## INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

### Área Departamental de Engenharia de Eletrónica e Telecomunicações e de Computadores


# Modelo de *Data Mining* para deteção de embolias pulmonares


## VIRGÍNIA VALENTE RAMALHO

Licenciada


Trabalho de Projeto para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Doutora Matilde Pós-de-Mina Pato
Mestre Nuno Miguel Soares Datia

Júri:

Presidente: Doutor Hélder Jorge Pinheiro Pita

Vogais: Doutor Daniel Pedro de Jesus Faria
Doutora Matilde Pós-de-Mina Pato

**Setembro, 2013**

## INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

### Área Departamental de Engenharia de Eletrónica e Telecomunicações e de Computadores


# Modelo de *Data Mining* para deteção de embolias pulmonares


## VIRGÍNIA VALENTE RAMALHO

Licenciada


Trabalho de Projeto para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Doutora Matilde Pós-de-Mina Pato
Mestre Nuno Miguel Soares Datia

Júri:

Presidente: Doutor Hélder Jorge Pinheiro Pita

Vogais: Doutor Daniel Pedro de Jesus Faria
Doutora Matilde Pós-de-Mina Pato

**Setembro, 2013**

*A todos os que tiveram muita paciência...*

# Agradecimentos

Aos professores Matilde Pato e Nuno Datia, orientadores deste projeto, pela disponibilidade e orientação prestada.

Ao Bruno pelo apoio constante, pelas críticas e sugestões ao longo do desenvolvimento do projeto.

Aos meus colegas da CGI por terem tido paciência para me ouvir a lamentar e resmungar durante um ano inteiro.

À família e amigos pelo apoio e compreensão pela minha ausência durante a elaboração do projeto.

*O essencial é invisível aos olhos...*

in *O Principezinho*,

Antoine de Saint-Exupéry

# Resumo

Este trabalho surge na sequência de um desafio proposto no *KDD Cup 2006*, detetar a presença de embolia pulmonar a partir de imagens médicas.

A embolia pulmonar é o bloqueio da artéria pulmonar ou de um de seus ramos. A rapidez no diagnóstico e tratamento de doentes com embolia pulmonar aguda permite reduzir a sua mortalidade. O desafio clínico, num cenário de emergência, é diagnosticar corretamente o indivíduo que apresenta a patologia, para se dar início ao tratamento. É neste ponto que técnicas de *Data Mining* podem ser usadas para produzir modelos que auxiliam o médico, radiologista, a tomar decisões. Este trabalho tem como objetivo apresentar modelos de classificação que tenham baixos rácios de falsos positivos na identificação de embolias pulmonares num indivíduo, mas apresentando valores altos de sensibilidade.

Foi criado um conjunto de dados, dividido em conjuntos de treino e de teste, que resultam da aplicação de técnicas de *Feature Selection* e de equilíbrio entre os números de casos de cada classe. Cada par foi utilizado em diferentes algoritmos de classificação. A cada combinação, conjunto de dados e algoritmo, foram aplicadas técnicas de pós-processamento, nomeadamente a alteração do ponto operacional, permitindo alterar as classificações produzidas. A avaliação dos resultados foi obtida através de métricas próprias do domínio do problema, métricas comuns em avaliação de algoritmos de classificação e uma métrica combinada produzida no âmbito deste trabalho.

Verifica-se que o algoritmo *nu-svm* com o tipo *kernel radial* pode produzir excelentes resultados perante este conjunto de dados.

**Palavras-chave:** Embolia Pulmonar, Tomografia Computorizada, *KDD Cup 2006*, *Data Mining*, *Feature Selection*, *Support Vector Machines*, classificação.

# Abstract

This work follows the challenge proposed in KDD Cup 2006, for detecting the presence of pulmonary embolism from medical images. A Pulmonary embolism is a blockage of the pulmonary artery or one of its branches. Its rapid diagnosis and treatment can reduce the mortality associated with this disease. The clinical challenge in an emergency setting, is to quickly diagnose the embolism, so the treatment can start. This is where data mining techniques can be used to produce models that help radiologists with their decisions. The goal of this paper is to present classification models that have low false positive ratios, but are high sensitive to detect pulmonary embolism in the patients.

During the development process, several data sets were created, divided in pairs of training and testing data, resulting from the application of Feature Selection techniques and balance between the numbers of cases of each class. Each pair was used with different classification algorithms, normally used in this domain. For each combination of algorithm and dataset, some post-processing techniques are used, including changing the operational point of the classifiers. The evaluation use domain metrics, common metrics for classifiers evaluation and a combined metric produced for this work.

The algorithm nu-kernel SVM with radial type, if correctly parameterized , can produce excellent results against this data set. Models that have produced the best results for this data set were implemented with this algorithm.

**Keywords:** Pulmonary Embolism, Computed Tomography, KDD Cup 2006, Data Mining, Feature Selection, Support Vector Machines, classification.

# Nomenclatura

Angio-TC Angiografia por tomografia computorizada

AUC *Area under the curve*

CAD *Computer-Aided Detection*

CRISP-DM *Cross Industry Standard Process for Data Mining*

FN *False negatives*

FP *False positives*

FR *Feature Ranking*

FS *Feature Selection*

KDD *Knowledge Discovery in Databases*

PE *Pulmonary embolism*

PV *Positive values*

ROC *Receiver Operating Characteristic*

SS *Subset Selection*

SVM *Support Vector Machines*

TAC Tomografia axial computorizada

TC Tomografia computorizada

TN *True nagatives*

TP *True positives*

# Conteúdo

**1 Introdução** **1**

1.1 Motivação . . . . . 1

1.2 Objetivos . . . . . 3

1.3 Anatomia e fisiologia do pulmão . . . . . 3

1.4 Embolia Pulmonar . . . . . 5

1.5 Tomografia Computorizada . . . . . 7

1.6 Os sistemas CAD . . . . . 9

1.7 Organização do documento . . . . . 9

**2 Descoberta de Conhecimento** **11**

2.1 Processo KDD por Fayyad . . . . . 12

2.2 Metodologia CRISP-DM . . . . . 13

2.3 *Feature Selection* . . . . . 15

2.3.1 Algoritmos de *Feature Ranking* . . . . . 15

2.4 *Data Mining* . . . . . 16

2.4.1 Algoritmos de *Data Mining* . . . . . 17

2.5 Trabalhos relacionados . . . . . 18

2.6 A escolha do processo KDD e tecnologia . . . . . 20

**3 Exploração dos dados** **21**

3.1 Compreensão do problema e dos dados . . . . . 21

3.2 Preparação dos dados . . . . . 22

3.2.1 Correlação entre variáveis . . . . . 25

3.2.2 Redução de variáveis . . . . . 26

3.2.3 O equilíbrio do conjunto de dados . . . . . 27

3.2.4 Indicadores de desempenho . . . . . 28

**4 Modelação e Avaliação** **29**

4.1 Modelação dos dados . . . . . 29

4.1.1 Classificação utilizando o conjunto de dados original . . . . 30

4.1.2 Classificação utilizando o conjunto de dados equilibrado . . 37

4.1.3 Reparametrização dos algoritmos SVM . . . . . 40

4.2 Avaliação . . . . . 43

**5 Conclusão** **49**

5.1 Trabalho futuro . . . . . 50

**Bibliografia** **54**

**A Anexos** **i**

# Lista de Figuras

1.1 O sistema respiratório humano. . . . . 4
1.2 Embolia pulmonar. . . . . 6
1.3 O *scanner*. . . . . 7
1.4 Angiografia por tomografia computorizada. . . . . 8

2.1 As etapas do processo KDD. . . . . 12
2.2 O ciclo CRISP-DM. . . . . 14

3.1 Apresentação dos candidatos em 2D. . . . . 23
3.2 Apresentação dos candidatos em 3D. . . . . 24

4.1 Curvas ROC usando todas as variáveis. . . . . 31
4.2 Diferença na curva ROC com a variação de conjunto de dados . . . 35
4.3 Curvas ROC usando todas as variáveis sobre o algoritmo SVM. . . 42

# Lista de Tabelas


3.1 Correlação entre variáveis. . . . . 25

4.1 Resultados obtidos com o conjunto de dados original. . . . . 32

4.2 Resultados obtidos com o conjunto de dados original. . . . . 32

4.3 Resultados após ajuste da classificação, alterando o ponto de operação. . . . . 34

4.4 Resultados após *Feature Ranking*. . . . . 34

4.5 Resultados após ajuste da classificação com *Feature Ranking*. . . . . 36

4.6 Os piores resultados após ajuste da classificação com *Feature Ranking*. 36

4.7 Melhores resultados usando o conjunto de dados original. . . . . 37

4.8 Resultados obtidos com o conjunto de dados equilibrado. . . . . 38

4.9 Resultados obtidos com o conjunto de dados equilibrado. . . . . 38

4.10 Resultados globais obtidos com o conjunto de dados equilibrado. . 39

4.11 Resultados globais obtidos com o conjunto de dados equilibrado. . 39

4.12 Melhores resultados do algoritmo SVM após otimização. . . . . 41

4.13 Melhores resultados do algoritmo *svm radial* com *v-classification*. . . 43

4.14 Melhores resultados do algoritmo SVM. . . . . 43

4.15 Maior *PE sensitivity* com limite de 2 *FP max*. . . . . 44

4.16 Maior *PA sensitivity* com limite de 2 *FP max*. . . . . 44

4.17 Maior *PE sensitivity* com limite de 4 *FP max*. . . . . 45

4.18 Maior *PA sensitivity* com limite de 4 *FP max*. . . . . 45
4.19 Maior *PE sensitivity* com limite de 10 *FP max*. . . . . 46
4.20 Maior *PA sensitivity* com limite de 10 *FP max*. . . . . 46
4.21 Resultado do concurso *KDD Cup 2006* para *PE sensitivity*. . . . . 48
4.22 Resultado do concurso *KDD Cup 2006* para *PA sensitivity*. . . . . 48
4.23 Resultados obtidos a partir deste estudo. . . . . 48

# Listagens

A.1 Função para determinar *PE sensitivity*. . . . . i
A.2 Função para determinar *PA sensitivity*. . . . . i
A.3 Função para fazer *Feature Ranking* sobre o conjunto de dados. . . . ii
A.4 Função para fazer o equilíbrio do conjunto de dados. . . . . iii
A.5 Função para determinar *FP max*. . . . . iv
A.6 Função para ajustar a classificação com base no ponto de operação. iv
A.7 Otimização do algoritmo SVM. . . . . iv

# 1

# Introdução

O trabalho apresentado neste estudo surge de um desafio proposto no *KDD Cup 2006*: deteção de embolia pulmonar a partir de imagens médicas - tomografia computorizada (TC). O objetivo é desenvolver um modelo usando técnicas de *Data Mining* que permita identificar automaticamente a presença da patologia - embolia pulmonar - num indivíduo.

Neste capítulo é apresentada a motivação e os objetivos do trabalho, assim como toda a informação necessária para compreender os passos efetuados durante este estudo. São descritos de forma simples os conceitos médicos que dão origem a este estudo, para que seja possível compreender a necessidade da criação deste novo modelo.

## 1.1 Motivação

A embolia pulmonar ocorre quando um coágulo de sangue se solta do vaso onde se formou e viaja na circulação sanguínea até aos pulmões, obstruindo uma artéria pulmonar e interrompendo o fluxo normal do sangue. Uma embolia pulmonar nem sempre é fatal, mas é uma das causas de morte inesperadas mais comuns [29]. A rapidez no diagnóstico e tratamento de doentes com embolia pulmonar aguda permite reduzir a sua mortalidade.

O desafio clínico, num cenário de emergência, é diagnosticar corretamente o indivíduo que apresenta a patologia para se dar início ao tratamento. Entre os

sintomas mais comuns destaca-se a dispneia (dificuldade em respirar, acompanhada de uma sensação de mal-estar) (80%), a dor torácica pleurítica (52%), tosse (20%) e hemoptise (11%) [41]. Sintomas como palpitações ou dor anginosa podem ocorrer menos frequentemente [23, 31]. A Angiografia Pulmonar por tomografia computorizada (Angio-TC Pulmonar) é considerada o melhor método de diagnóstico de embolias pulmonares [14]. Este exame pulmonar é obtido depois de uma inspiração profunda e após a injeção intra-venosa do agente de contraste (iodo). O agente de contraste não realça os coágulos, permitindo identificar e visualizar diretamente a região onde ocorre a obstrução do vaso [31, 33, 44].

A TC produz uma série de imagens que representam uma parte do corpo. Cada imagem corresponde a uma secção ou "fatia" da zona do corpo analisada. No caso do pulmão, o uso destas imagens permitem aos clínicos identificar embolias pulmonares. Um aspecto importante no diagnóstico efetuado pelos médicos é o elevado número de imagens de cada exame, que produzem em média 300-500 aquisições axiais por paciente [33, 44], tornando a sua análise muito demorada. Por este motivo, os métodos automáticos de visualização e segmentação de embolias pulmonares em Angio-TC Pulmonar permitem uma análise rápida de todas as secções do exame, auxiliando o diagnóstico médico.

A análise de uma TC para deteção de embolias pulmonares é um processo moroso se for feito exclusivamente por um analista. Por este motivo, pretende-se desenvolver um modelo, que permita identificar automaticamente a presença de embolia pulmonar, usando técnicas de *Data Mining*. Para que seja utilizável, o modelo, deve apresentar um número reduzido de falsos positivos.

A informação de uma TC pode ser apresentada sob a forma de imagem ou sob a forma de uma tabela de dados onde estão descritas todas as características da imagem. Muitas das variáveis presentes nestas tabelas podem ser redundantes, causando entropia sobre os restantes dados. As técnicas de *Data Mining* permitem realizar um estudo exaustivo sobre os dados e construir ou aperfeiçoar um modelo que permite a classificação dos casos com e sem a doença.

Os sistemas informáticos permitem a automatização de atividades manuais repetitivas em que os resultados podem ser calculados. Isto permite a agilização de muitos processos, assim como a diminuição de tempo de execução. Na medicina tem-se procurado agilizar os processos de diagnóstico através da criação de sistemas informáticos, capazes de obter esses resultados de forma rápida e fiável. Neste estudo, procura-se encontrar um modelo que possa ser usado por estes sistemas informáticos para diagnosticar embolias pulmonares.

Pretende-se construir um modelo que possa obter melhores resultados que outros estudos sobre o mesmo tema, de forma a que se possa contribuir para a evolução deste tipo de diagnóstico. Os estudos anteriores são utilizados como uma base de informação, de forma a que se possa obter melhores resultados.

## 1.2 Objetivos

O objetivo deste estudo foi a criação de um modelo, usando técnicas de *Data Mining*, que permita classificar, a partir de imagens médicas, a existência de embolia pulmonar. O modelo construído deve ter capacidade para classificar corretamente este tipo de dados. Tendo em conta os resultados dos vencedores do *KDD Cup 2006*, o objetivo passa por conseguir melhores resultados aplicando sempre algoritmos e técnicas já disponíveis, estudadas e experimentadas em estudos anteriores.

Para encontrar o modelo adequado a esta tarefa foram escolhidos, ao longo do estudo, alguns algoritmos que se utilizaram em vários ensaios. Em cada ensaio executaram-se vários tipos de teste. Os resultados dos testes são analisados e comparados, para que se possa avaliar a capacidade de classificação dos algoritmos em determinados conjuntos de treino.

Para determinar o(s) modelo(s) são, de acordo com o *KDD Cup 2006*, estabelecidos limites quanto ao número de falsos positivos permitidos. Uma vez que se trata de um estudo comparativo, os resultados obtidos devem classificar corretamente: (i) uma embolia pulmonar num paciente com um limite máximo de 2, 4 e 10 falsos positivos por paciente, e (ii) se um paciente apresenta ou não embolia pulmonar com um limite máximo de 2, 4 e 10 falsos positivos por paciente.

## 1.3 Anatomia e fisiologia do pulmão

Dá-se o nome de aparelho respiratório ao conjunto de órgãos que permitem a captação de oxigénio ($O_2$) e a eliminação de dióxido de carbono ($CO_2$) produzido na respiração interna. O objetivo da respiração é fornecer oxigénio para os tecidos e remover o dióxido de carbono. Podemos identificar quatro funções principais: (1) ventilação pulmonar, ou seja, a entrada e saída de ar que se dá entre a atmosfera e os alvéolos pulmonares; (2) difusão de $O_2$ e $CO_2$ entre os alvéolos e o sangue; (3) transporte de $O_2$ e $CO_2$ no sangue e fluidos do corpo de e para o tecido das células; (4) regulação da ventilação e outras facetas da respiração [20].

No ser humano, o processo respiratório tem como órgão central o pulmão. Este compõem-se, por sua vez, por dois órgãos de forma piramidal, localizados no interior do tórax, como apresentado na Figura 1.1. A base de cada pulmão apoia-se no diafragma, músculo esquelético, que serve de fronteira entre a cavidade torácica e a abdominal, promovendo, juntamente com os músculos intercostais, os movimentos respiratórios. Localizado logo acima do estômago, o nervo frénico controla os movimentos do diafragma.

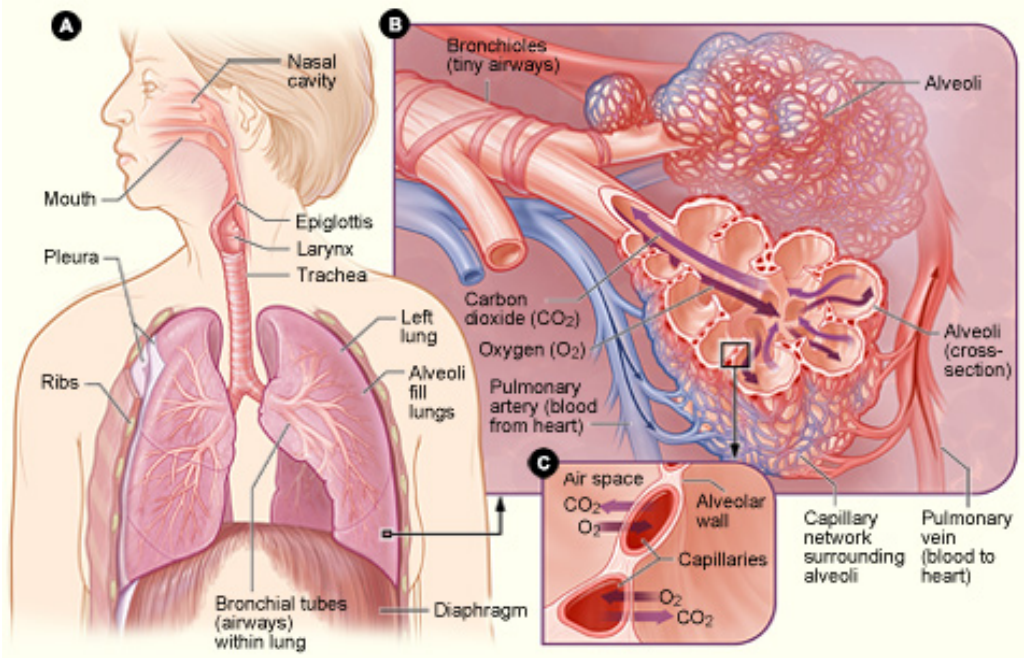



Figura 1.1: O sistema respiratório humano. (Fonte: National Heart, Lung, and Blood Institute)

Os pulmões estão envolvidos por uma membrana húmida chamada pleura, a qual reveste igualmente o interior da caixa torácica. Estas superfícies lubrificantes possibilitam a sua expansão e contração, bem como as trocas gasosas.

Os pulmões apresentam características morfológicas diferentes e estão divididos por fissuras, formando os lobos pulmonares. Cada lobo pulmonar é subdividido em vários segmentos pulmonares, ventilados por brônquios específicos. Os segmentos estão separados por planos de tecido conjuntivo, denominados septos intersegmentares, que se estendem para dentro do tecido pulmonar da pleura. Cada segmento pulmonar é formado por vários lóbulos pulmonares.

Cada pulmão apresenta na sua face interna uma grande fissura, o hilo pulmonar, através do qual os brônquios e vasos sanguíneos penetram no órgão. É através dos hilos que os brônquios principais transportam o sangue para o coração, e as

veias pulmonares entram no interior dos pulmões. Uma vez no interior, ver Figura 1.1, os brônquios principais ramificam-se em segmentos progressivamente mais pequenos. As últimas ramificações são os bronquíolos terminais, que chegam a todo o tecido pulmonar [26].

Na parte terminal, os bronquíolos abrem-se numa ampola formada por alvéolos. É a este nível que ocorrem as trocas gasosas entre o ar que chega do exterior e o sangue, passando o oxigénio, por difusão, do ar para o sangue, e fazendo o $CO_2$ o percurso inverso. Os alvéolos têm uma parede muito fina e permeável e são cobertos por uma rede capilar resultante da ramificação da artéria pulmonar. As artérias pulmonares acompanham a árvore brônquica, dividindo-se também por dicotomia, até aos alvéolos [20, 26].

## 1.4 Embolia Pulmonar

A embolia pulmonar é uma patologia que ocorre quando um êmbolo (coágulo) atinge os pulmões, bloqueando, total ou parcialmente, uma artéria pulmonar.

A coagulação do sangue é um mecanismo de proteção do organismo, que previne a perda de sangue e leva à formação de uma massa constituída por plaquetas e glóbulos vermelhos.

O êmbolo pode formar-se em qualquer parte do corpo, normalmente nos membros inferiores. O êmbolo é um fragmento de um trombo, um coágulo que se forma agarrado à parede de um vaso sanguíneo, que viaja pela corrente sanguínea. Em circulação, pode encontrar vasos sanguíneos de dimensão reduzida, podendo provocar obstrução, como representado na Figura 1.2.

Êmbolos de menores dimensões bloqueiam ramos de pequeno calibre, em regiões já na periferia do pulmão, sendo assintomáticos ou apresentando sintomas locais como dor ou tosse. Se a obstrução é provocada por êmbolos de dimensões superiores, ocorrendo em ramos maiores como o tronco da artéria pulmonar ou nos ramos lobares, a gravidade aumenta. Para além da dor e da tosse, o doente apresenta súbita falta de ar, palpitações e tosse com expetoração e sangue.

Assim que o sangue chega ao coração, é imediatamente bombeado para a artéria pulmonar, que, por sua vez, o irá distribuir por todo o pulmão para que este seja novamente oxigenado. Posteriormente, o sangue volta ao coração para ser bombeado novamente para o resto do corpo. Se ocorrer uma obstrução da circulação sanguínea durante este processo, pode levar a que o sangue seja devolvido ao

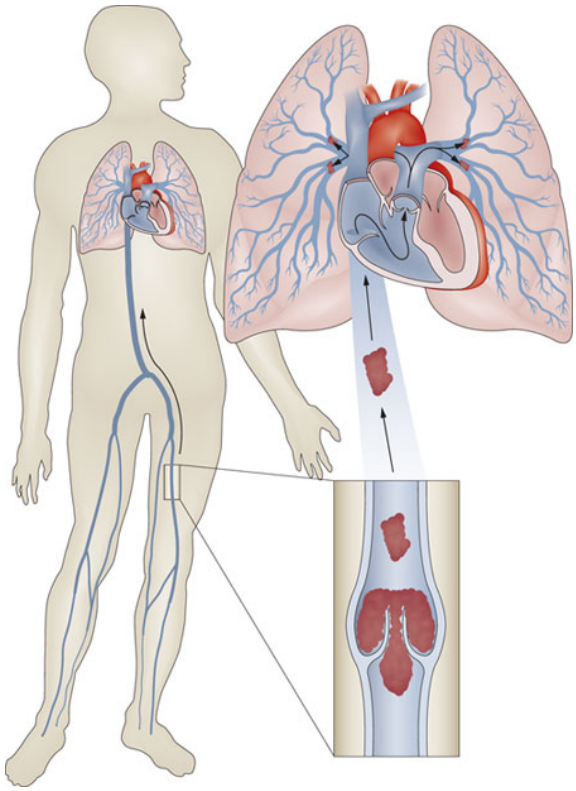


Figura 1.2: Embolia pulmonar. (Fonte: Nature.com)

coração, provocando um súbito aumento de pressão e uma rápida dilatação do coração, que pode resultar em morte súbita.

Os principais fatores que aumentam o risco de trombose venosa, ou seja, a oclusão total ou parcial de uma veia por um trombo, são longos períodos de inatividade, lesões ou cirurgias nos membros inferiores, tumores, gravidez, parto, obesidade, distúrbios hematológicos, tratamentos e medicamentos.

Aquando da descoberta da fisiopatologia e etiologia da embolia pulmonar por Virchow, este autor referiu que, a propósito das manifestações clínicas desta doença, "em caso de existirem grandes trombos dentro dos ramos principais da artéria pulmonar ocorre bloqueio imediato com consequente asfixia instantânea"[14]. No entanto, atualmente sabe-se que o diagnóstico clínico de embolia pulmonar é bastante difícil, podendo esta patologia apresentar-se clinicamente de várias formas. Como tal, as manifestações clínicas de embolia pulmonar oscilam dentro de um amplo espetro de sinais e sintomas cujos extremos vão desde formas assintomáticas até à morte súbita.

A prevalência e incidência exata de embolia pulmonar é de difícil obtenção, não só pela pouca especificidade dos sintomas, como também pelo facto de que esta muitas vezes pode ser assintomática ou ter como única manifestação a morte

súbita. Além disso, as estimativas da incidência de embolia pulmonar variam consideravelmente consoante a população estudada, os recursos disponíveis e os critérios de diagnóstico utilizados [9, 39].

A incidência de embolia pulmonar é de cerca de 120 casos por cada 100 000 habitantes por ano nos Estados Unidos da América. É aos pacientes internados em entidades hospitalares que cabe a maior incidência de embolia pulmonar. Verifica-se uma baixa incidência desta doença entre indivíduos com idades iguais ou inferiores a 14 anos, traduzida por valores inferiores a 1 por 100 000 pessoas por ano. Por sua vez, a incidência aumenta drasticamente a partir dos 50 anos, atingindo valores de incidência na ordem dos 1000 por cada 100 000 habitantes em indivíduos com 85 anos ou mais. A acumulação de múltiplos fatores de risco, como doenças concomitantes e diminuição da mobilidade, por exemplo, está na génese da maior prevalência desta doença em indivíduos idosos.

## 1.5 Tomografia Computorizada

A tomografia computorizada (TC) é um exame de diagnóstico que permite obter uma imagem do corpo humano. Cada imagem obtida, através da TC, representa uma secção ou "fatia" do corpo.

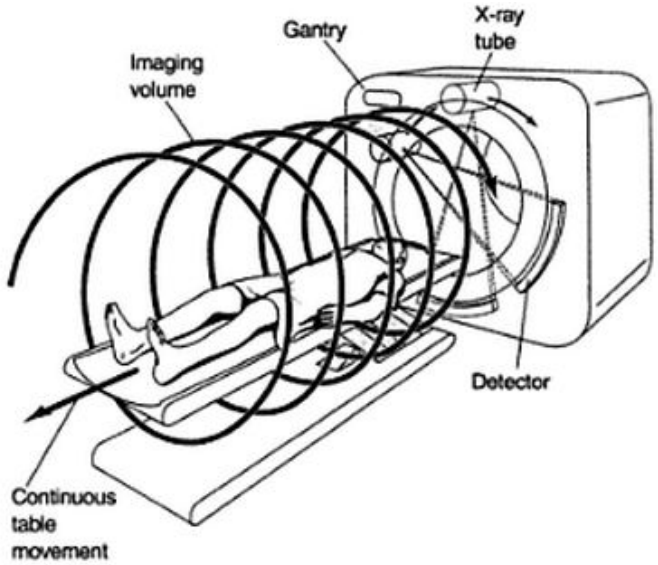



Figura 1.3: O *scanner*. (Fonte: Novelline e Squire, [30])

O nome original deste exame é tomografia axial computorizada (TAC), é assim que hoje em dia, ainda é conhecido pela maior parte das pessoas. A mudança de

nome ocorreu com o aparecimento dos sistemas helicoidais, que permitem obter o volume de cada parte do corpo apresentado na "fatia" retirada.

Para obter este tipo de imagens foi necessário alterar a forma como o raio X iria percorrer o corpo. A mesa onde está o paciente tem um movimento contínuo e passa através do *scanner* de onde são emitidos raios X a partir de um tubo que percorre o *scanner* em movimentos circulares e traçam uma espiral em volta do corpo. A Figura 1.3 apresenta a forma como são obtidas as imagens numa TC [30].

Através deste mecanismo é possível obter os dados em que cada unidade é o *voxel*, ou seja, um pixel volumétrico. É possível construir imagens de cada uma das secções analisadas em três dimensões e calcular o volume dos órgãos visíveis. Estes sistemas podem devolver os dados obtidos no exame através de imagens ou simplesmente através de tabelas de dados, que fornecem os valores obtidos para cada uma das características analisadas [37].

A Angio-TC permite a visualização da estrutura vascular auxiliada por injeções de contraste. Após o processamento da imagem obtida pela TC é possível ver todo o sistema vascular em três dimensões. Esta técnica permite ver os vasos mais pequenos, onde poderão existir problemas como aneurismas ou embolias. A Figura 1.4 mostra o resultado de uma angiografia por TC sobre o pulmão direito. A seta indica o local onde poderá existir uma embolia.

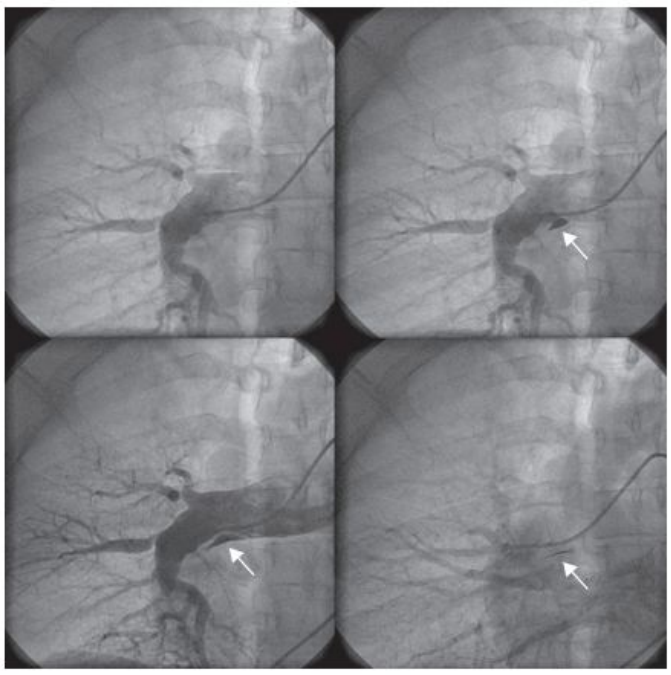


Figura 1.4: Angiografia por tomografia computorizada. (Fonte: Baptista, *et al* [2])

## 1.6 Os sistemas CAD

Os sistemas de deteção assistida por computador (CAD, do termo em inglês *Computer-Aided Detection*), são hoje em dia cada vez mais utilizados para deteção de patologias através de imagens médicas. As aplicações CAD podem fornecer desafios interessantes em *Data Mining*, pois os conjuntos de dados normalmente obtidos são grandes e sem balanceamento entre casos positivos e negativos.

Os exames médicos são convertidos para conjuntos de dados, em que cada característica encontrada na imagem é convertida numa coluna do conjunto de dados. Esta busca de características pode fazer com que sejam introduzidas no conjunto de dados variáveis irrelevantes e redundantes. Outro fator que torna os conjuntos de dados mais complicados de analisar, é o facto de os médicos especialistas na análise dos exames médicos colocarem marcas que podem introduzir ruído sobre os dados.

Normalmente, um sistema CAD deve demonstrar uma melhoria significativa no desempenho clínico. No limite, um sistema CAD e um médico especialista devem conseguir o mesmo resultado sobre a análise de um exame, sem que exista um aumento de falsos positivos. A diminuição de falsos positivos é importante para que não sejam efetuados exames médicos desnecessários e dispendiosos.

## 1.7 Organização do documento

O presente capítulo faz o enquadramento dos temas abordados neste estudo, mencionando a motivação, os objetivos propostos e o contexto médico. É também apresentada a estrutura do documento, sintetizando os assuntos abordados nos vários capítulos.

No segundo capítulo são analisadas as técnicas mais comuns para efetuar a descoberta de conhecimento em bases de dados.

No terceiro capítulo é descrito todo o desenvolvimento efetuado para compreender e preparar os dados para a modelação.

No quarto capítulo é demonstrado todo o desenvolvimento efetuado para construir um modelo de *Data Mining*, utilizando o conjunto de dados fornecido. Relatam-se os ensaios realizados para encontrar o melhor modelo. Por fim, é efetuado um estudo comparativo com os resultados do *KDD Cup 2006*.

No quinto capítulo são apresentadas as conclusões finais.

# 2

# Descoberta de Conhecimento em Base de Dados

A Descoberta de Conhecimento em Base de Dados, geralmente conhecida por KDD (tradução do inglês de *Knowledge Discovery in Databases*), tem como objetivo extrair conhecimento de bases de dados.

A utilização de processos de KDD tornou-se numa prática frequente com o aumento da quantidade de informação guardada em bases de dados. Além de ser em grande quantidade, a informação é complexa, tornando-se impossível de ser analisada manualmente por analistas. Assim, passou a existir a necessidade de extrair dessa informação todo o conhecimento possível de forma automática. O conhecimento adquirido é utilizado para apoio à decisão em variados processos, por exemplo nas decisões empresariais em relação aos serviços a fornecer aos seus clientes, pois é importante que a empresa saiba qual é o tipo de produto que os seus clientes estão à espera de encontrar. Para sistemas financeiros também é muito importante conseguir perceber as tendências do mercado (previsões para o futuro) através de acontecimentos anteriores. Estes sistemas obtêm esse conhecimento através de processos KDD nas bases de dados que registam todos os acontecimentos.

Destaca-se também a aplicação destes processos na investigação científica, biologia e medicina. Nestas áreas existe grande quantidade de dados, na maioria das

vezes quase impossível de analisar manualmente. Na medicina, na fase de diagnóstico, uma análise correta e rápida a esse grande volume de dados é importante na identificação de patologias.

De entre as várias metodologias existentes [24] para a descoberta de conhecimento em bases de dados, neste estudo são apresentados os processos KDD analisados do ponto de vista de Fayyad [19] e a metodologia CRISP-DM (Cross Industry Standard Process for Data Mining) [36], desenhada por um consórcio de empresas.

## 2.1 Processo KDD por Fayyad

O processo KDD é um processo iterativo e interativo, envolvendo vários passos e decisões do utilizador. O KDD pode ser visto como o processo da descoberta de novas correlações, padrões, modelos e tendências significativas por meio da análise minuciosa de grandes conjuntos de dados. Os padrões/modelos encontrados devem manter-se válidos quando aplicados sobre novos dados, para que seja possível interpretá-los corretamente [17].

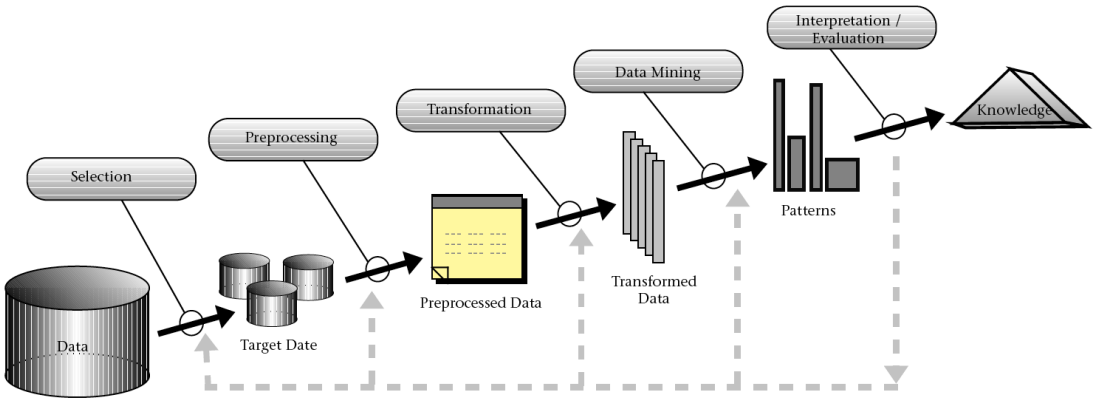



Figura 2.1: As etapas do processo KDD. (Fonte: Fayyad, Piatetsky-Shapiro e Smyth [17])

O processo KDD acompanha todo o percurso de descoberta de conhecimento em bases de dados, desde a forma como os dados são armazenados e acedidos, à análise do desempenho dos conjuntos de dados devido ao seu tamanho e à forma

como os resultados são interpretados e visualizados. Segundo Fayyad, Piatetsky-Shapiro e Smyth [17], o processo KDD tem nove etapas distintas, ilustradas na Figura 2.1. Nos seguintes pontos é efetuada uma breve descrição de cada etapa.

1. Definição do objetivo do processo do ponto de vista do cliente, definição do domínio da aplicação e identificação do pré-conhecimento relevante.
2. Seleção dos dados sobre os quais será efetuado o processo.
3. Pré-processamento e limpeza de dados. São efetuadas operações básicas sobre os dados, com o principal objetivo de eliminar ruído.
4. Redução e projeção dos dados de forma a encontrar variáveis que possam definir o objetivo da tarefa.
5. Escolha do método de *Data Mining* a utilizar, por exemplo classificação, regressão ou *clustering*.
6. Escolha dos algoritmos e parâmetros a utilizar.
7. *Data Mining*: pesquisa de padrões no conjunto de dados.
8. Interpretação dos padrões obtidos. Nesta etapa é avaliada a necessidade de nova iteração sobre as etapas do processo KDD.
9. Aplicação do conhecimento obtido. Usar o conhecimento diretamente noutros sistemas ou simplesmente documentá-lo.

## 2.2 Metodologia CRISP-DM

A metodologia CRISP-DM foi desenvolvida pelo consórcio CRISP-DM: NCR Engenharia de Sistemas de Copenhaga (EUA e Dinamarca), a DaimlerChrysler AG (Alemanha), a SPSS Inc. (EUA), e OHRA Verzekeringen en Bank Groep BV (Holanda). O consórcio publicou um guia [10] que apresenta esta metodologia.

Na Figura 2.2 é apresentado o ciclo de vida de um processo CRISP-DM. Este ciclo é composto por seis etapas mas, a sua sequência não é fixa. A passagem de uma etapa para a seguinte depende sempre do resultado da atual. Na avaliação dos resultados de uma etapa, pode verificar-se a necessidade de recuar mais do que uma etapa no processo. Este ciclo é independente da tecnologia usada para o desenvolvimento.

As etapas do processo CRISP-DM podem ser definidas da seguinte forma:

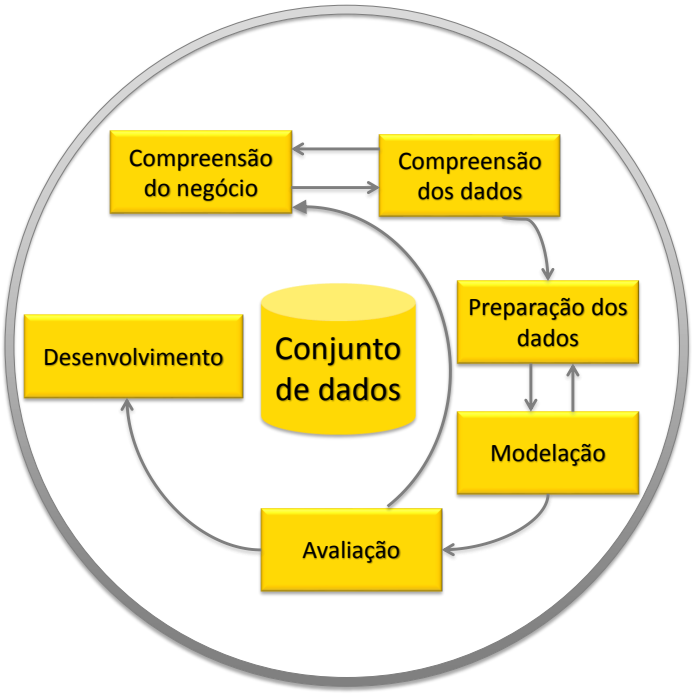



Figura 2.2: O ciclo CRISP-DM.

1. Compreensão do negócio - nesta primeira fase é necessário conhecer o objetivo do projeto do ponto de vista do negócio. Devem ser definidos os problemas, os riscos, os recursos e as tecnologias de desenvolvimento.

2. Compreensão dos dados - pretende-se obter e conhecer os dados, de forma a que seja possível identificar as suas qualidades e defeitos, e identificar subconjuntos relacionando-os com o objetivo.

3. Preparação dos dados - nesta fase deve ficar garantido que os dados contêm toda a informação necessária que permita gerar um modelo. Esta etapa tem como objetivo a criação de um conjunto de dados que será utilizado na etapa seguinte. São efetuadas tarefas de redução e projeção dos dados de forma a selecionar variáveis.

4. Modelação - são aplicadas várias técnicas de *Data Mining*, alterando os algoritmos e os seus parâmetros de forma a conseguir aperfeiçoar o modelo obtido. Nesta etapa pode verificar-se a necessidade de retroceder à etapa anterior se o conjunto de dados precisar de novos ajustes.

5. Avaliação - são avaliados os resultados obtidos na fase anterior. Com o modelo obtido deve ser verificado, usando um conjunto de dados de teste, se

este consegue responder aos objetivos do problema. Nesta fase pode existir a necessidade do processo voltar à primeira etapa, no caso dos resultados não serem os esperados.

6. Desenvolvimento - o modelo é disponibilizado ao cliente, para que este possa usá-lo. O analista deve certificar-se que o modelo gerado irá ser usado pelo cliente de forma correta, para que este possa obter os resultados pretendidos.

## 2.3 *Feature Selection*

O *Feature Selection* (FS) é usado durante a fase de preparação de dados de um processo KDD. É um processo de escolha de um conjunto de variáveis, de forma a que seja possível reduzir o conjunto original, de acordo com um determinado critério de avaliação.

O objetivo passa por remover variáveis irrelevantes e redundantes, aumentando a eficiência, desempenho e precisão nos resultados durante a fase de modelação. Tendo em conta o aumento de dados nas bases de dados, devido à massificação das aplicações informáticas, este tipo de seleção tornou-se muito importante para a modelação em *Data Mining*.

As técnicas de *Feature Selection* podem ser divididas em duas abordagens: *Feature Ranking* (FR) e *Subset Selection* (SS) [32, 35, 43]. Na primeira, é atribuído um peso às variáveis mediante um determinado critério, sendo selecionadas as N variáveis com pesos mais altos. Na segunda, é procurado o melhor subconjunto entre vários construídos com as variáveis do conjunto inicial. Para a avaliação do subconjunto é usado um algoritmo de classificação. A segunda abordagem requer um processamento computacional superior à primeira.

Neste estudo utiliza-se a abordagem de *Feature Ranking* de forma a encontrar as variáveis de maior importância no conjunto de dados. As variáveis selecionadas são posteriormente utilizadas nos algoritmos de *Data Mining*.

### 2.3.1 Algoritmos de *Feature Ranking*

O principal objetivo do *Feature Ranking* é reduzir a dimensão do conjunto de dados para diminuir o tempo de processamento computacional. Isto é muito importante principalmente em situações em que existe uma grande dimensão do

conjunto de dados. Por outro lado, deve garantir-se que, ao efetuar esta redução de dimensão, se irá manter ou melhorar, o desempenho do modelo.

Neste estudo utilizaram-se alguns dos algoritmos de *Feature Ranking* implementados na linguagem de programação R: *Chi-squared Filter*, *Information Gain*, *Gain Ratio* e *Symmetrical Uncertainty*, *OneR* e *Random Forest Filter*. Os algoritmos estão disponíveis no pacote *FSelector*. Este pacote disponibiliza funções que, dado um conjunto de dados, permitem a seleção de variáveis através de um processo de identificação e remoção, se a informação da variável é irrelevante ou redundante.

O algoritmo *Chi-squared Filter* calcula o peso das variáveis usando como base o teste do *chi*-quadrado. O *chi*-quadrado, representado por $\chi^2$, é um teste de hipóteses que se destina a encontrar o valor da dispersão para duas variáveis nominais, avaliando a associação existente entre variáveis qualitativas. Quanto maior o valor de *chi*-quadrado, mais significante é a relação entre a variável dependente e a variável independente [43].

Os algoritmos *Information Gain*, *Gain Ratio* e *Symmetrical Uncertainty* medem a diminuição de entropia quando determinada variável está presente ou ausente do conjunto de dados. Calculam o peso das variáveis baseando-se na sua correlação com a variável dependente [43].

O algoritmo *OneR* calcula o peso das variáveis usando o classificador *OneR*. Para cada variável é criada uma regra baseada unicamente nessa variável e é calculada a taxa de erro [43].

O algoritmo *Random Forest Filter* calcula o peso das variáveis usando o algoritmo *RandomForest*. Durante a classificação são alternadas as variáveis usadas e a quantidade de variáveis. Com o resultado da classificação é calculada a taxa de erro da classificação [8].

## 2.4 *Data Mining*

O *Data Mining* surge com a necessidade de extrair automaticamente informação potencialmente útil a partir das bases de dados, tipicamente de elevada dimensão e/ou complexidade. A informação obtida através de técnicas de *Data Mining* tem o propósito de revelar relações desconhecidas e disponibilizar os dados de forma útil e compreensível para os utilizadores. São utilizadas técnicas de inteligência artificial com o objetivo de encontrar padrões, anomalias ou regras. São designados por modelos os resultados da aplicação destas técnicas. Um modelo

pode adotar várias formas de representação, equações, regras, árvores ou grafos [18].

Neste estudo utilizaram-se técnicas de classificação, pois o objetivo passa por encontrar um modelo que possa identificar corretamente se um paciente tem ou não embolia pulmonar. A classificação tem como objetivo a construção de modelos que permitam atribuir uma classe, de um conjunto conhecido à partida, com base nos valores de um conjunto de variáveis. Existem dois tipos de análises que levam a que sejam utilizadas técnicas diferentes: classificação e regressão. A classificação envolve uma variável de saída discreta e exige a atribuição de uma classe a um conjunto de valores para as variáveis de entrada. Na regressão as variáveis de saída são contínuas e os modelos são construídos tipicamente com base em expressões matemáticas.

### 2.4.1 Algoritmos de *Data Mining*

Selecionaram-se alguns algoritmos de *Data Mining* para aplicar sobre o conjunto de dados. A escolha foi feita através da análise de alguns trabalhos efetuados na mesma área (ver no subcapítulo 2.5), tendo em conta os algoritmos com melhores resultados em situações semelhantes.

O algoritmo *Recursive Partitioning* constrói modelos em que o resultado pode ser representado por uma árvore de classificação [6]. Para construir a árvore, é procurada a variável que melhor divide o conjunto de dados em dois e os dados são separados. Aos subconjuntos criados é aplicado este procedimento recursivamente enquanto for útil dividir os subconjuntos.

O algoritmo *Conditional Inference Trees*, é muito semelhante ao anterior, usa testes estatísticos de hipóteses para determinar como vai fazer a próxima divisão do conjunto [22]. Todas as variáveis são avaliadas em cada passo. O modelo construído pelo algoritmo pode ser representado através de uma árvore de classificação.

O algoritmo *Bagging Classification, Regression and Survival Trees* combina vários modelos, com base em árvores de classificação, treinados com amostras do conjunto de dados original. O resultado da classificação é definido pela maioria dos votos [7].

O algoritmo *Classification and Regression with Random Forest* é um classificador onde é construído um conjunto de árvores de classificação [21]. Em cada nó de cada árvore, é utilizado um conjunto aleatório de variáveis fazendo com que a

divisão dos conjuntos seja diferente de árvore para árvore. O resultado final da classificação é definido por votação do resultado de todas as árvores.

O algoritmo *Random Forest* é uma combinação entre os algoritmos de classificação *Random Forest* e o *Bagging*, em que é utilizado o algoritmo *Conditional Inference Trees* como método de aprendizagem [8].

O algoritmo *Fit Multinomial Log-linear Models* constrói um modelo com base em redes neuronais [34].

O algoritmo *k-Nearest Neighbors classifier* pertence a um conjunto de algoritmos baseados em instâncias [1]. Utiliza o conjunto de dados para construir amostras com instâncias cuja classe é conhecida. Posteriormente, as instâncias cuja classe não é conhecida, serão classificadas com base numa votação efetuada relativa às *k* instâncias mais próximas. Este algoritmo é também conhecido por ser um *lazy learning*, visto que deixa a classificação para o momento da predição.

O algoritmo *Support Vector Machines* (SVM) tem como objetivo encontrar um hiperplano, num espaço com elevada dimensionalidade, que permita separar os diferentes vetores pela sua classe [13]. São chamados vetores de suporte aos vetores mais próximos do hiperplano traçado. Uma vez que, na maioria dos problemas não existe uma separação linear, recorre-se a uma função de *kernel* que efetua uma transformação não linear dos dados, para que estes sejam separáveis de modo linear.

## 2.5 Trabalhos relacionados

Estudaram-se alguns trabalhos relacionados com o tema deste estudo de forma a identificar as técnicas mais utilizadas e com melhores resultados. Através destes trabalhos foi efetuada a escolha dos algoritmos de *Data Mining* que foram mencionados no subcapítulo anterior.

Meyer, Leisch e Hornik [28] apresentam uma comparação entre *Support Vector Machines* (SVM) e outros algoritmos de classificação e regressão. O algoritmo SVM apresentou bons resultados tanto nas tarefas de classificação como de regressão. Contudo alguns dos outros algoritmos também apresentaram resultados muito competitivos. Nas tarefas de classificação, o algoritmo SVM apresentou os melhores resultados na maioria dos testes. Nas tarefas de regressão, os melhores resultados foram apresentados por algoritmos baseados em redes neuronais e pelo *Random Forest*. Os autores concluíram que o algoritmo SVM não pode ser considerado o melhor para todo o tipo de tarefas.

Parimala e Nallaswamy [32] realizaram um estudo em que apresentaram uma nova técnica de *Feature Selection* utilizando o pacote *FSelector* do R. A avaliação dos resultados foi feita utilizando o algoritmo SVM para classificar o conjunto de dados, utilizando as variáveis escolhidas por cada um dos algoritmos de *Feature Ranking*. Neste estudo concluiu-se que a precisão do algoritmo aumentou quando foram retiradas variáveis desnecessárias, reduzindo assim o processamento computacional e a dimensão do conjunto de dados.

Lavrač [25] efetuou um estudo com o objetivo de apresentar algumas técnicas de *Data Mining* que podem ser aplicadas na medicina e discutiu algumas das suas características usadas para resolver problemas médicos. Toda a sua análise teve em conta a alta precisão que deve resultar quando aplicadas as técnicas sobre um conjunto de dados. A sensibilidade, especificidade, probabilidade e o custo de classificação incorreto são apresentados como alternativas à precisão da classificação para avaliar a qualidade de um classificador.

Sluimer, *et al* [38] apresentam um estudo sobre os sistemas de diagnóstico auxiliados por computador utilizando imagens médicas. O estudo foi efetuado com base em tomografias computorizadas feitas aos pulmões. Com as imagens médicas obtidas foram construídos conjuntos de dados que posteriormente são classificados utilizando técnicas de *data Mining*, de forma a que seja possível identificar automaticamente quando existe uma anomalia no tecido pulmonar. Foram usados vários classificadores com vários subconjuntos e os resultados foram avaliados com recurso à análise ROC (*Receiver Operating Characteristic*) [16]. Os autores verificaram a necessidade de uma etapa de *Feature Selection* devido ao elevado número de variáveis do conjunto de dados. O classificador que obteve melhores resultados foi o *k-Nearest Neighbors classifier*, que conseguiu apresentar um desempenho muito semelhante à classificação manual feita por dois especialistas radiológicos.

Tikk, Kardkovács e Szidarovszky apresentam em [40] o método utilizado para resolver o desafio apresentado no *KDD Cup 2006*. Foi construído um comité de classificadores com regras próprias para determinar a decisão final. O processo de votação atribui pesos a cada um dos classificadores de acordo com o seu desempenho nos testes de *ten-fold cross-validation* e o seu nível de confiança na classificação. Foi sempre considerado um máximo de 4 falsos positivos por paciente e a solução foi otimizada tendo em conta este valor. Com esta solução, foram vencedores na segunda tarefa do *KDD Cup 2006* e ficaram na segunda posição na primeira tarefa.

## 2.6 A escolha do processo KDD e tecnologia

Verificou-se, na análise realizada sobre os processos KDD mais conhecidos e utilizados, que os dois identificam as mesmas tarefas num processo de descoberta de conhecimento sobre bases de dados. Ambos os processos são iterativos e interativos em diversas etapas. A diferença mais percetível entre estes dois processos está no número de etapas em que estão subdivididos.

Neste estudo será seguida a metodologia CRISP-DM para criação de um modelo *Data Mining*, para deteção de embolias pulmonares, num conjunto de dados obtido através de imagens médicas. A análise inicial que se fez ao problema levou a que fosse escolhida esta metodologia, pois esta era a que melhor se identificava com o passos definidos para o estudo.

A tecnologia utilizada é o R[1]. O R é ao mesmo tempo uma linguagem de programação e um ambiente para computação estatística e gráfica. Trata-se de uma tecnologia especializada em computação com dados.

[1] http://www.r-project.org/

# 3

# Exploração dos dados

Este estudo tem por base dois conjuntos de dados disponibilizados no *KDD Cup 2006*, realizado pela ACM SIGKDD[1]. Ambos foram construídos através da leitura das imagens obtidas por Angio-TC. Os resultados dos exames foram analisados por especialistas de forma a identificar quais os pacientes em que poderia estar presente a embolia pulmonar.

Os conjuntos de dados disponibilizados, de treino e de teste, foram construídos a partir de 69 exames. Em ambos os conjuntos existem casos positivos e negativos, ou seja, existem exames em que foi diagnosticada a embolia pulmonar e noutros não. O conjunto de treino contém 38 exames positivos e 8 exames negativos. O conjunto de teste contém 21 exames positivos e 2 exames negativos.

Tendo por base a metodologia CRISP-DM, foi efetuada a análise aos dados seguindo as etapas definidas por este processo.

## 3.1 Compreensão do problema e dos dados

Utilizando o conjunto de dados fornecido é necessário construir um modelo *Data Mining* que permita identificar corretamente pacientes com embolia pulmonar. Para que esta tarefa seja possível é necessário conhecer e compreender o conjunto de dados que contém os resultados dos exames feitos aos pacientes.

[1] http://www.sigkdd.org

Ao observar os dados verifica-se que por cada paciente analisado existe um conjunto de registos. Os 69 exames deram origem a 4429 registos, dos quais 3038 estão no conjunto de treino. Cada registo representa um candidato a *Pulmonary embolism* (PE).

Cada candidato é um *cluster* de *voxels* que representa uma "fatia" ou secção do pulmão obtida através da Angio-TC. Assim, cada paciente é representado por vários candidatos que representam todas as "fatias" do seu pulmão. Cada candidato contém 116 variáveis em que três delas representam a sua posição (x, y , z) em relação ao pulmão. As restantes variáveis são retiradas nas imagens recolhidas na Angio-TC e representam a intensidade do *voxel* do candidato, a intensidade da sua vizinhança e a forma em 3D .

Quando a marcação de embolia pulmonar é feita pelo especialista, é guardada a posição da marca efetuada. Esta marca é usada para definir se um candidato deve ser ou não identificado como positivo. Se o candidato estiver a uma determinada distância da marca efetuada pelo especialista, o candidato é marcado como positivo. Esta identificação é efetuada por um gerador automático.

O conjunto de dados tem duas variáveis para indicar o paciente analisado e se é um candidato positivo. A coluna que indica se o paciente tem embolia pulmonar contém o valor zero no caso de ser um candidato negativo e contém um valor positivo no caso de ser um candidato positivo. Um paciente pode ter mais do que uma embolia pulmonar, em que cada uma é identificada com um valor positivo único.

## 3.2 Preparação dos dados

Utilizando a linguagem de programação R é efetuada toda a análise e tratamento dos dados. O conjunto de dados é uma tabela de dados em que cada coluna é considerada uma variável. As variáveis estão identificadas no conjunto de dados de V1 a V118. Segue-se uma descrição sucinta das variáveis.

1. **Identificador do paciente** (V1) - Cada paciente em que foi feita uma Angio-TC é identificado por um número inteiro positivo entre 3000 e 28010. Esse identificador está presente em todos os candidatos pertencentes ao paciente.

2. **Embolia pulmonar (positivo/negativo)** (V2) - O valor zero indica que se está perante um candidato sem embolia pulmonar (negativo). Os valores

superiores a zero representam um candidato com embolia pulmonar (positivo). Esta variável será considerada a variável dependente que será usada no modelo de classificação. Tendo em conta que a informação dada por esta variável tem apenas dois valores possíveis, negativo ou positivo, todos os valores superiores a zero são substituídos pelo valor 1.


Figura 3.1: Apresentação dos candidatos em 2D.

3. **Posição em X** (V3) - Esta variável representa a posição do candidato em $x$. Todos os valores presentes nesta coluna são valores inteiros positivos.

4. **Posição em Y** (V4) - Esta variável representa a posição do candidato em $y$. Todos os valores presentes nesta coluna são valores inteiros positivos.

5. **Posição em Z** (V5) - Esta variável representa a posição do candidato em $z$. Todos os valores presentes nesta coluna são valores inteiros positivos.

6. **Características da imagem** (de V6 a V118) - Estas variáveis são constituídas por valores numéricos compreendidos entre 1 e -1 com média em 0. Não

existem valores indefinidos (NA). Estes valores são características proprietárias do sistema que constrói o conjunto de dados com base nas imagens. Não existe informação precisa sobre o que cada uma das variáveis representa.

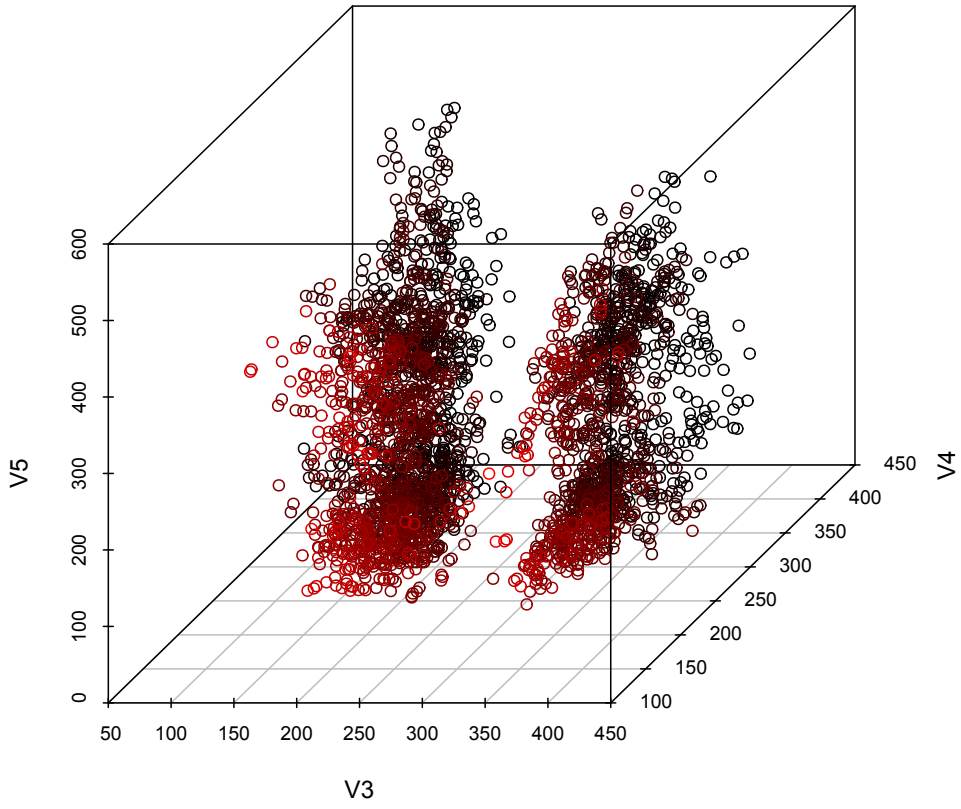



Figura 3.2: Apresentação dos candidatos em 3D.

As Figuras 3.1 e 3.2 apresentam os pontos onde estão localizados os candidatos no pulmão. As imagens são produzidas através do processamento de quatro variáveis, V2, V3, V4 e V5. Por observação da Figura 3.1 é possível perceber que não existem zonas fixas no pulmão para a ocorrência de embolias pulmonares, pois os candidatos classificados como positivos encontram-se espalhados uniformemente pela imagem.

### 3.2.1 Correlação entre variáveis

A correlação entre variáveis é uma forma de medir a relação entre as mesmas. A correlação entre as variáveis é normalmente expressa pelo coeficiente de correlação de *Pearson*. Este coeficiente varia entre -1 e 1, onde o primeiro valor indica uma forte correlação negativa, o segundo valor uma forte correlação positiva e o valor 0 indica que as variáveis não estão correlacionadas.

Utilizando como suporte o R é possível calcular a correlação entre as variáveis automaticamente, sendo o resultado apresentado sob a forma de uma matriz que relaciona todas as variáveis. Utilizou-se o conjunto de dados original para determinar a correlação linear direta entre as variáveis independentes e a variável dependente. Neste estudo não se verificou nenhuma dependência deste tipo.

Apresenta-se na tabela 3.1, os pares de variáveis que apresentaram uma correlação de 0.9 e 0.95. Esta informação é útil para efetuar a redução de variáveis num conjunto de dados, ou seja, quando duas variáveis estão fortemente correlacionadas a eliminação de uma delas poderá melhorar o desempenho de um algoritmo sem afetar os resultados.

| **0.9** | | **0.95** | |
|---|---|---|---|
| V29 | V23 | V66 | V3 |
| V35 | V29 | V36 | V30 |
| V62 | V56 | V57 | V56 |
| V62 | V57 | V60 | V59 |
| V96 | V89 | V63 | V56 |
| V96 | V94 | V63 | V57 |
| V110 | V107 | V63 | V62 |
| V114 | V113 | V75 | V74 |
| V116 | V113 | V80 | V79 |
| | | V89 | V88 |
| | | V94 | V88 |
| | | V94 | V89 |
| | | V102 | V99 |
| | | V114 | V112 |
| | | V116 | V112 |
| | | V116 | V114 |

Tabela 3.1: Correlação entre variáveis.

### 3.2.2 Redução de variáveis

A grande quantidade de variáveis do conjunto de dados original, que devem ser analisadas pelo modelo, podem não trazer vantagens. É aceite pela comunidade científica que, em certos casos a classificação é melhor se for feita sobre um espaço mais reduzido. Mas para muitos algoritmos, o treino e teste é mais eficiente num espaço dimensional menor. Beyer *et al* [4] diz que para um conjunto de dados ser considerado de grande dimensão, basta que o número de variáveis seja superior a 10 ou 15. Visto que se verificou existir elevada correlação entre algumas variáveis, pode ser útil reduzir a dimensão do conjunto de dados, sem que sejam afetados os indicadores de desempenho dos algoritmos.

Para construir conjuntos de dados de menor dimensão utiliza-se uma das abordagens de *Feature Selection*, o *Feature Ranking*. Os algoritmos de *ranking* são utilizados para atribuir a cada variável um valor que identifica a importância desta na determinação da variável dependente. A redução de variáveis no conjunto de dados é feita com base neste valor de importância, isto é, são removidas aquelas que apresentam o menor valor.

O objetivo passa por construir automaticamente regras que possam ser usadas nos algoritmos de *Data Mining*. As regras construídas indicam quais as variáveis que serão usadas para determinar a variável dependente. As regras têm o seguinte formato: (variável dependente) ˜ (variáveis independentes que levam à determinação da variável dependente). Na prática, uma regra tem a seguinte estrutura: (V2) ˜ (V4 + V5 + V6 + V7).

Criou-se uma função em R que determina as variáveis de maior importância através da utilização de vários algoritmos de *ranking*. Os algoritmos utilizados foram *Chi-squared Filter*, *Information Gain*, *Gain Ratio*, *Symmetrical Uncertainty*, *OneR* e *Random Forest Filter*, sendo que o seu funcionamento está descrito no subcapítulo 2.3.1. O código da função implementada consta em anexo na listagem A.3.

Cada um dos algoritmos, tendo em conta o seu tipo de avaliação, permite uma ordenação através do peso que atribui a cada variável. Para construir a regra, são obtidas as variáveis de maior peso através de cada um dos algoritmos. A quantidade de variáveis obtidas está limitada a um número previamente definido. Dos vários conjuntos de variáveis mais significativos, obtidos pelos algoritmos, são considerados para o conjunto de dados final os mais frequentes.

Sabendo que o conjunto de dados original contém 118 variáveis, construíram-se 11 regras, em que a quantidade de variáveis independentes pode variar entre 10

e 110. Estas regras são usadas sobre o conjunto de dados original, para construir novos conjuntos de dados apenas com as variáveis escolhidas. Os conjuntos de dados passam a ser mencionados da seguinte forma: (i) DORG representa o conjunto de dados original; (ii) DXXX representa um conjunto de dados com uma seleção de variáveis independentes onde XXX indica o número de variáveis selecionadas, por exemplo D010 ou D110.

O resultado da função implementada é uma regra que é posteriormente aplicada nos algoritmos de *Data Mining*, utilizando o conjunto de dados original. Na prática não é necessária a criação manual do conjunto de dados, o algoritmo faz isso automaticamente.

### 3.2.3 O equilíbrio do conjunto de dados

O conjunto de dados fornecido contém maioritariamente pacientes com embolia pulmonar, o que poderá influenciar os resultados durante a modelação dos dados [11, 42]. Durante a preparação dos dados houve a necessidade de criar uma função que, dado um conjunto de dados, o pudesse tornar equilibrado, onde o número de pacientes com e sem embolia pulmonar fosse igual.

Neste estudo, as técnicas usadas para ajustar a distribuição dos casos num conjunto de dados são o *oversampling* e *undersampling* [12]. O código da listagem A.4, presente nos anexos, apresenta a função *equi* construída em R que permite aplicar esta técnica a um conjunto de dados.

Utilizou-se a função *equi* para construir um conjunto de dados com base no conjunto de dados original em que o número de casos positivos e casos negativos fosse igual. Construiu-se um conjunto de dados com 40 casos, 20 casos positivos e 20 casos negativos. Como no conjunto de dados original apenas existem 8 casos negativos, para construir o conjunto de dados é necessário repetir muitos destes casos.

Tendo em conta o conjunto de treino, o *oversampling* é utilizado para selecionar pacientes sem embolia pulmonar até atingir metade do número de casos pretendidos. Utilizando o *undersampling* são selecionados pacientes aleatoriamente até atingir metade do número de casos pretendidos.

O novo conjunto de dados balanceado obtido passa a ser mencionado como DBAL.

### 3.2.4 Indicadores de desempenho

Para avaliar os resultados obtidos após a utilização dos algoritmos de *Data Mining* sobre um conjunto de dados usam-se algumas métricas conhecidas como o TN, FP, FN, TP e o valor da área abaixo da curva ROC (AUC) [16]. Foram também utilizadas outras métricas mais específicas neste tipo de problemas:

- ***PE sensitivity*** - Representa o número de embolias pulmonares identificadas por paciente. Para que seja considerada uma embolia pulmonar, basta que seja classificado corretamente um candidato positivo dessa embolia. Em anexo a listagem A.1 apresenta a implementação desta métrica em R.
- ***PA sensitivity*** - Representa o número de pacientes aos quais foi diagnosticado embolia pulmonar. Para que um paciente seja considerado positivo, basta que seja classificado corretamente um candidato positivo desse paciente. Em anexo a listagem A.2 apresenta a implementação desta métrica em R.
- ***FP max*** - Indica a maior quantidade de falsos positivos encontrados num paciente. Em anexo a listagem A.5 apresenta a implementação desta métrica em R.

Verificou-se a necessidade de construir uma nova métrica que pudesse ser, simultaneamente, um indicador de desempenho e um fator de ordenação. O novo indicador tem o nome de **PV *(Positive values)*** visto que é calculado com base em todas as métricas positivas mencionadas. As métricas utilizadas são aquelas em que quanto maior é o seu valor, melhor é o algoritmo.

As métricas usadas foram multiplicadas por um valor que representa o seu peso na determinação do novo indicador. O cálculo efetuado para determinar o valor de PV é o seguinte:

$$\text{PV} = \text{AUC} * 0.2 + \text{TN} * 0.1 + \text{TP} * 0.1 + \text{PE sens} * 0.3 + \text{PA sens} * 0.3$$

As métricas *PE sensitivity* e *PA sensitivity* têm uma maior importância perante todas as outras, pois estas são as métricas utilizadas no *KDD Cup 2006*, possibilitando assim a comparação de resultados. O novo indicador de desemprenho apenas permite a comparação entre testes realizados com diferentes algoritmos sobre o mesmo conjunto de testes.

# 4

# Modelação e Avaliação

Tendo em conta o processo de desenvolvimento CRISP-DM, após as etapas de compreensão e preparação de dados, seguem-se a construção e avaliação do modelo. Assim sendo, nos próximos subcapítulos são descritos os passos para efetuar estas últimas etapas.

## 4.1 Modelação dos dados

Considerando as etapas anteriormente percorridas para obtenção de conhecimento sobre os dados e a criação de conjuntos de dados reduzidos, este é o momento indicado para iniciar a construção do modelo. Nos próximos subcapítulos apresentam-se as várias etapas efetuadas para encontrar o modelo que melhor deteta embolias pulmonares.

A abordagem adotada passou pelos seguintes passos:

1. Selecionar os algoritmos de *Data Mining* a utilizar na classificação.
2. Aplicar os algoritmos de *Data Mining* selecionados sobre o conjunto de dados, considerando todas as variáveis.
3. Aplicar os algoritmos de *Data Mining* selecionados sobre o conjunto de dados, aplicando regras para redução do número de variáveis independentes.

4. Aplicar os algoritmos de *Data Mining* selecionados sobre o conjunto de dados equilibrado.

5. Alterar as parametrizações por omissão dos algoritmos de *Data Mining* com melhor desempenho.

6. Ajuste do ponto de operação dos modelos.

Para selecionar os algoritmos de *Data Mining* foram efetuados vários testes utilizando o conjunto de dados original. O único fator de comparação usado foi o número de falsos positivos em cada uma das classificações. Os algoritmos escolhidos foram aqueles que apresentaram menor quantidade de falsos positivos, utilizando todos os seus parâmetros com os valores por omissão. Os restantes algoritmos foram excluídos deste estudo e por essa razão os seus resultados não foram descritos os testes efetuados.

Na lista seguinte apresentam-se os algoritmos de *Data Mining* utilizados, estando uma breve descrição destes no subcapítulo 2.4.1. Nos próximos subcapítulos passam a ser mencionados da seguinte forma:

- **rpart** - *Recursive Partitioning*
- **ctree** - *Conditional Inference Trees*
- **cforest** - *Random Forest*
- **bagging** - *Bagging Classification, Regression and Survival Trees*
- **multinom** - *Fit Multinomial Log-linear Models*
- **randomForest** - *Classification and Regression with Random Forest*
- **svm** - *Support Vector Machines*
- **IBk** - *k-Nearest Neighbors classifier*

### 4.1.1 Classificação utilizando o conjunto de dados original

Os algoritmos foram aplicados sobre o conjunto de dados original, isto é, todas as variáveis são utilizadas para a determinação da variável dependente. Nesta fase, utilizaram-se todos os algoritmos escolhidos com os parâmetros por omissão, exceto o algoritmo SVM. Este último, permite a escolha do tipo de *kernel*, ou

seja, permite escolher a função de *kernel* que será utilizada para mapear os dados para um espaço de dimensão superior. Assim, utilizou-se o algoritmo SVM nas quatro versões de *kernel* disponíveis (*linear*, *polynomial*, *radial basis* e *sigmoid*), de forma a analisar a melhor função de *kernel* tendo em conta os dados. Todos os outros parâmetros mantiveram os valores por omissão.

Os resultados obtidos serviram de base de comparação do desempenho dos algoritmos e permitiram escolher aqueles que foram alvo de posteriores avaliações, com outras parametrizações.

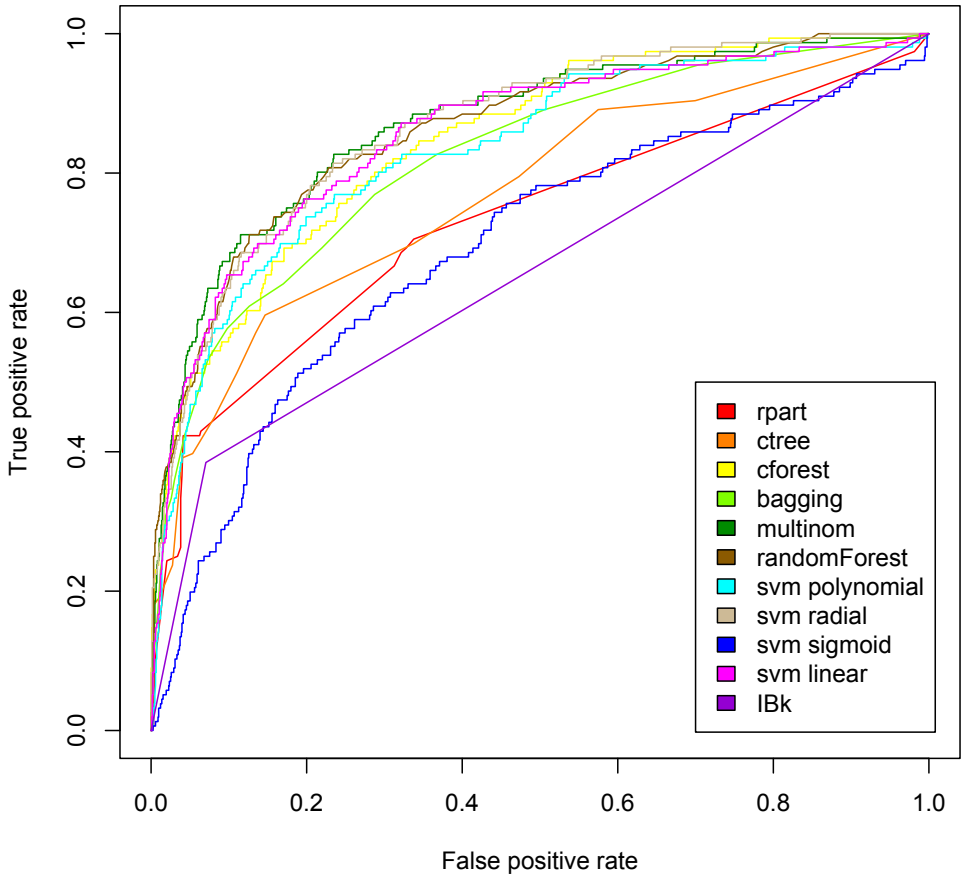



Figura 4.1: Curvas ROC obtidas com o conjunto de dados original.

A Figura 4.1 apresenta as curvas ROC determinadas para cada um dos algoritmos usados, tendo em conta o conjunto de treino utilizado. É possível verificar que os algoritmos que apresentam uma área abaixo da curva inferior são os mesmos que apresentaram na sua classificação um maior número de valores errados.

As tabelas 4.1 e 4.2 apresentam os valores obtidos e as métricas calculadas através

do resultado da classificação. Tendo em conta os resultados obtidos, foi possível retirar algumas conclusões sobre o comportamento dos algoritmos perante este conjunto de treino.

| **Algoritmo** | **TN** | **FP** | **FN** | **TP** |
|---|---|---|---|---|
| svm radial | 1229 | 6 | 121 | 35 |
| randomForest | 1221 | 14 | 106 | 50 |
| cforest | 1215 | 20 | 107 | 49 |
| svm polynomial | 1215 | 20 | 113 | 43 |
| bagging | 1201 | 34 | 93 | 63 |
| svm linear | 1200 | 35 | 89 | 67 |
| rpart | 1188 | 47 | 104 | 52 |
| multinom | 1187 | 48 | 81 | 75 |
| ctree | 1186 | 49 | 95 | 61 |
| IBk | 1136 | 99 | 96 | 60 |
| svm sigmoid | 1132 | 103 | 115 | 41 |

Tabela 4.1: Resultados obtidos com o conjunto de dados original.

| **Algoritmo** | **AUC** | **PE Sens** | **PA Sens** | **FP max** | **PV** |
|---|---|---|---|---|---|
| multinom | 0,870 | 1,826 | 17 | 11 | 129,168 |
| randomForest | 0,860 | 1,391 | 15 | 3 | 129,150 |
| bagging | 0,832 | 1,739 | 16 | 8 | 129,071 |
| svm linear | 0,856 | 1,609 | 15 | 6 | 129,043 |
| cforest | 0,845 | 1,435 | 16 | 3 | 128,635 |
| svm radial | 0,867 | 1,043 | 14 | 1 | 127,888 |
| svm polynomial | 0,833 | 1,261 | 14 | 3 | 127,555 |
| ctree | 0,769 | 1,739 | 16 | 6 | 127,358 |
| rpart | 0,730 | 1,696 | 16 | 16 | 126,588 |
| IBk | 0,657 | 1,609 | 17 | 15 | 122,193 |
| svm sigmoid | 0,692 | 1,130 | 16 | 20 | 119,066 |

Tabela 4.2: Resultados obtidos com o conjunto de dados original.

Observando a tabela 4.1 é possível verificar que todos os algoritmos apresentam um grande número de acertos na classificação, mas alguns deles obtiveram um número bastante elevado de falsos positivos, fazendo assim com que o uso destes algoritmos não seja tão fiável como desejado. Tendo em conta os resultados, os algoritmos *ctree*, *rpart*, *IBk* e *svm sigmoid* apresentam um número muito elevado de classificações falsas, tanto de falsos positivos como de falsos negativos. Estes resultados indicam que estes algoritmos não são adequados para o conjunto de treino usado, e não cumprem os objetivos mínimos pretendidos para o classificador.

Pela tabela 4.2 verifica-se que o algoritmo *multinom* apresenta os melhores resultados, tendo como referência as métricas usadas para medir o desempenho do algoritmo neste conjunto de dados. Os resultados deste algoritmo são claramente influenciados pelos valores dos indicadores *PE* e *PA sensitivity*. Contudo, é um dos algoritmos com maior número de falsos positivos por paciente.

Para encontrar o algoritmo que proporcione os resultados mais equilibrados entre a quantidade de acertos e o menor número de classificações erradas é necessário avaliar todas as métricas conjuntamente. Os algoritmos que apresentaram, nesta fase, produzir os resultados mais equilibrados foram o *randomForest*, *cforest* e *svm radial*. O algoritmo *svm radial* não apresentou um *PE Sensitivity* tão alto como os outros algoritmos, mas por outro lado, teve o *FP max* mais baixo entre todos os algoritmos.

#### 4.1.1.1 Ajuste da classificação, alterando o ponto de operação do modelo

A aplicação do modelo ao conjunto de teste produz dois valores para cada caso: a probabilidade do candidato ser negativo e a probabilidade do candidato ser positivo. Construiu-se uma função em R que dado o resultado de uma classificação e a probabilidade mínima para ser declarado positivo, é devolvida uma nova classificação. Esta função tem como objetivo diminuir o número de falsos positivos. Quando um caso é classificado como positivo é verificado se a sua probabilidade de ser positivo é superior ao valor da probabilidade mínima, se essa condição não se verificar é alterada a classificação do candidato para negativo. Em anexo a listagem A.6 apresenta a função descrita.

A nova função foi aplicada sobre os resultados obtidos através do ensaio do subcapítulo 4.1.1. Efetuaram-se cinco testes para cada algoritmo, em que a probabilidade pretendida para que o candidato fosse positivo variasse pelos seguintes valores: 0.5, 0.6, 0.7, 0.8 e 0.9. Os melhores resultados, tendo em conta o fator de ordenação PV, para cada um dos algoritmos, estão apresentados na tabela 4.3.

A capacidade de classificar corretamente não sofreu grandes alterações face ao ensaio anterior, visto que os valores de *PE sensitivity* se mantiveram muito idênticos.

Verificou-se que a maioria dos algoritmos melhorou a sua capacidade de classificação nos testes em que o ajuste foi feito com um ponto de operação a 0.5. Todos os algoritmos apresentaram valores muito elevados de falsos negativos nos testes em que o ajuste foi efetuado com o ponto de operação igual a 0.9, assim este valor deixou de ser considerado nos ensaios que se seguiram.

| Algoritmo | % | AUC | PE Sens | PA Sens | FP max | PV |
|---|---|---|---|---|---|---|
| multinom | 0,5 | 0,870 | 1,826 | 17 | 11 | 129,168 |
| randomForest | 0,5 | 0,864 | 1,348 | 15 | 3 | 128,992 |
| cforest | 0,5 | 0,847 | 1,391 | 16 | 3 | 128,773 |
| bagging | 0,5 | 0,815 | 1,609 | 16 | 5 | 128,280 |
| svm radial | 0,5 | 0,867 | 1,043 | 14 | 1 | 127,888 |
| svm linear | 0,5 | 0,856 | 1,217 | 13 | 6 | 127,695 |
| ctree | 0,5 | 0,769 | 1,739 | 16 | 6 | 127,358 |
| rpart | 0,8 | 0,730 | 1,478 | 15 | 3 | 126,942 |
| svm polynomial | 0,5 | 0,833 | 0,391 | 8 | 2 | 124,248 |
| IBk | 0,5 | 0,657 | 1,609 | 17 | 15 | 123,493 |

Tabela 4.3: Resultados após ajuste da classificação, alterando o ponto de operação.

O algoritmo *svm sigmoid* classificou todos os candidatos com a mesma classe, independentemente do valor dado para a probabilidade mínima. Com este algoritmo foram classificados todos os candidatos como negativos, logo, foram desconsiderados estes valores.

#### 4.1.1.2 Diminuição de variáveis independentes através de *Feature Ranking*

Durante a fase de preparação de dados, no subcapítulo 3.2.2, foram construídas as regras a aplicar sobre o conjunto de dados original de forma a construir novos conjuntos de dados com menor dimensão.

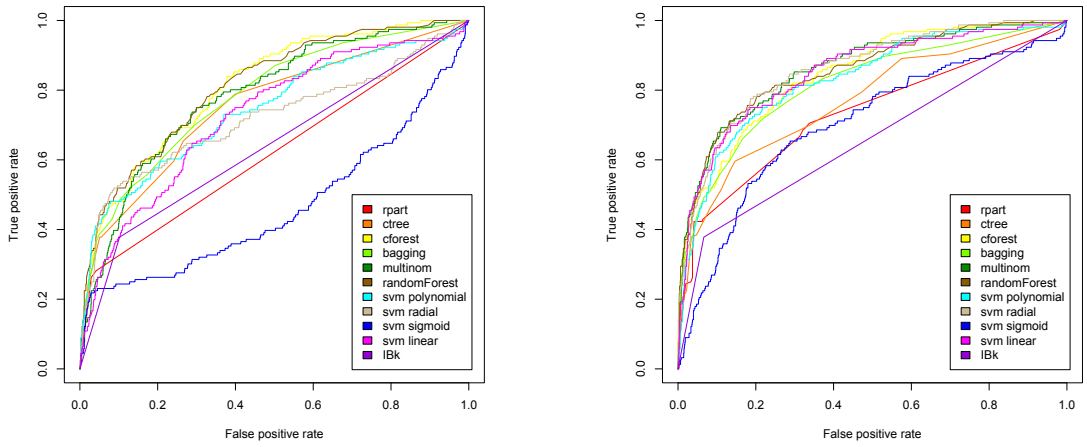
Neste ensaio foram aplicados os mesmos testes do subcapítulo 4.1.1, onde foi utilizado o conjunto de dados original, a todos os novos conjuntos de dados de menor dimensão.

| Algoritmo | AUC | PE Sens | PA Sens | FP max | FR | PV |
|---|---|---|---|---|---|---|
| randomForest | 0,867 | 1,478 | 15 | 3 | D050 | 129,669 |
| multinom | 0,861 | 1,826 | 17 | 15 | D110 | 129,266 |
| bagging | 0,835 | 1,565 | 16 | 4 | D030 | 129,221 |
| cforest | 0,863 | 1,522 | 16 | 4 | D050 | 128,864 |
| svm radial | 0,848 | 1,087 | 14 | 1 | D080 | 128,239 |
| ctree | 0,797 | 1,696 | 16 | 5 | D040 | 128,001 |
| svm polynomial | 0,831 | 1,261 | 15 | 3 | D100 | 127,768 |
| svm linear | 0,850 | 1,304 | 14 | 4 | D110 | 127,713 |
| rpart | 0,731 | 1,696 | 16 | 16 | D110 | 126,688 |
| IBk | 0,672 | 1,696 | 16 | 19 | D060 | 123,276 |
| svm sigmoid | 0,684 | 1,174 | 16 | 21 | D100 | 118,827 |

Tabela 4.4: Resultados após *Feature Ranking*.

Com este ensaio verificou-se que a capacidade de classificar corretamente continua a apresentar valores semelhantes aos ensaios anteriores. A tabela 4.4 apresenta os resultados dos testes em que o fator de ordenação PV é superior para cada um dos algoritmos usados. Verificou-se que os melhores resultados dos algoritmos são obtidos com dimensões diferentes, ou seja, não existe nenhuma dimensão que seja mais eficiente para todos os algoritmos. Também se verifica uma melhoria generalizada na classificação quando se utilizam conjuntos de dados com maior dimensão.

Na Figura 4.2 é apresentada a diferença de aspecto da curva ROC tendo em conta os resultados obtidos com os conjuntos de dados de maior e menor dimensão. Verificou-se que quando é utilizado um menor número de variáveis independentes a área abaixo da curva diminui.




(a) Conjunto de dados D010 (b) Conjunto de dados D110

Figura 4.2: Diferença na curva ROC com a variação de conjunto de dados

#### 4.1.1.3 Ajuste da classificação e diminuição de variáveis independentes

Tendo por base os dois ensaios anteriores, verificou-se que poderia ser benéfico para a análise, a combinação destes dois tipos de teste. Efetuou-se um novo ensaio para fazer o ajuste da classificação, tendo em conta um ponto de operação, sobre o resultado da classificação efetuada por cada um dos algoritmos, utilizando todos os conjuntos de dados construídos. Os valores a aplicar no ponto de operação são: 0.5, 0.6, 0.7 e 0.8. Na tabela 4.5 apresentam-se os melhores resultados para cada um dos algoritmos tendo em conta o fator de ordenação PV.

| Algoritmo | % | AUC | PE Sens | PA Sens | FP max | FR | PV |
|---|---|---|---|---|---|---|---|
| multinom | 0,6 | 0,861 | 1,652 | 16 | 5 | D100 | 129,951 |
| bagging | 0,5 | 0,828 | 1,652 | 18 | 5 | D060 | 129,842 |
| randomForest | 0,5 | 0,866 | 1,435 | 15 | 3 | D060 | 129,410 |
| cforest | 0,5 | 0,847 | 1,565 | 16 | 4 | D090 | 128,823 |
| svm radial | 0,5 | 0,848 | 1,087 | 14 | 1 | D080 | 128,239 |
| ctree | 0,5 | 0,797 | 1,696 | 16 | 5 | D040 | 128,001 |
| rpart | 0,8 | 0,731 | 1,478 | 15 | 3 | D110 | 127,042 |
| svm linear | 0,5 | 0,850 | 1,130 | 12 | 3 | D110 | 126,991 |
| svm polynomial | 0,6 | 0,810 | 0,826 | 12 | 2 | D080 | 126,354 |
| IBk | 0,5 | 0,656 | 1,522 | 16 | 14 | D110 | 124,122 |

Tabela 4.5: Resultados após ajuste da classificação com *Feature Ranking*.

Através dos resultados obtidos, em relação aos ensaios anteriores, verificou-se uma melhoria nos valores de PV, ou seja, é possível obter uma maior capacidade de acerto na classificação. Também se verificou uma diminuição nos valores de falsos positivos, o que indica que o ajuste da classificação obtida utilizando menos variáveis é benéfica para esta modelação.

Na tabela 4.6 são apresentados, para cada algoritmo, os valores mínimos de PV produzidos durante a classificação. É possível verificar que apesar de alguns algoritmos apresentarem um valor inferior de *FP max*, também apresentam um valor inferior de *PE Sensitivity* em relação aos seus testes que produziram um PV mais alto.

| Algoritmo | % | AUC | PE Sens | PA Sens | FP max | FR | PV |
|---|---|---|---|---|---|---|---|
| ctree | 0,7 | 0,753 | 0,609 | 11 | 19 | D010 | 124,553 |
| cforest | 0,8 | 0,806 | 0,304 | 7 | 5 | D20 | 124,453 |
| svm radial | 0,8 | 0,743 | 0,348 | 7 | 6 | D020 | 124,368 |
| randomForest | 0,8 | 0,855 | 0,217 | 3 | 2 | D100 | 124,130 |
| bagging | 0,9 | 0,802 | 0,174 | 3 | 1 | DORG | 124,107 |
| svm linear | 0,6 | 0,727 | 0,304 | 6 | 6 | D010 | 123,910 |
| rpart | 0,7 | 0,618 | 0,609 | 10 | 23 | D020 | 123,771 |
| svm polynomial | 0,5 | 0,744 | 0,087 | 2 | 2 | D010 | 123,664 |
| multinom | - | 0,784 | 0,696 | 11 | 32 | D010 | 123,445 |
| IBk | - | 0,631 | 1,478 | 17 | 118 | D020 | 119,688 |
| svm sigmoid | - | 0,395 | 1,217 | 13 | 150 | D040 | 114,703 |

Tabela 4.6: Os piores resultados após ajuste da classificação com *Feature Ranking*.

#### 4.1.1.4 Avaliação global do ensaio

Foram efetuados 671 casos de teste sobre o conjunto de dados que estão descritos nos subcapítulos anteriores. Com base nestes casos, fez-se uma análise detalhada de forma a obter informações que pudessem ser úteis para a descoberta dos algoritmos que produzem melhores resultados sobre este conjunto de dados. Observando todos os resultados é possível retirar algumas conclusões sobre o comportamento de determinados algoritmos.

| Algoritmo | % | AUC | PE Sens | PA Sens | FP max | FR |
|---|---|---|---|---|---|---|
| svm radial | 0,5 | 0,848 | 1,087 | 14 | 1 | D080 |
| randomForest | - | 0,867 | 1,478 | 15 | 3 | D050 |
| rpart | 0,8 | 0,731 | 1,478 | 15 | 3 | D110 |
| svm polynomial | - | 0,831 | 1,261 | 15 | 3 | D100 |
| cforest | - | 0,863 | 1,522 | 16 | 4 | D050 |
| ctree | - | 0,797 | 1,696 | 16 | 5 | D040 |
| multinom | 0,6 | 0,861 | 1,652 | 16 | 5 | D100 |
| bagging | 0,5 | 0,828 | 1,652 | 18 | 5 | D060 |
| svm linear | - | 0,856 | 1,609 | 15 | 6 | DORG |

Tabela 4.7: Melhores resultados usando o conjunto de dados original.

Tendo em conta que os algoritmos escolhidos no final desta análise não podem conter nenhum paciente com mais de 10 falsos positivos, foram excluídos da análise todos os testes com *FP max* maior do que 10. A tabela 4.7 resume os melhores resultados obtidos nos ensaios anteriores, para cada um dos algoritmos.

Os algoritmos *svm sigmoid* e *IBk* apresentam valores de *PE sensitivity* elevados, mas por outro lado, apresentam uma grande quantidade de falsos positivos. Visto que em todos os casos de teste estes algoritmos obtiveram mais de 10 *FP max*, os algoritmos foram desconsiderados. O algoritmo *svm radial* obteve o menor valor de *FP max*, mas apresentou também o menor valor de *PE Sensitivity*.

### 4.1.2 Classificação utilizando o conjunto de dados equilibrado

O conjunto de dados original utilizado neste estudo apresenta um desequilíbrio entre casos positivos e casos negativos. Verificou-se a necessidade de analisar o desempenho dos algoritmos tendo por base um conjunto de treino mais equilibrado. Efetuou-se um ensaio igual ao do subcapítulo anterior, mas considerando um conjunto de dados equilibrado. Nas tabelas 4.8 e 4.9 estão presentes os melhores resultados para cada um dos algoritmos usando todas as variáveis do conjunto de dados.

| Algoritmo | TN | FP | FN | TP |
|---|---|---|---|---|
| svm radial | 1231 | 4 | 131 | 25 |
| randomForest | 1228 | 7 | 121 | 35 |
| cforest | 1228 | 7 | 126 | 30 |
| svm polynomial | 1227 | 8 | 125 | 31 |
| bagging | 1210 | 25 | 119 | 37 |
| svm sigmoid | 1207 | 28 | 121 | 35 |
| svm linear | 1198 | 37 | 82 | 74 |
| multinom | 1188 | 47 | 75 | 81 |
| rpart | 1188 | 47 | 137 | 19 |
| ctree | 1171 | 64 | 103 | 53 |
| IBk | 1157 | 78 | 97 | 59 |

Tabela 4.8: Resultados obtidos com o conjunto de dados equilibrado.

| Algoritmo | AUC | PE Sens | PA Sens | FP max | PV |
|---|---|---|---|---|---|
| svm linear | 0,857 | 1,826 | 15 | 8 | 129,837 |
| multinom | 0,861 | 1,826 | 15 | 8 | 129,537 |
| randomForest | 0,856 | 0,957 | 14 | 2 | 127,676 |
| svm polynomial | 0,816 | 1,087 | 15 | 2 | 127,431 |
| svm radial | 0,866 | 0,870 | 14 | 1 | 126,869 |
| cforest | 0,831 | 0,826 | 12 | 2 | 126,858 |
| bagging | 0,767 | 1,043 | 14 | 7 | 126,168 |
| svm sigmoid | 0,733 | 1,043 | 15 | 4 | 125,755 |
| ctree | 0,755 | 1,217 | 16 | 18 | 124,304 |
| IBk | 0,662 | 1,652 | 17 | 15 | 124,260 |
| rpart | 0,694 | 0,652 | 9 | 15 | 121,367 |

Tabela 4.9: Resultados obtidos com o conjunto de dados equilibrado.

Utilizando este conjunto de treino, os algoritmos apresentaram uma diminuição de falsos positivos, sem que a capacidade de classificação sofresse uma alteração significativa em comparação aos testes realizados sobre o conjunto de dados original (subcapítulo 4.1.1).

#### 4.1.2.1 Avaliação global do ensaio

Na avaliação global do ensaio são analisados os resultados dos 671 testes feitos sobre os conjuntos de dados. Excluíram-se todos os testes em que o número de *FP max* foi superior a 10, pois estes não cumprem os objetivos do estudo.

Nas tabelas 4.10 e 4.11 é possível ver os melhores resultados para cada um dos algoritmos tendo em conta a regra descrita anteriormente.

Verificou-se que os algoritmos continuam, face ao ensaio anterior, a apresentar

| Algoritmo | TN | FP | FN | TP |
|---|---|---|---|---|
| svm radial | 1231 | 4 | 126 | 30 |
| randomForest | 1229 | 6 | 117 | 39 |
| svm polynomial | 1225 | 10 | 117 | 39 |
| cforest | 1223 | 12 | 113 | 43 |
| ctree | 1221 | 14 | 130 | 26 |
| bagging | 1209 | 26 | 104 | 52 |
| svm sigmoid | 1207 | 28 | 121 | 35 |
| svm linear | 1206 | 29 | 84 | 72 |
| rpart | 1204 | 31 | 123 | 33 |
| multinom | 1202 | 33 | 78 | 78 |
| IBk | 1151 | 84 | 112 | 44 |

Tabela 4.10: Resultados globais obtidos com o conjunto de dados equilibrado.

| Algoritmo | % | AUC | PE Sens | PA Sens | FP max | FR |
|---|---|---|---|---|---|---|
| randomForest | - | 0,855 | 1,130 | 14 | 1 | D070 |
| svm radial | - | 0,850 | 0,826 | 13 | 1 | D050 |
| ctree | - | 0,802 | 0,739 | 12 | 2 | D060 |
| cforest | - | 0,862 | 1,130 | 15 | 3 | D040 |
| svm polynomial | - | 0,808 | 1,130 | 14 | 3 | D040 |
| rpart | - | 0,681 | 1,087 | 14 | 4 | D060 |
| svm sigmoid | - | 0,733 | 1,043 | 15 | 4 | DBAL |
| multinom | 0,6 | 0,861 | 1,783 | 15 | 6 | DBAL |
| bagging | - | 0,844 | 1,348 | 14 | 6 | D050 |
| svm linear | - | 0,859 | 1,652 | 14 | 7 | D110 |
| IBk | 0,5 | 0,642 | 1,087 | 15 | 10 | D020 |

Tabela 4.11: Resultados globais obtidos com o conjunto de dados equilibrado.

valores semelhantes de *FP max* e *PE Sensitivity*. Os algoritmos com menor valor de *FP max* continuam a ser aqueles que apresentam maior valor de *PE Sensitivity*. Assim, verificou-se que a utilização deste conjunto de dados, não proporcionou melhores resultados que os obtidos anteriormente.

Este conjunto de treino equilibrado é construído automaticamente no início de cada ensaio. Assim, sempre que um novo ensaio é executado, é criado um novo conjunto de dados que será diferente do anterior.

Tendo em conta que se construiu um conjunto com 20 casos, os 10 casos positivos são escolhidos aleatoriamente entre os 38 existentes, logo, não existem garantias que outra combinação de casos não possa produzir um conjunto de treino em que os algoritmos possam produzir resultados completamente diferentes. Com os casos negativos acontece uma situação semelhante, pois como existem menos casos

do que aqueles que são necessários para construir o novo conjunto de dados, os casos selecionados podem repetir-se várias vezes.

Sabendo que o objetivo passa por encontrar o algoritmo que sobre um determinado conjunto de treino produz uma classificação o mais fiável possível, verificou-se que para usar esta técnica de criação do conjunto de treino apenas poderia ser vantajosa se o novo conjunto de dados quando produzisse bons resultados fosse guardado. Se o novo conjunto de dados não for guardado, nada garante que o próximo conjunto de dados gerado possa dar resultados semelhantes. Logo, nesta situação não seria possível comparar resultados.

Sabendo que o próximo conjunto de dados criado nestas condições pode não devolver resultados tão bons como os apresentados, excluiu-se dos testes seguintes esta forma de criação de conjunto de treino.

### 4.1.3 Reparametrização dos algoritmos SVM

O algoritmo SVM apresenta uma grande diversidade de resultados apenas efetuando a variação do tipo *kernel*. Tendo em conta que num conjunto de testes está sempre a ser utilizado o mesmo conjunto de treino, verificou-se que dependendo do tipo de *kernel* este algoritmo de *Data Mining* pode apresentar bons e maus resultados.

Para cada tipo de *kernel* efetuou-se a otimização do algoritmo usando a função *tune.svm* do pacote *e1071* [1] do R. Esta função determina os melhores valores a utilizar nos parâmetros do algoritmo tendo em conta os dados, cuja otimização é feita utilizando o *Grid Search*. Tendo em conta o resultado da otimização do algoritmo, foram considerados os valores obtidos para os parâmetros *gamma* e *cost*.

Excluiu-se o *svm linear* pois o seu tempo de execução para este conjunto de treino é muito extenso. A otimização do algoritmo demora cerca de 2 dias e a criação do modelo cerca de 5 horas. Tendo em conta o tempo de execução e face aos resultados obtidos, não se justifica a continuação do uso deste algoritmo sabendo que pode tornar os testes demasiado demorados. Em termos de resultados, este algoritmo apresenta valores semelhantes a outros.

Efetuaram-se várias execuções da otimização do algoritmo SVM com os 3 tipos de *kernel* escolhidos e fazendo variação dos parâmetros *gamma* e *cost*. A execução

[1] http://cran.r-project.org/web/packages/e1071/index.html

que definiu os valores escolhidos está apresentada no código A.7 apresentado nos anexos.

Depois de efetuada a otimização do algoritmo, repetiu-se todo o conjunto de testes dos subcapítulos anteriores, parametrizando os algoritmos da seguinte forma:

- ***svm polynomial*** tipo *C-classification* com *gamma* = 0.1 e *cost* = 0.01
- ***svm radial*** tipo *C-classification* com *gamma* = 0.001 e *cost* = 100
- ***svm sigmoid*** tipo *C-classification* com *gamma* = $10e^{-5}$ e *cost* = $10e^{+5}$

O algoritmo *svm radial* é aquele que melhores resultados apresenta comparativamente aos outros. O valor de AUC, *PE sensitivity*, *PA sensitivity* e PV são sempre elevados. Apresenta também um número de falsos positivos muito reduzido. Na Figura 4.3 está apresentada a curva ROC usando todas as variáveis do conjunto de dados para o algoritmo SVM com os 3 tipos de *kernel* escolhidos.

O algoritmo *svm sigmoid* apresenta valores muito baixos de classificações corretas e valores muito altos de falsos negativos. Tal como se pode verificar neste teste e nos anteriores, este algoritmo não é adequado para modelar este conjunto de dados.

Na tabela 4.12 estão apresentados os melhores resultados de cada um dos algoritmos. Os testes apresentados são aqueles que obtiveram um maior valor no fator de ordenação PV.

| **Algoritmo** | **%** | **AUC** | **PE Sens** | **PA Sens** | **FP max** | **FR** | **PV** |
|---|---|---|---|---|---|---|---|
| svm radial | 0,5 | 0,854 | 1,652 | 17 | 5 | D090 | 130,199 |
| svm polynomial | - | 0,830 | 1,348 | 14 | 3 | D060 | 128,064 |
| svm sigmoid | 0,5 | 0,708 | 0,217 | 4 | 2 | DORG | 123,920 |

Tabela 4.12: Melhores resultados do algoritmo SVM após otimização.

O algoritmo *svm radial* apresenta sempre bons resultados em todos os testes efetuados, nomeadamente, obteve sempre um valor de *FP max* muito baixo. Por esta razão procuraram-se melhores resultados utilizando este algoritmo. Optou-se por alterar o tipo de classificação deste algoritmo e verificar qual o impacto no seu desempenho.

O parâmetro *v* do algoritmo *svm* permite ter uma maior controlo sobre o número de vetores de suporte. Quanto maior a quantidade de vetores de suporte, maior será a margem entre eles, o que pode originar um maior número de classificações

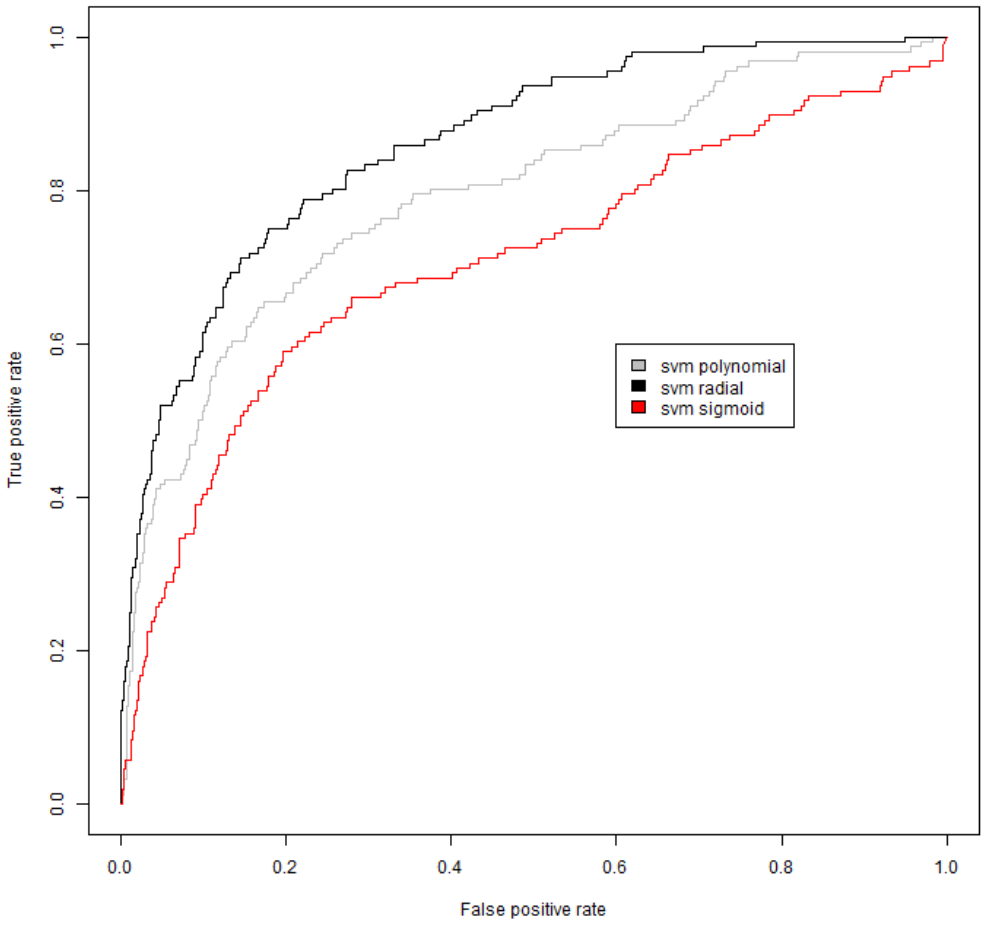



Figura 4.3: Curvas ROC usando todas as variáveis sobre o algoritmo SVM.

incorretas. Este parâmetro permite controlar a capacidade de classificar correta ou incorretamente [27]. Otimizou-se o algoritmo para procurar os valores mais adequados de *gamma* e *cost*.

Parametrizou-se o algoritmo SVM da seguinte forma:

- ***svm radial*** tipo *v-classification* com *gamma* = 0.01, *cost* = 10 e $\nu = 0.1$
- ***svm radial*** tipo *v-classification* com *gamma* = 0.0001, *cost* = 10 e $\nu = 0.2$

Efetuou-se novo ensaio sobre o conjunto de dados utilizando este algoritmo re-parametrizado e os melhores resultados, tendo em conta o maior valor do fator de ordenação PV, estão apresentados na tabela 4.13.

Neste conjunto de testes é possível encontrar um dos maiores valores de PV conseguido sobre este conjunto de dados. Esta informação indica que, tal como se

tem verificado ao longo dos testes, este algoritmo é um dos que produz melhores resultados nesta classificação.

| Algoritmo | *v* | % | AUC | PE Sens | PA Sens | FP max | FR | PV |
|---|---|---|---|---|---|---|---|---|
| svm radial | 0,2 | - | 0,843 | 1,870 | 17 | 3 | D090 | 131,129 |
| svm radial | 0,1 | - | 0,836 | 1,826 | 16 | 14 | DORG | 128,197 |

Tabela 4.13: Melhores resultados do algoritmo *svm radial* com *v-classification*.

#### 4.1.3.1 Avaliação global do ensaio

Verificou-se que a reparametrização do algoritmo *svm radial* permitiu obter melhores resultados face aos apresentados anteriormente. A otimização do algoritmo permitiu apurar quais os melhores valores de *gamma* e *cost* tendo em conta os restantes parâmetros. Visto não serem permitidos algoritmos com *FP max* maior que 10, foram excluídos todos os resultados que apresentavam esses valores. Na tabela 4.14 estão apresentados os melhores valores de PV.

| Algoritmo | *v* | % | AUC | PE Sens | PA Sens | FP max | FR |
|---|---|---|---|---|---|---|---|
| svm radial | 0,2 | - | 0,843 | 1,870 | 17 | 3 | D090 |
| svm radial | - | 0,5 | 0,854 | 1,652 | 17 | 5 | D090 |
| svm radial | 0,1 | 0,5 | 0,832 | 1,522 | 15 | 6 | D110 |
| svm polynomial | - | - | 0,830 | 1,348 | 14 | 3 | D060 |
| svm sigmoid | - | 0,5 | 0,708 | 0,217 | 4 | 2 | DORG |

Tabela 4.14: Melhores resultados do algoritmo SVM.

O algoritmo SVM com *kernel* do tipo *radial* é aquele que produz melhores resultados perante este conjunto de dados, independentemente do tipo de parametrização, comparativamente aos outros tipos do mesmo algoritmo.

## 4.2 Avaliação

Testaram-se todos os modelos criados utilizando o conjunto de teste fornecido. A avaliação final foi realizada sobre os resultados de todas as execuções de algoritmos efetuadas nos subcapítulos anteriores. Ao todo foram considerados 976 casos de teste sobre o conjunto de dados, ao longo de 7 ensaios.

Tendo em conta que este estudo resulta de um desafio proposto no *KDD Cup 2006* foram seguidas as regras deste concurso para a avaliação dos resultados. A listagem seguinte apresenta o método de avaliação seguido neste concurso:

- Maior valor de *PE sensitivity* com limite de *FP max* a 2, 4 e 10.
- Maior valor de *PA sensitivity* com limite de *FP max* a 2, 4 e 10.

Tendo em conta os objetivos, analisaram-se os resultados de forma a que fosse possível escolher a melhor modelação para cada um dos limites de *FP max*. Para cada um dos limites escolheram-se os 5 melhores casos.

#### 4.2.0.2 *FP max* limitado a 2

Nas tabelas 4.15 e 4.16 apresentam-se os melhores resultados com *PE sensitivity* e *PA sensitivity* mais elevados, tendo como restrição um limite de 2 *FP max*.

| **Algoritmo** | *v* | **%** | **PE Sens** | **PA Sens** | **FP max** | **FR** |
|---|---|---|---|---|---|---|
| svm radial | 0,2 | 0,6 | 1,348 | 16 | 2 | DORG |
| svm radial | 0,2 | 0,5 | 1,261 | 14 | 2 | D090 |
| bagging | - | 0,6 | 1,217 | 15 | 2 | D030 |
| cforest | - | 0,6 | 1,130 | 16 | 2 | DORG |
| svm radial | - | 0,7 | 1,130 | 14 | 2 | D090 |

Tabela 4.15: Maior *PE sensitivity* com limite de 2 *FP max*.

| **Algoritmo** | *v* | **%** | **PE Sens** | **PA Sens** | **FP max** | **FR** |
|---|---|---|---|---|---|---|
| svm radial | 0,2 | 0,6 | 1,348 | 16 | 2 | DORG |
| cforest | - | 0,6 | 1,130 | 16 | 2 | DORG |
| bagging | - | 0,6 | 1,217 | 15 | 2 | D030 |
| svm radial | - | 0,5 | 1,043 | 15 | 2 | D040 |
| svm radial | - | 0,6 | 1,043 | 15 | 2 | D040 |

Tabela 4.16: Maior *PA sensitivity* com limite de 2 *FP max*.

Tendo em conta os resultados apresentados verifica-se que o modelo que apresenta os melhores valores de *PE sensitivity* e *PA sensitivity* é o mesmo.

Para obter estes resultados usou-se o algoritmo SVM com o conjunto de dados original como conjunto de treino. Para a modelação foram consideradas todas as variáveis e o algoritmo foi executado com os seguintes parâmetros:

- *kernel radial*
- *type v-classification*
- *gamma* = 0.001

- *cost* = 100
- $\nu = 0.2$

Após a classificação, feita pelo algoritmo, o resultado desta foi ajustado utilizando uma probabilidade mínima de 0.6.

#### 4.2.0.3 *FP max* limitado a 4

Nas tabelas 4.17 e 4.18 apresentam-se os melhores resultados com *PE sensitivity* e *PA sensitivity* mais elevados, tendo como restrição um limite de 4 *FP max*.

| **Algoritmo** | $\nu$ | **%** | **PE Sens** | **PA Sens** | **FP max** | **FR** |
|---|---|---|---|---|---|---|
| svm radial | 0,2 | - | 1,870 | 17 | 3 | D090 |
| bagging | - | 0,5 | 1,609 | 18 | 4 | D010 |
| bagging | - | - | 1,565 | 15 | 3 | D110 |
| bagging | - | 0,5 | 1,565 | 17 | 4 | D110 |
| bagging | - | - | 1,565 | 16 | 4 | D030 |

Tabela 4.17: Maior *PE sensitivity* com limite de 4 *FP max*.

| **Algoritmo** | $\nu$ | **%** | **PE Sens** | **PA Sens** | **FP max** | **FR** |
|---|---|---|---|---|---|---|
| bagging | - | 0,5 | 1,609 | 18 | 4 | D010 |
| svm radial | 0,2 | - | 1,870 | 17 | 3 | D090 |
| bagging | - | 0,5 | 1,565 | 17 | 4 | D110 |
| svm radial | 0,2 | 0,6 | 1,348 | 16 | 2 | DORG |
| cforest | - | 0,6 | 1,130 | 16 | 2 | DORG |

Tabela 4.18: Maior *PA sensitivity* com limite de 4 *FP max*.

Nas duas tabelas os dois primeiros modelos são os mesmos. A única diferença está na posição que ocupam na respetiva tabela. Sendo assim, a escolha do melhor algoritmo é feita tendo em conta o valor de *FP max*. Sabendo que a perfeição do algoritmo seria atingida se não existissem falsos positivos, foi considerado o modelo com menor valor de *FP max*.

O modelo que melhor responde aos dois objetivos usa o algoritmo SVM com o conjunto de dados original como conjunto de treino. Para a modelação apenas foram consideradas as 90 variáveis de maior peso (escolhidas através de *Feature Ranking*) e o algoritmo foi executado com os seguintes parâmetros:

- *kernel radial*

- *type ν-classification*
- *gamma* = 0.001
- *cost* = 100
- $\nu = 0.2$

#### 4.2.0.4 *FP max* limitado a 10

Nas tabelas 4.19 e 4.20 apresentam-se-se os melhores resultados conseguidos com *PE sensitivity* e *PA sensitivity* mais elevados, tendo como restrição um limite de 10 *FP max*.

| Algoritmo | $\nu$ | % | PE Sens | PA Sens | FP max | FR |
|---|---|---|---|---|---|---|
| svm radial | 0,2 | - | 1,870 | 17 | 3 | D090 |
| svm radial | 0,2 | - | 1,783 | 17 | 6 | D100 |
| svm radial | - | - | 1,783 | 17 | 8 | DORG |
| ctree | - | 0,5 | 1,739 | 16 | 6 | DORG |
| ctree | - | - | 1,739 | 16 | 6 | DORG |

Tabela 4.19: Maior *PE sensitivity* com limite de 10 *FP max*.

| Algoritmo | $\nu$ | % | PE Sens | PA Sens | FP max | FR |
|---|---|---|---|---|---|---|
| bagging | - | 0,5 | 1,609 | 18 | 4 | D010 |
| bagging | - | 0,5 | 1,652 | 18 | 5 | D060 |
| randomForest | - | - | 1,478 | 18 | 5 | D010 |
| svm radial | 0,1 | - | 1,652 | 18 | 9 | D010 |
| svm radial | 0,2 | - | 1,870 | 17 | 3 | D090 |

Tabela 4.20: Maior *PA sensitivity* com limite de 10 *FP max*.

Tendo em conta os resultados, verificou-se que apenas um dos modelos é coincidente entre os melhores valores de *PE sensitivity* e *PA sensitivity*. Este modelo utiliza o algoritmo SVM com o conjunto de dados original como conjunto de treino. Para a modelação apenas foram consideradas as 90 variáveis de maior peso (escolhidas através de *Feature Ranking*) e o algoritmo foi executado com os seguintes parâmetros:

- *kernel radial*
- *type ν-classification*

- *gamma* = 0.001
- *cost* = 100
- $\nu = 0.2$

A principal vantagem deste modelo é a de poder ser utilizado para os dois objetivos, apresentando o menor número de *FP max*.

#### 4.2.0.5 Ajuste da classificação através de votação

O ajuste da classificação através de votação, consiste na utilização simultânea de alguns dos modelos criados. Depois da classificação ser feita por todos os modelos escolhidos, os resultados são analisados em conjunto. Esta técnica é conhecida por *Ensemble* [15].

Utilizaram-se 3 dos modelos que melhores resultados apresentaram para efetuar a modelação. Para que uma classificação num candidato seja considerada positiva, é necessário que pelo menos 2 dos modelos tenham classificado o candidato como positivo.

Verificou-se, pelos testes efetuados, não existirem melhorias na classificação utilizando esta técnica. Sendo assim, os resultados não foram considerados para a avaliação final da modelação.

#### 4.2.0.6 Comparação de resultados com o *KDD Cup 2006*

Os resultados do concurso *KDD Cup 2006* estão publicados[2], logo, é possível efetuar comparações entre os resultados obtidos pelos vencedores e os resultados obtidos neste estudo.

Nas tabelas 4.21 e 4.22 apresentam-se os resultados para os primeiros três classificados, para a determinação de maior *PE sensitivity* e maior *PA sensitivity*, respetivamente. A tabela 4.23 apresenta os resultados obtidos neste estudo, tendo em conta os maiores de valores de *PE sensitivity* e *PA sensitivity*.

Verifica-se que neste estudo, foi possível encontrar um modelo que produzisse melhores resultados que os apresentados pelos vencedores.

Comprovou-se, ao longo dos testes, que os modelos baseados no algoritmo SVM com o tipo *kernel radial*, tendo sido parametrizado adequadamente, produziu excelentes resultados perante este conjunto de dados.

[2]http://www.sigkdd.org/kdd-cup-2006-pulmonary-embolisms-detection-image-data

| Rank | PE sensitivity 2 FP max | PE sensitivity 4 FP max | PE sensitivity 10 FP max | Média |
|---|---|---|---|---|
| 1 | 1.17 | 1.31 | 1.58 | 1.35 |
| 2 | 1.00 | 1.25 | 1.58 | 1.28 |
| 3 | 0.93 | 1.36 | 1.51 | 1.27 |

Tabela 4.21: Resultado do concurso *KDD Cup 2006* para *PE sensitivity*.

| Rank | PE sensitivity 2 FP max | PE sensitivity 4 FP max | PE sensitivity 10 FP max | Média |
|---|---|---|---|---|
| 1 | 11.50 | 14.34 | 14.90 | 13.58 |
| 2 | 11.56 | 13.74 | 15.39 | 13.56 |
| 3 | 11.18 | 13.74 | 15.39 | 13.44 |

Tabela 4.22: Resultado do concurso *KDD Cup 2006* para *PA sensitivity*.

| | 2 FP max | 4 FP max | 10 FP max | Média |
|---|---|---|---|---|
| PE sens | 1,348 | 1,870 | 1,870 | 1,696 |
| PA sens | 16 | 17 | 17 | 16,667 |

Tabela 4.23: Resultados obtidos a partir deste estudo.

# 5

# Conclusão

Este estudo focou-se na criação de um modelo, utilizando técnicas de *Data Mining*, que permitisse classificar, a partir de imagens médicas, a existência de embolia pulmonar.

Inicialmente foram estudadas as técnicas que poderiam ser utilizadas de forma a cumprir o objetivo, e implementou-se uma sequência de testes que permitisse avaliar e comparar resultados. Esta sequência de teste foi usada em cada um dos ensaios realizados sobre cada um dos conjuntos de dados. O primeiro ensaio foi efetuado sobre o conjunto de dados original e os resultados deste passaram a ser tomados como referência. Foram construídos novos conjuntos de dados com base no original e foram efetuados ensaios sobre estes.

Tendo em conta o facto de o conjunto de dados original não se encontrar equilibrado em termos de casos positivos e negativos, também se optou por criar um conjunto de dados equilibrado para que pudesse ser testada a classificação sobre este cenário. Também nesta situação, não se encontraram melhorias significativas perante a classificação feita com o conjunto de dados original. Os resultados foram muito semelhantes, logo, não se verificou a necessidade de continuar a realizar ensaios com este conjunto de dados.

Por outro lado, verificou-se que a aplicação de algoritmos de *Feature Ranking* para definir as variáveis com maior importância no conjunto de dados, poderia ser determinante para a criação de novos conjuntos. Assim, foram criados conjuntos de dados de várias dimensões utilizando as variáveis com maior importância

para a determinação da variável dependente.

Uma das técnicas que obteve melhores resultados foi o ajuste do ponto de operação após a classificação. Verificou-se que as técnicas de pós-processamento podem ser vantajosas visto que é possível manipular o resultado da classificação, de forma a se conseguir obter melhores resultados.

Verificou-se que o algoritmo SVM com *kernel* do tipo *radial* foi aquele que apresentou melhores resultados tendo em conta os objetivos. Este algoritmo apresentou um *PE sensitivity* e um *PA sensitivity* elevados e ao mesmo tempo um baixo valor de falsos positivos por paciente. Os bons resultados deste algoritmo tornaram-se evidentes quando este foi reparametrizado e afinado. A utilização de uma métrica própria, construída a pensar neste problema, facilitou a ordenação automática dos resultados, permitindo verificar facilmente quais os testes que melhoram o desempenho dos modelos. Esta métrica foi importante na tomada de decisões sobre cada um dos algoritmos e permitiu superar os resultados existentes para o mesmo conjunto de dados.

Verificou-se que o caminho escolhido para encontrar o melhor modelo possível, pode tornar-se num labirinto em que devem ser escolhidas as portas corretas para se poder avançar. Quando uma das técnicas utilizadas não apresenta bons resultados, deve ser descartada e aplicada uma estratégia diferente. A única forma de saber se uma determinada técnica é melhor do que outra é efetuando testes e comparando resultados. A descoberta de conhecimento em bases de dados pode tornar-se num processo exaustivo na busca de melhores resultados.

## 5.1 Trabalho futuro

Para que fosse possível implementar sistemas CAD perfeitos, seria necessário conseguir obter classificações totalmente corretas sobre os conjuntos de dados.

Com base neste estudo, o trabalho futuro passa por continuar a aplicar técnicas de *Data Mining* que permitam melhorar os resultados obtidos. Por exemplo, construir novos conjuntos de dados, aplicar e reparametrizar outros algoritmos de *Data Mining* ou efetuar novas técnicas de pós-processamento.

Tendo por base os resultados deste estudo, uma das técnicas de pós-processamento que poderá ser mais interessante é o *Ensemble*. Neste estudo foram utilizados os 3 modelos com melhores resultados para construir um *Ensemble*, que não devolveram a excelentes resultados. No futuro podem ser utilizados outros modelos, que não sendo os melhores, podem levar a atingir melhores resultados.

# Bibliografia


[1] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] Rui Baptista, Graça Castro, António Marinho da Silva, and Luís A. Providência. Pulmonary dissection during diagnostic pulmonary angiography. *Revista Portuguesa de Cardiologia*, May 2012.

[3] Ricardo Barandela, Rosa M Valdovinos, J Salvador Sánchez, and Francesc J Ferri. The imbalanced training sample problem: Under or over sampling? In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 806–814. Springer, 2004.

[4] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In *Database Theory—ICDT'99*, pages 217–235. Springer, 1999.

[5] Andrew P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145 – 1159, 1997. ISSN 0031-3203.

[6] Leo Breiman. *Classification and regression trees*. CRC press, 1993.

[7] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[8] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[9] J.L. Carson, M.A. Kelley, A. Duff, J.G. Weg, W.J. Fulkerson, H.I. Palevsky, J.S. Schwartz, B.T. Thompson, J. Popovich Jr, T.E. Hobbins, et al. The clinical course of pulmonary embolism. *New England Journal of Medicine*, 326(19): 1240–1245, 1992.

[10] Pete Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinartz, Colin Shearer, and Rudiger Wirth. CRISP-DM 1.0 step-by-step data mining guide. 2000.

[11] Nitesh V Chawla. C4. 5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In *Proceedings of the ICML*, volume 3, 2003.

[12] Nitesh V Chawla. Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, pages 875–886. Springer, 2010.

[13] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995. ISSN 0885-6125.

[14] J.E. Dalen. Pulmonary embolism: what have we learned since Virchow? treatment and prevention. *CHEST Journal*, 122(5):1801–1817, 2002.

[15] Thomas G Dietterich. Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer, 2000.

[16] Tom Fawcett. An introduction to ROC analysis. *Pattern recognition letters*, 27 (8):861–874, 2006.

[17] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39 (11):27–34, 1996.

[18] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37, 1996.

[19] Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, et al. Knowledge discovery and data mining: Towards a unifying framework. *Knowledge Discovery and Data Mining*, pages 82–88, 1996.

[20] Arthur C. Guyton and John E. Hall. *Textbook of Medical Physiology*.

[21] Tin Kam Ho. The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):832–844, 1998. ISSN 0162-8828.

[22] Torsten Hothorn, Kurt Hornik, and Achim Zeileis. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical Statistics*, 15(3), 2006.

[23] C. Kearon. Natural history of venous thromboembolism. *Circulation*, 107(23 suppl 1):I–22, 2003.

[24] L.A. Kurgan and P. Musilek. A survey of knowledge discovery and data mining process models. *Knowledge Engineering Review*, 21(1):1–24, 2006.

[25] Nada Lavrač. Selected techniques for data mining in medicine. *Artificial intelligence in medicine*, 16(1):3–23, 1999.

[26] Sylvia S Mader and Patrick L Galliart. *Understanding human anatomy and physiology*. McGraw-Hill Higher Education, 1997.

[27] David Meyer and Technische Universität Wien. Support vector machines. the interface to libsvm in package e1071. online-documentation of the package e1071 for r, 2001.

[28] David Meyer, Friedrich Leisch, and Kurt Hornik. Benchmarking Support Vector Machines. 2002.

[29] Lung National Heart and Blood Institute (NHLBI. Pulmonary embolism @ONLINE, July 2011. URL http://www.nhlbi.nih.gov/health/.

[30] R.A. Novelline and L.F. Squire. *Squire's fundamentals of radiology*. Belknap Press, 2004.

[31] Daniel R Ouellette, Annie Harrington, and Nader Kamangar. Pulmonary embolism @ONLINE, February 2013. URL http://emedicine.medscape.com/article/300901-overview.

[32] R Parimala and R Nallaswamy. A study of spam e-mail classification using feature selection package. *Global Journal of Computer Science and Technology*, 11(7), 2011.

[33] E. Pichon, C.L. Novak, A.P. Kiraly, and D.P. Naidich. A novel method for pulmonary emboli visualization from high-resolution ct images. 2004.

[34] Brian D Ripley. *Modern applied statistics with S*. Springer, 2002.

[35] Piotr Romanski and Maintainer Lars Kotthoff. Package 'fselector'. 2013.

[36] Colin Shearer. The CRISP-DM model: the new blueprint for data mining. *Journal of Data Warehousing*, 5(4):13–22, 2000.

[37] Medical Solutions Siemens AG. Computed tomography - its history and technology. www.SiemensMedical.com.

[38] Ingrid C Sluimer, Paul F van Waes, Max A Viergever, and Bram van Ginneken. Computer-aided diagnosis in high resolution CT of the lungs. *Medical Physics*, 30:3081, 2003.

[39] P.D. Stein and F. Matta. Acute pulmonary embolism. *Current Problems in Cardiology*, 35(7):314–376, 2010.

[40] Domonkos Tikk, Zsolt T Kardkovács, and Ferenc P Szidarovszky. Voting with a parameterized veto strategy: Solving the KDD cup 2006 problem by means of a classifier committee. *ACM SIGKDD Explorations Newsletter*, 8(2): 53–62, 2006.

[41] A. Torbicki, A. Perrier, S. Konstantinides, G. Agnelli, N. Galiè, P. Pruszczyk, F. Bengel, A.J.B. Brady, D. Ferreira, U. Janssens, et al. Guidelines on the diagnosis and management of acute pulmonary embolism the task force for the diagnosis and management of acute pulmonary embolism of the european society of cardiology (esc). *European heart journal*, 29(18):2276–2315, 2008.

[42] Gary M Weiss and Foster J Provost. Learning when training data are costly: the effect of class distribution on tree induction. *J. Artif. Intell. Res.(JAIR)*, 19: 315–354, 2003.

[43] Ian H Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[44] R. Wittenberg, J.F. Peters, J.J. Sonnemans, M. Prokop, and C.M. Schaefer-Prokop. Computer-assisted detection of pulmonary embolism: evaluation of pulmonary CT angiograms performed in an on-call setting. *European radiology*, 20(4):801–806, 2010.

# Anexos

```r
predict_sensitivity_t1 <- function(pred){

# Create dataset (ID, predict, true)
  ss <- as.data.frame(cbind(testLabel$V1, as.numeric(as.character(pred)), testLabel$V2))

# Select all positive candidates, one for PE
  ss <- unique(subset(ss[,1:3], V2 != 0 & V3 != 0))

# Average of PEs by patient
  return(length(ss$V1) / num_pacientes)
}
```

Listagem A.1: Função para determinar *PE sensitivity*.

```r
predict_sensitivity_t2 <- function(pred){

# Create dataset (ID, predict, true)
  ss <- as.data.frame(cbind(testLabel$V1, as.numeric(as.character(pred)), testLabel$V2))

# Select all positive candidates
  ss <- subset(ss[,1:3], V2 != 0 & V3 != 0)

# Number of patients with at least one PE
  return(length(unique(ss$V1)))
}
```

Listagem A.2: Função para determinar *PA sensitivity*.

```
feature_ranking <- function(d, n){

# Determine the most important variables
  weights <- chi.squared(V2~., d)
  ss <- cutoff.k(weights, n)
  r <- ss
  weights <- information.gain(V2~., d)
  ss <- cutoff.k(weights, n)
  r <-append(r, ss)
  weights <- gain.ratio(V2~., d)
  ss <- cutoff.k(weights, n)
  r <-append(r, ss)
  weights <- symmetrical.uncertainty(V2~., d)
  ss <- cutoff.k(weights, n)
  r <-append(r, ss)
  weights <- oneR(V2~., d)
  ss <- cutoff.k(weights, n)
  r <-append(r, ss)
  weights <- random.forest.importance(V2~., d, importance.type = 1)
  ss <- cutoff.k(weights, n)
  r <-append(r, ss)

  r <- as.data.frame(r)
  r <- count(r, "r")
  colnames(r)<-c("V","N")
  r <- r[with(r, order(N, decreasing=T)),]

# Select N most important variables and formula construction
  fs <- "V2 ~"
  for(i in 1:n)
    if(i == n)
      fs <- paste(fs, r$V[i])
    else
      fs <- paste(fs, r$V[i], " +")

  return(fs)
}
```

Listagem A.3: Função para fazer *Feature Ranking* sobre o conjunto de dados.

```r
equi <- function(d, n){
# d -> data set
# n -> number of records required for each type

# Determine the number of positive and negative cases
 num_pacientes <- as.numeric(length(unique(d$V1)))

 ppos <- unique(subset(d, V2>0)$V1)
 npos <- as.numeric(length(ppos))

 pneg <- setdiff(unique(d$V1), ppos)
 nneg <- as.numeric(length(pneg))

# Complete subset of negative
 if (n > nneg){
# oversampling
  x <- n-nneg
  neg <- pneg[sample(length(pneg), x, replace = TRUE)]
  neg <- append(neg, pneg)
 }else{
# undersampling
  neg <- pneg[sample(length(pneg), n, replace = FALSE)]
 }

# Complete subset of positive
 if (n > npos){
# oversampling
  x <- n-npos
  pos <- ppos[sample(length(ppos), x, replace = TRUE)]
  pos <- append(pos, ppos)
 }else{
# undersampling
  pos <- ppos[sample(length(ppos), n, replace = FALSE)]
 }

# join the negatives with positives
 casos <- append(neg, pos)
# shuffle
 casos <- sample(casos)

# Put in data set all candidates of selected patients
 r <- rbind()

 for(i in casos)
  r <- rbind(r, subset(d, V1==i))

 return(as.data.frame(r))
}
```

Listagem A.4: Função para fazer o equilíbrio do conjunto de dados.

```
predict_fp_max <- function(pred){

# Create dataset (ID, predict, true)
 ss <- as.data.frame(cbind(testLabel$V1, as.numeric(as.character(pred)), testLabel$V2))

# select false positive
 ss_fp <- subset(ss[,1:3], V2 != 0 & V3 == 0)

# Count PEs by patient
 ss_fp <- count(ss_fp, c("V1"))

# Max of FPs by patient
 return(max(ss_fp$freq))
}
```

Listagem A.5: Função para determinar *FP max*.

```
ajust_predict <- function(pred, prob, percent){

 pred <- as.numeric(as.character(pred))

 for(i in 1:length(pred)){
  if(pred[i] == 1 && prob[i] > 1 - percent){
   pred[i] <- 0
  }
 }
 return(pred[])
}
```

Listagem A.6: Função para ajustar a classificação com base no ponto de operação.

```
tuned2 <- tune.svm(V2~., kernel = "polynomial", data = data, gamma = 10^(-3:-1),
    cost = 10^(-2:1))

tuned3 <- tune.svm(V2~., kernel = "radial", data = data, gamma = 10^(-4:-2),
    cost = 10^(1:3))

tuned4 <- tune.svm(V2~., kernel = "sigmoid", data = data, gamma = 10^(-7:-5),
    cost = 10^(5:7))
```

Listagem A.7: Otimização do algoritmo SVM.