uma mais valia para a concepção de métodos de detecção de *outliers*.

Capítulo 4

Um Método para a Previsão de Valores Extremos Raros

A previsão de fenómenos raros tem em vista a concepção de modelos que consigam descrever o conceito subjacente à ocorrência desse tipo de fenómenos num certo domínio de aplicação. Estes modelos deverão ser capazes de detectar as situações dadas como fora do comum. No fundo pretende-se, dado um conjunto de dados, ser capaz de detectar pontos com frequência muito baixa e que se distinguem por serem significativamente diferentes do resto dos pontos que constituem o conjunto de dados. Os pontos nestas condições são designados de *outliers*.

Nos vários exemplos já mencionados no Capítulo 2, e no Capítulo 3, constatou-se que a maior parte das aplicações de problemas de previsão de fenómenos raros está associada a catástrofes, a situações cuja ocorrência, apesar de rara, pode ser gravosa. Entre as variadas aplicações podemos citar a previsão de catástrofes ecológicas, de alterações bruscas de preços no mercado bolsista, de falhas ou intrusão em grandes redes de computadores, de fraudes bancárias, etc.. Idealmente pretenderíamos ser capazes de prever todas estas e outras situações de emergência, de tal modo que houvesse o tempo necessário para accionar mecanismos de prevenção e minimizar, assim, as consequências da sua ocorrência.

O que em muitos destes casos acontece é que, às situações de emergência raras, estão associados valores extremos de uma determinada variável. Tomemos o exemplo do mercado bolsista: são quedas ou subidas bruscas dos preços que fazem desencadear uma série de acções de compra e venda que poderão destabilizar o mercado, invertendo por completo a sua tendência. É necessário estar atento a todos os factores que possam afectar de forma radical o mercado. Assim como esta, muitas outras aplicações poderiam ser referidas onde o interesse reside na previsão de fenómenos caracterizados quer pela raridade, quer pelos valores extremos que assumem. A amplitude de um *outlier* poderá ser relevante para o domínio de aplicação.

Foi com o objectivo de conseguir fazer a previsão de fenómenos raros de valor extremos, de uma forma precisa, que foi desenvolvido, no âmbito desta tese, um método capaz de lidar com este tipo de problemas. É neste capítulo que se faz uma descrição deste método. Assim, na Secção 4.1 começa-se por descrever um problema concreto, que se enquadra no tipo de problemas que pretendemos resolver, e que serviu de motivação à concepção do método proposto. Na Secção 4.2 é descrito o conceito de valor extremo raro que se usou. Ele poderá ser considerado com um caso especial do conceito de *outlier* já mencionado no Capítulo 2, na Subsecção 2.1.1. Na Secção 4.3 faz-se um sumário do tipo de abordagens que poderiam ser seguidas na resolução deste tipo de problemas, explicitando, na Secção 4.4, a abordagem por nós seguida. Dela referimos os seus objectivos principais e a descrição do método implementado.

4.1 Motivação

A preocupação do impacto que a acção do Homem tem sobre o ambiente e processos biológicos inerentes tem sido, progressivamente, alvo de vários estudos. As descargas de lixos tóxicos nas águas dos rios é uma das questões levantadas por vários ecologistas. Em alguns desses estudos foi possível registar grandes concentrações de alguns tipos de algas que pioravam a qualidade da água de rios, afectando a sua fauna e flora. Por razões ainda desconhecidas, essas algas surgem principalmente no verão em climas temperados e em grandes quantidades dando origem ao fenómeno de *blooms* de algas. Felizmente, estes *blooms* possuem ainda assim um carácter raro. Os *blooms* destas algas reduzem a claridade da água e os níveis de oxigénio. Os peixes do rio acabam por morrer em massa degradando, assim, a qualidade da água. Neste contexto, e porque é dos rios que se retira muita da água que se tornará potável, será de extrema importância a previsão destes *blooms* tentando perceber o que está subjacente à sua ocorrência.

A partir do registo das várias análises químicas feitas à água, interessa ser capaz de perceber quais as componentes químicas que parecem interferir na ocorrência das algas nocivas.

A ideia consiste em ser capaz de prever quando um certo tipo de alga nociva assumirá valores de concentração bastante elevados. Sendo esta ocorrência rara e sendo apenas os *blooms* considerados preocupantes, do ponto de vista ecológico, estamos, pois, perante um problema de previsão de fenómenos raros associados a valores extremamente elevados de uma variável (a ocorrência de algas nocivas).

Para responder a situações como esta, e sendo a variável a prever numérica, será necessário conceber um método de regressão capaz de prever, de uma forma precisa, *outliers* de valor extremo (neste caso elevado). Para além disso, será desejável obter uma descrição compreensível do fenómeno expressa através de outras variáveis que explique a causa dos valores que caracterizam o fenómeno.

Recorrendo a técnicas de *data mining* usando registos históricos deste tipo de problemas, será, em princípio, possível obter modelos que expliquem a ocorrência dos *blooms* das algas nocivas e prever o valor da sua concentração. Isto evita a necessidade de recorrer a análises microscópicas efectuada por especialistas, de modo a inferir o estado biológico da água. Se tal tarefa vier um dia a ser cumprida, ela será certamente mais rápida e mais barata.

Este foi, de facto, um dos motivos que deu origem à realização de uma competição[1] internacional de análise de dados, com vista à resolução de um problema semelhante. Os dados fornecidos para a competição diziam respeito ao resultado de análises químicas feitas a amostras de água retiradas durante aproximadamente um ano e de diferentes rios europeus. Para além de medidas químicas, foram registadas características dos rios como a sua velocidade, o seu tamanho e a estação do ano a que diz respeito a análise. A cada registo foram associados os valores de concentração de sete comunidades de algas diferentes, todas elas nocivas. Na verdade, cada amostra de água diz respeito à agregação de várias amostras de água durante mais ou menos três meses no mesmo rio e durante a mesma estação do ano. Por questões de simplicidade estas agregações foram, igualmente, denominadas de amostras. Em suma, a cada amostra de água eram associados os valores descritos na Tabela 4.1.

Foram recolhidas, analisadas e disponibilizadas para a competição, 340 amostras de água, cada uma delas contendo dezoito valores. Cada amostra de água é descrita por três atributos nominais, referentes às características, e oito atributos contínuos, referentes aos parâmetros químicos medidos. Os restantes sete valores dizem respeito à concentração das

[1] http://www.erudit.de/erudit/competitions/ic-99/

Características	**Parâmetros Químicos**		**Algas Nocivas**
- estação do ano (season)	- valor máximo de PH (mxPH)	- valor médio de Azoto Amoniacal (NH4)	- $a_1 \rightarrow a_7$
- tamanho do rio (size)	- valor mínimo de Oxigénio (mnO2)	- valor médio de Ortofosfato (oPO4)	
- velocidade do rio (speed)	- valor médio de Cloro (Cl)	- valor médio de Fosfato (PO4)	
	- valor médio de Nitratos (NO3)	- valor médio de Clorofila (Chla)	

Tabela 4.1: Atributos associados a cada amostra de água.

sete comunidades de algas nocivas. Das 340 amostras de água, 200 constituíram o conjunto de treino e 140 foram deixadas para teste, sem a informação de valores de concentração de qualquer uma das sete algas.

Neste problema interessa prever, de uma forma tão precisa quanto possível, os valores das raras e elevadas concentrações de algas nocivas, isto é, os *blooms* dessas algas. No fundo, acabam por ser formulados sete problemas semelhantes, cada um com o objectivo de prever a concentração de uma das sete algas.

4.2 Conceito de valor extremo raro

Tomando a classe de problemas a que o exemplo apresentado na secção anterior pertence, estamos interessados em obter modelos capazes prever com exactidão valores raros e extremos. O carácter raro dos registos em que ocorrem valores significativamente diferentes na variável numérica que pretendemos prever e que descrevem o fenómeno a aprender, concede-lhes o nome de *outliers* univariados.

Como já foi referido no Capítulo 2, o conceito de *outlier* teve origem no campo da Estatística. Foi, portanto, à Estatística que recorremos, para obter uma formalização da noção de *outlier* que pretendemos.

Os gráficos têm um grande poder de resumo da informação, servindo como uma poderosa ferramenta para a análise de dados. Eles transmitem, de uma forma empírica, alguns conceitos estatísticos.

Tal como refere Cleveland (1993), o *box-and-whisker plot*, mais frequentemente denominado por *box plot*, inventado por J. Tukey, consiste num gráfico que sumaria de uma forma mais ou menos simples o conjunto de dados, expressando a distribuição de valores de uma variável. Este gráfico é também denominado por "caixa de bigodes" por se apresentar sob a forma de uma caixa, cujos limites são o 1° quartil e o 3° quartil. Esta caixa é dividida por uma linha que representa a mediana. Existem também dois eixos, ou "bigodes", ligados à caixa que se estendem até ao valor mínimo e máximo dos dados, excluindo os valores discrepantes (*outliers*). Este gráfico mostra a forma como o conjunto de dados se espalha em torno da mediana, através de cinco conceitos estatísticos importantes.

Um exemplo de um *box plot* é apresentado na Figura 4.1.

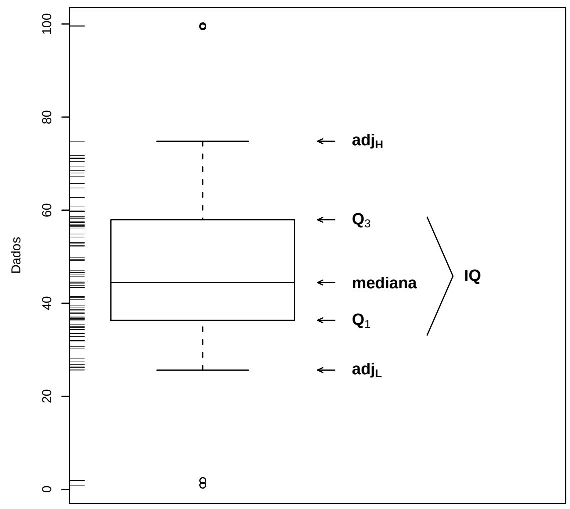


Figura 4.1: *Box plot.*

Para melhor compreender o gráfico, convém relembrar os conceitos estatísticos expressos no gráfico.

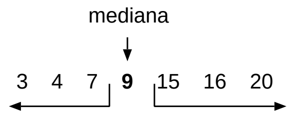
- Q_1 (1° quartil): valor abaixo do qual estão 25% dos valores da variável.
- Q_3 (3° quartil): valor abaixo do qual estão 75% dos valores da variável.
- IQ (intervalo inter-quartil): intervalo onde estão situados 50% dos dados, isto é, $Q_3 - Q_1$.
- adj_L (*adjacent Low*): ponto adjacente inferior, isto é, o menor valor superior ou igual a $Q_1 - 1.5 \cdot IQ$; é também denominado de *lower whisker*.
- adj_H (*adjacent High*): ponto adjacente superior, isto é, o maior valor menor ou igual a $Q_3 + 1.5 \cdot IQ$; é também denominado de *upper whisker*.

Diferentes convenções são, muitas vezes, utilizadas para o cálculo dos quartis. Duas dessas convenções são o método original de Tukey e o método de Moore e McCabe. Basicamente a diferença reside na opção de incluir ou não a mediana no conjunto de dados quando se acha Q_1 e Q_3.

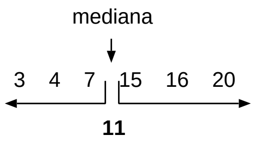
A mediana de um conjunto de dados é o valor que se situa no centro da distribuição dos dados, ordenados por ordem crescente ou decrescente. Ou seja, é o valor, tal que o número de elementos inferior ou superior a ele é exactamente igual. O cálculo da mediana varia de acordo com o número (n) de elementos que compõem o conjunto.

Exemplo 1: Cálculo da mediana.

- Se n é ímpar: a mediana coincide com o valor central do conjunto ordenado de dados.



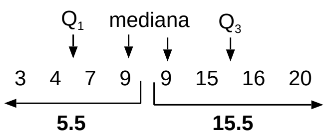
- Se n é par: a mediana é a média aritmética dos dois elementos centrais do conjunto ordenado de dados.



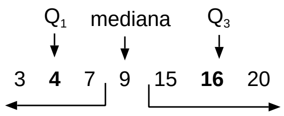
Abaixo de Q_1 estão 25% dos dados e acima de Q_3 estão também 25% dos dados. Q_1 e Q_3 correspondem à mediana da metade inferior e superior dos dados, respectivamente. Para o cálculo de Q_1 e Q_3 e no caso de n ser ímpar, é tão legítimo incluir a mediana do conjunto de dados, como excluí-la.

Exemplo 2: Cálculo de Q_1 e Q_3.

- Método de Tukey: inclui a mediana. Aqui são, muitas vezes, atribuídos nomes de *lower hinge* e *upper hinge* a Q_1 e Q_3, respectivamente.



- Método de Moore and McCabe: não inclui a mediana.



Contudo, e considerando que se está a fazer um sumário de um grande conjunto de dados, as diferenças dos valores obtidos para Q_1 e Q_3 usando cada um destes dois métodos não será significativa. No caso da implementação do *box plot* por nós utilizada[2] é usado o método de Tukey.

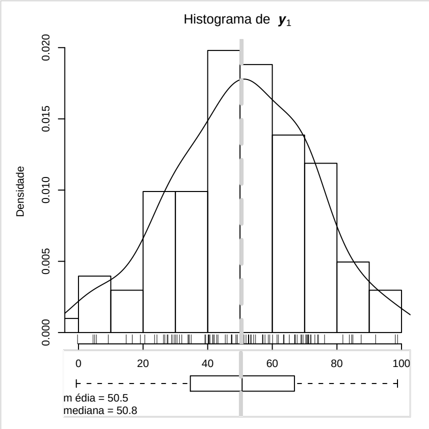
Uma vez encontrados os cinco "números-chave" para o conjunto de dados, são considerados *outliers* de valor extremo todos os pontos cujo valor se encontra abaixo ou acima de adj_L e adj_H, respectivamente. Nos *box plots* eles são representados por círculos. No caso da Figura 4.1 temos dois *outliers* de valor extremamente baixo e dois *outliers* de valor extremamente alto.

Nas Figuras 4.2 (a) e (b) são apresentadas duas variáveis, cada uma delas com o seu respectivo histograma, função de densidade de probabilidade e *box plot*. Conforme se pode constatar, a variável y_1 segue um distribuição normal, ao passo que a variável y_2 possui uma distribuição "enviesada" pela presença de *outliers*. A presença de vários *outliers* distorce o valor da média que deixa assim de poder ser considerada como o valor mais representativo do conjunto de valores.

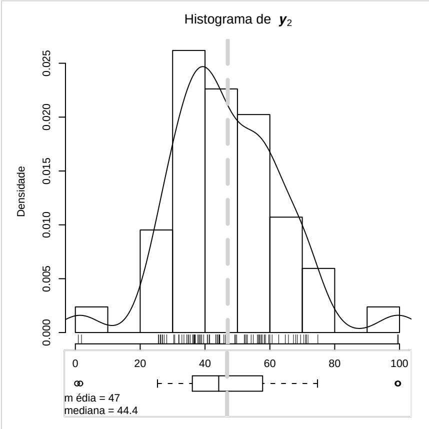
Aliás, tal como já referido no Capítulo 3, Secção 3.1, assumindo uma distribuição normal para os dados, é através a observação da curva normal que se podem detectar a presença *outliers*.

No conjunto de *outliers*, estabeleceu-se dois "graus" diferentes: todos os pontos que sejam superiores (inferiores) a adj_H (adj_L), mas que não sejam superiores (inferiores) a $Q_3+1.5 \cdot IQ$ ($Q_1 - 1.5 \cdot IQ$), dizem-se *"suspect outliers"*. Pontos que estejam fora desses limites, isto é, que sejam superiores (inferiores) ou iguais a $Q_3 + 3 \cdot IQ$ ($Q_1 - 3 \cdot IQ$) dizem-se *"highly suspect outliers"*. No entanto, e por que nos interessa tentar prever todos os *outliers* e com alguma precisão no seu valor, não fazemos aqui essa distinção. Estabelecemos apenas dois tipos de outliers: o conjunto de *outliers* com valor extremo baixo (O_L) e o conjunto de

[2] Para o nosso estudo foi usada a implementação de *box plot* feita pela linguagem R (www.R-project.org).



(a) variável s/ *outliers* extremos



(b) variável c/ *outliers* extremos

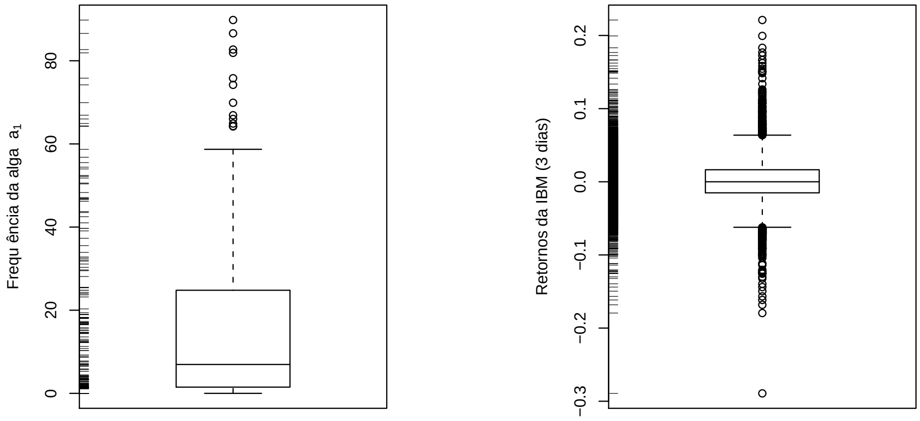
Figura 4.2: Histogramas de duas variáveis sem e com *outliers* extremos.

outliers com valor extremo elevado (O_L). A reunião destes dois conjuntos fornece-nos o conjunto de *outliers* O (*c.f* Equação 4.1).

$$\begin{aligned} O &= O_L \cup O_H \\ O_L &= \{y \in D \mid y < adj_L\} \\ O_H &= \{y \in D \mid y > adj_H\} \end{aligned} \tag{4.1}$$

Dependendo do domínio de aplicação do problema, poderemos ter O_L ou O_H como conjunto vazio. Todas as aplicações em que O é um conjunto vazio não nos interessam: pretendemos prever *outliers* de valor extremo.

Na Figura 4.3 temos o exemplo de duas aplicações já referidas com dois tipo de *outliers* diferentes. O objectivo de ambas consiste na previsão de *outliers* de valor extremo. A Figura 4.3 (a) apresenta os valores de frequência de uma das sete algas nocivas do problema referido na secção anterior; *outliers* representam os *blooms* de algas e assumem por isso apenas valores extremos elevados. A Figura 4.3 (b) apresenta os retornos de três dias com os preços de fecho da IBM ao longo de um certo período de tempo; os *outliers* acontecem sempre que há uma descida ou subida brusca de preços e que devem desencadear alguma acção especial.



(a) variável com *outliers* extremos apenas do conjunto O_H

(b) variável com *outliers* extremos dos conjuntos O_H e O_L

Figura 4.3: *Box plots* de duas variáveis.

4.3 Abordagens relacionadas com a previsão de valores extremos raros

Desde a detecção de *outliers* até à detecção de subconjuntos de dados interessantes, passando pela classificação de fenómenos raros e monitorização de actividades para a detecção de anomalias, o conjunto de abordagens propostas para a resolução de cada um destes problemas são bastante variadas, tal como referido no Capítulo 3. Embora sendo díspares, todas elas se encontram, de alguma forma, relacionadas com o tipo de problema alvo de estudo nesta tese e, nesse sentido, forneceram-nos alguma pista quanto à direcção a tomar na sua resolução.

Fazendo uma reflexão sumária dos trabalhos recolhidos, podemos constatar que, embora ainda sendo uma área relativamente pouco explorada, o estudo de problemas relacionados com eventos raros, isto é *outliers*, têm vindo cada vez mais a suscitar interesse nas mais variadas áreas de aplicação.

Relembremos o problema, descrito na Secção 4.1, que nos serviu de motivação à concepção de um método vocacionado para a sua resolução: a previsão da ocorrência de *blooms* de algas nocivas nas águas dos rios. A ocorrência destes *blooms* é registada por uma frequência elevada anormal de certas algas e é, felizmente, rara. É neste contexto que podemos incluir este problema numa classe mais geral de problemas: problemas de previsão de fenómenos raros.

Do conjunto de abordagens *"standard"* feitas a este tipo de problemas, a abordagem que mais está relacionada com o nosso problema, terá a ver com a detecção de um padrão irregular, de uma anomalia, no registo da actividade de um sistema ao longo do tempo. No fundo, o que se pretende neste tipo de problemas é, com base num registo contínuo de dados, descobrir algo de surpreendente e tentar aprender a razão da sua ocorrência para que mais tarde seja possível antever uma situação fora do que é considerado normal no sistema monitorizado. Porque se trata de um problema de previsão, a correlação temporal está intrínseca aos dados que constituem este tipo de problemas. Uma previsão só fará sentido quando incluída num período de previsão que lhe está associado. O nosso problema está, desta forma, relacionado com a monitorização de actividades.

É importante não desprezar a correlação temporal dos dados e, se o método de aprendizagem não é sensível a essa correlação, a informação temporal deve ser embutida nos dados, recorrendo para tal a diferentes técnicas de *data mining* (Dietterich, 2002). Desta forma, é constituído um novo conjunto de exemplos, o qual será submetido a métodos de aprendizagem automática.

A questão que se pode colocar no que se refere à aprendizagem, é se esta se trata de uma aprendizagem supervisionada ou não. A resposta está intimamente ligada ao domínio de aplicação: se neste existe o conhecimento à priori daquilo que constitui um fenómeno raro. No caso desse conceito ser desconhecido, então técnicas de *outlier mining*, como as referidas na no Capítulo 3, Secção 3.1 serão aplicáveis. Mas, sendo nosso objectivo prever concentrações elevadas e raras de um certo tipo de algas, o fenómeno a aprender é univariado e sobre ele poderá ser aplicado o conceito estatístico de *outlier* de valor extremo, já referido na secção anterior. Nesse caso, podemos identificar à priori os exemplos cujos valores de concentração de uma determinada alga correspondem a *blooms*, isto é, aquilo que pretendemos prever. Assim sendo, serão aplicados métodos de aprendizagem supervisionada.

Na grande parte das abordagens estudadas, uma vez constituído o conjunto de exemplos, procede-se à aplicação de métodos de classificação supervisionada. Sendo ou não um problema colocado num contexto de uma monitorização, ele passa a reduzir-se a um problema de classificação com apenas duas classes: a classe rara, e portanto minoritária, e a classe que engloba todos os restantes casos, dita a classe maioritária. O conjunto de exemplos associados à classe rara (*outliers*) descrevem o fenómeno que se pretende aprender. O objectivo consiste em conceber um modelo suficientemente sensível na detecção da classe rara, no sentido de não falhar a detecção de um *outlier*, e suficientemente selectivo, no sentido de conseguir discriminar bem aquilo que é um *outlier* do que é uma situação comum. Este tipo de problemas possui um conjunto de características que fazem dele um problema de classificação especial: as duas classes existentes serão naturalmente desequilibradas, a sensibilidade e selectividade exigidas ao modelo a conceber implicam o uso de medidas de avaliação vocacionadas para tal.

Particularidades tornam as abordagens de classificação referidas não aplicáveis ao nosso problema, pelo menos não directamente, essencialmente pelo facto de estarmos perante a previsão de um valor numérico. Na verdade, poderíamos adaptar o nosso problema a um problema de classificação, onde uma vez reconhecidos os valores numéricos *outliers*, eles seriam simplesmente rotulados como tal, constituindo a classe rara. No entanto, convém não esquecer que não estamos apenas interessados em prever *outliers*, queremos prever com alguma exactidão os seus valores. No tipo de aplicações a que pretendemos dar resposta, a magnitude dos valores previstos pode ser importante na medida em que pode levar ao desencadeamento de diferentes acções. A importância da magnitude dos valores na previsão, e em particular de valores raros extremos, é constatável em vários domínios. No mercado bolsista, por exemplo, uma subida ou descida de 10% de retorno do investimento deverá, certamente, levar a diferentes acções daquelas que seriam tomadas perante uma subida ou descida de 2%, ainda que ambos correspondam a retornos de valor especialmente extremo.

É da natureza do conceito de *outlier* assumir valores significativamente diferentes dos restantes valores. Com base no conceito de *outlier* extremo formalizado na secção anterior, poderemos afirmar ainda a existência de dois grupos de *outliers* especialmente díspares: aqueles que pertencem ao subconjunto O_L e a O_H, segundo a definição apresentada na Equação 4.1. Para que dois *outliers* sejam efectivamente diferentes, basta que pertençam a extremos opostos da gama de valores assumida pela variável que descreve o fenómeno. A fim de tentar adaptar o nosso problema a um problema de classificação, uma solução seria constituir não uma, mas duas classes de *outliers* extremos: classe rara de valores extremamente elevados e classe rara de valores extremamente baixos. Mas ainda assim poderíamos perder muito em termos de precisão: poderão existir dois *outliers* que pertencendo à mesma classe, isto é, ao mesmo extremo, tenham valores bastante diferentes. Esta situação poderá ser drástica se a amplitude do *outlier* determinar a decisão a tomar. Para colmatar esse facto poderíamos subdividir mais essas duas classes de acordo com intervalo de valores, mas isso iria agravar o problema da aprendizagem sobre um conjunto de dados com um desequilíbrio acentuado de classes como referido na Secção 3.2 do Capítulo 3. Posto isto, a inadequação de técnicas de classificação levou-nos a optar por uma abordagem de regressão.

O objectivo de prever valores raros e extremos, aproximou-nos das abordagens propostas por Buja e Lee (2001). De facto, não estamos perante um problema de regressão, dito *standard*, onde se pretende prever todos os valores assumidos por uma variável objectivo de uma forma globalmente boa. Estamos interessados num certo subconjunto de valores assumidos por essa variável: valores raros e extremos. Neste contexto, o objectivo consiste em modelar particularmente bem um certo subconjunto de dados interessante. Convém, contudo, relembrar que os métodos propostos por Buja e Lee (2001) são vocacionados para a detecção de subconjuntos interessantes sob uma perspectiva por eles estabelecida: variância mínima ou extremidade de valores. Destes dois métodos talvez o segundo pudesse dar alguma resposta ao nosso problema, mas o que é facto é que na explanação das suas abordagens, Buja e Lee (2001) não associam de forma directa o conceito de raridade ao conceito de interessante, restrição essencial ao nosso problema. Por esta razão, tal como já referido na Secção 3.4 do Capítulo 3, estes métodos poderão falhar na detecção de *outliers*. Não deixam, no entanto, de ser métodos que nos suscitaram algum interesse. Os fenómenos raros, não deixam de ser interessantes e, por essa razão, abordagens concebidas para a detecção de conjunto de dados interessantes podem ser úteis. A abordagem de Buja e Lee (2001) foi tomada como ponto de partida para a concepção de um método vocacionado para a resolução do nosso problema.

4.4 Abordagem seguida

Formalizando o nosso tipo de problema alvo e sendo D um conjunto dados constituído por n casos, $\{\langle \mathbf{x}_i, y_i \rangle\}_{i=1}^{n}$, onde $\mathbf{x}_i$ é um vector de p variáveis contínuas ou discretas, y_i é um valor da variável objectivo Y que descreve o fenómeno raro a prever. Dado que a nossa variável objectivo é contínua, estamos pois perante um problema de regressão.

4.4.1 Objectivos

O estudo desta tese centrou-se na concepção de um método vocacionado para aplicações de *data mining* que apresentem as características de um problema de previsão de fenómenos raros de valor extremo.

Os nossos objectivos contemplam a obtenção de modelos interpretáveis e capazes de prever com alguma precisão valores raros e extremos de Y, definidos segundo os conceitos estatísticos referidos na Secção 4.2.

4.4.2 Critérios de avaliação

A noção de relevância está relacionada com o valor de utilidade associado a uma reposta. Ela está na base de medidas de avaliação, que tiveram origem na área de *Information Retrieval*: *precision* e *recall*. Tal como já foi referido no Capítulo 3, Subsecção 3.2.1, estas medidas avaliam quão útil é um resultado na medida em que expressam o seu valor tendo em conta quais são os valores relevantes retornados e que parte eles constituem no conjunto dos valores retornados. Estas medidas acabam por ser aplicáveis a contextos onde o que se pretende prever é raro. A raridade do fenómeno atribui-lhe o carácter de relevante e é sobre ele que o modelo se deve concentrar pois é lá que reside o seu sucesso. *Precision* e *recall* são as medidas de avaliação mais importantes a reter neste tipo de problemas. O modelo será tão melhor quão maior for o número de fenómenos raros que consegue prever correctamente e quão maior for a sua cobertura sobre esses mesmos fenómenos.

Ao optar por manter o problema como sendo um problema originalmente de regressão, colocou-se-nos um novo obstáculo: teríamos de transpor os conceitos subjacentes às medidas de *precision* e *recall*, concebidas sob uma perspectiva de classificação, adequadas à previsão de situações raras, para uma perspectiva de regressão. Fazendo uso destas medidas, o nosso modelo fará, idealmente, as escolhas óptimas.

De acordo com as definições convencionadas para estas medidas no âmbito da classificação tínhamos, tal como definido pelas Equações 3.2 e 3.3 do Capítulo 3, que:

$$precision = \frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FP}} \qquad recall = \frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FN}}$$

No contexto do nosso problema, e de uma forma informal, redefinimos as medidas de *precision* e *recall* da seguinte forma:

- a *precision* é definida como a proporção de *outliers* extremos que foram previstos correctamente pelo modelo;
- o *recall* é definido como a proporção de *outliers* extremos existentes no domínio que foram previstos como tal, isto é, cobertos pelo modelo.

Formalizando estas noções temos que sendo $\hat{Y}$ o conjunto de previsões $\hat{y}$ feitas pelo nosso modelo e $\hat{Y}_O$ as previsões para os casos *outliers* extremos O obtemos:

$$precision = \frac{\mid \{\hat{y} \in \hat{Y}_O \mid (y \in O_H \wedge \ \hat{y} > adj_H) \vee (y \in O_L \wedge \hat{y} < adj_L)\} \mid}{\mid \{\hat{y} \in \hat{Y} \mid \hat{y} < adj_L \vee \hat{y} > adj_H\} \mid} \tag{4.2}$$

$$recall = \frac{\mid \{\hat{y} \in \hat{Y}_O \mid (y \in O_H \wedge \ \hat{y} > adj_H) \vee (y \in O_L \wedge \hat{y} < adj_L)\} \mid}{\mid O \mid} \tag{4.3}$$

No entanto e apesar de esta ser a correspondência directa dos conceitos de *precision* e *recall* para o contexto do nosso problema, a definição de *precision* da Equação 4.2 não responde, de facto, aos objectivos por nós estabelecidos. Tomemos o exemplo de duas previsões ($\hat{y}_a$ e $\hat{y}_b$) feitas para um mesmo *outlier* de valor extremo, apresentado na Figura 4.4

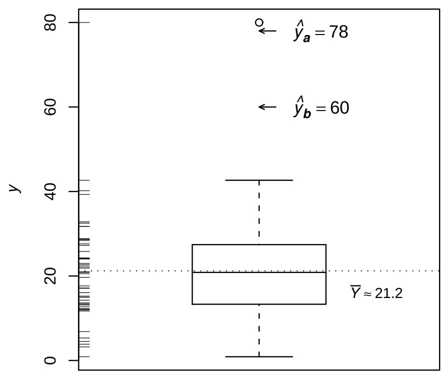


Figura 4.4: Exemplo de duas previsões feitas para um *outlier* de valor extremo.

Para ambas as previsões, $\hat{y}_a$ e $\hat{y}_b$, teríamos um valor de *recall* igual a 1 e um valor de *precision* também igual 1. Contudo, apesar da definição de *precision* (*c.f.* Equação 4.2) não o demonstrar, é visível que a previsão $\hat{y}_a$ é bem mais precisa do que a previsão $\hat{y}_b$. Pretendemos que a diferença entre estas duas previsões seja, de facto, contabilizada. Caso contrário não faria sentido tratar este problema segundo uma abordagem de regressão.

Assim, procedemos à reformulação da definição de *precision*, aproximando-a mais do conceito de regressão. A nova definição tem em conta a distância das previsões aos valores verdadeiros, no conjunto dos *outliers* extremos.

As medidas MAD (*Mean Absolute Deviation*) e MSE (*Mean Square Error*) são duas das medidas de avaliação mais usadas no âmbito da regressão. Ambas as medidas tiveram origem no campo da estatística, tendo como intuito definir quão melhor é um estimador δ para um conjunto de valores. Se ele for uma boa medida de centralidade, ele representa a distribuição dos valores, melhor do qualquer outro valor. A melhor medida de centralidade será aquela que minimiza as medidas MAD ou MSE, definidas pelas Equações 4.4 e 4.5, respectivamente.

$$MAD = \frac{1}{n}\sum_{i=1}^{n} |\delta - y_i| \tag{4.4}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (\delta - y_i)^2 \tag{4.5}$$

Um modelo de regressão é um estimador: ele tenta estimar o valor associado a cada um dos exemplos. O principal objectivo de um método de regressão é conceber o "melhor" modelo segundo uma medida da estimativa do erro. Neste contexto, MAD e MSE podem fornecer uma estimativa da capacidade de previsão de um modelo. Dado que estamos perante um cenário de regressão, é possível estimar o erro associado às previsões feitas por um modelo através da diferença destas para os valores reais de um conjunto de exemplos que constitui uma amostra da população que descreve o fenómeno a prever.

Tal como referido por Torgo (1999), estas medidas quando aplicadas à avaliação de modelos de regressão, servem diferentes propósitos. Se na previsão de valores for aceitável cometer alguns erros extremos, desde que nos aproximemos na maior parte das vezes do valor real, a medida MAD é a mais adequada. Se, pelo contrário, for crucial não cometer erros extremos, mesmo que possam existir erros pequenos e frequentes, então a medida MSE deverá ser preferível. Isto acontece porque o MSE utiliza o quadrado das distâncias das previsões aos valores reais. Desta forma, grandes distâncias serão amplificadas relativamente às pequenas distâncias.

No contexto do nosso problema, o MSE parece ser a medida mais adequada. Preferencialmente, não queremos cometer grandes erros, mesmo que pequenos erros possam ocorrer. Os valores raros e extremos que pretendemos prever poderão ser significativamente diferentes entre eles: tratam-se de *outliers*. Queremos ser precisos na previsão destes valores.

Decidimos usar o MSE como medida base na avaliação da nossa *precision*. No entanto, tal como a *accuracy*, em problemas com uma distribuição de classes desequilibrada, esta medida por si só mostra-se inadequada ao nosso problema. Ela terá que ser embebida na definição de *precision*, a fim de dar relevância apenas aos valores raros e extremos. Não nos interessa tanto que o nosso modelo seja preciso em todos os valores assumidos pela variável que descreve o fenómeno, mas apenas nos seus valores raros e extremos. O MSE passa assim a ser calculado sobre o conjunto O.

Contudo, as medidas de *precision* e *recall* definidas no âmbito da *Information Retrieval*, assumem valores no intervalo 0..1. Se estabelecermos a média do conjunto de valores da variável objectivo como o estimador δ, obtemos o erro do modelo mais simples. Nestas condições, e tal como foi referido na Secção 4.2, a média não será, à partida, uma boa medida de representatividade para a variável objectivo Y, já que a presença de *outliers* faz com que o conjunto de valores que descrevem o fenómeno não possua uma distribuição Normal. Em especial, e para o caso do conjunto O, conjunto sobre o qual queremos ser precisos, a média de Y será um mau estimador.

Assim sendo, parte-se do princípio que o erro das previsões feitas pelo nosso modelo será inferior ao de simplesmente prever a média dos valores de Y. Optamos então por definir a *precision* com base no $NMSE$ (*Normalized Mean Square Error*) apresentado pela Equação 4.6, normalizando o MSE com base no modelo mais simples de forma assumir valores no intervalo 0..1. A nossa medida de $precision_{regr}$ é definida pela Equação 4.7.

Temos então

$$NMSE_O = \frac{\sum\limits_{y_i \in O} \left(\hat{y}_i - y_i\right)^2}{\sum\limits_{y_i \in O} \left(\bar{Y} - y_i\right)^2} \tag{4.6}$$

onde $\bar{Y}$ é a média de Y no conjunto de treino.

A medida $precision_{regr}$ será o complementar de $NMSE_0$:

$$precision_{regr} = 1 - NMSE_O \tag{4.7}$$

$NMSE_O$ irá, regra geral, assumir valores no intervalo 0..1. No entanto, é possível que em alguns casos assuma um valor superior a 1. Tais casos significam que a previsão feita é pior do que simplesmente prever sempre a média. Nessa situação estabelecemos que a $precision_{regr}$ do modelo é igual a 0.

Se voltarmos ao exemplo acima referido (*c.f.* Figura 4.4), obteríamos os valores para $precision_{regr}$ de $\backsimeq 0.998$ e de $\backsimeq 0.884$, para as previsões $\hat{y}_a$ e $\hat{y}_b$, respectivamente (valores que já reflectem a diferença das duas previsões).

Idealmente, gostaríamos que as medidas de *precision* e *recall* assumissem ambas um valor igual a 1. Isso significaria que todos *outliers* de valores extremos e apenas esses seriam previstos correctamente pelo nosso modelo. Assim, recorremos à *F-measure*, também definida no âmbito da *Information Retrieval* (*c.f.* Equação 3.6, página 42), para a avaliação da *performance* do modelo.

$$F = \frac{(\beta^2 + 1) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall}$$

Esta medida assume, igualmente, valores entre 0 e 1, retornando valores tão valiosos quão mais valiosos forem os valores de *precision* e *recall*. O compromisso estabelecido entre as duas medidas é especificável pelo parâmetro β, que expressa a importância de *recall* relativamente à *precision*. Substituindo *precision* por $precision_{regr}$, esta será a medida que guiará a procura do "melhor" modelo.

4.4.3 Descrição do método

Uma das nossas metas na concepção de um novo método consiste em tentar adquirir novos conhecimentos ao perceber a razão da ocorrência dos fenómenos raros de valor extremo. Nesse sentido, interessa que o nosso método seja capaz de conceber modelos interpretáveis para pessoas ligadas ao domínio da aplicação.

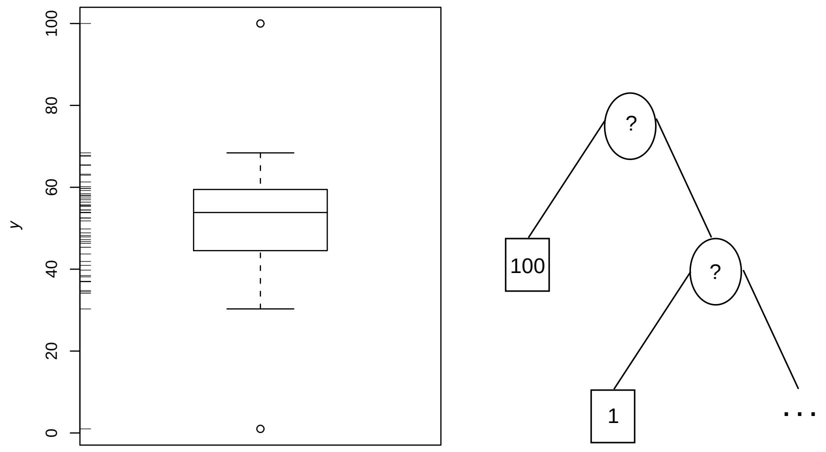
A obtenção de tais modelos interpretáveis foi uma das razões que esteve na base da escolha das árvores de regressão para o desenho do nosso método. Para além de darem origem a modelos altamente interpretáveis, as árvores são computacionalmente eficientes e constituem um poderoso método pela sua flexibilização de ajuste ao problema. Foi sobre esses mesmos propósitos que assentou a abordagem de Buja e Lee (2001). Para melhorar a *performance* das árvores na detecção de subconjuntos de dados interessantes, eles propuseram novos critérios de *split* para o crescimento das árvores de regressão e classificação. Também nós propomos aqui um novo critério de *split* para as árvores de regressão orientado aos objectivos por nós estabelecidos.

Uma árvore consiste numa estrutura que liga um conjunto de nós através de ramos resultantes de uma partição recursiva dos dados. Esta partição dos dados é feita através de testes contidos em cada um dos nós, sobre o valor de variáveis independentes (contínuas ou discretas) e escolhidos no sentido de minimizar uma determinada estimativa de erro do que se pretende prever. As folhas são os nós terminais e contêm as previsões feitas pelo modelo. O conjunto de caminhos da raiz da árvore a cada uma das suas folhas constitui o conjunto de "regras" pelas quais a árvore faz as suas previsões. No caso das árvores de regressão, a variável a prever é uma variável contínua e, segundo o critério mais *standard* (Breiman et al., 1984), consiste na média dos valores dessa variável na partição de dados incluída em cada folha. Para além disso, cada nó dá origem a apenas dois ramos, de acordo com o resultado do teste (verdadeiro ou falso).

Os modelos baseados em árvores constituem um conjunto de "regras" de fácil entendimento para o utilizador final, onde são destacadas as variáveis para a previsão mais relevantes no domínio. Estes modelos têm assim a capacidade de sumariar grandes conjuntos de dados multivariados.

A interpretabilidade das árvores reside, essencialmente, no topo das mesmas. Idealmente, gostaríamos de conceber um critério de *split* capaz de isolar, logo que possível, os *outliers* de valor extremo. Desta forma obteríamos árvores mais desequilibradas, mas mais interpretáveis sob o ponto de vista do nosso problema. A Figura 4.5 (b) apresenta o exemplo de uma árvore bastante interpretável por identificar, logo no topo, as condições que levam à ocorrência de valores raros e extremos da variável objectivo y, de acordo com o *box plot* apresentado na Figura 4.5 (a), discernindo-os dos restantes valores.

A par dos objectivos estabelecidos devem, então, ser definidos os critérios que conduzirão a procura do "melhor" modelo. Para tal, e no caso concreto das árvores de que falamos aqui, torna-se necessária a definição de dois critérios: critério de *split* e critério de paragem.



(a) *box plot* da variável y

(b) árvore de regressão para a previsão de y

Figura 4.5: Árvore que isola *outliers* extremos no topo.

Estes dois critérios dizem respeito à fase de crescimento da árvore. O primeiro critério tem a ver com a forma como são escolhidos testes para cada partição e que irão constituir cada nó da árvore. No fundo, é responsável pela procura da árvore "perfeita" para o conjunto de dados disponível para treino. O segundo critério consiste na condição pela qual se decide parar o crescimento da árvore por se considerar que este não trará nenhuma mais-valia significativa ao modelo, segundo parâmetros específicos ao problema. Mais do que isso, o prosseguimento do crescimento da árvore poderá traduzir-se num sobre-ajustamento indesejável aos dados de treino estando, nesse caso, sob o problema de *overfitting*.

4.4.3.1 Critério de *split*

Uma vez definidos os critérios de avaliação dos modelos, devem ser estabelecidos critérios que orientem a concepção do modelo nesse sentido através da escolha dos *splits* mais valiosos. Não faria sentido fazer crescer as árvores segundo outros critérios que não aqueles que consideramos ser os mais adequados para a avaliação de soluções retornadas.

Torgo (1999) refere a existência de dois critérios base usados na concepção de árvores de regressão: *LS* (*Least Squares Regression Trees*) e *LAD* (*Least Absolute Deviation Regression Trees*). Estes dois critérios apostam na minimização das estatísticas MSE

e MAD, respectivamente. Nas árvores LS os valores previstos correspondem à média dos valores assumidos pela variável objectivo em cada uma das partições atribuídas a cada folha. Nas árvores LAD, em vez da média, é colocada a mediana dos valores da variável objectivo na partição representada por cada folha. Estas últimas árvores tornam-se menos sensíveis a *outliers* relativamente às árvores LS. Tal acontece pelo facto da média ser uma estatística bastante enviesada pela distribuição dos valores e portanto mais sensível à presença de *outliers*. Por essa razão, e porque pretendemos obter modelos precisos segundo a medida definida pela Equação 4.7, tomámos as árvores LS como ponto de partida para a concepção do nosso método.

O sistema CART (Breiman et al., 1984) constitui uma referência *standard* nas árvores de regressão e classificação. No caso da regressão, as árvores são construídas segundo o critério LS. Segundo este critério o "melhor" *split* será aquele que maximiza a diferença entre o erro quadrado médio do nó actual e o erro quadrado médio obtido nas duas partições geradas, ponderado pelo tamanho de cada uma delas (*c.f.* Equação 3.10, Secção 3.4, Capítulo 3).

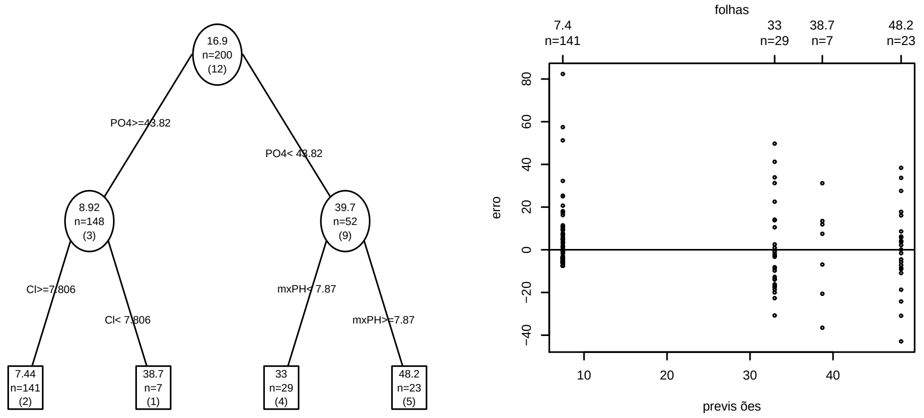
Contudo, este critério mostra-se inadequado aos nossos objectivos, da mesma forma que se mostrava inadequado ao objectivos de Buja e Lee (2001). Não pretendemos obter uma boa *performance* global sobre todos os dados, mas apenas sobre um subconjunto específico deles. Não nos interessa avaliar a qualidade de um *split* a partir da qualidade conjunta das duas partições a que dá origem. Interessa-nos, sim, um *split* que dê origem a uma partição muito valiosa independentemente da outra partição criada. Basta que isolemos o maior número de *outliers* possível numa partição. Para além disso, e segundo o critério LS, a presença de valores extremos irá certamente distorcer a média presente nas folhas da árvore, fazendo com que esta deixe de ser um valor representativo da partição, piorando, automaticamente, a qualidade do *split*.

No Figura 4.6 (a) é apresentada a árvore[3] gerada pelo CART para o problema de previsão de concentrações da alga a_1, cujo *box plot* foi apresentado na Figura 4.3. Apesar de existirem folhas que indicam valores elevados de concentração dessa alga, nenhuma delas faz previsões superiores a 58.7 (o valor de adj_H neste problema). Isto significa que esta árvore não só se mostra incapaz de captar os valores de ocorrência extremos desta alga, como também deixa que estes exerçam uma influência negativa sobre as suas previsões, na medida em que estas se tornam bastante enviesadas. A Figura 4.6 (b) comprova isso mesmo. As folhas que contêm as previsões mais elevadas para a ocorrência da alga a_1, para além de

[3]Os números entre parêntesis correspondem ao número de *outliers* de valor extremo em cada nó.

não serem indicadoras de *blooms* desta alga, têm a si associados erros também elevados. Em particular, a folha com o valor 38.7, apesar de representar uma pequena partição dos dados de treino, não se revela um bom valor representativo para nenhum dos exemplos nela contidos. Por outro lado, a previsão de frequência de ocorrência da alga a_1 de 7.44, “encobre” a presença de dois *outliers* de valor extremo que fazem associar a esta folha alguns valores residuais bastante elevados.

O modelo retornado pelo CART mostra-se assim muito pobre face aos nossos objectivos: os *blooms* da alga a_1 nunca serão captados e muito menos serão perceptíveis as condições sob as quais estes ocorrem.



(a) árvore CART (b) valores residuais

Figura 4.6: Árvore de regressão obtida pelo CART para a previsão de concentração da alga a_1.

Para que se possa dar uma resposta mais adequada aos nossos objectivos, torna-se, então, necessária uma adaptação do critério LS. À semelhança da proposta de Buja e Lee (2001), pretendemos obter partições o mais “puras” possível, independentemente das medidas obtidas para outras partições que consideramos ser menos relevantes. Assim sendo, um *split* será tão melhor quão maior for a medida de pureza de uma das duas partições geradas.

No entanto, o conceito de pureza aqui utilizado não é coincidente com os utilizados por Buja e Lee (2001) nos métodos *one-sided purity* e *one-sided extremes*. De acordo com os critérios de avaliação já por nós estabelecidos no contexto do tipo de problemas alvo desta tese, o conceito de pureza terá que estar estritamente ligado aos conceitos de *precision* e *recall*. Pretendemos maximizar a *F-measure*. Neste contexto, propomos que o melhor *split* seja:

$$s^*(D_t) = \max_{s \in S} \ \max\left(F(D_{t_L}), F(D_{t_R})\right) \tag{4.8}$$

onde S é o conjunto de *splits* candidatos[4] para um nó t; D_{t_L} é o subconjunto de casos em t (D_t) que satisfaz o *split* s (i.e. o ramo esquerdo de t) e D_{t_R} contém os restantes casos (i.e. $D_{t_R} = D_t - D_{t_L}$); $F(D)$ é a *F-measure* para um conjunto de exemplos.

Para avaliarmos um determinado *split*, necessitamos de obter o valor de *F-measure* associado a cada uma das duas partições geradas. Para tal, teremos de calcular a *precision* e *recall* de cada partição.

A avaliação da *precision* de uma dada partição tem, por base, a definição $precision_{regr}$ dada pela Equação 4.7 da Subsecção 4.4.2. O seu cálculo é, igualmente, baseado na medida $NMSE_O$, apenas com uma pequena diferença: não são, necessariamente, usados todos os *outliers* extremos existentes na partição de dados.

Imagine-se que, numa partição sob avaliação, existem *outliers* de ambos os extremos da variável. Será impossível que esta partição tenha uma *precision* elevada dado que terá, certamente, um $NMSE_0$ elevado. No entanto, penalizar a qualidade do *split* que originou uma partição nessas condições, poderia ser prematuro no sentido em que se poderia estar desprezar informação potencialmente útil sobre esses mesmos *outliers* extremos.

Para salvaguardar o valor de partições como esta, optou-se por incidir o cálculo da *precision* sobre apenas um dos tipos de *outliers* extremos, O_L ou O_H, em cada momento. A decisão do tipo de *outliers* extremos sobre o qual se vai calcular a *precision* da partição, é feita com base na posição relativa da média da partição em relação à mediana global da variável Y. De uma forma um pouco heurística, estabelecemos que, se a média da partição se encontra abaixo da mediana global, então é provável que esse enviesamento em relação à mediana decorra de valores pertencentes a O_L. O mesmo foi estabelecido para O_H.

No caso da própria média constituir um *outlier* extremo, então é certo que essa partição fará a previsão de *outliers* do mesmo tipo. Se a média não for um *outlier* extremo, então, estando abaixo ou acima da mediana global, vemos esta partição como uma tentativa de prever *outliers* extremos baixos ou elevados, respectivamente. Mas se a média da partição coincide com a mediana global, então recorre-se ao tipo do maior número de *outliers*

[4] O conjunto de *splits* candidatos a um nó é o mesmo usado em árvores de regressão *standard*.

extremos existentes no domínio para determinar, de uma forma novamente heurística, o subconjunto de *outliers* extremos a usar (O_L e O_H).

É claro que no caso da partição não possuir quaisquer *outliers* extremos, ou a média se encontrar no lado errado da mediana, em relação ao tipo de *outliers* extremos existentes no domínio, a *precision* é zero.

Resumindo a medida $precision_{regr_t}$ representa a *precision* de um nó t e é obtida da seguinte forma:

$$precision_{regr_t} = \begin{cases} 0 & \begin{array}{c} \text{if } (O_H(D_t) \cup O_L(D_t) = \phi) \vee \\ (\bar{y}_t < \tilde{Y} \wedge O_L(D_t) = \phi) \vee \\ (\bar{y}_t > \tilde{Y} \wedge O_H(D_t) = \phi) \end{array} \\ 1 - \dfrac{\sum\limits_{y_i \in O_L(D_t)} (\bar{y}_t - y_i)^2}{\sum\limits_{y_i \in O_L(D_t)} (\bar{Y} - y_i)^2} & \text{if } \bar{y}_t < \tilde{Y} \;\vee (\; \bar{y}_t = \tilde{Y} \wedge |O_L| > |O_H| \;) \\ 1 - \dfrac{\sum\limits_{y_i \in O_H(D_t)} (\bar{y}_t - y_i)^2}{\sum\limits_{y_i \in O_H(D_t)} (\bar{Y} - y_i)^2} & \text{if } \bar{y}_t > \tilde{Y} \;\vee (\; \bar{y}_t = \tilde{Y} \wedge |O_H| > |O_L| \;) \end{cases} \tag{4.9}$$

onde $O_L(D_t)$ ($O_H(D_t)$) é o conjunto de casos do nó t que pertencem a $O_L(O_H)$; $\bar{y}_t$ é a média dos valores da variável Y presentes no nó; e $\bar{Y}$ e $\tilde{Y}$ são a média e a mediana de Y no conjunto de dados, respectivamente.

Em relação ao *recall*, o seu cálculo é mais transparente e mais próximo da definição dada pela Equação 4.3. Se a média da partição não for um *outlier* extremo, então o seu *recall* é zero: não é previsto nenhum *outlier* extremo. Se média da partição avaliada for um *outlier* extremo baixo então é calculada a proporção de valores pertencentes a O_L que estão presentes nesta partição. Um processo equivalente é feito no caso da média da partição ser um *outlier* extremo elevado.

A medida $recall_t$ representa o *recall* para um nó t e é obtido da seguinte forma:

$$recall_t = \begin{cases} 0 & \text{if } \bar{y}_t \geq adj_L \ \wedge \ \bar{y}_t \leq adj_H \\ \frac{\mid y \in D_t \ \wedge \ y \in O_H \mid}{\mid O_H \mid} & \text{if } \bar{y}_t > adj_H \\ \frac{\mid y \in D_t \ \wedge \ y \in O_L \mid}{\mid O_L \mid} & \text{if } \bar{y}_t < adj_L \end{cases} \tag{4.10}$$

No entanto, esta definição de *recall* coloca-nos um novo problema: sempre que a média da partição não for um *outlier* extremo, o $recall_t$ será zero e, independentemente do valor de $precision_{regr}$, a *F-measure* será também zero.

O que acontece é que nem sempre é fácil encontrar um *split* que logo à partida dê origem a uma partição cuja média é um *outlier* extremo. Nesse caso poderá ser obtida uma *F-measure* de zero para todos os *splits* candidatos para um dado nível da árvore. Situações como estas ocorrem sobretudo no topo da árvore onde as partições geradas são grandes e se torna mais difícil obter uma partição cuja média seja um *outlier* extremo. Isto é ainda mais problemático no caso de atributos nominais, onde o número de *splits* possíveis é menor.

Contudo, a medida $precision_{regr_t}$ fornece-nos uma relação de qualidade entre os diferentes *splits* mesmo que o seu *recall* seja zero. Para que o $recall_t$ não anule o valor de $precision_{regr_t}$, optámos por fazer um pequeno ajuste à definição de *F-measure* para um dado nó t. Ao parâmetro $recall_t$ passa a ser adicionado um *threshold*[5] Δ que impede que a medida *F-measure* seja nula quando o $recall_t$ é zero (*c.f.* Equação 4.11).

$$F(D_t) = \frac{(\beta^2 + 1) \cdot precision_{regr_t} \cdot (recall_t + \Delta)}{\beta^2 \cdot precision_{regr_t} + recall_t + \Delta} \tag{4.11}$$

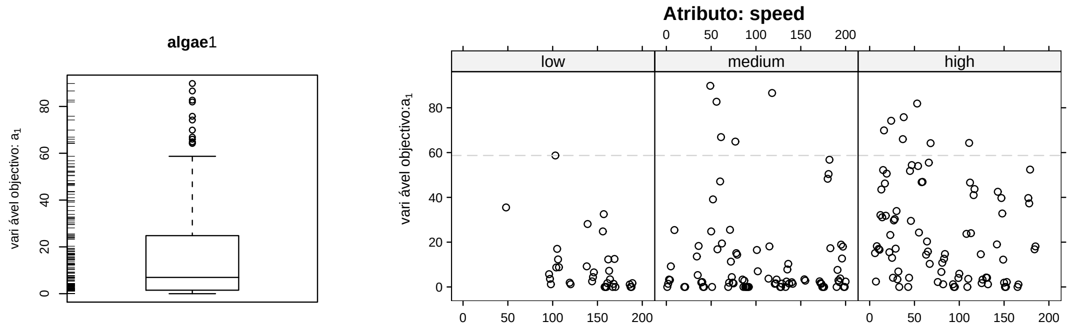
O parâmetro β é um parâmetro especificável pelo utilizador[6], de acordo com a importância relativa que se pretende atribuir ao $recall_t$ face à $precision_{regr_t}$.

A fim de dar algumas ilustrações do comportamento do nosso método no que toca à escolha de *splits*, recorremos mais uma vez ao problema de previsão de *blooms* da alga a_1. Dos exemplos abaixo apresentados é já notada uma diferença vincada nos propósitos que o nosso método pretende servir face àqueles estabelecidos pelo CART.

[5] Nas nossas experiências o *threshold* Δ usado foi de 0.001.

[6] Por defeito $\beta = 1$.

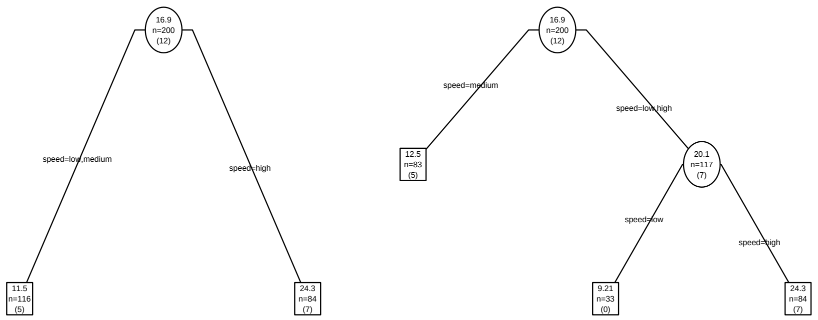
O atributo `speed`, é um atributo discreto com apenas três valores. Os valores assumidos pela variável objectivo a_1 de acordo com os três valores deste atributo encontram-se abaixo apresentados no gráfico da Figura 4.7 (b).



(a) *box plot* da variável a_1 (b) velocidade do rio

Figura 4.7: Concentrações da alga a_1 registadas em águas com velocidades diferentes.

Da observação do gráfico é possível constatar que os *blooms* parecem ocorrer quando a velocidade dos rios é média ou elevada. A fim de melhor podermos diferenciar o nosso método do CART, gerámos as árvores para a previsão de concentração da alga a_1 com base, unicamente, no atributo `speed`. As árvores obtidas encontram-se nas Figuras 4.8 (a) e (b).



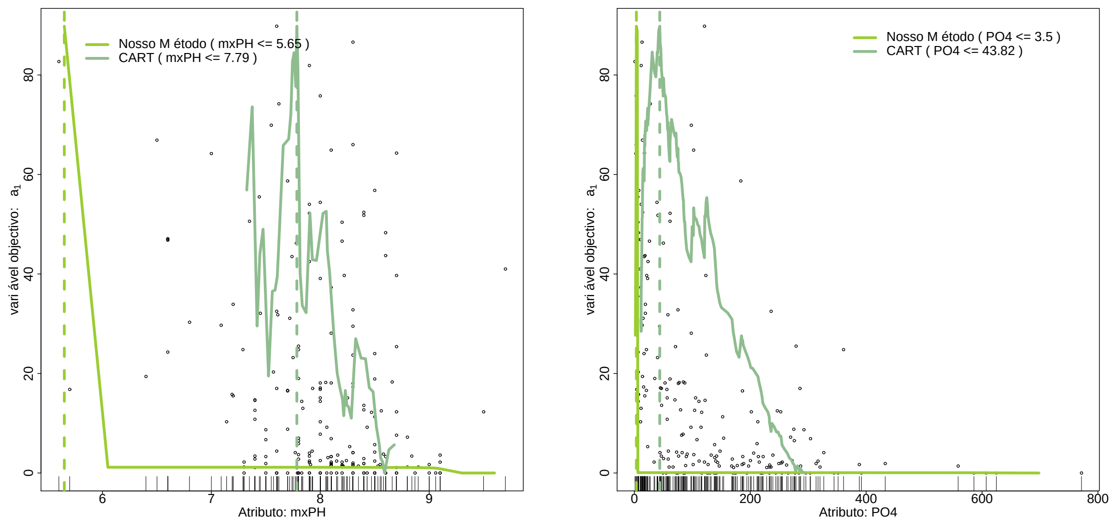
(a) CART (b) Nosso Método

Figura 4.8: Árvores de regressão, geradas a partir do atributo `speed`, pelo CART e pelo Nosso Método para a previsão de concentração da alga a_1.

Face aos requisitos do problema, a nossa árvore (Figura 4.8 (b)) seria uma árvore bem mais "útil" do que aquela gerada pelo CART (Figura 4.8 (a)). Isto porque o nosso método foi capaz de discernir a situação onde muito provavelmente não virão a acontecer esses *blooms* das situações onde eles, potencialmente, poderão vir a acontecer.

Ainda com o objectivo de vincar a diferença de comportamento do nosso método e do CART face a uma partição de dados, simulámos a avaliação de *splits* feita por cada um dos métodos para a previsão de concentração da alga a_1, para dois atributos: mxPH e PO4.

Nas Figuras 4.9 (a) e (b) são apresentados os gráficos dos valores de concentração da alga a_1 em relação aos valores dos atributos mxPH e PO4, respectivamente. Em cada um dos gráficos são também apresentadas as linhas de valorização atribuídas ao conjunto de *splits* possíveis para o atributo em questão, por cada um dos métodos.



(a) *splits* sobre valores máximos de PH

(b) *splits* sobre valores médios de Fosfato

Figura 4.9: Avaliação de *splits* feita pelo CART e pelo Nosso Método para os atributos mxPH e PO4 para a previsão de concentração da alga a_1.

Conforme se pode observar, o CART falha na constituição de um *split* capaz de isolar apenas *outliers* extremos. Isso acontece, fundamentalmente, pela simples razão de que os *blooms* da alga a_1 se encontram na cauda da distribuição de valores das variáveis mxPH e PO4. O problema dos *end-cut splits* (já mencionado no Capítulo 3, Secção 3.4) impede o CART de, sequer, considerar *splits* que dêem origem a pequenas partições. Este facto é evidente para o caso do atributo mxPH, onde o nosso método conseguiu seleccionar um split que isola um *outlier* extremo. Este *split* terá, certamente, uma boa *precision*. No caso do atributo PO4, o nosso método consegue seleccionar um *split* que isola não apenas um, mas quatro *outliers* extremos numa só partição. Este é um *split* ainda mais valioso no contexto do nosso problema, já que possui também um bom *recall*.

4.4.3.2 Critério de paragem

Tal como já foi referido, uma árvore é concebida segundo um critério de *split* que avalia a pureza das partições geradas e que guia o seu crescimento. Sem um critério de paragem pré-definido, a árvore crescerá até que seja alcançada uma partição absolutamente "pura" ou que contenha apenas um exemplo. Esta será, certamente, a árvore que melhor representará o conjunto de treino e a mais exacta nas previsões feitas sobre esses exemplos. No entanto, a par deste bom desempenho no conjunto de treino, ela tenderá a ter um mau desempenho num conjunto de dados independente. Tal acontece pelo facto da árvore concebida ser um modelo demasiadamente sobre-ajustado aos dados de treino, possuindo assim uma má capacidade de generalização. Este é o denominado problema de *overfitting* que se coloca na concepção dos modelos.

A questão que se coloca então é: quando devemos parar o crescimento da árvore?

Esta é, de facto, uma questão complexa. Se deixarmos crescer árvores extremamente profundas, temos o problema de *overfitting*. Mas, se obtivermos árvores demasiado pequenas, elas poderão não conseguir capturar a estrutura presente nos dados, possuindo assim uma pobre *performance* no domínio. A problemática coloca-se quanto ao instante em que se obtém um modelo suficientemente preciso e suficientemente geral, de modo a possuir sempre uma boa *performance*.

Idealmente, um modelo deve aprender o conceito embebido no conjunto de exemplos de treino de um modo suficientemente geral para que, perante novos exemplos de um conjunto de teste, seja capaz de aplicar o conceito aprendido e, com base nele, fazer as suas previsões de forma suficientemente precisa.

Para ajudar no processo de escolha do "melhor" modelo, tendo em conta o compromisso entre a precisão do modelo e a capacidade de generalização deste, recorre-se à técnica de *pruning*.

Existem duas estratégias base para o *pruning* das árvores: *pre-pruning* e *post-pruning*. O *pre-pruning* envolve decidir, durante o processo de crescimento da árvore, quando é que este deve parar de acordo com algum critério de pureza, por exemplo a estimativa do erro. Usando *post-pruning* a árvore é concebida em duas fases: numa primeira fase é obtida a maior árvore possível e numa segunda fase ela é cortada no sentido de melhorar a capacidade de generalização da mesma. Nesta fase são eliminados testes demasiado específicos ao conjunto de treino e cuja existência não se traduz num acréscimo de pureza no modelo, como seja por exemplo um decréscimo da estimativa do erro do modelo.

A maior parte dos sistemas usa o *post-pruning*. Isto acontece porque apesar do *pre-pruning* ser bem mais eficiente, esta estratégia não consegue obter os bons resultados oferecidos pelo *post-pruning*. De facto, dado o carácter *greedy* do algoritmo de construção das árvores, o *post-pruning* parece trazer algumas vantagens. O *split* escolhido é sempre aquele que optimiza o critério de pureza no contexto local à escolha, não tendo em conta uma optimização global da *performance* da árvore. Um *split* que, à partida, não pareça ter muito para contribuir, pode vir a revelar-se, mais tarde, num *split* "poderoso" quando combinado com um outro conjunto de *splits* subsequentes. A decisão de paragem num algoritmo *greedy* será sempre um risco, podendo sempre ser prematura.

Normalmente, o tamanho "ideal" da árvore é decidido com base em estratégias como *cross-validation* ou *holdout* de forma a obter uma estimativa mais fiável da capacidade de generalização do modelo. A estimativa do erro do modelo deve ser avaliada num conjunto independente de dados. Só desta forma poderemos tomar uma decisão racional quanto ao tamanho da árvore. Usar a estimativa do erro no conjunto de treino não levará a qualquer *pruning* visto a árvore estar a ser construída expressamente para esse conjunto de dados.

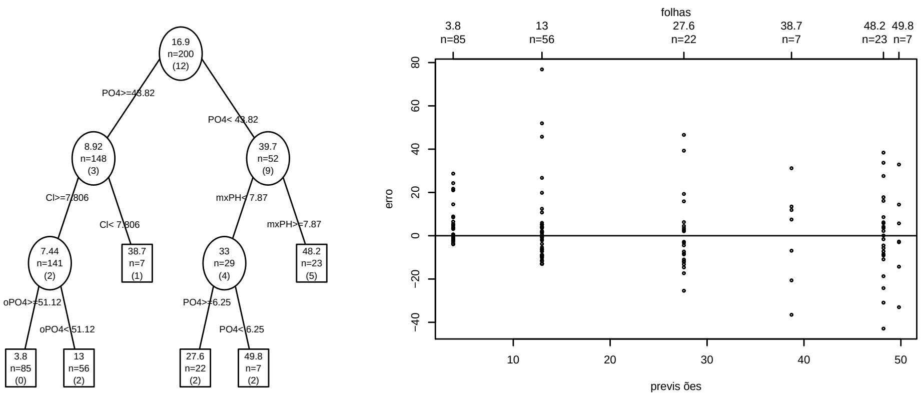
O sistema CART (Breiman et al., 1984), à semelhança do que acontece com outros sistemas, recorre à estratégia de *post-pruning*. A concepção das árvores é assim feita em dois passos.

O *pruning* usado pelo CART é feito através da minimização de uma medida que determina o custo da árvore e que combina dois factores: a complexidade (tamanho) da árvore e a sua estimativa de erro. O nível de *pruning* é controlado pelo parâmetro *cp* (*complexity parameter*) que define o "peso" associado às folhas. Quanto maior for o valor de *cp*, mais "pesadas" serão as folhas e maior será o *pruning* feito. A partir da árvore completa, são criadas todas as sub-árvores possíveis com um *cp* maior ou igual ao estabelecido. O processo é feito de forma iterativa conduzindo à obtenção de uma sequência de sub-árvores.

Todas as sub-árvores geradas são depois avaliadas segundo uma estimativa de erro. Para que possa ser obtida uma estimativa mais fiável do erro é usado o *cross-validation*, sendo repetido o processo acima descrito para cada um dos *folds*. O objectivo é encontrar a menor sub-árvore que minimiza a estimativa do erro da árvore.

A árvore CART gerada com a regra 0-SE corresponde à menor sub-árvore, isto é, com maior *cp*, e menor estimativa de erro. A árvore CART gerada com a regra 1-SE corresponde à menor sub-árvore cuja estimativa de erro não excede $\hat{e} + SE(\hat{e})$, onde $\hat{e}$ é a menor estimativa de erro obtida no conjunto de sub-árvores geradas e $SE(\hat{e})$ é o erro *standard* da estimativa $\hat{e}$. Este último método conduz à concepção de árvores mais simples, apesar de, possivelmente, mais fracas em termos de *performance*. Esta é a árvore habitualmente retornada pelo CART.

A árvore apresentada na Figura 4.10 (a) resultou da regra 0-SE. Para o mesmo problema, a árvore gerada pelo CART com a regra 1-SE é aquela apresentada na Figura 4.6 (a). Conforme é possível constatar, a árvore CART (0-SE), quando comparada com a árvore CART (1-SE), é mais sobre-ajustada aos dados de treino sendo, por isso, mais profunda. Apesar dos erros se encontrarem espalhados por mais folhas, as previsões continuam a sofrer da presença de *outliers* que provocam o enviesamento da média colocada em cada folha (*c.f.* Figura 4.10 (b)).



(a) árvore CART (0-SE)

(b) valores residuais

Figura 4.10: Árvore de regressão obtida pelo CART (0-SE) para a previsão de concentração da alga a_1.

No entanto, este método de *post-pruning* mostra-se contra-intuitivo face aos nossos objectivos. Basicamente, ele procura simplificar modelos que sejam, possivelmente, desnecessariamente complexos. Esta simplificação é feita através da remoção de testes de pouca significância estatística, tal que seja minimizado o tamanho da árvore e a estimativa do erro, tentando, desta forma, chegar a um modelo compacto e preciso.

Mas, no nosso problema, o que realmente nos interessa no modelo é a previsão de *outliers* extremos, isto é, valores raros e de natureza insignificante do ponto de vista estatístico. Os testes que o fazem, apesar de serem responsáveis pela previsão correcta de poucos exemplos serão também, certamente, responsáveis pela previsão errada de muitos mais exemplos podendo, por essa razão, serem alvo de eliminação. Contudo, eles constituem o ponto fulcral da nossa modelação e não deverão, por isso, ser, simplesmente, eliminados.

Por essa razão, e porque os métodos de *post-pruning* de modelos se baseiam na significância estatística, optámos por não recorrer a esta estratégia. Esta opção é também fundamentada por um dos estudos feitos por Weiss e Hirsh (1998b, 2000b) acerca do problema com os chamados *small-disjuncts* e o impacto que eles têm no processo de aprendizagem.

A aprendizagem feita a partir de um conjunto de exemplos resulta, na maior parte dos casos, num conjunto de disjunções de testes sobre valores de variáveis independentes. Os *small-disjuncts* constituem a disjunções que apenas cobrem um pequeno subconjunto dos exemplos de treino. Weiss (1995) mostrou que os *small-disjuncts* correspondem aos casos raros no domínio em estudo. Por essa razão, o *pruning* poderá ser considerado questionável quando se aprende a partir dos *small-disjuncts*. O facto é que, em conjuntos de dados com ruído ou exemplos que descrevem um fenómeno raro, os *small-disjuncts* têm a si associada uma maior taxa de erro, quando comparada com a taxa de erro associada aos *large-disjuncts* (que cobrem os casos mais gerais), ainda que sejam necessários no sentido de alcançar uma elevada *accuracy*. A significância dos *small-disjuncts* é inversamente proporcional à sua taxa de erro.

É neste contexto que se encontra inserido o nosso tipo de problemas: pretendemos aprender a descrição de fenómenos raros com valor extremo, representado por um conjunto escasso de exemplos e que, portanto, darão origem à formulação de *small-disjuncts*.

Os *small-disjuncts* levantam questões não só a nível de *pruning* dos modelos, mas também a nível da quantidade de dados disponível e da presença de ruído nos dados e de como isso afecta o processo de aprendizagem. Experiências realizadas por Weiss e Hirsh (2000b), mostraram que, efectivamente, muitos dos *small-disjuncts* são eliminados pelo *pruning*. No *post-pruning* o objectivo centra-se em reduzir ao máximo o número de disjunções de tal modo que isso não se reflicta num aumento da estimativa da taxa de erro do modelo. Por esta razão, os *small-disjuncts* acabam por ser, na sua maioria, eliminados já que a eles está associada uma elevada taxa de erro, ainda que sejam responsáveis pela previsão correcta de um pequeno e restrito conjunto de exemplos.

Esta eliminação acaba por traduzir-se de forma mais significativa no tamanho médio das disjunções que constituem o modelo, do que propriamente na redução do erro. Aliás, resultados apresentados no trabalho Weiss e Hirsh (2000b) revelaram que o *pruning* poderá ajudar em algumas situações, mas em outras situações pode, de facto, vir a reflectir-se num aumento da taxa de erro. Situações onde pretendamos obter uma boa *performance* num subconjunto de dados específico e não estejamos tão interessados na *performance* global. Nestas situações, tal como sugerido por Weiss e Hirsh (2000b), o *pruning* deve estar ligado ao uso de uma métrica específica à tarefa de aprendizagem.

Este é o caso das situações onde o que estamos interessados em prever é descrito por um conjunto escasso de exemplos. Posto isto, o nosso método obtém as árvores num só passo, de acordo com as seguintes condições de paragem, que estabelecem o *pruning* das árvores:

- o nó não contém *outliers* extremos ($D_t \cap O = \phi$);
- a *F-measure* do nó é superior a um determinado *threshold* f.

O parâmetro f poderá assumir valores no intervalo 0..1 e é mais um parâmetro especificável pelo utilizador[7]. Com o parâmetro f igual a um, valor máximo da função *F-measure*, a árvore não parará o seu crescimento senão pela inexistência de *outliers* extremos na partição de dados.

4.4.4 Interpretabilidade dos modelos

Tal como já foi referido anteriormente, um dos pontos-chave que tivemos em consideração na elaboração deste método tem exactamente a ver com a interpretabilidade dos modelos obtidos.

[7]Por defeito $f = 0.7$.

Definimos um critério de *split* que tenta isolar, o mais cedo possível, os *outliers* de valor extremo para que deste modo, as árvores, apesar de possivelmente desequilibradas, se tornassem mais interpretáveis.

A título de exemplo, comparemos então os modelos retornados pelo sistema CART (Breiman et al., 1984) e pelo método que acabamos de descrever, para o problema de previsão de concentração da alga a_1.

Na Figura 4.11 é apresentada a árvore obtida pelo nosso método com *pruning* de $f = 0.4$.

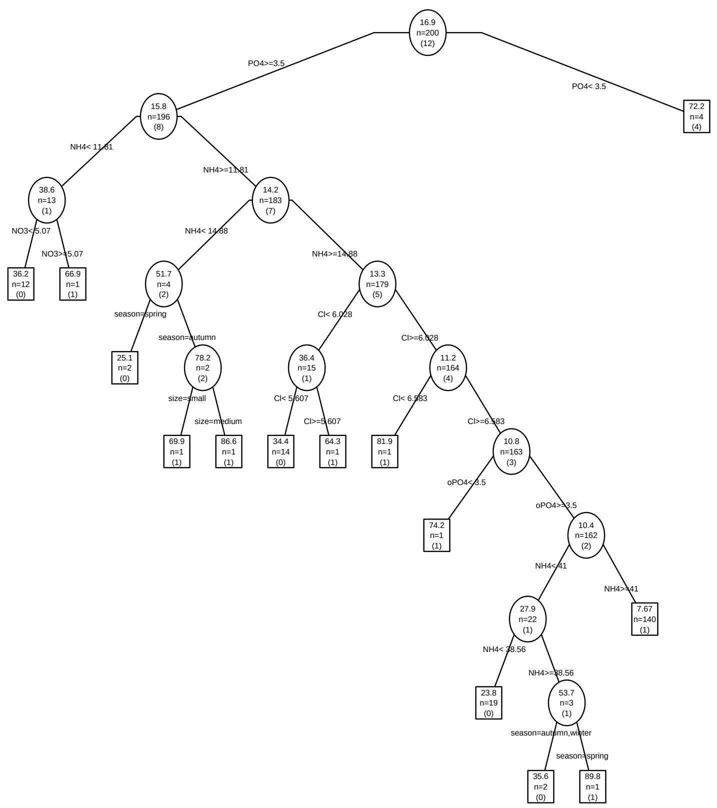


Figura 4.11: Árvore de regressão obtida pelo nosso método com *pruning* de $f = 0.4$ para a previsão de concentração da alga a_1.

De facto, e comparando com a árvore apresentada na Figura 4.6, esta árvore faz logo de início uma escolha muito mais útil do ponto de vista da previsão de *blooms* da alga a_1: é identificada logo à partida uma partição "pura" de quatro *ouliers* extremos, através do *split* apresentado como mais valioso na figura 4.9 para o atributo PO4. Os restantes oito *outliers* extremos encontram-se isolados em níveis mais profundos da árvore, portanto de forma menos clara. Mesmo assim todos eles são distinguidos dos restantes valores. A excepção reside na folha com o valor 7.67 que contém um *outlier* extremo que não foi distinguido da partição de 140 exemplos que a constituem. Mas, neste contexto, pode ser colocada a questão de estarmos realmente perante um *outlier* extremo de concentração da alga ou simplesmente ruído.

A árvore possui uma estrutura bastante desequilibrada, mas bastante interpretável no sentido de perceber as condições que levam à ocorrência de *blooms*, pelo menos no que se refere ao primeiro *split* feito.

A informação mais importante que este modelo transmite sobre o domínio refere-se a valores baixos de fosfato (PO4) levarem a valores elevados de ocorrência da alga a_1. Nas águas não poluídas os níveis de fosfato são relativamente baixos. Isto acontece porque, sendo os fosfatos um importante fertilizante para as plantas, elas rapidamente o absorvem. Para sobreviver as algas precisam de quantidades pequenas de fosfato. Ora, se houver uma subida dos níveis de fosfato, isto é excesso de fosfato, então as plantas irão consumi-lo provocando o crescimento desenfreado de algas. Talvez por essa razão, a maior parte dos *outliers* extremos de concentração da alga a_1, surja para valores elevados de PO4 Contudo, com base nesta teoria, a primeira informação transmitida pelo modelo parece ser contraditória. Na realidade, e por ser contraditória, essa informação poderá vir a revelar-se, relevante para especialistas no domínio. Ela decorreu dos dados e se a validade destes não for colocada em causa, então este será um indicador de que uma análise mais cuidada, por parte dos peritos no domínio, poderá trazer novos resultados e mais conclusivos.

É também possível observar que valores elevados de NH4 influenciam, de alguma maneira, a concentração elevada desta alga, o que aliás faz todo o sentido, já que o azoto amoniacal (NH4) indica a contaminação/poluição da água e pode provocar o desenvolvimento de bactérias que conferem uma má qualidade à água. Em ambientes adversos, as algas conseguem sobreviver consumindo os escassos recursos da água. Uma outra conclusão a retirar do modelo prende-se com o facto de valores elevados de Cloro (Cl) serem indicadores da presença de algas. É possível que, pela sua presença deteriorar a qualidade da água, as necessidades de desinfecção sejam maiores.

Este modelo representa algum do conhecimento sobre o aparecimento de *blooms* da alga a_1 que as árvores do CART não foram capaz de induzir.

A árvore apresentada na Figura 4.12 corresponde à árvore obtida para o mesmo problema mas sem *pruning*, isto é, $f = 1$.

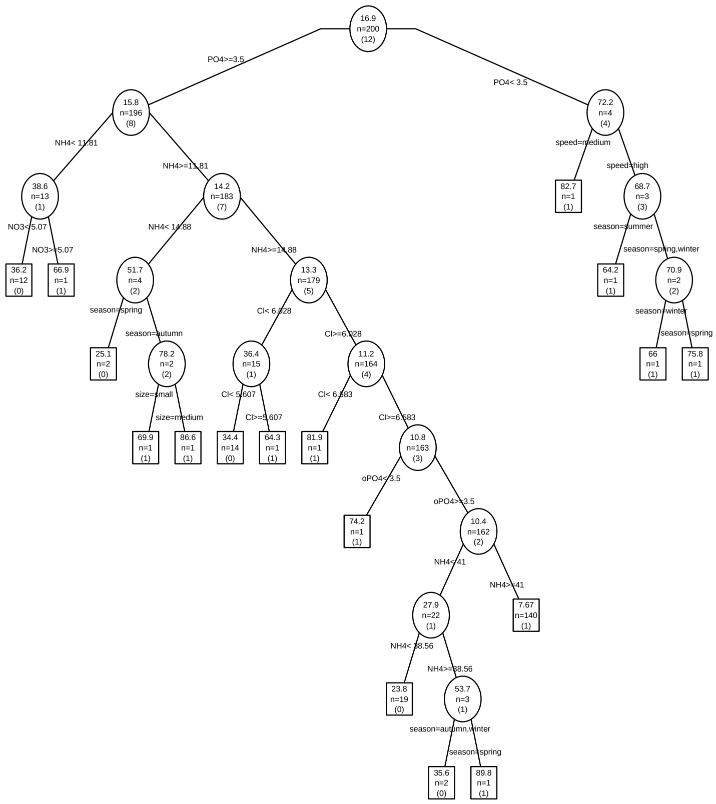


Figura 4.12: Árvore de regressão obtida pelo nosso método sem *pruning*, para a previsão de concentração da alga a_1.

O esforço adicional do método concentrou-se em melhorar a *precision* da partição "pura" de quatro *outliers* extremos, já que o seu *recall* é muito bom. Baseado na velocidade do rio e na estação do ano, foram criadas quatro partições a partir dessa, cada uma delas contendo um dos *outliers* extremos. Segundo o modelo, os maiores *blooms* registam-se quando a velocidade do rio é média ou, em alternativa, a velocidade do rio é elevada e a estação do ano é a Primavera.

Ainda que, à primeira vista, possa parecer que estas duas árvores estão a fazer *overfitting* aos dados, o facto é que elas obtêm melhores valores estimados de $precision_{regr}$, $recall$ e *F-measure*, do que as árvores geradas pelo CART (1-SE) e pelo CART (0-SE), como veremos na Capítulo 5, Secção 5.1.

A questão da interpretabilidade dos modelos gerados pelo nosso método é abordada novamente no Apêndice A, no contexto de outras aplicações.

Capítulo 5

Análise Experimental do Método

Uma vez descrito o problema e o método proposto para a sua resolução, há que fazer uma avaliação dos resultados obtidos. Tentou-se, por esta razão obter estimativas fiáveis dos resultados do método em relação a um conjunto de medidas, usando vários conjuntos de dados.

Neste capítulo pretende-se comparar o nosso método com métodos *standard* de árvores de regressão disponibilizados através do sistema CART (Breiman et al., 1984). Descreve-se a metodologia experimental por nós utilizada e o conjunto de dados sobre os quais se realizaram as experiências. Faz-se também alguma análise comparativa dos resultados obtidos. Por último, é feito um pequeno estudo relativo à sensibilidade do nosso método face a alguns dos seus parâmetros.

5.1 Metodologia experimental

Para a análise experimental do método, recorremos à sua aplicação sobre vinte e cinco domínios reais cuja variável objectivo é contínua e possui valores extremos raros, isto é, domínios onde podem ser estabelecidos objectivos semelhantes àqueles que serviram de motivação à concepção deste método. Os conjuntos de dados encontram-se descritos na Tabela 5.1. Sete correspondem ao problema da previsão de *blooms* de sete comunidades de algas nocivas, já referido no Capítulo 4, Secção 4.1 (algae1 $\rightarrow$ algae7). Os restantes conjuntos de dados dizem respeito a aplicações obtidas de repositórios *standard* de algumas aplicações comerciais[1] e de acesso restrito[2].

Conjunto de Dados	# exs.	# atributos		$\vert O\vert$	$\vert O_L\vert$	$\vert O_H\vert$
		contínuos	nominais			
servo	167	0	4	30	0	30
triazines	186	60	0	9	9	0
algae1	200	8	3	12	0	12
algae2	200	8	3	10	0	10
algae3	200	8	3	22	0	22
algae4	200	8	3	16	0	16
algae5	200	8	3	13	0	13
algae6	200	8	3	19	0	19
algae7	200	8	3	21	0	21
machine.cpu	209	6	0	23	0	23
china	217	9	0	19	0	19
Boston	506	13	0	37	0	37
onekm	710	14	3	8	6	2
cw.drag	1449	12	2	52	1	51
co2.emission	1558	19	8	23	0	23
acceleration	1732	11	3	26	0	26
available.power	1802	7	8	121	0	121
bank8FM	4499	8	0	69	0	69
delta.ailerons	7129	5	0	107	41	66
ibm	8166	10	0	325	140	185
cpu.small	8192	12	0	430	430	0
delta.elevators	9517	6	0	132	60	72
cal.housing	20460	8	0	1071	0	1071
add	30000	10	0	63	0	63
fried.delve	40768	10	0	25	6	19

Tabela 5.1: Descrição dos conjuntos de dados

A fim de obter um termo de comparação submetemos estes domínios, não só à aplicação do nosso método, mas também à aplicação do método de árvores de regressão *standard*.

Para a árvore de regressão *standard* usamos o *package rpart* da linguagem R[3] com *pruning* 1-SE por *error complexity* com *cross-validation*, conforme apresentado por Breiman et al. (1984).

O nosso método foi implementado aproveitando o código base fornecido pelo *package rpart* do R, adaptando o critério de *split*, o critério de paragem de crescimento da árvore, introduzindo novos parâmetros ao sistema (β e f) e não fazendo qualquer *post-pruning*. Por defeito o nosso método assume as valorizações $\beta = 1$, atribuindo a mesma importância

[1] Disponíveis em *http://www.niaad.liacc.up.pt/~ltorgo/Regression/Datasets/*
[2] Disponíveis em *http://www.liacc.up.pt/ML/METAL/*
[3] *http://www.R-project.org*

à *precision* e *recall* no cálculo da *F-measure* durante a concepção do modelo, e $f = 0.7$ como *threshold* para o critério de paragem da árvore.

5.1.1 Medidas de avaliação

De acordo com o que foi descrito no capítulo anterior, as medidas de avaliação utilizadas foram, *precision*, *recall* e *F-measure*. Com os conjuntos de dados presentes na Tabela 5.1, corremos cinco repetições de *10-fold cross-validation*, a fim de obter uma estimativa fiável das medidas *precision*, *recall* e *F-measure* obtidas pelo nosso método e pelas árvores de regressão *standard*. Para a *F-measure* utilizada na avaliação dos resultados, fixámos $\beta = 1$. Desta forma, atribuímos igual peso à *precision* e ao *recall*.

5.2 Avaliação de resultados

Dos valores obtidos para as medidas *precision*, *recall* e *F-measure*, para cada um dos conjuntos de dados, apenas nos interessam destacar aqueles que apresentam um diferença significativa quando comparados ao método *standard* utilizado nas árvores de regressão. Assim, recorremos aos *paired t-tests* para atribuir valores de significância às diferenças observadas entre os valores retornados nos 5x10 *folds* corridos, obtidos por amostragem, com o mesmo *seed* (o que viabiliza a aplicação do *paired t-test*).

Este teste permite saber se a diferença entre as médias de duas amostras é suficientemente grande, tal que que nos dê um certo grau de confiança de que são provenientes de populações com médias, necessariamente, diferentes.

A diferença entre as duas médias é significativa a um certo nível de probabilidade de acordo com o $p - value$. O $p - value$ é um valor retornado pelo *paired t-test*, que corresponde à probabilidade da diferença das médias reais das populações seja, de facto, zero. Se p é maior do que 0.05, então é porque existe mais do que uma possibilidade em vinte das duas médias serem, de facto, iguais e a diferença apenas decorrer de erros de amostragem. Se p é menor ou igual a 0.05, então existe menos de uma possibilidade em vinte das duas médias serem iguais e se tratarem apenas de amostras não representativas. Existe um maior grau de certeza de que as médias são tão diferentes como se tivessem sido obtidas de diferentes populações. Mais do que isso, se p for menor ou igual a 0.001, então, existirá menos de uma possibilidade em 1000 de que as duas médias sejam iguais.

Este teste, tal como outros testes estatísticos, não permite provar ou rejeitar nada. Um resultado significativo a um nível de probabilidade de 95% diz-nos apenas que os nossos dados são suficientes para suportar uma conclusão com 95% de confiança, existindo uma possibilidade em vinte da nossa conclusão estar errada. O mesmo se aplica para um nível de probabilidade de 99.9%, onde existe apenas uma possibilidade em 1000 da nossa conclusão estar errada.

Por convenção, estabeleceu-se que as diferenças entre médias a um nível de 95% são significativas, e diferenças entre médias a um nível de 99.9% são muito significativas.

As diferenças significativas evidenciadas pelos resultados são representadas através de diferentes sinais conforme descrito na Tabela 5.2: dependem do nível de probabilidade e do carácter da diferença ser benéfica ou prejudicial face ao nosso método. No caso de não existir qualquer diferença significativa não é utilizado nenhum sinal.

Tanto a *precision*, como o *recall* e a *F-measure* constituem medidas que pretendemos maximizar. Portanto um sinal – (+) associado ao nosso método significa que o valor por nós obtido para uma dessas medidas é significativamente melhor (pior) do que aquele obtido pelo método que estamos a comparar.

Sinal	**Descrição**	
++	diferença significativa a nível de 99.9%	método que serviu de comparação
+	diferença significativa a nível de 95%	possui valor signif. superior
s/ sinal	diferença não significativa	
–	diferença significativa a nível de 95%	método que serviu de comparação
– –	diferença significativa a nível de 99.9%	possui valor signif. inferior

Tabela 5.2: Sinais de significância atribuídos pelo *paired t-test.*

Os resultados obtidos pelo CART (1-SE) e pelo nosso método para cada um dos conjuntos de dados da Tabela 5.1 são apresentados na Tabela 5.3.

Da observação dos resultados é nos possível constatar uma clara vantagem do nosso método em relação ao CART (1-SE) em termos de *F-measure*, a medida utilizada no crescimento das árvores no nosso método. Se olharmos para o *recall*, os resultados são ainda mais

Conjuntos de Dados	***precision$_{regr}$***		***recall***		***F-measure***	
	CART $(1-SE)$	Nosso Método $(\beta=1)$	CART $(1-SE)$	Nosso Método $(\beta=1)$	CART $(1-SE)$	Nosso Método $(\beta=1)$
servo	0.7616598	0.5062348 $++$	0.8713333	0.9800000 $--$	0.8053263	0.6612034 $++$
triazines	0.1474294	0.2988911 $-$	0.0400000	0.2233333 $--$	0.0371048	0.2282423 $--$
algae1	0.2947144	0.4180734 $-$	0.0000000	0.3366667 $--$	0.0000000	0.3279042 $--$
algae2	0.0000000	0.1244440 $-$	0.0000000	0.0900000 $-$	0.0000000	0.0685473
algae3	0.0000000	0.1190006 $-$	0.0000000	0.1906667 $--$	0.0000000	0.1186311 $-$
algae4	0.0000000	0.1210823 $--$	0.0000000	0.1150000 $-$	0.0000000	0.0636982 $-$
algae5	0.0000000	0.1571761 $--$	0.0000000	0.1366667 $-$	0.0000000	0.0970766 $-$
algae6	0.0000000	0.1029830 $-$	0.0000000	0.1606667 $--$	0.0000000	0.0805630 $-$
algae7	0.0127059	0.0443510	0.0100000	0.0590000	0.0111917	0.0476719
machine.cpu	0.5517675	0.6870716 $--$	0.8186667	0.8950000 $-$	0.6266528	0.7591475 $--$
china	0.0000000	0.0450076 $-$	0.0000000	0.0530000 $-$	0.0000000	0.0401994 $-$
Boston	0.8243590	0.8192831	0.7580000	0.7595000	0.7675711	0.7346941
onekm	0.4001506	0.3350779	0.0166667	0.2600000 $--$	0.0059831	0.2701772 $--$
cw.drag	0.9250750	0.8355772 $++$	0.8419906	0.9801429 $--$	0.8734481	0.8976743
co2.emission	0.8052664	0.8068307	0.4213333	0.6326667 $--$	0.4770925	0.6775423 $--$
acceleration	0.8964751	0.8652793 $+$	0.5600000	0.7963333 $--$	0.6287861	0.7864996 $-$
available.power	0.9668409	0.8567897 $++$	0.9091224	1.0000000 $--$	0.9353684	0.9217586
bank8FM	0.9781688	0.9425374 $++$	0.6532389	0.6798909	0.7592038	0.7594744
delta.ailerons	0.6442916	0.5531748 $++$	0.1293077	0.1760511 $-$	0.1933579	0.2462873
ibm	0.0000000	0.0104953 $--$	0.0000000	0.0055804 $-$	0.0000000	0.0035846 $-$
cpu.small	0.9939290	0.9855883 $++$	0.8519882	0.8886580 $--$	0.9165213	0.9341849 $--$
delta.elevators	0.5980803	0.5978288	0.0419848	0.1620447 $--$	0.0719734	0.2445691 $--$
cal.housing	0.8425739	0.5871524 $++$	0.3140477	0.4608993 $--$	0.4561053	0.5155620 $--$
add	0.93204126	0.78503524 $++$	0.0000000	0.1240159 $--$	0.0000000	0.1823070 $--$
fried.delve	0.8427616	0.5371731 $++$	0.1141905	0.09674603	0.1356259	0.1161262

Tabela 5.3: Avaliação de resultados obtidos pelo CART vs Nosso Método.

vantajosos para o nosso método. Os modelos gerados pelo nosso método parecem capturar uma boa proporção dos *outliers* extremos existentes na maior parte dos domínios.

No entanto, os resultados obtidos para a medida $precision_{regr}$ não se apresentam tão vantajosos como os obtidos para as medidas de *recall* e *F-measure*. Os ganhos obtidos pelo nosso método são em menor número e assumem valores baixos, sendo, inclusive, registadas algumas perdas significativas em relação à $precision_{regr}$ obtida pelo CART (1-SE), especialmente para os grandes conjuntos de dados.

Uma das nossas suspeitas para a explicação destes resultados reside no critério de paragem. As árvores concebidas pelo nosso método poderão estar a ser sub-ajustadas aos dados, no sentido em que não estão a ser suficientemente precisas. Se, de facto, for essa a razão então o *tuning* do *threshold* f poderá melhorar os resultados obtidos para a medida $precision_{regr}$. Na Subsecção 5.3.1 exploramos um pouco mais esta hipótese no contexto da análise de sensibilidade do método.

Existem ainda algumas observações a fazer aos resultados: para alguns conjuntos de dados vemos uma $precision_{regr}$ de zero. Estes correspondem aos casos onde $NMSE_O$ é superior a um, isto é, o modelo está a fazer pior do que simplesmente prever a média. Nesse caso a $precision_{regr}$ estabelecida é zero, tal como definido no capítulo anterior, Secção 4.4. Também para o *recall* surgem alguns zeros. Tal acontece quando os modelos não prevêem nenhum *outlier* extremo, isto é, não existe nenhuma folha cujo valor seja um *outlier* extremo.

5.2.1 Discussão

Da observação da Tabela 5.3 é óbvia a tendência das perdas em termos de $precision_{regr}$ recaírem sobre os maiores conjuntos de dados. Isto poderá estar relacionado com o problema, já abordado no capítulo anterior, dos *small-disjuncts* (Holte et al., 1989; Weiss, 1995; Weiss e Hirsh, 1998b, 2000b).

As árvores constituem um dos algoritmos de aprendizagem que criam conceitos formados por um conjunto de disjunções. A cobertura dessas disjunções está associada ao número de exemplos do conjunto de treino que cobrem. À semelhança do que acontece noutros algoritmos, as árvores tendem a constituir *large-disjuncts* para cobrir os casos mais gerais e *small-disjuncts* para cobrir os casos mais raros.

Estudos realizados por Weiss e Hirsh (1998b, 2000b) indicam que o tamanho do conjunto de treino exerce uma influência sobre o processo de aprendizagem quando se está na presença simultânea de casos raros e comuns.

Segundo experiências realizadas por Weiss (1995), grandes conjuntos de dados trazem uma dificuldade acrescida à aprendizagem a partir de uma distribuição desequilibrada dos dados. Nestes casos, o efeito dos *small-disjuncts* sobre a estimativa do erro é mais gravosa. Abaixo de um determinado tamanho máximo de conjunto de dados, o contrário acontece. Quando o conjunto de treino é pequeno, mesmo para distribuições desequilibradas dos dados, os casos mais comuns são ainda assim possíveis de ser bem aprendidos, dotando o modelo de, talvez, uma maior capacidade de selectividade. No entanto, para grandes conjuntos de dados, não é possível obter mais exemplos do fenómeno raro cuja aprendizagem se mantém, assim, deficiente. Neste último contexto será, possivelmente, difícil dotar o modelo de uma maior sensibilidade face à presença de um fenómeno raro. Sendo assim, à medida que o tamanho do conjunto de dados cresce, os casos raros, e portanto os *small-disjuncts*, são responsáveis por uma cada vez maior percentagem do erro do modelo. Uma vez que o nosso problema consiste em prever um fenómeno raro, não podemos simplesmente eliminar os *small-disjuncts*: eles poderão estar na base dos bons valores de *recall* obtidos. Contudo, parecem ter um efeito penalizador sobre a *precision*.

Esta pode ser uma das razões que está na base da explicação dos resultados obtidos para a medida $precision_{regr}$. No entanto, é possível que não seja a única. Tal como já referimos, o critério de paragem da árvore que definimos poderá ter de ser alvo de algum *tuning*, ou simplesmente, não ser o mais adequado. Para além disso, o nosso método é sensível, não só ao critério de paragem (estabelecido pelo *threshold* f), mas também ao parâmetro β que define a importância relativa do *recall* face à *precision*. A fim de tentar perceber a sensibilidade do método a estes parâmetros iniciámos uma análise exploratória. Alguns dos resultados são apresentados de seguida na Secção 5.3.

5.3 Análise de sensibilidade do método

A avaliação dos resultados feita na secção anterior, demonstrou que o nosso método apresenta não só uma boa *performance* em relação à *F-measure*, mas sobretudo, em relação ao *recall*. No entanto, colocaram-se algumas questões quanto a alguns maus resultados obtidos para a medida $precision_{regr}$. Nesta secção descrevemos algumas experiências iniciadas no sentido de tentar colmatar essa deficiência, explorando a sensibilidade do método aos parâmetros f e β.

5.3.1 Critério de paragem

A fim de observar até que ponto o critério de paragem que utilizamos para o nosso método pode estar a ser responsável por um efeito negativo sobre a medida $precision_{regr}$, decidimos comparar o nosso método com o sistema CART (Breiman et al., 1984) com diferentes níveis de *pruning*: 1-SE, 0-SE e sem qualquer *pruning*. Esta comparação serve também o propósito de verificar em que medida o nosso método poderá ser considerado melhor ou pior em relação a uma árvore *standard* sem *pruning*.

As mesmas experiências realizadas para o CART (1-SE) e para o nosso método (*c.f.* Tabela 5.3), foram repetidas para o CART (0-SE) e para o CART sem *pruning*. Os resultados obtidos estão apresentados nas Tabelas B.1 e B.2 do Apêndice B. Contabilizámos o número de resultados significativos (perdas e ganhos) e não significativos, denotados por N.S., e sintetizamos essa informação na Tabela 5.4.

Comparação CART (1-SE) **vs Nosso Método** ($\beta = 1$)							
Medidas	++	+	Subtotal	N.S.	–	– –	Subtotal
$precision_{regr}$	9	1	**10**	5	6	4	**10**
recall	0	0	**0**	4	7	14	**21**
F-measure	1	0	**1**	8	7	9	**16**
Comparação CART (0-SE) **vs Nosso Método** ($\beta = 1$)							
Medidas	++	+	Subtotal	N.S.	–	– –	Subtotal
$precision_{regr}$	9	3	**12**	9	3	1	**4**
recall	0	0	**0**	8	8	9	**17**
F-measure	1	1	**2**	12	6	5	**11**
Comparação CART (*s/ pruning*) **vs Nosso Método** ($\beta = 1$)							
Medidas	++	+	Subtotal	N.S.	–	– –	Subtotal
$precision_{regr}$	11	4	**15**	10	0	0	**0**
recall	2	4	**6**	13	6	0	**6**
F-measure	4	3	**7**	17	1	0	**1**

Tabela 5.4: Resumo de resultados obtidos pelo sistema CART vs Nosso Método.

A primeira coisa a destacar da observação desta tabela é que, nos vinte e cinco domínios testados, o nosso método consegue sempre ganhar em termos de *F-measure* e, sobretudo, em termos de *recall*, às árvores CART com *pruning*. Contudo, no que toca a $precision_{regr}$, o cenário apresenta-se mais desfavorável. À medida que o nível de *pruning* do CART diminui, o número de perdas do nosso método aumenta, chegando, inclusive, a não obter nenhuma vitória quando comparado a árvores sem *pruning*. Esta situação leva-nos a crer que o critério de paragem do nosso método talvez necessite de alguma "afinação".

Para uma melhor contextualização do nosso método face ao CART (1-SE), CART (0-SE) e CART sem *pruning*, criámos um *mosaic plot* a partir dos valores contidos na tabela acima. O gráfico resultante é apresentado na Figura 5.1.

Este gráfico representa a *performance* do nosso método comparada à *performance* do CART com três níveis de *pruning* diferentes e segundo três medidas de avaliação. As três medidas de avaliação são $precision_{regr}$, *recall* e a *F-measure* e representam as três colunas principais presentes no gráfico. Os três níveis de *pruning* do CART utilizados para comparação são

1-SE, 0-SE e sem *pruning* e estão representados por três cores diferentes. No gráfico existem, ainda, três linhas: a primeira representa os ganhos significativos do nosso método (– – /–), a segunda representa os resultados não significativos obtidos (N.S.) e a última representa as perdas significativas do nosso método (+ + /+).

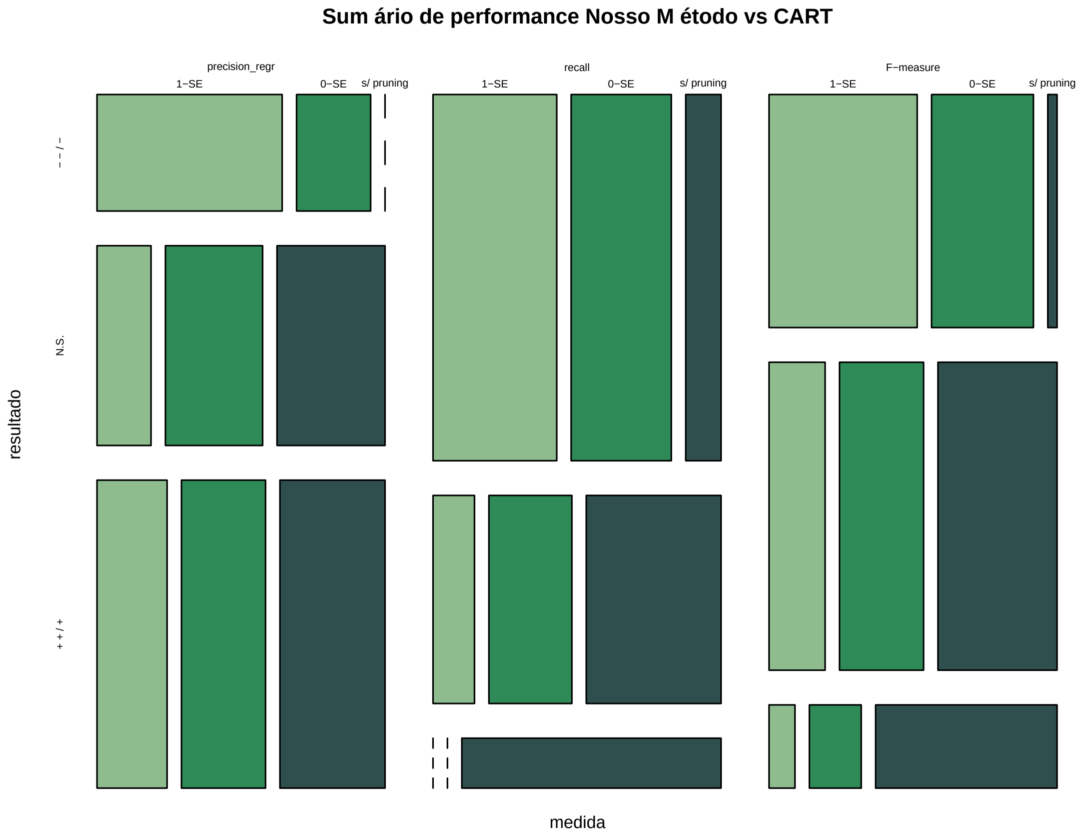


Figura 5.1: Comparação de *performance* do Nosso Método face ao CART com diferentes níveis de *pruning*.

A informação contida neste gráfico deverá ser analisada em duas fases. Numa primeira fase, e fazendo uma avaliação por coluna, observamos que, qualquer que seja o termo de comparação, o nosso método apresenta diferentes distribuições de ganhos e perdas significativas de acordo com a medida. Para a medida $precision_{regr}$, por exemplo, grande parte dos resultados obtidos foram de perda significativa do nosso método. Uma situação oposta acontece na coluna seguinte, quando se analisa o *recall*: aí é evidente que a maioria dos resultados obtidos representam ganhos significativos do nosso método. É possível ainda notar a influência mínima que as perdas significativas têm nesta medida. Por último, a terceira coluna apresenta os resultados obtidos para a *F-measure*, a medida com base na qual construímos os nossos modelos. O gráfico leva-nos a fazer uma apreciação razoável do nosso método para as experiências realizadas. Ainda que o número de ganhos significativos seja, ligeiramente, inferior ao número de resultados não significativos, o número de perdas significativas do nosso método constituem apenas uma pequena parte dos resultados.

Feita esta primeira análise, podemos agora estudar a influência que cada um dos três níveis de *pruning* do CART exerce sobre as medidas avaliadas. A observação do gráfico deve agora ser feita: por linha, para especificar o tipo de resultado; por coluna, para especificar a medida em análise; e por cor para especificar o nível de *pruning* do CART que serviu de termo de comparação. Assim, e seguindo este procedimento, podemos constatar, por exemplo, que em relação à medida $precision_{regr}$, o CART sem *pruning* tem uma influência nula sobre os ganhos significativos do nosso método, sendo o CART (1-SE) o principal responsável por eles. No caso concreto do *recall*, todas as nossas perdas significativas tiveram origem no CART sem *pruning*. Uma situação semelhante acontece para a medida *F-measure*. De facto, grande parte das nossas perdas significativas dizem respeito à comparação com árvores sem *pruning* ao mesmo tempo que grande parte dos nossos ganhos significativos são relativos às árvores com mais *pruning*, isto é, CART (1-SE).

Este resultado seria previsível dado que, quão menos *pruning* sofrer a árvore, mais específica ela se tornará e, portanto, mais exemplos de treino ela cobrirá mesmo aqueles que constituem *outliers*, neste caso, de valor extremo.

Neste contexto, será então interessante comparar o tamanho das árvores geradas pelo nosso método e pelo CART (1-SE), CART (0-SE) e CART sem *pruning*. No processo 5x10 *cross-validation* foram guardados os tamanhos das árvores geradas, sendo que na Figura 5.2 se apresenta o tamanho médio das árvores obtidas para cada conjunto de dados. As médias e desvios-padrão do tamanho das árvores de todos os conjuntos de dados encontram-se no Apêndice B, Tabela B.3.

Da observação dos gráficos da Figura 5.2, é possível constatar que as árvores geradas pelo CART sem *pruning* são significativamente maiores em relação a todas as restantes árvores e, conforme já foi referido, isso traz custos a nível de interpretabilidade. As árvores mais pequenas são as árvores geradas pelo CART (1-SE). Para conjuntos de dados mais pequenos o nosso método apresenta modelos relativamente grandes mas, em contrapartida, para grandes conjuntos de dados ele apresenta modelos, geralmente, inferiores aos gerados pelo CART (0-SE) e portanto, à partida, mais interpretáveis.

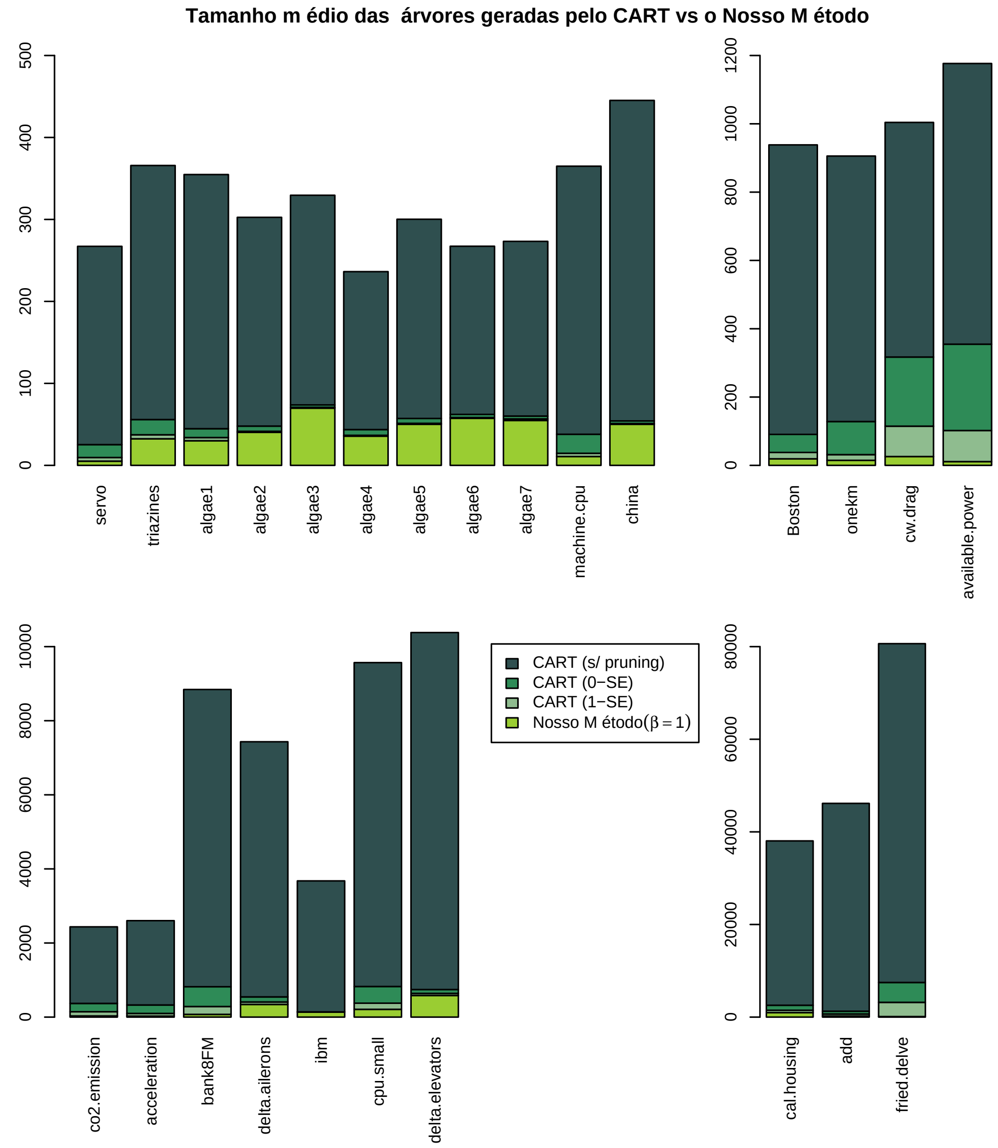


Figura 5.2: Comparação do tamanho médio de árvores geradas pelo CART vs o Nosso Método

Vejamos o que acontece no caso do problema `algae1`: para além do nosso método obter valores significativamente melhores do que o CART sem *pruning* para o *recall* e para a *F-measure*, tal como mostra a Tabela B.2, o tamanho da nossa árvore é cerca de 10% do tamanho da árvore gerada pelo CART sem *pruning*. As árvores geradas pelo nosso método, neste domínio, serão mais interpretáveis.

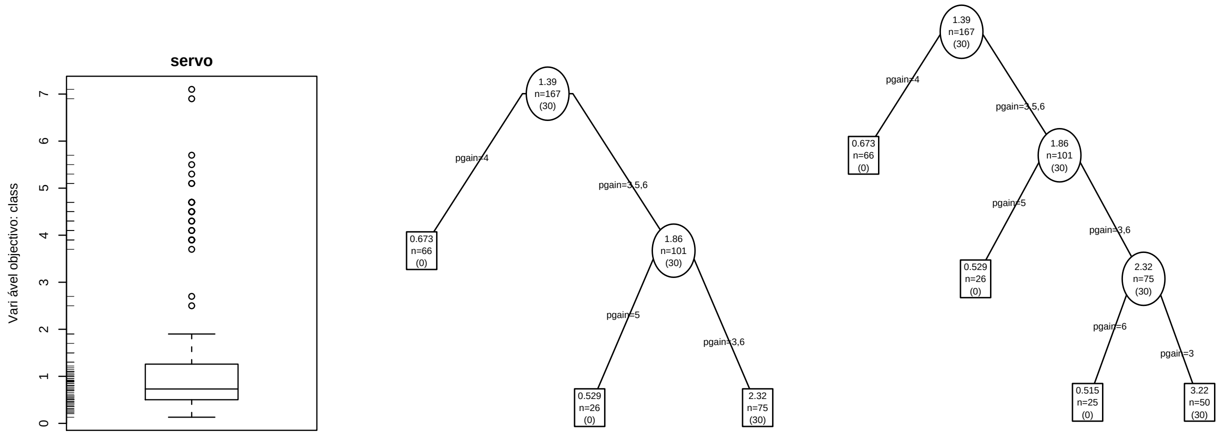
Um outro facto interessante é aquilo que acontece para o conjunto de dados ibm. Este constitui um problema sobre previsão de retornos dos preços de fecho da IBM no mercado da bolsa e de reconhecida dificuldade de modelação para os algoritmos de aprendizagem mais comuns. Os sistemas *standard* CART (1-SE) e CART (0-SE) apenas conseguem gerar uma folha com a média de todos os valores da variável objectivo. O tamanho médio das árvores sem *pruning* geradas para este problema, possuem um número de nós na ordem dos 3000. O nosso método por seu lado conseguiu gerar modelos mais compactos de em média 138 nós. Pelos resultados apresentados na Tabela B.2 do Apêndice B, é possível constatar que, ainda assim, o nosso método conseguiu obter valores de $precision_{regr}$ ligeiramente melhores, perdendo um pouco em termos de *recall*.

No entanto, ainda que certamente menores, as nossas árvores não apresentam, na sua maioria, melhores valores de *performance* nos domínios avaliados, quando comparadas com as árvores geradas pelo CART sem *pruning*.

Dados estes resultados, poderemos supor que um dos problemas do nosso método estará relacionado com questões de *pruning*, que no nosso caso dizem respeito ao critério de paragem escolhido.

A escolha do parâmetro f poderá, em alguns casos, fazer a diferença. Tomemos o exemplo do conjunto de dados servo. Neste conjunto de dados obtivemos maus resultados de $precision_{regr}$ em comparação com a árvore com mais *pruning* (CART (1-SE)) (*c.f.* Tabela 5.3). Por defeito, o nosso método assume o valor 0.7 para o parâmetro f. Experimentámos obter, através do nosso método, a árvore gerada para o conjunto de dados servo, sem *pruning*, isto é, com o parâmetro f igual a um. As árvores obtidas pelo nosso método, por defeito e sem *pruning*, são apresentadas nas Figuras 5.3 (b) e (c), respectivamente.

Ambas as árvores conseguem constituir uma folha com uma partição que contém todos os *outliers* extremos presentes no conjunto de dados. Quando não é imposto nenhum critério de paragem à árvore, senão alcançar o valor máximo da *F-measure* num nó ou parar quando não existirem mais *outliers*, a nossa árvore cresce apenas mais um nível, atingindo um melhor valor de *recall*. Ao olharmos para a distribuição dos *outliers* extremos apresentados na Figura 5.3 (a), facilmente nos apercebemos de que, para além do valor de *recall* ter sido melhorado, o mesmo aconteceu, certamente, com o valor de $precision_{regr}$. No entanto, e caso estivéssemos interessados numa maior precisão na previsão dos valores, bastava que, para além de estabelecer $f = 1$, impuséssemos $\beta = 0.5$ para assistirmos a um crescimento exponencial da árvore. Este constitui um exemplo onde a escolha do parâmetro f poderá marcar a diferença.



(a) *box plot* da variável objectivo

(b) árvore c/ $f = 0.7$

(c) árvore c/ $f = 1$

Figura 5.3: Árvores obtidas pelo Nosso Método para o problema servo com $f = 0.7$ e $f = 1$.

Decidimos, então, fazer um pequeno estudo da influência da variação do parâmetro f na avaliação do nosso método quando comparado com CART sem *pruning*. Repetimos as experiências com valores de f mais elevados, de forma a possibilitar que a paragem do crescimento das árvores ocorresse mais tarde. Para tentar evitar a constituição de árvores sobre-ajustadas, corremos as experiências para valores de f iguais a 0.85 e 0.95. Os resultados são apresentados nas Tabelas B.4, B.5 e B.6 do Apêndice B.

Contrariamente ao que seria de esperar, maiores valores de f não conduziram a resultados globalmente melhores, especialmente para grandes conjuntos de dados. Isto leva-nos a crer que, nos domínios mais problemáticos, as árvores não estarão a parar devido parâmetro f mas sim pelo facto de chegar a partições sem *outliers* extremos. Nesse caso, o parâmetro f poderá não ser o único responsável pelos resultados obtidos.

5.3.2 O parâmetro β

O parâmetro β estabelece o compromisso entre as medidas *recall* e *precision*. É com base nesse compromisso que o nosso método orienta o crescimento das árvores na escolha dos melhores *splits*. Por defeito, o nosso método atribui igual importância ao *recall* e à *precision*, com $\beta = 1$.

A fim de ver como o método reagia à variação do parâmetro β, repetiram-se as experiências para $\beta = 0.5$ e $\beta = 2$. Os resultados estão apresentados nas Tabelas B.7, B.8 e B.9 do Apêndice B. As Figuras 5.4, 5.5 e 5.6, abaixo apresentadas, representam a distribuição dos valores obtidos para as medidas $precision_{regr}$, *recall* e *F-measure* durante o processo de 5x10 *cross-validation*.

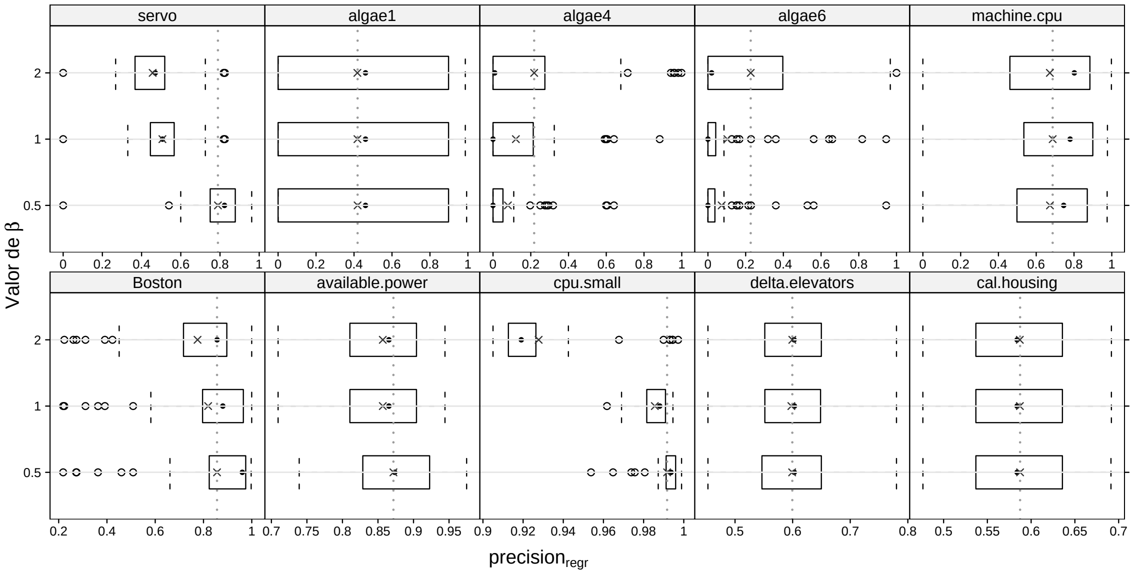


Figura 5.4: Variação do parâmetro β: valores de $precision_{regr}$ obtidos no 5x10 *cross-validation*

Uma análise sumária permite denotar um ganho, ainda que pouco significativo, da medida $precision_{regr}$ para $\beta = 0.5$. Aliás este seria o comportamento esperado para todos os domínios, já que fazendo $\beta = 0.5$ se está atribuir ao *recall* metade da importância da *precision* na escolha do melhor *split*. Estranhamente, o comportamento observado para alguns domínios é exactamente o oposto. A Figura 5.4 apresenta conjuntos de dados nessas condições, sendo o algae4 e algae6 os casos mais evidentes. Mas, se no caso de $\beta = 2$ também seria de esperar melhores valores de *recall*, por atribuir ao *recall* o dobro da importância da *precision*, e se isso acontece na maior parte dos casos, em alguns casos são obtidos piores valores de *recall* do que para parametrizações $\beta = 1$ ou $\beta = 0.5$. Nesta situação encontram-se domínios como machine.cpu e Boston, apresentados na Figura 5.5. Todas estas excepções contra-intuitivas não nos permitem extrair um padrão de comportamento do nosso método.

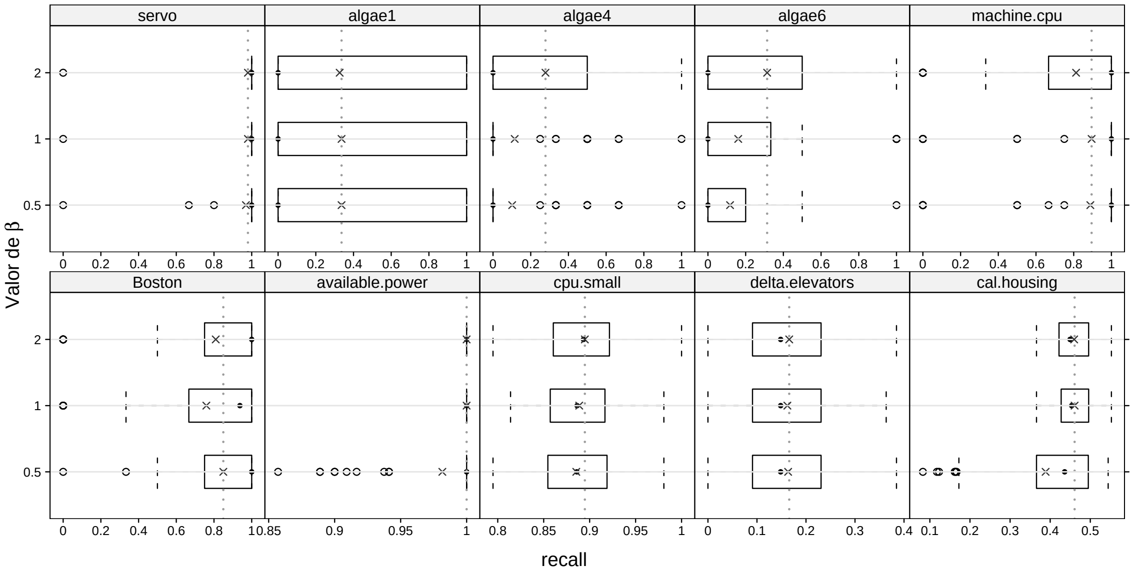


Figura 5.5: Variação do parâmetro β: valores de *recall* obtidos no 5x10 *cross-validation*

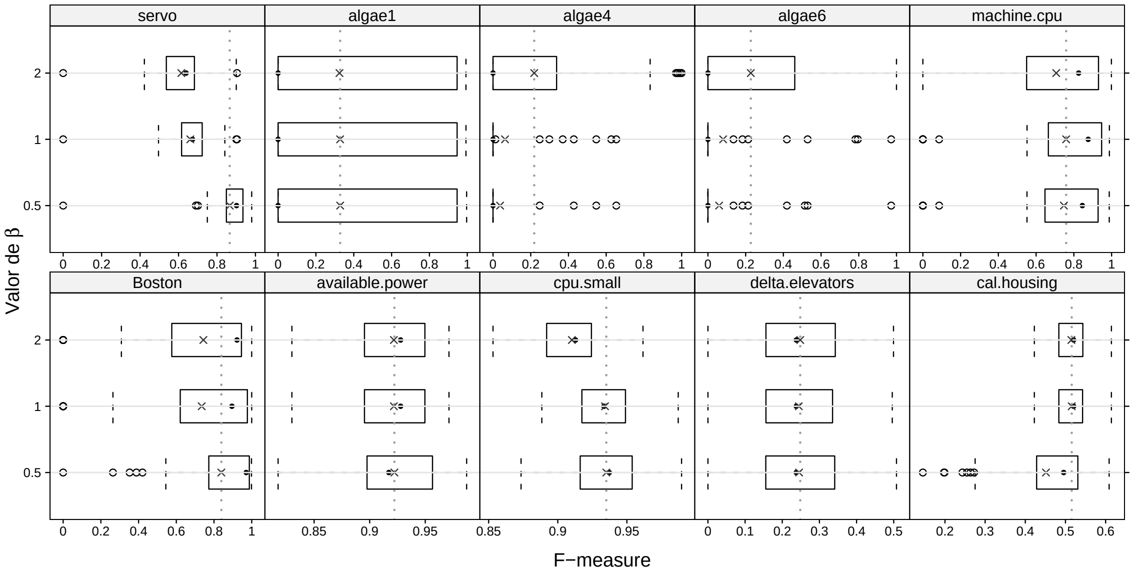


Figura 5.6: Variação do parâmetro β: valores de *F-measure* obtidos no 5x10 *cross-validation*

Domínios como algae1 e delta.elevators, não parecem ser sensíveis à variação testada para o parâmetro β. Tal como é possível observar na Figura 5.6, não existe uma tendência bem definida. Os resultados destas experiências de variação do parâmetro não foram, portanto, muito conclusivas.

Contudo, a distribuição dos valores por cada umas das medidas em cada um dos *folds* apresenta, também ela, muitos *outliers* extremos. Isso leva-nos a reflectir sobre o método como deveria ter sido implementado o processo de *cross-validation* em domínios como estes.

Nas nossas experiências utilizámos uma estratégia de *cross-validation* para obter estimativas mais fiáveis das medidas $precision_{regr}$, *recall* e *F-measure*. No entanto, deveríamos ter recorrido a um *cross-validation* estratificado. Uma das principais características do fenómeno que pretendemos prever é a sua raridade. Assim sendo, teremos de assegurar exemplos da sua ocorrência, quer na fase de treino, quer na fase de teste. Sendo nosso objectivo aprender a descrição de fenómenos descritos por *outliers* extremos, facilmente poderiam surgir *folds* sem nenhum *outlier*. Estes *folds*, quando usados como conjunto de teste, dariam origem a um valor zero para a $precision_{regr}$ e para o *recall* e, consequentemente, para a *F-measure*, dada a inexistência de *outliers* extremos. Da mesma forma, poderíamos cair na situação oposta em que esse mesmo *fold*, usado para estimação das medidas, possuiria todos os *outliers* extremos do conjunto de dados. Nesse caso, o método não disporia de exemplos para aprender a descrição do fenómeno. Estas situações enviesariam muito a estimativa das medidas e talvez esta seja a explicação que esteja na base da distribuição de valores de medidas apresentadas nas Figuras 5.4, 5.5 e 5.6. Uma solução passaria, tal como referimos, por recorrer à estratégia de *cross-validation* estratificado. No entanto, seria impraticável, senão vejamos: para assegurar que todos os *folds* possuiriam *outliers* extremos seria necessária a existência de pelo menos k *outliers* extremos para executar *k-fold cross-validation*. No contexto da raridade inerente aos *outliers*, isso poderá não ser possível.

Uma alternativa ao processo de *cross-validation* utilizado deveria ser estudada.

Em suma, os resultados obtidos pelas nossas experiências mostram uma vantagem do nosso método na previsão de *outliers* extremos em relação ao CART (1-SE). Existe, no entanto, ainda muito espaço para melhorias, especialmente no que toca à medida $precision_{regr}$. Em comparação com o CART sem *pruning*, o nosso método ainda que produzindo modelos mais pequenos e portanto mais interpretáveis, não consegue superar os resultados obtidos por este. A escolha do critério de paragem e o compromisso estabelecido entre $precision_{regr}$ e *recall* (parâmetro β) devem ser alvo de um estudo mais aprofundado, com particular atenção à metodologia experimental a usar de modo a garantir uma maior estabilidade de resultados nas diferentes repetições.

Capítulo 6

Conclusão

O problema de previsão de fenómenos raros é de reconhecida importância em algumas áreas de aplicação: sejam elas a economia, a saúde, a ecologia, a sociologia, etc. Por vezes, esses fenómenos são acompanhados pelo registo de valores numéricos extremos para determinada variável.

Tendo como motivação a resolução de um problema ecológico grave, propôs-se um novo método para fazer a previsão de fenómenos raros e de valor extremo. O problema tem a ver com o aparecimento de *blooms* de algas nocivas, sob circunstâncias ainda desconhecidas, em rios de onde se extrai água potável que abastece grandes centros urbanos. Estes *blooms* são raros e caracterizados por valores extremamente elevados de frequência das ditas algas. Concebemos um método que pretende ser mais eficaz na previsão de fenómenos como este, quando comparado com os métodos *standard* de aprendizagem automática.

Neste capítulo, voltamos a fazer referência ao novo método por nós proposto e aos resultados obtidos em diferentes experiências. Uma reflexão sumária dos resultados permite-nos identificar, por agora, alguns dos benefícios apresentados por este método, bem como indicar possíveis melhorias, para os seus "pontos fracos". É com base nas conclusões retiradas, que, de seguida, são traçadas algumas perspectivas futuras quanto ao possível prosseguimento deste trabalho.

6.1 Sumário

Ao longo de todos os capítulos anteriores, caracterizámos e contextualizámos o problema abordado nesta tese. Falámos de problemas relacionados e apresentámos a proposta de um novo método para a resolução do mesmo. Fizemos uma análise experimental do método, apresentando e discutindo os resultados obtidos de diferentes experiências realizadas.

Nesta secção, sumariamos aquilo que acreditamos serem as principais motivações e contribuições desta tese.

6.1.1 Motivação da tese

Uma das razões que esteve na base da concepção do novo método apresentado nesta tese, foi o facto de grande parte dos métodos de aprendizagem automática se encontrarem mais vocacionados para a aprendizagem de conceitos descritos por situações comuns ao invés de situações raras. Daí a sua reduzida eficácia na aprendizagem de conceitos cuja ocorrência é rara e portanto descrita por um conjunto de *outliers*.

Estabelecendo como meta a aprendizagem de conceitos que representam fenómenos de ocorrência rara, apresentam-se, logo à partida, dois desafios importantes na resolução deste tipo de problemas:

- a identificação dos *outliers* no contexto do domínio de aplicação;
- a escassez de número de exemplos para a aprendizagem.

No caso concreto dos problemas que pretendemos abordar, sabemos que os *outliers* assumem um valor numérico extremo. Com base neste pressuposto, pudemos, através do *box plot*, identificar os "nossos" *outliers*, pela distribuição dos valores da variável que contém a descrição do fenómeno raro a prever.

Uma vez ultrapassado o primeiro desafio, pudemos concentrarmos na concepção de um método de aprendizagem supervisionada. A investigação feita na área da previsão de fenómenos raros, e que tenta minimizar o problema da escassez de exemplos, recai, sobretudo, sobre domínios de classificação.

A principal motivação desta tese foi a concepção de um método com vista à previsão de fenómenos raros num domínio de regressão. Em particular, focámo-nos sobre os fenómenos raros que se apresentam sob valores extremos da variável objectivo. Assim sendo, estabelecemos como objectivos:

- a previsão de *outliers* extremos da variável objectivo com alguma precisão;
- a obtenção de modelos interpretáveis que consigam explicar sob que circunstâncias os *outliers* extremos ocorrem.

6.1.2 Contribuições da tese

O método de aprendizagem proposto no âmbito desta tese teve por paradigma base as árvores de regressão. No sentido de responder a uma classe específica de problemas, foi concebido um novo critério de *split*. Este novo critério pretende levar à concepção de árvores de regressão que maximizem quer o número de *outliers* extremos capturados pelo modelo quer a precisão na previsão dos seus valores. Para tal, recorremos a definições da área da *Information Retrieval* como *precision*, *recall* e *F-measure*, que preenchiam os nossos requisitos. Para que fossem aplicáveis ao nosso problema, as medidas de *precision* e *recall* foram por nós adaptadas. Em particular, a medida *precision* foi reformulada no sentido de dar uma resposta mais eficaz num âmbito de regressão como o dos nossos problemas, dando origem à medida $precision_{regr}$.

Uma vez definidas as medidas consideradas relevantes, procedemos à definição e implementação de um critério de *split* que obedecesse aos requisitos por nós impostos. A disponibilização do código fonte do *package rpart* da linguagem R, permitiu-nos tirar partido de todo sistema base que suporta a concepção de árvores, adaptando o critério de *split* e de paragem que guiam a construção das árvores de regressão e, desta forma, a constituição de um novo *package*. Este novo *package* acaba por representar um novo sistema de aprendizagem de árvores de regressão, que recebe, como *input*, dois novos parâmetros adicionais: f e β.

A concepção de um novo critério de *split*, embebido num poderoso método de aprendizagem automática a nível de eficiência e interpretabilidade como as árvores de regressão, permitiu a obtenção de modelos vocacionados para a previsão de *outliers* de valor extremo.

Para avaliar a qualidade do método concebido, face aos objectivos estabelecidos, este foi aplicado sobre vinte e cinco domínios reais diferentes e comparado com as árvores de regressão *standard* (Breiman et al., 1984).

Os resultados obtidos pelas experiências realizadas revelaram que, tipicamente, o nosso método consegue obter um bom desempenho em termos das medidas de avaliação consideradas relevantes para este tipo de problemas: *precision*, *recall* e *F-measure*.

Face ao sistema Breiman et al. (1984), o nosso método parece conseguir capturar uma boa percentagem de *outliers* extremos existentes na maioria dos domínios testados. Contudo, no que respeita à *precision* e especialmente para grandes conjuntos de dados, os resultados não se mostraram favoráveis ao nosso método. As árvores de regressão *standard* apresentaram-se como modelos mais precisos. Mesmo assim, os bons valores de *recall*, concedem ao nosso método bons valores de *F-measure*, tornando-o mais eficaz que o sistema Breiman et al. (1984) na previsão de *outliers* de valor extremo.

No entanto, este cenário favorável não se mantém face a todas as árvores geradas pelo sistema CART. À medida que é diminuído o nível de *pruning* das árvores CART, elas tornam-se mais específicas e, com isso, conseguem melhorar os seus resultados, sobretudo em termos de *precision*. Ainda assim, e apesar dos pobres valores de *precision*, os bons valores de *recall* obtidos pelo nosso método conduzem a uma apreciação razoável do mesmo. Na sua maioria, as comparações realizadas entre o nosso método e o sistema CART traduziram-se em valores de *F-measure* não significativos, sendo o número de ganhos significativos em número superior ao número de perdas significativas.

No que toca à interpretabilidade dos modelos gerados, o nosso método parece estar em clara vantagem. Para os domínios testados, o nosso método gerou, em média, árvores mais pequenas do que aquelas geradas pelo CART (Breiman et al., 1984), especialmente em relação às árvores resultantes do CART sem *pruning*, o que nos leva a concluir que os nossos modelos são, em teoria, mais interpretáveis. Para alguns dos domínios, a análise de alguns dos modelos gerados permitiu-nos constatar que as árvores geradas pelo nosso método são bastante desequilibradas e, talvez por isso mesmo, sejam mais interpretáveis.

6.2 Trabalho futuro

As características do problema, alvo de estudo nesta tese, em conjunto com os resultados obtidos, permitem-nos apontar algumas possíveis direcções para prosseguimento de trabalho.

- **Estimação dos parâmetros f e β.**

  Uma das primeiras abordagens a considerar no prosseguimento deste trabalho, poderia centrar-se na investigação das razões do fraco desempenho do método em termos de *precision*. Aparentemente, ou não está a ser dada à *precision* a importância devida, através do parâmetro β, ou as nossas árvores estarão a parar de crescer demasiado cedo, através do *threshold* f. Dados os resultados da análise sumária feita da sensibilidade do método a estes dois parâmetros, a nossa sugestão actual reside na estimação conjunta dos parâmetros β e f para cada um dos domínios. Esta estimação, feita por *cross-validation* ou *holdout*, poderá trazer, a nosso ver, algumas melhorias nos resultados.

- **Comparação de diferentes parametrizações usando *REC Curves*.**

  Outras medidas de avaliação poderiam ser utilizadas para a determinação da melhor parametrização do nosso método para um dado domínio, recorrendo à *ROC analysis*, mas aplicada à área da regressão. Bi e Bennett (2003) propõem as *Regression Error Characteristic (REC) Curves* como generalização das *ROC Curves* à regressão, tirando partido de todas as potencialidades que estas possuem na avaliação da qualidade de classificadores, independentemente da distribuição dos dados.

  As *REC Curves* fazem o *plot* da tolerância ao erro no eixo dos xx' e da *accuracy* da função de regressão no eixo dos yy', isto é, da percentagem de previsões consideradas correctas num certo intervalo de tolerância do erro. De acordo com Bi e Bennett (2003), a *accuracy* é calculada de acordo com uma determinada *loss function* cujo valor pode ser o erro quadrado ou o desvio absoluto. Segundo os autores, supondo que temos um conjunto de dados com m exemplos, $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$, onde $\mathbf{x}_i \in \Re^n$ são variáveis independentes e $y \in \Re$ é a variável objectivo, a *accuracy* numa tolerância ϵ é definida por:

$$acc(\epsilon) := \frac{|(\mathbf{x}, y) : loss(f(\mathbf{x}_i), y) \leq \epsilon, i = 1, \ldots, m|}{m} \tag{6.1}$$

As *REC Curves* devem caracterizar a qualidade de um modelo de acordo com diferentes níveis de tolerância ao erro. Normalmente, estamos interessados em obter modelos de regressão precisos nas suas previsões. No nosso caso concreto, estamos interessados em fazer, maioritariamente, previsões acertadas, aceitando uma certa tolerância ao erro, desde que isso evite falhas "catastróficas".

Poderíamos, então, definir a *loss function* como o complementar da *F-measure*. As curvas presentes no canto superior esquerdo, representarão os modelos com melhor *performance*, isto é, aqueles que obtêm melhores valores de *accuracy* com menor tolerância de erro.

Da mesma forma que as *ROC Curves* forneciam uma medida da *performance* esperada através da AUC, as *REC Curves* também fornecem uma medida válida da estimativa do erro esperado associado ao modelo de regressão, através da AOC (*Area Over the Curve*). A estimativa do erro diz respeito ao erro expresso pela *loss function*.

A implementação desta técnica poderia ajudar a diagnosticar problemas com os modelos e ajudar na selecção dos mesmos.

- **Comparação de *performance* com outros métodos.**

  Seria também interessante comparar a *performance* do nosso método com outros métodos de regressão que não as árvores de regressão *standard* do sistema CART (Breiman et al., 1984). A título de exemplo poderiam ser utilizados para comparação: as redes neuronais (Ripley, 1996); as SVM (Cortes e Vapnik, 1995); o MARS (Friedman, 1991); o CUBIST (Quinlan, 1999), entre outros métodos. Estes métodos estão, actualmente em voga e são reconhecidos pelos bons resultados que têm obtido em diversos domínios.

  Convém, contudo, salientar que a interpretabilidade dos modelos obtidos pelos métodos acima referidos não seria comparável aos modelos gerados pelo nosso método, uma vez que constituem métodos de "natureza" diferente. Em particular, e no que respeita aos métodos de redes neuronais e das SVM, por exemplo, eles geram modelos do tipo *"caixa negra"*: fornecido um conjunto de *inputs* é obtido um conjunto de *outputs*, que comprova que método funciona, mas não se sabe, exactamente, como actuou para chegar aos resultados. Os modelos produzidos por estes métodos são difíceis de interpretar.

- **Integração no sistema *Random Forests*.**

  A questão da interpretabilidade dos modelos poderá, no entanto, ser subjectiva. Um modelo expresso através de uma árvore com 1000 nós, por exemplo, não será, certamente, interpretável. O que realmente poderá ser considerado interpretável é o facto de poder apresentar uma justificação, com base nos valores dos atributos, para a previsão feita. Posto isto, poderemos concentrarmo-nos em melhorar a *performance* das árvores.

  Foi partindo deste pressuposto que Breiman (2001) propôs um novo sistema: as *Random Forests* (RF). Com este sistema, Breiman, pretendeu melhorar a *performance* das árvores (classificação ou regressão), ao mesmo tempo que tirava partido da sua interpretabilidade. Em linhas muito gerais, poder-se-á dizer que este método consiste na geração de n árvores sem *pruning* a partir de amostras de dados diferentes e considerando em cada instante apenas um subconjunto, aleatoriamente seleccionado, de atributos. A *"lei dos grandes números"* assegura que estas árvores não correm o risco de fazer *overfit*. Resultados apresentados em Breiman (2001) demonstraram que, em domínios de classificação, as RF são bastante competitivas em relação ao *AdaBoost*, SVM e redes neuronais.

  Conforme já referido, os métodos actualmente utilizados, que apresentam melhores resultados a nível de previsões, constituem métodos de *"caixa negra"*. Com a apresentação das RF, Breiman, mostra que a *performance* e a interpretabilidade não têm, necessariamente, que caminhar em direcções opostas. Breiman, enquadrou as RF na classe de algoritmos a que apelidou de *"Two-eyed Algorithms"*. A ideia consiste em manter um olho sobre a máquina de classificação/regressão e outro dentro dessa máquina. O primeiro olho visualiza todo o processo de obtenção, manipulação de dados e fornecimento de previsões precisas. O segundo olho visualiza o que se passa dentro da *"caixa negra"*, detectando, por exemplo, o conjunto de atributos relevantes no processo que ocorre na máquina, ou até, a existência de eventuais *outliers*.

  Desta forma, não só obtermos excelentes resultados a nível de *performance* como também a nível de interpretabilidade. Segundo Breiman, os algoritmos que actualmente apresentam melhores resultados a nível de previsões são as SVM e as RF.

  Neste contexto, a integração no sistema RF do novo critério de *split* aqui proposto, poderá também ser uma das linhas de investigação a seguir.

- **Implementação da abordagem de classificação.**

  Numa outra perspectiva, poderá ser estudada e implementada a adaptação do método aqui proposto para o contexto das árvores de decisão na aplicação a domínios de classificação, de onde, aliás, foram "herdados" os conceitos de *precision*, *recall* e *F-measure*.

  Com esta "re-implementação" do nosso método, as melhorias, acima referidas, como a estimação dos parâmetros f e β, a comparação e visualização de diferentes parametrizações do método recorrendo às *ROC Curves*, a comparação com outros métodos de classificação que não as árvores de decisão *standard* e a integração do método no sistema RF, serão, todas elas, aplicáveis.

6.3 Considerações finais

As contribuições desta tese situam-se a dois níveis: científico e pessoal. A nível científico, pensamos que foi dado um primeiro passo na tentativa de previsão de *outliers* num âmbito de regressão que, mesmo que não seja seguido, poderá fornecer algumas pistas nesta área. A nível pessoal, as contribuições deste trabalho foram imensas: alargaram-se conhecimentos, exploraram-se vários caminhos, alguns sem sucesso, outros reconhecidamente interessantes, mas todos eles valiosos para as conclusões aqui apresentadas.

Apêndice A

Algumas Aplicações

Neste apêndice é feita alguma análise no que toca à interpretabilidade dos modelos gerados pelo nosso método, quando aplicado a alguns domínios com conjuntos de dados reais, apresentados na Tabela 5.1 do Capítulo 5. Como termo de comparação, utiliza-se as árvores *standard* geradas pelo sistema CART (Breiman et al., 1984) usando a regra 1-SE. Pretende-se assim evidenciar a diferença dos modelos gerados no que respeita à sensibilidade a *outliers* extremos.

A.1 Machine Cpu

Machine Cpu[1] constitui um conjunto de dados cujo objectivo reside em prever a *performance* de um CPU, tendo em conta algumas características de *hardware*.

A Tabela A.1 apresenta a descrição dos atributos que compõem o conjunto de dados.

A Figura A.1 apresenta o *box plot* da variável objectivo prp. A variável objectivo prp só possui *outliers* extremos elevados, isto é, pertencentes a O_H, que correspondem a máquinas de elevada *performance*.

A árvore apresentada na Figura A.2 foi gerada pelo sistema CART. Sob ponto de vista da detecção de *outliers* extremos, esta árvore consegue, de facto, captar as principais características de *hardware* que estão na base de uma boa performance do CPU. São elas

[1] Disponível no *UCI machine learning repository* (http://www.ics.uci.edu/~mlearn/MLSummary.html).

Machine Cpu	
Atributo	**Descrição**
MYCT	tempo do ciclo da máquina em nanossegundos
MMIN	tam. mínimo da memória principal (kilobytes)
MMAX	tam. máximo da memória principal (kilobytes)
CACH	tam. memória cache (kilobytes)
CHMIN	número mínimo de canais (units)
CHMAX	número máximo de canais (units)
prp	*performance* relativa publicada

Tabela A.1: Descrição do conjunto de dados *Machine Cpu*.

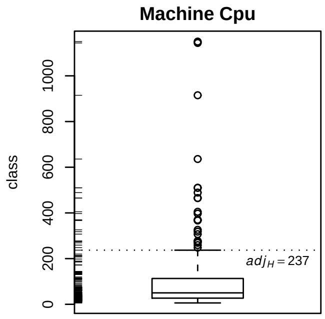


Figura A.1: *Box plot* da variável objectivo prp do problema Machine Cpu.

o tamanho da memória principal e o tamanho da cache. Máquinas com um máximo de memória principal superior a 28000K possuem uma *performance* em média de 408 e constituem, na sua maioria, *outliers* extremos. Através desta árvore, é possível ainda distinguir dois "tipos" de *outliers* extremos. Máquinas nestas condições, com uma memória cache superior a 80k, possuem uma *performance* especialmente elevada. Este último teste distingue as máquinas com uma *performance* média de 299 das máquinas com uma *performance* média de 667.

No entanto, existem ainda nesta árvore três máquinas cuja *performance* constitui um *outlier* extremo e que não foram explicadas pelo modelo. Elas estão incluídas no conjunto de máquinas com uma memória principal inferior a 28000K.

Na Figura A.3 é apresentada a árvore gerada pelo nosso método para o mesmo problema. Os três *outliers* extremos ignorados pelo CART são aqui explicados. O facto é que máquinas com memória principal máxima inferior a 28000K podem, ainda assim, obter bons valores de *performance* dependendo do tamanho da cache. Ainda que a *performance* seja boa, ela é um pouco mais comedida.

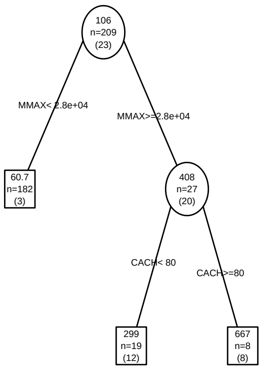


Figura A.2: Árvore de regressão obtida pelo CART para o problema Machine Cpu.

Para além disso, o nosso modelo consegue distinguir ainda mais as máquinas de *performance* especialmente elevada, isto é, superior a 667 em média. As máquinas com uma memória principal superior a 48000K possuem uma performance média de 961. Com base no tamanho da cache é possível distinguir diferentes valores de *performance*.

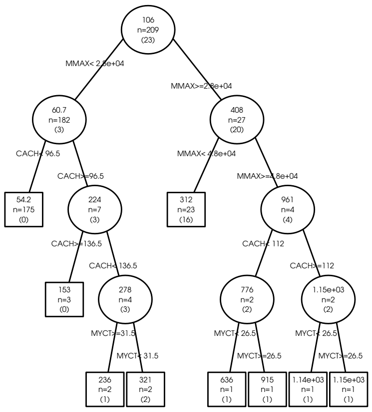


Figura A.3: Árvore de regressão obtida pelo Nosso Método para o problema Machine Cpu.

Em suma, o nosso modelo parece ser mais consistente com a distribuição de valores apresentada pelo *box plot* da Figura A.1. Nesse contexto, trata-se de um modelo com boa interpretabilidade. Mais do que isso, consegue obter melhores resultados do que o CART, conforme demonstram os resultados apresentados no Capítulo 5, Tabela 5.3.

A.2 Boston Housing

Boston Housing[2] é um conjunto de dados constituído com o propósito inicial de investigar a maneira como a concentração da poluição do ar afectava o preço das casas nos subúrbios da cidade de Boston. Actualmente, é um conjunto de dados bastante usado para experiências de novos métodos de previsão. Modelos concebidos revelaram diferente relações de influência de variados factores sobre o preço das casas.

A Tabela A.2 apresenta a descrição dos atributos que compõem o conjunto de dados.

Boston Housing	
Atributo	**Descrição**
crim	índice de criminalidade
zn	% de área para loteamento (lotes com mais de $25m^2$)
indus	% de negócios não-comerciais em acres por subúrbio
chas	1 se situada à beira do rio Charles, 0 caso contrário
nox	Concentração de óxido de nitrogénio (em pphm)
rm	número médio de quartos por habitação
age	% de casas ocupadas pelo dono, construídas antes de 1940
dis	distância ponderada a cinco centros de emprego em Boston
rad	índice de acessibilidade às auto-estradas circundantes
tax	imposto municipal
ptratio	rácio aluno/professor por subúrbio
black	$1000(Bk - 0.63)^2$ onde Bk é a % de população negra por subúrbio
lstat	% de população de estrato social baixo
medv	Valor médio das casas na área ocupadas pelo dono em $1000

Tabela A.2: Descrição do conjunto de dados Boston Housing.

A Figura A.4 apresenta o *box plot* da variável objectivo medv.

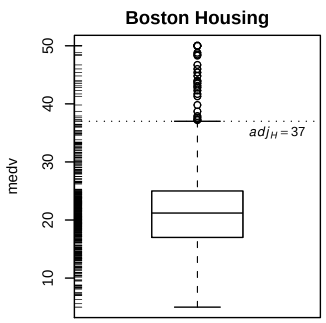


Figura A.4: *Box plot* da variável objectivo medv do problema Boston Housing.

[2]Disponível no *UCI machine learning repository* (http://www.ics.uci.edu/~ mlearn/MLSummary.html).

O modelo gerado pelo CART, apresentado na Figura A.5, consegue captar a informação de que o número de quartos de uma casa exerce uma grande influência sobre o preço da mesma. Assim, e segundo este modelo, casas com um número de quartos, em média, superior ou igual a 6.941 possuem, na sua maioria, um preço elevado, constituindo *outliers*. Se o número de quartos for inferior a 6.941, então serão as casas incluídas em zonas habitacionais de estrato social mais ou menos elevado e próximas do centro que terão preços especialmente elevados. Algumas casas com cerca de seis, sete quartos podem surgir em zonas com uma percentagem reduzida de habitantes de estrato social baixo com um preço elevado. No entanto não é bem explicado pelo modelo as situações em que estas realmente ocorrem, acabando por ser desprezadas.

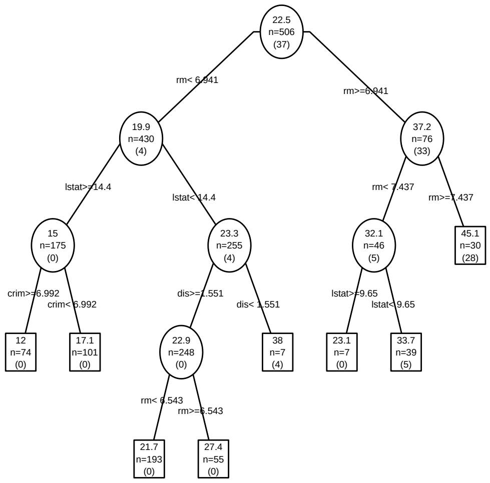


Figura A.5: Árvore de regressão obtida pelo CART para o problema Boston Housing.

Na Figura A.6 é apresentado o modelo obtido pelo nosso método para o mesmo problema. A mesma informação relativa à influência do número de quartos sobre o preço das casas é captada. No entanto, dado que a maioria das casas de preço elevado possuem, em média, um número de quartos superior ou igual a 6.941, o nosso modelo pára a explicação de casas nessas condições, por considerar que já alcançou um bom *recall*. De facto, no conjunto de dados de treino, trinta e três das trinta e sete casas cujo preço constitui um *outlier* extremo, encontram-se nessas condições. O modelo concentra-se mais em tentar explicar em que condições as casas com menos de 6.941 quartos podem ser extremamente caras. O modelo explica que zonas próximas do centro com uma criminalidade baixa ou zonas com habitantes de um estrato social elevado fazem elevar o preço das casas. O que também é transmitido pelo modelo é que bairros caros são zonas de alguma criminalidade, o que, de certo modo, faz sentido.

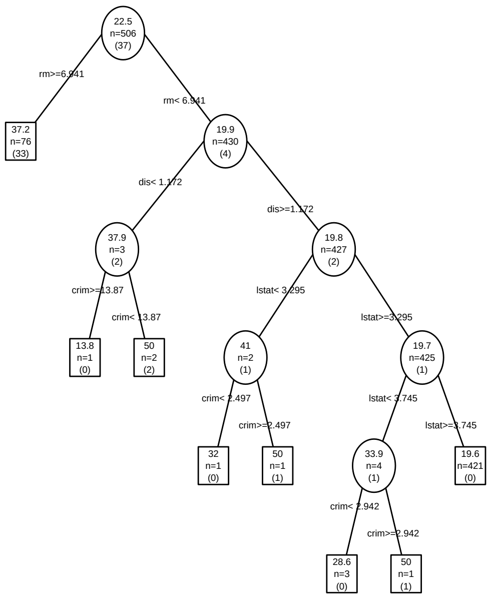


Figura A.6: Árvore de regressão obtida pelo Nosso Método para o problema Boston Housing.

Este é mais um exemplo de uma árvore desequilibrada mas altamente interpretável, de acordo com os objectivos por nós estabelecidos.

Os resultados apresentados no Apêndice B, Tabelas B.7 e B.8, demonstram, contudo, que o nosso método com $\beta = 1$, apesar de ganhar em *recall*, perde em termos de *precision* com o CART.

Se exigirmos um modelo mais preciso na previsão dos preços das casas, então o nosso método gera um modelo diferente. Estabelecendo $\beta = 0.5$, isto é, dando à *precision* o dobro da importância do *recall*, obtemos o modelo apresentado na Figura A.7.

Da observação do modelo é possível constatar, logo à partida, uma escolha diferente para o primeiro nó. É feito um teste mais específico e mais restritivo quanto ao número de quartos. Este teste dá origem a uma folha com um bom *recall*, ainda que um pouco inferior ao obtido para a mesma folha no modelo com $\beta = 1$ (*c.f.* Figura A.6), uma vez que capta vinte e oito dos trinta e sete *outliers* extremos existentes no conjunto de treino. Mas esta é também uma folha com uma partição mais "pura" e portanto com uma *precision* maior.

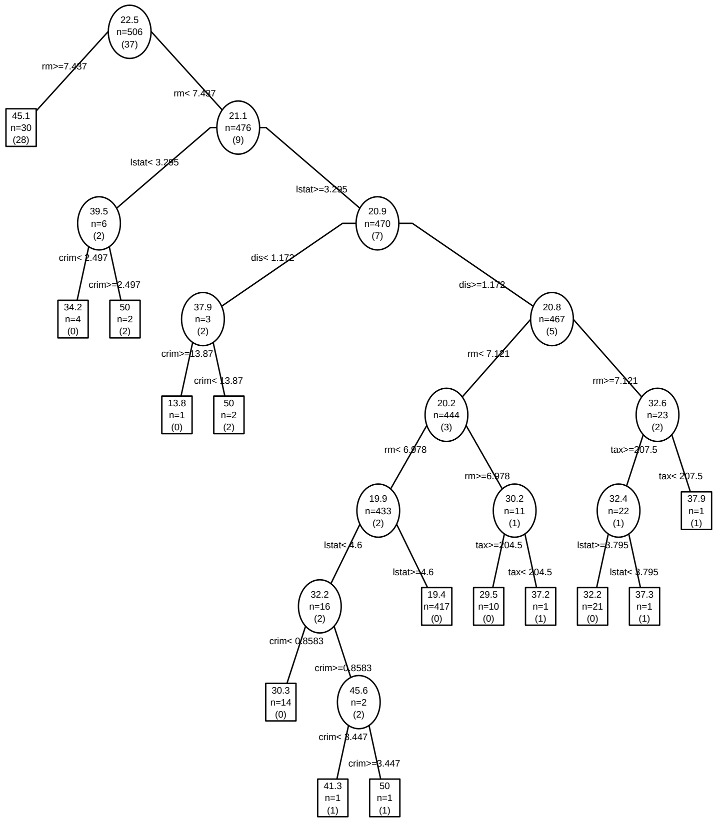


Figura A.7: Árvore de regressão obtida pelo Nosso Método com $\beta = 0.5$, para o problema Boston Housing.

As restantes casas de preço elevado são descritas, essencialmente, com base na percentagem de pessoas de estrato social baixo, na criminalidade e na distância ao centro. Estes são, aliás, os atributos que aliados ao número de quartos parecem governar o preço das casas, já que estão presentes em todos os modelos (*c.f.* Figuras A.5, A.6 e A.7).

Com esta parametrização, o nosso método concebeu modelos que ganharam um pouco em *precision* relativamente à parametrização anterior de $\beta = 1$, chegando a superar o CART (tal como os resultados apresentados no Apêndice B, Tabelas B.7, B.8 e B.9 o mostram).

Em jeito de conclusão, no que toca à identificação das condições que levam à existência de casas com preço extremamente elevado, os nossos modelos parecem ser mais eficazes. Obtemos uma explicação mais pormenorizada dessas condições.

Apêndice B

Alguns Resultados

Neste apêndice apresentam-se alguns dos resultados obtidos nas experiências realizadas.

Conjuntos de Dados	$precision_{regr}$			recall			F-measure		
	CART $(0-SE)$	Nosso Método $(\beta=1)$		CART $(0-SE)$	Nosso Método $(\beta=1)$		CART $(0-SE)$	Nosso Método $(\beta=1)$	
servo	0.8023867	0.5062348	++	0.8493333	0.9800000	−−	0.8201367	0.6612034	++
triazines	0.2565775	0.2988911		0.0600000	0.2233333	−	0.0552234	0.2282423	−−
algae1	0.2939821	0.4180734	−	0.0000000	0.3366667	−−	0.0000000	0.3279042	−−
algae2	0.1118913	0.1244440		0.0000000	0.0900000	−	0.0000000	0.0685473	
algae3	0.0521137	0.1190006		0.0630000	0.1906667	−	0.0537133	0.1186311	
algae4	0.1208642	0.1210823		0.0866667	0.1150000		0.0631145	0.0636982	
algae5	0.0630465	0.1571761	−	0.0000000	0.1366667	−	0.0000000	0.0970766	−
algae6	0.1122685	0.1029830		0.1273333	0.1606667		0.0974198	0.0805630	
algae7	0.0614083	0.0443510		0.0700000	0.0590000		0.0635197	0.0476719	
machine.cpu	0.6025036	0.6870716	−	0.8036667	0.8950000	−	0.6500510	0.7591475	−
china	0.0542900	0.0450076		0.0000000	0.0530000	−	0.0000000	0.0401994	−
Boston	0.8474018	0.8192831		0.7785000	0.7595000		0.7874833	0.7346941	
onekm	0.4211353	0.3350779	+	0.2933333	0.2600000		0.2939498	0.2701772	
cw.drag	0.9299020	0.8355772	++	0.8419906	0.9801429	−−	0.8758283	0.8976743	
co2.emission	0.8071862	0.8068307		0.4213333	0.6326667	−−	0.4776471	0.6775423	−−
acceleration	0.8993252	0.8652793	+	0.5900000	0.7963333	−	0.6554287	0.7864996	−
available.power	0.9672531	0.8567897	++	0.8780178	1.0000000	−−	0.9178064	0.9217586	
bank8FM	0.9787010	0.9425374	++	0.6446674	0.6798909		0.7533743	0.7594744	
delta.ailerons	0.6808727	0.5531748	++	0.1704292	0.1760511		0.2534065	0.2462873	
ibm	0.0000000	0.0104953	−−	0.0000000	0.0055804	−	0.0000000	0.0035846	−
cpu.small	0.9944129	0.9855883	++	0.8596401	0.8886580	−−	0.9213565	0.9341849	−
delta.elevators	0.6202172	0.5978288	+	0.0962118	0.1620447	−−	0.1530586	0.2445691	−−
cal.housing	0.8531954	0.5871524	++	0.3999709	0.4608993	−−	0.5416227	0.5155620	+
add	0.9352271	0.7850352	++	0.0140000	0.1240159	−−	0.0196822	0.1823070	−−
fried.delve	0.8843362	0.5371731	++	0.0935714	0.0967460		0.1224614	0.1161262	

Tabela B.1: Avaliação de resultados obtidos pelo CART $(0-SE)$ vs Nosso Método.

Conjuntos de Dados	***$precision_{regr}$***		***recall***		***F-measure***	
	CART (*s/ pruning*)	Nosso Método ($\beta = 1$)	CART (*s/ pruning*)	Nosso Método ($\beta = 1$)	CART (*s/ pruning*)	Nosso Método ($\beta = 1$)
servo	0.9144078	0.5062348 ++	0.9166667	0.9800000 −	0.9148829	0.6612034 ++
triazines	0.3712013	0.2988911	0.1783333	0.2233333	0.1824408	0.2282423
algae1	0.4573431	0.4180734	0.2333333	0.3366667 −	0.2313154	0.3279042 −
algae2	0.1979685	0.1244440	0.1066667	0.0900000	0.1039575	0.0685473
algae3	0.1704799	0.1190006	0.1466667	0.1906667	0.1034266	0.1186311
algae4	0.1477303	0.1210823	0.1350000	0.1150000	0.0858976	0.0636982
algae5	0.2130087	0.1571761	0.1006667	0.1366667	0.1019625	0.0970766
algae6	0.1362754	0.1029830	0.1573333	0.1606667	0.0886959	0.0805630
algae7	0.0911165	0.0443510	0.1490000	0.0590000 +	0.0810744	0.0476719
machine.cpu	0.8137943	0.6870716 ++	0.8026667	0.8950000 −	0.7876936	0.7591475
china	0.1376418	0.0450076 +	0.0780000	0.0530000	0.0688116	0.0401994
Boston	0.9274704	0.8192831 +	0.8605000	0.7595000 +	0.8737304	0.7346941 +
onekm	0.4342629	0.3350779 +	0.2000000	0.2600000	0.2062307	0.2701772
cw.drag	0.9515258	0.8355772 ++	0.9555397	0.9801429 −	0.9508629	0.8976743 ++
co2.emission	0.8467644	0.8068307 +	0.6716667	0.6326667	0.7116916	0.6775423
acceleration	0.9320429	0.8652793 ++	0.8023333	0.7963333	0.8368396	0.7864996
available.power	0.9810829	0.8567897 ++	0.9787801	1.0000000 −	0.9792208	0.9217586 ++
bank8FM	0.9771989	0.9425374 ++	0.6113454	0.6798909	0.7293388	0.7594744
delta.ailerons	0.6774042	0.5531748 ++	0.2300351	0.1760511 +	0.3209157	0.2462873 +
ibm	0.0057480	0.0104953	0.0325151	0.0055804 ++	0.0045179	0.0035846
cpu.small	0.9944256	0.9855883 ++	0.8744840	0.8886580 −	0.9301252	0.9341849
delta.elevators	0.5757086	0.5978288	0.1700251	0.1620447	0.2503409	0.2445691
cal.housing	0.8461150	0.5871524 ++	0.5528001	0.4608993 ++	0.6676467	0.5155620 ++
add	0.92722943	0.78503524 ++	0.1594603	0.1240159	0.2460639	0.1823070
fried.delve	0.8532134	0.5371731 ++	0.2265238	0.09674603 +	0.3032233	0.1161262 +

Tabela B.2: Avaliação de resultados obtidos pelo CART (s/ *pruning*) vs Nosso Método.

Conjuntos de Dados	Tamanho Médio das Árvores			
	Nosso Método	CART		
	($\beta = 1$)	$(1 - SE)$	$(0 - SE)$	*(s/ pruning)*
servo	5.12 ± 0.48	4.44 ± 1.21	15.72 ± 6.15	241.92 ± 6.21
triazines	32.40 ± 5.62	4.88 ± 3.27	18.56 ± 7.90	309.96 ± 3.75
algae1	30.00 ± 5.98	3.88 ± 1.52	10.84 ± 6.90	310.04 ± 4.22
algae2	40.32 ± 5.46	1.20 ± 1.01	6.36 ± 6.71	254.68 ± 6.20
algae3	69.80 ± 7.69	1.00 ± 0.00	2.92 ± 2.91	255.80 ± 5.71
algae4	35.64 ± 8.81	1.00 ± 0.00	7.00 ± 6.80	192.68 ± 5.68
algae5	50.12 ± 5.03	1.16 ± 0.68	5.92 ± 7.33	243.04 ± 5.65
algae6	57.28 ± 9.74	1.00 ± 0.00	3.88 ± 3.57	205.16 ± 5.80
algae7	54.72 ± 8.41	1.88 ± 1.67	3.52 ± 2.24	213.12 ± 7.34
machine.cpu	10.76 ± 1.74	3.96 ± 1.29	23.20 ± 6.51	327.04 ± 5.44
china	50.04 ± 6.20	1.00 ± 0.00	3.16 ± 2.68	391.00 ± 0.00
Boston	19.16 ± 3.63	18.60 ± 4.52	52.88 ± 17.98	847.64 ± 7.14
onekm	14.72 ± 2.65	16.64 ± 7.17	96.84 ± 12.85	777.40 ± 13.82
cw.drag	25.76 ± 3.79	88.92 ± 15.66	202.44 ± 9.32	686.92 ± 14.59
co2.emission	30.56 ± 6.12	114.00 ± 11.52	224.00 ± 19.56	2067.28 ± 16.37
acceleration	26.52 ± 6.73	71.64 ± 17.10	228.60 ± 33.43	2275.96 ± 24.58
available.power	10.96 ± 0.28	91.00 ± 13.50	252.80 ± 9.70	821.80 ± 9.71
bank8FM	73.08 ± 24.55	211.40 ± 36.38	534.60 ± 106.65	8024.44 ± 4.93
delta.ailerons	342.40 ± 25.37	65.36 ± 13.15	137.20 ± 35.05	6886.96 ± 63.43
ibm	138.00 ± 7.51	1.00 ± 0.00	1.00 ± 0.00	3536.40 ± 807.56
cpu.small	209.40 ± 15.28	165.92 ± 63.06	449.72 ± 85.46	8743.64 ± 53.73
delta.elevators	583.84 ± 22.23	55.28 ± 10.50	104.32 ± 24.70	9637.04 ± 62.80
cal.housing	950.72 ± 87.59	517.32 ± 66.97	1083.32 ± 259.95	35494.24 ± 60.54
add	234.12 ± 10.99	433.00 ± 36.79	595.44 ± 53.33	44888.40 ± 33.55
fried.delve	112.92 ± 8.54	3068.16 ± 157.49	4289.96 ± 345.69	73167.40 ± 53.53

Tabela B.3: Tamanho médio das árvores obtidas por *cross-validation*.

Conjuntos de Dados	$precision_{regr}$						
	CART	Nosso Método					
	(s/ pruning)	$(f = 0.7)$		$(f = 0.85)$		$(f = 0.95)$	
servo	0.9144078	0.5062348	++	0.7788837	++	0.7788837	++
triazines	0.3712013	0.2988911		0.2339836		0.2339836	
algae1	0.4573431	0.4180734		0.3737079		0.3737079	
algae2	0.1979685	0.1244440		0.1315870		0.1315870	
algae3	0.1704799	0.1190006		0.1295480		0.1295480	
algae4	0.1477303	0.1210823		0.1221831		0.1221831	
algae5	0.2130087	0.1571761		0.1352760		0.1352760	
algae6	0.1362754	0.1029830		0.0936290		0.0936290	
algae7	0.0911165	0.0443510		0.0915245		0.0915245	
machine.cpu	0.8137943	0.6870716	++	0.5857648	++	0.5986705	++
china	0.1376418	0.0450076	+	0.0127149	++	0.0127149	++
Boston	0.9274704	0.8192831	+	0.7698865	++	0.7642168	++
onekm	0.4342629	0.3350779	+	0.3658710		0.3626117	
cw.drag	0.9515258	0.8355772	++	0.8218117	++	0.8218117	++
co2.emission	0.8467644	0.8068307	+	0.7524325		0.7524325	
acceleration	0.9320429	0.8652793	++	0.8621148		0.8791512	
available.power	0.9810829	0.8567897	++	0.8501017	++	0.8501017	++
bank8FM	0.9771989	0.9425374	++	0.9341487	++	0.9341487	++
delta.ailerons	0.6774042	0.5531748	++	0.5712251	++	0.5712251	++
ibm	0.0057480	0.0104953		0.0189094	−	0.0189094	−
cpu.small	0.9944256	0.9855883	++	0.9914920		0.9914920	
delta.elevators	0.5757086	0.5978288		0.6033829		0.6033829	
cal.housing	0.84611498	0.58715243	++	0.58059997	++	0.58059997	++
add	0.92722943	0.78503524	++	0.78378371	++	0.78378371	++
fried.delve	0.8532134	0.5371731	++	0.5344990	++	0.5344990	++

Tabela B.4: Variação do parâmetro f: avaliação de $precision_{regr}$.

Conjuntos de Dados	*recall*						
	CART	Nosso Método					
	(s/ pruning)	$(f = 0.7)$		$(f = 0.85)$		$(f = 0.95)$	
servo	0.9166667	0.9800000	−	0.9800000		0.9800000	
triazines	0.1783333	0.2233333		0.1466667		0.1466667	
algae1	0.2333333	0.3366667	−	0.3266667		0.3266667	
algae2	0.1066667	0.0900000		0.0966667		0.0966667	
algae3	0.1466667	0.1906667		0.2063333		0.2063333	
algae4	0.1350000	0.1150000		0.1440000		0.1440000	
algae5	0.1006667	0.1366667		0.0766667		0.0766667	
algae6	0.1573333	0.1606667		0.1696667		0.1696667	
algae7	0.1490000	0.0590000	+	0.0906667		0.0906667	
machine.cpu	0.8026667	0.8950000	−	0.7243333		0.6736667	
china	0.0780000	0.0530000		0.0233333	+	0.0233333	+
Boston	0.8605000	0.7595000	+	0.7760476		0.7703333	
onekm	0.2000000	0.2600000		0.3033333		0.3033333	
cw.drag	0.9555397	0.9801429	−	0.9811667		0.9811667	
co2.emission	0.6716667	0.6326667		0.6576667		0.6576667	
acceleration	0.8023333	0.7963333		0.6936667		0.6896667	
available.power	0.9787801	1.0000000	−	0.9911111		0.9911111	
bank8FM	0.6113454	0.6798909		0.6850859		0.6850859	
delta.ailerons	0.2300351	0.1760511	+	0.1631381	+	0.1631381	+
ibm	0.0325151	0.0055804	++	0.0096351	++	0.0096351	++
cpu.small	0.8744840	0.8886580	−	0.8837459		0.8837459	
delta.elevators	0.1700251	0.1620447		0.1548121		0.1548121	
cal.housing	0.55280013	0.46089927	++	0.45219977	++	0.45219977	++
add	0.1594603	0.1240159		0.1531349		0.1531349	
fried.delve	0.2265238	0.09674603	+	0.1375238	+	0.1375238	+

Tabela B.5: Variação do parâmetro f: avaliação de *recall*.

Conjuntos de Dados	*F-measure*						
	CART	Nosso Método					
	(s/ pruning)	$(f = 0.7)$		$(f = 0.85)$		$(f = 0.95)$	
servo	0.9148829	0.6612034	++	0.8644117		0.8644117	
triazines	0.1824408	0.2282423		0.1519461		0.1519461	
algae1	0.2313154	0.3279042	−	0.3172723		0.3172723	
algae2	0.1039575	0.0685473		0.0834489		0.0834489	
algae3	0.1034266	0.1186311		0.1415184		0.1415184	
algae4	0.0858976	0.0636982		0.0991550		0.0991550	
algae5	0.1019625	0.0970766		0.0585721		0.0585721	
algae6	0.0886959	0.0805630		0.0781744		0.0781744	
algae7	0.0810744	0.0476719		0.0822707		0.0822707	
machine.cpu	0.7876936	0.7591475		0.6205883	+	0.5970384	+
china	0.0688116	0.0401994		0.0091351	+	0.0091351	+
Boston	0.8737304	0.7346941	+	0.7387424	+	0.7323323	+
onekm	0.2062307	0.2701772		0.3032746		0.3015983	
cw.drag	0.9508629	0.8976743	++	0.8876936	+	0.8876936	+
co2.emission	0.7116916	0.6775423		0.6700393		0.6700393	
acceleration	0.8368396	0.7864996		0.7064438	+	0.7123285	
available.power	0.9792208	0.9217586	++	0.9143349	++	0.9143349	++
bank8FM	0.7293388	0.7594744		0.7736796		0.7736796	
delta.ailerons	0.3209157	0.2462873	+	0.2307000	+	0.2307000	+
ibm	0.0045179	0.0035846		0.0085958		0.0085958	
cpu.small	0.9301252	0.9341849		0.9338217		0.9338217	
delta.elevators	0.2503409	0.2445691		0.2355574		0.2355574	
cal.housing	0.6676467	0.51556198	++	0.50795193	++	0.50795193	++
add	0.2460639	0.1823070		0.2318075		0.2318075	
fried.delve	0.3032233	0.1161262	+	0.1610023	+	0.1610023	+

Tabela B.6: Variação do parâmetro f: avaliação de *F-measure*.

Conjuntos de Dados	$precision_{regr}$			
	CART	Nosso Método		
	$(1-SE)$	$(\beta=0.5)$	$(\beta=1)$	$(\beta=2)$
servo	0.7616598	0.7894392	0.5062348 ++	0.4564660 ++
triazines	0.1474294	0.2988911 −	0.2988911 −	0.2988911 −
algae1	0.2947144	0.4188416 −	0.4180734 −	0.4176917 −
algae2	0.0000000	0.1244440 −	0.1244440 −	0.1244440 −
algae3	0.0000000	0.1190006 −	0.1190006 −	0.1190006 −
algae4	0.0000000	0.0789215 −	0.1210823 −−	0.2180226 −−
algae5	0.0000000	0.1571761 −−	0.1571761 −−	0.1571761 −−
algae6	0.0000000	0.0727176 −	0.1029830 −	0.2268465 −−
algae7	0.0127059	0.0496159	0.0443510	0.0379399
machine.cpu	0.5517675	0.6727902 −	0.6870716 −−	0.6722179 −
china	0.0000000	0.0450076 −	0.0450076 −	0.0450076 −
Boston	0.8243590	0.8560680	0.8192831	0.7753407
onekm	0.4001506	0.3350779	0.3350779	0.3350779
cw.drag	0.9250750	0.8355772 ++	0.8355772 ++	0.8355772 ++
co2.emission	0.8052664	0.8006652	0.8068307	0.8077473
acceleration	0.8964751	0.8651399 +	0.8652793 +	0.8652793 +
available.power	0.9668409	0.8717602 ++	0.8567897 ++	0.8567897 ++
bank8FM	0.9781688	0.9463860 ++	0.9425374 ++	0.9398156 ++
delta.ailerons	0.6442916	0.5534367 ++	0.5531748 ++	0.5527013 ++
ibm	0.0000000	0.0104953 −−	0.0104953 −−	0.0104953 −−
cpu.small	0.9939290	0.9917059 +	0.9855883 ++	0.9277362 ++
delta.elevators	0.5980803	0.5991528	0.5978288	0.5994543
cal.housing	0.8425739	0.5874014 ++	0.5871524 ++	0.5870872 ++
add	0.9352271	0.7850352 ++	0.7850352 ++	0.7850352 ++
fried.delve	0.8427616	0.5371731 ++	0.5371731 ++	0.5371731 ++

Tabela B.7: Variação do parâmetro β: avaliação de $precision_{regr}$.

Conjuntos de Dados	$recall$			
	CART	Nosso Método		
	$(1-SE)$	$(\beta=0.5)$	$(\beta=1)$	$(\beta=2)$
servo	0.8713333	0.9693333 −	0.9800000 −−	0.9800000 −−
triazines	0.0400000	0.2233333 −−	0.2233333 −−	0.2233333 −−
algae1	0.0000000	0.3366667 −−	0.3366667 −−	0.3266667 −−
algae2	0.0000000	0.0900000 −	0.0900000 −	0.0900000 −
algae3	0.0000000	0.1906667 −−	0.1906667 −−	0.1906667 −−
algae4	0.0000000	0.1016667 −	0.1150000 −	0.2783333 −−
algae5	0.0000000	0.1366667 −	0.1366667 −	0.1366667 −
algae6	0.0000000	0.1173333 −−	0.1606667 −−	0.3140000 −−
algae7	0.0100000	0.0823333 −	0.0590000	0.0690000
machine.cpu	0.8186667	0.8883333 −	0.8950000 −	0.8126667
china	0.0000000	0.0530000 −	0.0530000 −	0.0530000 −
Boston	0.7580000	0.8495000 −	0.7595000	0.8093333
onekm	0.0166667	0.2600000 −−	0.2600000 −−	0.2600000 −−
cw.drag	0.8419906	0.9801429 −−	0.9801429 −−	0.9801429 −−
co2.emission	0.4213333	0.6326667 −−	0.6326667 −−	0.6326667 −−
acceleration	0.5600000	0.7963333 −−	0.7963333 −−	0.7963333 −−
available.power	0.9091224	0.9816106 −−	1.0000000 −−	1.0000000 −−
bank8FM	0.6532389	0.7605930 −	0.6798909	0.6438592
delta.ailerons	0.1293077	0.1760511 −	0.1760511 −	0.1743844 −
ibm	0.0000000	0.0055804 −	0.0055804 −	0.0055804 −
cpu.small	0.8519882	0.8854643 −−	0.8886580 −−	0.8947880 −−
delta.elevators	0.0419848	0.1635832 −−	0.1620447 −−	0.1658054 −−
cal.housing	0.3140477	0.3885872 −−	0.4608993 −−	0.4597903 −−
add	0.0140000	0.1240159 −−	0.1240159 −−	0.1240159 −−
fried.delve	0.1141905	0.09674603	0.09674603	0.09674603

Tabela B.8: Variação do parâmetro β: avaliação de *recall*.

Conjuntos de Dados	***F-measure***						
	CART	Nosso Método					
	$(1-SE)$	$(\beta = 0.5)$		$(\beta = 1)$		$(\beta = 2)$	
servo	0.8053263	0.8662569	−	0.6612034	++	0.6146256	++
triazines	0.0371048	0.2282423	−−	0.2282423	−−	0.2282423	−−
algae1	0.0000000	0.3281298	−−	0.3279042	−−	0.3245128	−−
algae2	0.0000000	0.0685473		0.0685473		0.0685473	
algae3	0.0000000	0.1186311	−	0.1186311	−	0.1186311	−
algae4	0.0000000	0.0375353		0.0636982	−	0.2183882	−−
algae5	0.0000000	0.0970766	−	0.0970766	−	0.0970766	−
algae6	0.0000000	0.0592960	−	0.0805630	−	0.2273751	−−
algae7	0.0111917	0.0551332		0.0476719		0.0453404	
machine.cpu	0.6266528	0.7473006	−	0.7591475	−−	0.7067602	
china	0.0000000	0.0401994	−	0.0401994	−	0.0401994	−
Boston	0.7675711	0.8388169	−	0.7346941		0.7436105	
onekm	0.0059831	0.2701772	−−	0.2701772	−−	0.2701772	−−
cw.drag	0.8734481	0.8976743		0.8976743		0.8976743	
co2.emission	0.4770925	0.6754074	−−	0.6775423	−−	0.6780212	−−
acceleration	0.6287861	0.7864102	−	0.7864996	−	0.7864996	−
available.power	0.9353684	0.9222137		0.9217586		0.9217586	
bank8FM	0.7592038	0.8109842		0.7594744		0.7341129	
delta.ailerons	0.1933579	0.2463114		0.2462873		0.2441828	
ibm	0.0000000	0.0035846	−	0.0035846	−	0.0035846	−
cpu.small	0.9165213	0.9351127	−−	0.9341849	−−	0.9102072	
delta.elevators	0.0719734	0.2452588	−−	0.2445691	−−	0.2484020	−−
cal.housing	0.4561053	0.4514434		0.5155620	−−	0.5147584	−−
add	0.0196822	0.1823070	−−	0.1823070	−−	0.1823070	−−
fried.delve	0.1356259	0.1161262		0.1161262		0.1161262	

Tabela B.9: Variação do parâmetro β: avaliação de *F-measure*.

Referências

Aggarwal, C. C. e Yu, P. S. (2001). Outlier detection for high dimensional data. In *Proceedings of the ACM SIGMOD International Conference in Management of Data*, pages 37 – 46. ACM Press.

Angiulli, F. e Pizzuti, C. (2002). Fast outlier detection in high dimensional space. In *Proceedings of the Sixth European Conference, PKDD 2002*, pages 15–26.

Arning, A., Argrawal, C., e Raghavan, P. (1996). A linear method for deviation detection in large databases. In *Proceedings of the Second ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 164–169.

Barnett, V. e Lewis, T. (1978). *Outliers in Statistical Data.* John Wiley and Sons Ltd.

Bi, J. e Bennett, K. P. (2003). Regression error characteristic curves. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-03)*, page ?

Breiman, L. (2001). Random forests. Technical report, Statistics Department, University of California.

Breiman, L., Friedman, J., Olshen, R., e Stone, C. (1984). *Classification and Regression Trees.* Statistics/Probability Series. Wadsworth & Brooks/Cole Advanced Books & Software.

Breunig, M. M., Kriegel, H. P., Ng, R., e Sander, J. (1999). Optics-of: Identifying local outliers. *Lecture Notes in Computer Science*, 1704:262–270.

Breunig, M. M., Kriegel, H. P., Ng, R., e Sander, J. (2000). Lof: Identifying density-based local outliers. In *Proceedings of ACM SIGMOD 2000 International Conference on Management of Data*.

Buja, A. e Lee, Y. (2001). Data mining criteria for tree based regression and classification. In *Proceedings of Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD-2001*, pages 27–36.

Chan, P. e Stolfo, S. (1998). Toward scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. In *Proceedings of the Fourth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 164–168. AAAI Press.

Chawla, N., Bowyer, K., et al. (2000). Smote: synthetic minority over-sampling technique. In *Proceedings of the International Conference on Knowledge Based Computer Systems (KBCS-2000)*.

Cleveland, W. (1993). *Visualizing Data*. Horbart Press.

Cohen, W. (1995). Fast effective rule induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123.

Cortes, C. e Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Dietterich, T. (2002). *Machine Learning for Sequential Data: a Review*. Springer-Verlag.

Domeniconi, C., shing Perng, C., Vilalta, R., e Ma, S. (2002). A classification approach for prediction of target events in temporal sequence. In *Proceedings of the Sixth European Conference, PKDD 2002*, pages 125–137.

Egan, J. (1975). *Signal detection theory and ROC analysis*. New York: Academic Press.

Fan, W., Stolfo, S., Zhang, J., e Chan, P. K. (1999). Adacost: Misclassification cost-sensitive boosting. In *Proceedings of Sixteenth International Conference on Machine Learning, ICML99*.

Fawcett, T. e Provost, F. (1999). Activity monitoring: Noticing interesting changes in behavior. In Chaudhuri, S. e Madigan, D., editors, *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 53–62. ACM.

Friedman, J. (1991). Multivariate additive regression splines. In *Annals of Statistics*, volume 19, pages 1–66.

Gopen, G. D. e Swan, J. A. (1990). The science of scientific writing. In *American Scientist (Nov-Dec, 1990)*, volume 78, pages 550–558. Scientist's Bookshelf, Menlo Park, CA.

Guimarães, R. C. e Cabral, J. A. S. (1997). *Estatística*. McGraw-Hill.

Hawkins, D. M. (1980). *Identification of Outliers*. Chapman and Hall, 11 New Fetter Lane, London EC4P 4EE.

Holte, R., Acker, L., e Porter, B. (1989). Concept learning and the problem of small disjuncts. In Sridharan, N., editor, *Proceedings of the Eleventh International Conference on Artificial Intelligence*. Morgan Kaufmann.

Jin, W., Tung, A., e Han, J. (2001). Mining top-n local outliers in large databases. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 293–298. ACM Press.

Joshi, M. V., Agarwal, R. C., e Kumar, V. (2002). Predicting rare classes: Comparing two-phase rule induction to cost-sensitive boosting. In *Proceedings of the Sixth European Conference, PKDD 2002*, pages 237–249.

Kent, A., Berry, M., Leuhrs, F., e Perry, J. (1955). Machine literature searching viii. operational criteria for designing information retrieval systems. In *American Documentation*, volume 6, pages 93–101.

Keogh, E., Lonardi, S., e Chiu, W. (2002). Finding surprising patterns in a time series database in linear time and space. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 550–556.

Knorr, E. M. e Ng, R. T. (1998). Algorithms for mining distance-based outliers in large datasets. In *Proceedings of Twenty Fourth International Conference on Very Large Data Bases (VLDB 1998)*, pages 392–403. Morgan Kaufmann, San Francisco, CA.

Kubat, M. e Matwin, S. (1997). Addressing the curse of imbalanced training sets: one-sided selection. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186.

Meadow, C., Boyce, B., e Kraft, D. (2000). *Text Information Retrieval Systems*. Academic Press, 2nd edition.

Metz, C. E. (1978). *Basic principles of ROC analysis*, volume Semin Nucl Med 8.

Provost, F. e Fawcett, T. (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 43–48. AAAI Press.

Provost, F. e Fawcett, T. (2001). Robust classification for imprecise environments. In *Machine Learning Journal, March 2001*, volume 42 No 3.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*.

Quinlan, J. R. (1999). Data mining with cubist. http://www.rulequest.com/cubist-info.html.

Quinlan, J. R. e Cameron-Jones, R. M. (1993). FOIL: A midterm report. In *Machine Learning: ECML-93, European Conference on Machine Learning, Proceedings*, pages 3–20.

Ramaswamy, S., Rastoji, R., e Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. In *Proceedings of ACM International Conference on Management of Data (SIGMOD 2000)*, pages 427–438.

Rijsbergen, C. V. (1979). *Information Retrieval*. Dept. of Computer Science, University of Glasgow, 2nd edition.

Ripley, B. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press.

Torgo, L. (1999). *Inductive Learning of Tree-based Regression Models*. PhD thesis, Universidade do Porto.

Torgo, L. (2001). A study on end-cut preference in least squares regression trees. In Brazdil, P. e Jorge, A., editors, *Proceedings of the Tenth Portuguese AI Conference (EPIA 2001)*, number 2258 in LNAI, pages 104–115. Springer.

Weiss, G. (1995). Learning with rare cases and small disjuncts. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 558–565. Morgan Kaufmann.

Weiss, G. (2002). *Predicting Telecommunication Equipment Failures from Sequences of Network Alarms*. Oxford University Press.

Weiss, G. e Hirsh, H. (1998a). Learning to predict rare events in categorical time-series data. In *AAAI Workshop on Predicting the Future: AI Approaches to Time-Series Problems*, volume WS-98-07, pages 83–90. AAAI Press, Menlo Park, CA.

Weiss, G. e Hirsh, H. (2000a). Learning to predict extremely rare events. In *AAAI Workshop on Learning from Imbalanced Data Sets*, volume WS-00-05, pages 64–68. AAAI Press, Menlo Park, CA.

Weiss, G. e Provost, F. (2001). The effect of class distribution on classifier learning: an empirical study. Technical Report Technical Report ML-TR-44, Department of Computer Science, Rutgers University.

Weiss, G. M. e Hirsh, H. (1998b). The problem with noise and small disjuncts. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML98)*. Morgan Kaufmann Publishers.

Weiss, G. M. e Hirsh, H. (2000b). A quantitative study of small disjuncts. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*. AAAI Press/MIT Press.

Witten, I. H. e Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Applications*. The Morgan Kaufmann Series in Data Management Systems.

Zhang, T., Ramakhrishnan, R., e Livny, M. (1996). Birch: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, Montreal, Quebec, Canada*, pages 103–114. ACM Press.

Zyl, J. (2001). Modelling chaotic systems with neural networks: Application to seismic event predicting in gold mines. Master's thesis, University of Stellenbosch.