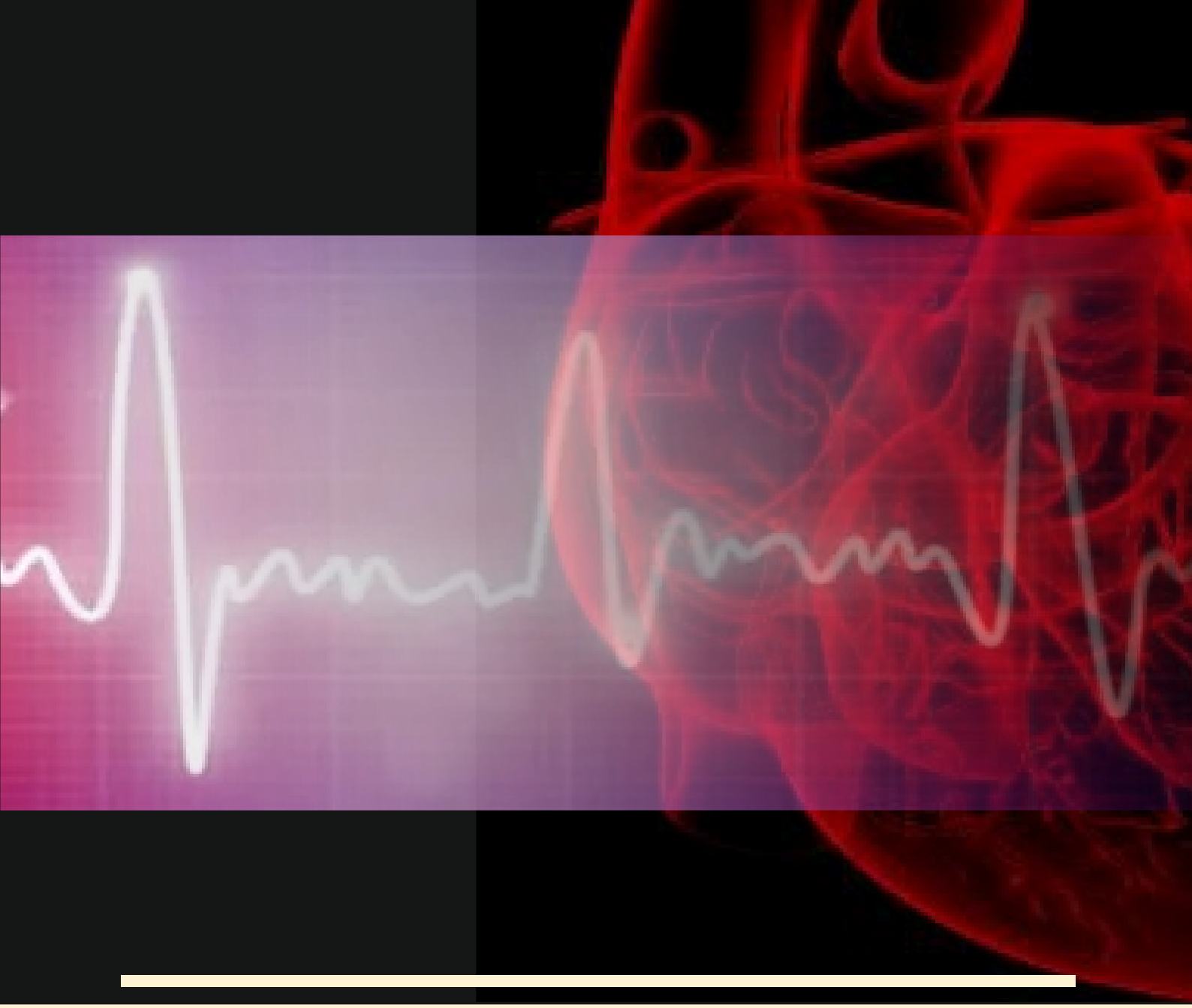


Ana Teresa Fonseca Salgueiro

Deteção de problemas cardíacos usando sinais do eletrocardiograma (ECG)

Dissertação de Mestrado Integrado em Engenharia Electrotécnica e de Computadores.

Coimbra
Outubro, 2020

1 2 9 0

UNIVERSIDADE D
COIMBRA

Universidade de Coimbra

Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica e de Computadores

Deteção de problemas cardíacos usando sinais do eletrocardiograma (ECG)

Ana Teresa Fonseca Salgueiro

Dissertation submitted to the Electrical and Computer Engineering Department of the Faculty of Science and Technology of the University of Coimbra in partial fulfillment of the requirements for the Degree of Master of Science in Electrical and Computer Engineering.

Supervisor: Prof. Doutor Rui Alexandre de Matos Araújo

Co-Supervisor: Doutor Francisco Alexandre Andrade de Souza

Jury: Prof. Doutor Jaime Baptista dos Santos

Prof. Doutor Fernando Manuel dos Santos Perdigão

Prof. Doutor Rui Alexandre de Matos Araújo

Outubro, 2020

"I do not know what I may appear to the world, but to myself I seem to have been only like a boy playing on the sea-shore, and diverting myself in now and then finding a smoother pebble or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me."

Sir David Brewster

Agradecimentos

Em primeiro lugar, gostaria de expressar o meu mais sincero agradecimento aos meus orientadores Prof. Doutor Rui Alexandre de Matos Araújo e Doutor Francisco Alexandre Andrade de Souza, por todo o apoio e ajuda prestados durante o desenvolvimento da minha tese. O encorajamento e *feedback* recebidos foram, igualmente, essenciais para a finalização do meu trabalho.

De seguida, um agradecimento muito especial aos meus amigos, em particular, Ana Cláudia, Carlos e Sandra que conheci nestes últimos anos e que representaram um grande apoio. Sei que continuarão a estar presentes na minha vida e que iremos manter para sempre as memórias que compartilhamos nesta cidade. Gostaria de agradecer também à Beatriz e ao Tiago, os resistentes que perduram desde a infância, e às suas respetivas famílias. Que continuemos a ter sempre os nosso almoços e jantares.

Por último, mas certamente não menos importante, quero expressar a minha profunda gratidão à minha família, especialmente aos meus pais, Fátima e João Paulo, e à minha irmã Ana Rita por todo o apoio e confiança dispensados nas fases mais desafiantes do processo de escrita da presente dissertação; nunca esquecendo os avós, por acreditarem sempre em mim e pelo seu apoio incondicional, pois a realização de um trabalho desta natureza não é concretizável sem a colaboração de diversas pessoas e a conversa amiga de outras.

Resumo

Recentemente, devido ao aumento do número de mortes por doenças cardiovasculares, o diagnóstico de doenças cardíacas tem sido um foco de bastante interesse no mundo computacional. Isto porque, a deteção de doenças cardíacas num estágio inicial pode prolongar a vida através de um tratamento adequado.

Inúmeros métodos para fazer a monitorização das condições cardíacas foram introduzidos no mercado, sendo que o mais utilizado é a eletrocardiograma (ECG). O ECG é o registo da variação da atividade bioelétrica do coração, que representa as contrações e relaxamentos cíclicos do músculo cardíaco humano. Este fornece informações importantes sobre os aspetos funcionais do coração e do sistema cardiovascular. No entanto, ler grandes quantidades de sinais de ECGs é um processo demorado. Por isso, a deteção automática de anomalias nos sinais do eletrocardiograma atua como um assistente para os médicos diagnosticarem uma condição cardíaca.

As irregularidades presentes no batimento cardíaco no formato do ECG são geralmente chamadas de arritmias. Arritmia é um termo comum para qualquer distúrbio cardíaco que difere do ritmo normal. A análise automática do sinal de ECG para deteção de batimentos cardíacos é difícil devido à grande variação nas características morfológicas e temporais das formas de onda do ECG entre pacientes diferentes, bem como nos mesmos pacientes.

Esta dissertação tem como objetivo desenvolver um método, que através da extração e classificação de *features* consiga fazer a deteção da fibrilhação auricular, que é um tipo de arritmia, através do sinal do eletrocardiograma. Deste modo, a metodologia proposta baseia-se na extração de um conjunto de características ("features") dos ECG, e na sua classificação através de diferentes tipos de classificadores baseados em *machine learning*, que consiste na execução de algoritmos que criam de modo automático modelos com base num conjunto de dados.

O método desenvolvido foi utilizado em 2000 ECGs de maneira a determinar a eficácia de cada um dos classificadores na deteção da doença cardíaca. Deste modo o presente documento inclui um estudo sobre os parâmetros do modelo escolhido e os resultados de classificação. Também é apresentado uma análise sobre as diferentes métricas de desempenho dos classificadores para o conjunto de teste.

Por sua vez, os resultados obtidos apoiam o uso de classificadores baseados em *machine learning* como ferramenta de classificação, na área de deteção de doenças cardíacas. O

sistema desenvolvido classifica 2000 ECGs, provenientes de duas classes, normal e fibrilhação auricular, com uma taxa de *accuracy* global de 93%, para o conjunto de teste. Na deteção particular da fibrilhação auricular, registou-se uma *sensitivity* de 97%, *precision* de 79.5% e *specificity* de 91.67% para o conjunto de teste.

Palavras Chave

Arritmias, Classificadores, Eletrocardiograma(ECG), Extração de *features*, Fibrilhação Auricular(FA), *Machine Learning*, Sinais do Eletrocardiograma

Abstract

Recently, due to the increase in the number of deaths from cardiovascular diseases, the diagnosis of heart disease has been a focus of great interest in the computational world. This is because the detection of heart disease at an early stage can prolong life through proper treatment.

Numerous methods for monitoring cardiac conditions have been introduced in the market, the most used being the electrocardiogram (ECG). The ECG is the recording of the variation in the bioelectric activity of the heart, which represents the contractions and cyclical relaxations of the human cardiac muscle. It provides important information about the functional aspects of the heart and the cardiovascular system. However, processing large amounts of raw electrocardiogram signals from sensors is time consuming. Therefore, the automatic detection of abnormalities in the electrocardiogram signals acts as an assistant for doctors to diagnose a heart condition.

The irregularities present in the heartbeat in the ECG format are generally called arrhythmia. Arrhythmia is a common term for any heart disorder that differs from the normal rhythm. The automatic analysis of the ECG signal to detect heartbeat is difficult due to the great variation in the morphological and temporal characteristics of the ECG waveforms between different patients, as well as in the same patients. This dissertation aims to develop a method of detecting atrial fibrillation, which is a type of arrhythmia, using the electrocardiogram signal.

The proposed methodology is based on the extraction of a set of characteristics (features) from ECG, and on their classification through different types of classifiers based on machine learning. These classifiers consists of the execution of algorithms that automatically create representation models based on a set of data.

The method developed was used on 2000 ECG in order to determine the effectiveness of each of the classifiers in detecting heart disease. Thus, this document includes a study on the parameters of the chosen model and the corresponding classification results. Not only, it presents an analysis of the influence of certain factors on the performance of the system, namely the size of the data set and the set of features used in the representation.

In turn, the results obtained support the use of classifiers based on machine learning as a classification tool, in the area of heart disease detection. Therefore, the developed system classifies 2000 ECG, as two classes: normal and atrial fibrillation, with an overall accuracy

rate of 93 %, on the test set. In the particular detection of atrial fibrillation, there was an sensitivity of 97 %, precision of 79.5% and specificity of 91.67% for the test set.

Keywords

Arrhythmias, Atrial Fibrillation (AF), Classifiers, Electrocardiogram (ECG), Electrocardiogram Signals, Features Extraction, Machine Learning

Lista de Abreviações

Siglas

ANN	*Artificial Neural Network*
AUC	*Area under the ROC curve*
AV	*auriculoventricular*
CMI	*Condition Mutual Information*
CMIM	*Conditional Mutual Info Maximisation*
CNN	*Convolutional Neural Network*
DISR	*Double Input Symmetrical Relevance*
DT	*Decision Tree*
DWT	*Discrete Wavelet Transform*
ECG	*Eletrocardiograma*
FA	*Fibrilhação Auricular*
FFN	*Feedforward Neural Network*
FNR	*False Negative Rate*
FP	*False Positive*
FPR	*False Positive Rate*
FN	*False Negative*
HCTSA	*Highly Comparative Time Series Analysis*
JMI	*Joint Mutual Information*
KNN	*K-Nearest Neighbors*
MIFS	*Mutual Information Feature Selection*
MIM	*Mutual Information Maximisation*
MRMR	*Max-Relevance Min-Redundancy*
NSR	*Ritmo Sinusal Normal*
RMSSD	*Root Mean Square of the Successive Differences*
ROC	*Receiver Operating Characteristic*
SA	*Sinoauricular*
SVM	*Support Vector Machine*
TN	*True Negative*
TP	*True Positive*

TPR	*True Positive Rate*
WESN	*Wireless ECG Sensor Nodes*
WPT	*Wavelet Packet Transform*

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

No decorrer dos últimos anos assistiu-se a um notável crescimento, tanto na área da tecnologia como na área da ciência, que conduziu ao exponencial aparecimento de máquinas computorizadas na sociedade e à sua consequente popularização. Devido a este facto, surgiu uma necessidade de interação com estas novas tecnologias e, desta feita, criaram-se condições para as colocar ao serviço da saúde, como é exemplo, a deteção de doenças cardíacas por meios automáticos, tendo por base os sinais do eletrocardiograma (ECG). Isto porque, no tempo presente, e de acordo com os dados da Organização Mundial de Saúde, os distúrbios cardiovasculares representam uma elevada percentagem do total de mortes no mundo (cerca de 30 %), sendo, portanto, considerado um dos grandes flagelos da atualidade [Valentin Fuster, 2010].

O Eletrocardiograma pode ser definido como uma representação gráfica de impulsos elétricos produzidos no coração [Zipes et al., 2018]. É um teste de baixo custo, não invasivo e eficaz no exame de alterações do ritmo cardíaco normal tendo, por conseguinte, condições para se revelar e afirmar como uma ferramenta de diagnóstico padrão [De Chazal et al., 2004]. Devido à natureza deste sinal, o estudo manual de um elevado número de ECGs é um processo lento e este facto aliado ao crescimento do número de pacientes cardiovasculares torna necessária a implementação de métodos automáticos de diagnóstico médico. Portanto, há uma necessidade de métodos computacionais poderosos para maximizar as informações extraídas de conjuntos de dados de ECG abrangentes [Obermeyer and Emanuel, 2016]. Na verdade, também a variedade existente de formatos de ECG e as suas diferentes aplicações clínicas começam a exigir uma diversidade de técnicas computacionais para dar resposta a essa necessidade [Lyon et al., 2018].

Face ao exposto, os métodos computacionais irão permitir, avaliar o diagnóstico, previsão e deteção de patologias cardíacas através dos valores obtidos a partir da interpretação do ECG. As vantagens destes métodos residem no facto de a análise computacional permitir uma avaliação mais célere dos resultados obtidos através do ECG, e, por meio de comparação com uma base de dados existente, a deteção de eventuais doenças cardíacas que, dada ainda a sua precocidade, poderiam passar despercebidas ao olhar atento de um clínico.

Por fim, este trabalho tem como objetivo a procura de um método automático que seja eficiente, eficaz e que possa ajudar a prevenir e diagnosticar doenças cardíacas no mais curto espaço de tempo, neste caso, irá concentrar-se concretamente na deteção da Fibrilhação Auricular (FA).

1.1 Motivação e Contexto

A Fibrilhação Auricular é a arritmia crónica mais frequente e uma das doenças cardíacas de maior prevalência na população, conduzindo ao aumento do número de ataques cardíacos e a uma elevada taxa de mortalidade. Desta forma, é necessário que esta enfermidade seja detetada relativamente cedo de forma a ser possível prevenir tais acontecimentos [Academic and Center, 2011]. Tanto a análise como a classificação dos sinais de um eletrocardiograma são processos fulcrais para o diagnóstico de uma doença cardíaca.

Um dos problemas desta classificação é a falta de padronização e a elevada variabilidade entre os diferentes parâmetros que se pretendem analisar num ECG (*features*), isto é, características predominantes de um sinal que são utilizadas para a classificação do mesmo. É ainda de salientar que, apesar de existir um banco de dados de ECG, o mesmo não contém informação suficiente, o que dificulta a validação de resultados, sendo por vezes necessários exames mais profundos para averiguar as anormalidades do ECG ao nível do órgão e da célula.

A deteção de problemas cardíacos pode ser efetuada através de métodos baseados em *machine learning*, que consistem na execução de algoritmos que criam de modo automático modelos de representação de conhecimento com base num conjunto de dados; ou deep learning que é um método que possui a capacidade de "aprender", automaticamente, as *features* de médio e alto nível a partir de imagens não treinadas [Lyon et al., 2018]. De referir que, na presente dissertação, será apenas utilizada a temática *machine learning*.

No caso particular da tipologia da doença em estudo, foram desenvolvidas várias abordagens para a deteção automática de indícios de fibrilhação auricular. Estas baseiam-se sobretudo na análise da onda P [Firoozabadi et al., 2018], na irregularidade dos intervalos RR [Dallali et al., 2011] ou, ainda, na combinação destas duas características [Kropf et al., 2017]. Estas abordagens requerem o reconhecimento prévio da onda P e/ou do pico R, são elementos que fazem parte de um sinal do ECG, no fundo trata-se de uma etapa de pré-processamento e, consequentemente, o resultado final terá menos eficácia caso os picos mencionados não sejam notados ou, por outra, detetados erradamente.

Contudo, começam a surgir novas abordagens que não requerem a deteção prévia de ondas ou intervalos do ECG para fazer o diagnóstico de deteção da doença. Assim, a presente dissertação tem como objetivo implementar um método de classificação de sinais do ECG, em que não seja necessário a deteção prévia de ondas ou intervalos, para fazer a deteção da fibrilhação auricular. Este método vai se basear na extração massiva de *features* de séries

temporais do ECG. Pode-se definir uma série temporal como uma sequência de registo de números em intervalos regulares durante um período de tempo.

1.2 Objetivos

A presente dissertação tem como objetivo, a utilização de um método que se baseia na extração de *features* dos sinais do ECG, e numa segunda fase, na seleção do conjunto de *features* que permite a maior eficácia de classificação com o objetivo de detetar problemas de fibrilhação auricular. Com efeito, o objetivo que se pretende alcançar é o de determinar as condições para as quais a eficácia da classificação é maximizada, recorrendo, para tal, a um estudo da relação entre o número de *features* extraídas e utilizadas. Este estudo realizou-se, portanto, com o intuito de proporcionar um diagnóstico mais efetivo na deteção da fibrilhação auricular.

1.3 Implementações e Contribuições Principais

Todo o trabalho prático necessário foi realizado no ambiente de computação *MATLAB*® [The MathWorks, 1994-2020a]. Foram testados vários cenários com a finalidade de avaliar a eficácia das *features* extraídas através da biblioteca *HCTSA (Highly Comparative Time-Series Analysis)*. A seleção de *features* foi efetuada através da *Toolbox Feast*, que fornece um conjunto de implementações de algoritmos de seleção de *features* de filtro teórico de informação e uma implementação de *RELIEF* para fins de comparação.

O capítulo 4, descreve como foram utilizados todos os classificadores de machine learning disponíveis na *Toolbox Statistics and Machine Learning* do *Matlab*, sendo que no final foram selecionados os que obtiveram melhor desempenho na diferenciação entre batimento cardíaco normal e fibrilhação auricular.

A base de dados de ECG utilizada para este estudo foi disponibilizada pela *PhysioNet Challenge 2017* [Clifford et al., 2017]. *PhysioNet* é um recurso de pesquisa do NIH (*National Institutes of Health*) para sinais fisiológicos complexos e é apoiado pelo Instituto Nacional de Ciências Médicas Gerais (NIGMS) e pelo Instituto Nacional de Imagens Biomédicas e Bioengenharia (NIBIB).

Por fim, todos os testes efetuados e códigos mencionados anteriormente, foram desenvolvidos num computador portátil com as seguintes especificações: CPU *Intel Core*®i7-6700HQ 4x2.6GHz e GPU *NVIDIA*® *GeForce*® GTX 960M.

1.4 Estrutura

Estruturalmente, a presente dissertação encontra-se dividida em 5 capítulos, sendo o conteúdo de cada um deles o seguinte:

- O Capítulo 1 contempla uma breve introdução, o contexto e a motivação sobre o tema em estudo e os principais objetivos desta dissertação;
- O Capítulo 2 corresponde ao estado da arte, o mesmo contém informações básicas sobre os sinais do ECG, sobre a descrição clínica da patologia em estudo, a fibrilhação auricular, e engloba ainda uma explicação sobre *machine learning*;
- O Capítulo 3 apresenta a explicação dos métodos desenvolvidos para a extração e seleção de *features*, juntamente com a implementação prática e uma descrição dos classificadores utilizados;
- O Capítulo 4 contém todos os resultados experimentais (treino e teste), obtidos para a classificação dos batimentos cardíacos, neste caso relativos à fibrilhação auricular, e a apresentação e discussão desses mesmos resultados;
- O Capítulo 5 encerra a conclusão desta dissertação juntamente com algumas sugestões de possíveis melhorias para trabalhos futuros.

Capítulo 2

Estado da Arte

Este capítulo fornece informações básicas acerca da anatomia e fisiologia do coração, da obtenção do sinal do electrocardiograma e da patologia clínica de fibrilhação auricular, bem como uma visão geral no que concerne aos diferentes tipos de *machine learning*.

2.1 O Coração e o Sistema de Aquisição do ECG

O coração é o músculo responsável por bombear o sangue, sendo que este último depois irá circular por todo o corpo e, assim, assegurar que as quantidades adequadas de oxigénio e nutrientes são fornecidos às células. Estruturalmente é constituído por quatro câmaras. Concretizando, tanto o lado esquerdo como o lado direito albergam, cada um, uma aurícula e um ventrículo. Na verdade, as câmaras superiores, aurícula direita e aurícula esquerda, atuam como câmaras recetoras, isto é, ao receberem o sangue proveniente de diversas partes do corpo contraem-se e impulsionam-no para as câmaras inferiores, o ventrículo direito e o ventrículo esquerdo [Gordon Betts et al., 2013]. Este processo deve ser rápido e sequencial para maximizar a ativação simultânea do miocárdio(tecido que constitui a parte contráctil da parede do coração, formado por músculo cardíaco). Isto é, o coração encerra um sistema de condução elétrico composto por células miocárdicas especializadas, estas células formam feixes de fibras que, por sua vez, atuam como cordões elétricos que difundem o potencial de ação rápido e sequencial, para a contração do miocárdio, ao nível das aurículas e dos ventrículos [LLC, 2018]. A figura 2.1 ilustra os componentes relevantes do sistema de condução.

No coração, o ciclo cardíaco começa quando as células do nó Sinoauricular (SA) descarregam um potencial de ação, que se espalha como um impulso elétrico pelo sistema de condução do coração e inicia a contração do miocárdio. O potencial de ação inclui uma despolarização (ativação) seguida de uma repolarização (recuperação). Com efeito, existem correntes iónicas específicas que se encontram localizadas nas membranas celulares e que, respetivamente, vão abrir e fechar, durante a despolarização e repolarização, de forma a que os iões (Na^2, K^2, Ca^{2+}) possam fluir entre os compartimentos intra e extracelular. Assim, compreende-se que, o potencial de ação ao envolver o movimento de iões, que são partículas

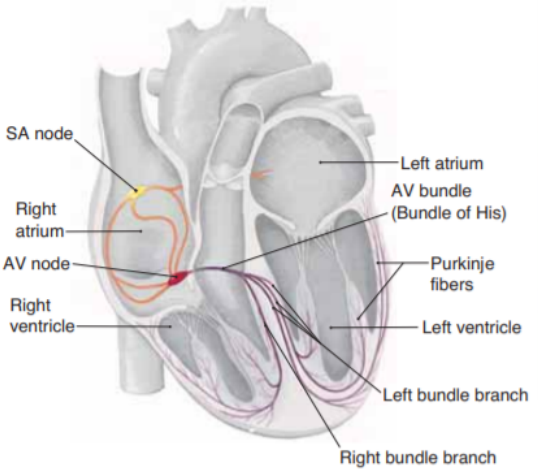


Figura 2.1: Componentes condutores do coração [Valerie C. Scanlon, 2007].

carregadas, irá gerar uma corrente elétrica [LLC, 2018].

Por sua vez, as correntes eléctricas geradas no miocárdio são transmitidas até à pele, lugar onde podem ser detetadas por elétrodos. Estes irão registar a atividade elétrica do coração e as formas de onda resultantes. Dependendo do tipo de máquina utilizada e do número de elétrodos colocados, poderão ser registadas através do eletrocardiograma, várias visualizações da atividade elétrica do coração [Sampson and McGrath, 2015].

Acontece que, com base no número de elétrodos utilizados e de acordo com a disposição destes pelo corpo, podem ser encontrados diferentes terminais. Uma *lead* é, assim, uma linha imaginária entre dois elétrodos, usada para ilustrar a atividade elétrica de uma dada perspetiva do coração [Potter, 2011]. Com efeito, uma *lead* fornece uma nova "visão" do coração e pode incluir informações que não são encontradas nas outras *leads*. De referir que, todas essas diferentes ligações podem somar, no total, 12 visões diferentes do coração [Thaler, 2012].

Um ECG padrão tem, portanto, 12 *leads*, sendo que estas se encontram divididas em 2 grupos de 6 *leads* cada um. Concretizando, o primeiro contém 3 *leads* bipolares e 3 *leads* unipolares aumentadas, ao passo que, o segundo, é unicamente constituído por 6 *leads* torácicas (precordiais). Na figura 2.2 podemos observar as posições das *leads* através do triângulo de Einthoven [Davey and Sharman, 2018]. Neste caso, os elétrodos colocados no braço esquerdo, no braço direito e na perna esquerda são usados para obter, respetivamente, a *lead I*, *lead II* e *lead III*. Estas vão registar a diferença de potencial entre dois pontos um com terminal positivo e outro negativo. Por outro lado, as *leads* do membro aumentado servem para registar a diferença de potencial entre um ponto teórico no centro do triângulo de Einthoven, com um valor de 0, e os elétrodos em cada extremidade. São conhecidas como *leads* unipolares aumentadas e designadas por: aVR, aVL e aVF (aV para a *lead* aumentada, R para o braço direito, L para o braço esquerdo e F para o perna esquerda).

Por último, existem as *leads* de Wilson que são as torácicas, que se vão posicionar no lado esquerdo do tórax num plano quase horizontal. Para fazer gravações com as *leads* de Wilson é necessário que estas estejam ligadas com as três *leads* do membro, para que se forme um elétrodo indiferente com altas resistências isto é, um elétrodo que praticamente não apresente variação de potencial durante a atividade cardíaca. De salientar que, os elétrodos torácicos detetam principalmente vetores potenciais direcionados para as costas e que os mesmos são dificilmente detetáveis no plano frontal [Khan, 2004].

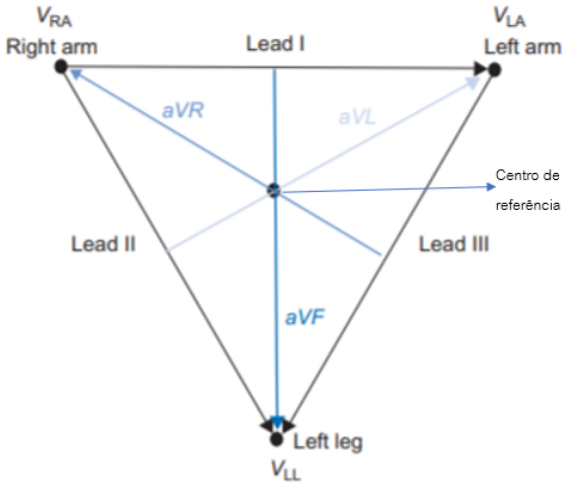


Figura 2.2: O triângulo de Einthoven [Davey and Sharman, 2018].

Quando se trata de registar um ECG existem vários dispositivos que oferecem diferentes possibilidades. A utilização do método das 12 derivações do ECG (*twelve lead ECG*) é a ferramenta padrão utilizada atualmente por cardiologistas para detetar diferentes disfunções cardiovasculares. No entanto, os problemas cardíacos nem sempre conseguem ser observados num registo padrão de 10 segundos de medições do ECG, através do método de 12 derivações, vulgarmente realizado em hospitais e clínicas. Contudo, foi possível um avanço na prática de registo e análise de sinais do ECG graças aos dispositivos portáteis de gravação deste exame que fazem uma monitorização constante. Exemplos, como o *Holter*, Apple Watch [Apple, 2018], AliveCor [AliveCor, 2020], Omron HeartScan [Omron Healthcare, 2020], QardioMD [Qardio, 2018], *Wireless ECG Sensor Nodes (WESNs)* [Park et al., 2006] e a *Astroskin Smart Shirt* [Carre Technologies, 2012] estão a revolucionar o diagnóstico cardíaco, através da medição da atividade cardíaca e transmissão destas informações para um serviço de onde serão armazenadas e processadas remotamente.

2.2 O sinal do ECG

Os sinais do eletrocardiograma (ECG) dão lugar a um registo da atividade bioelétrica do sistema cardíaco e algumas, ou todas, as partes desses mesmos sinais correspondem a alterações da condição normal, no caso de se tratar de um paciente com patologia cardíaca [Vafaie et al., 2014].

Com efeito, o ECG regista o sinal elétrico através da diferença de tensão entre os elétrodos colocados na pele. Este é um sinal não estacionário e variável no tempo, em que os intervalos dos batimentos cardíacos adjacentes variam, igualmente, com o tempo. É uma técnica não invasiva, ou seja, o sinal que é utilizado na identificação de cardiopatias é medido na superfície do corpo humano [De Chazal et al., 2004]. Tanto a amplitude como a duração da onda P-QRS-T contêm informações úteis sobre a natureza da doença que afeta o coração, estes serão definidos nas sub-secções seguintes.

2.2.1 Ondas, Intervalos e Segmentos do ECG

A representação de um ciclo cardíaco pelo ECG apresenta as características morfológicas conforme ilustrado na figura 2.3, onde se encontram refletidos os principais componentes: a

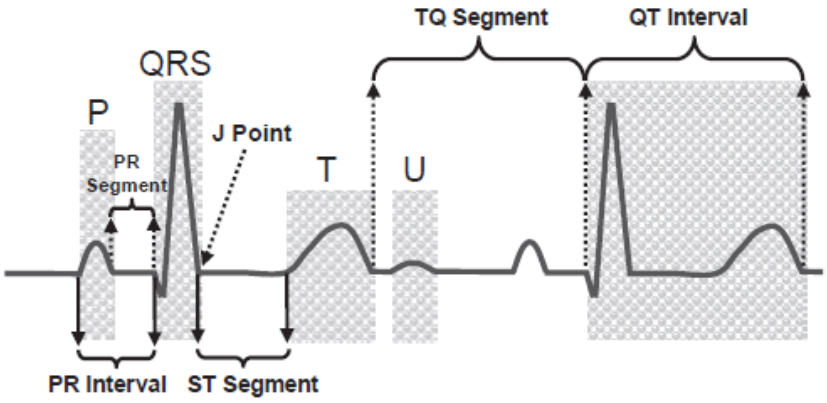


Figura 2.3: Representação esquemática da forma de onda do ECG normal [F.Baltazar, 2009].

onda P, o complexo QRS e a onda T. Por vezes, acontece que pode aparecer também uma onda U, esta contém informações úteis sobre a natureza das doenças que afetam o coração. Com efeito, a configuração de um ECG normal é composta por ondas, complexos, segmentos e intervalos, todos estes parâmetros são chamados de *features* e encontram-se gravados como tensão (eixo vertical) em função do tempo (eixo horizontal) [Jambukia et al., 2015].

De seguida, irão ser descritas as diferentes ondas, intervalos e segmentos. A duração de cada um no sinal do ECG está representada na Tabela 2.1.

Tabela 2.1: Duração das ondas, intervalos e segmentos do sinal de ECG [Jambukia et al., 2015].

Features	Duração (ms)
Onda P	60-80
Onda T	120-160
Complexo QRS	80-120
Segmento PR	50-120
Segmento ST	80-120
Intervalo PR	120-200
Intervalo QT	300-420
Intervalo RR	(0.4-1.2) s

2.2.1.1 Onda P

A onda P é a primeira deflexão (primeira curva representada na figura 2.3) no eletrocardiograma e ocorre quando acontece a despolarização auricular, isto é, ativação sequencial das aurículas. Como o nó sinusal está localizado na parte superior da aurícula direita, o impulso sinusal viaja desde a aurícula direita até à aurícula esquerda e, posteriormente, até aos ventrículos. A primeira metade da onda P (inicio da curva e despolarização da aurícula direito) deve-se à ativação da aurícula direita. Já a segunda metade (fim da curva e despolarização da aurícula esquerda) é devida à ativação da aurícula esquerda. Uma onda P nítida, isto é, que não teve nenhum problema na contração das aurículas, antes do complexo QRS, representa o ritmo normal. Por outro lado, a ausência desta onda pode sugerir fibrilhação auricular ou outro tipo de doença cardíaca [F.Baltazar, 2009].

2.2.1.2 Onda T

A Onda T corresponde a uma repolarização ventricular rápida. Esta pode estar representada por uma deflexão para baixo (negativa) ou por picos altos e pontiagudos. O facto da onda ser alta ou baixa (picos de maior ou menor amplitude) pode estar relacionado com a ocorrência de doenças cardíacas tais como bloqueio do ramo esquerdo ou direito, hipertrofia ventricular e ansiedade [E. et al., 2016].

2.2.1.3 Onda U

O fim da onda T completa o ciclo cardíaco normal que inclui, por esta ordem, a onda P, o complexo QRS e a onda T. Esta última, no entanto, pode frequentemente ser seguida por uma pequena deflexão positiva chamada onda U. Esta nem sempre está presente, mas pode ser o último complexo no ECG a ser registado. Acontece que as ondas U são recorrentemente vistas em indivíduos normais, contudo são consideradas anormais sempre que se encontram invertidas ou, quando igualam ou excedem a onda T. Esta situação ocorre num contexto de hipocalémia [F.Baltazar, 2009].

2.2.1.4 Complexo QRS

O Complexo QRS representa a despolarização ventricular, que está associada à ativação dos ventrículos. Este gera a maior deflexão no ECG, uma vez que, os ventrículos contêm a maior massa de células musculares do coração, denominadas coletivamente de miocárdio. O complexo QRS é medido desde o início da primeira deflexão, pode começar com uma onda Q ou uma onda R, dependendo da situação, e estende-se até o final da última deflexão. A duração do QRS é aumentada quando há hipertrofia ventricular, bloqueio de ramo ou quando ocorre excitação prematura dos ventrículos [F.Baltazar, 2009].

2.2.1.5 Segmento PR

O Segmento PR começa no final da onda P e estende-se até ao início do complexo QRS. Corresponde ao tempo que o impulso leva para ir do nó AV até aos ventrículos. Contudo, o intervalo PR é clinicamente mais relevante do que o segmento PR para a deteção de qualquer anomalia [F.Baltazar, 2009].

2.2.1.6 Segmento ST

O segmento ST começa do ponto J (2.3) e estende-se até ao início da onda T. O segmento ST é plano ou isoelétrico (carga elétrica igual a zero) e corresponde à fase 2 do potencial de ação das células miocárdicas ventriculares. Tem lugar no final da despolarização ventricular, mas antes do início da repolarização. Consoante a deflexão da onda T são identificadas doenças como isquemia miocárdica e hipertrofia ventricular esquerda [F.Baltazar, 2009].

2.2.1.7 Intervalo PR

O Intervalo PR é medido desde o início da onda P até ao inicio do complexo QRS. Se o complexo QRS começa com uma onda Q, o intervalo PR é medido desde o início da onda P até o início da onda Q (intervalo P-Q), mas mesmo assim é chamado intervalo PR. Inclui o tempo que leva para o impulso sinusal ir das aurículas aos ventrículos. A duração do intervalo PR muda com a frequência cardíaca, isto é, diminui com o aumento da frequência cardíaca [F.Baltazar, 2009].

2.2.1.8 Intervalo QT

O intervalo QT representa a repolarização da membrana. É medido desde o início do complexo QRS até ao final da onda T. A eventual presença de uma onda U não está incluída na medição. Para avaliar a duração do intervalo QT devem ser selecionadas várias *leads* e, o intervalo QT eleito para a medição, será aquele que for o mais longo, de todo o registo de ECG de 12 derivações. Consoante a duração do intervalo (longo ou curto) podem ocorrem, respetivamente, doenças como taquiarritmias ventriculares, morte súbita e doença de Graves [F.Baltazar, 2009].

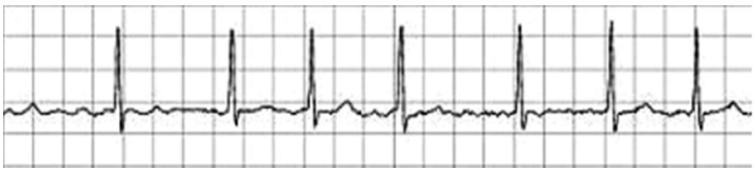
2.2.2 Arritmias no sinal do ECG

O ritmo normal do coração, onde não existem doenças ou distúrbios na morfologia do sinal do ECG, é chamado de ritmo normal, a frequência cardíaca deste ritmo varia entre 60 a 100 batimentos por minuto. A regularidade do intervalo RR (intervalo entre picos do ECG) está dependente do ciclo respiratório prende-se com a existência de uma onda P normal seguida por um complexo QRS, também ele, normal. Qualquer distúrbio do ritmo cardíaco ou alteração no padrão morfológico indicam a existência de uma arritmia cardíaca, esta pode ser detetada pela análise da forma das ondas registadas no ECG.

Quando a frequência cardíaca aumenta acima de 100 batimentos por minuto, o ritmo é conhecido como taquicardia. Caso contrário, se a frequência cardíaca for muito baixa, trata-se de uma bradicardia. Na primeira situação, em que a frequência cardíaca é muito rápida, acontece que os ventrículos não se encontram completamente cheios aquando da contração, o que diminui a eficiência de bombeamento. Os ritmos que se desviam do normal são chamados de arritmias, pois são anormais e disfuncionais [Acharya et al., 2007].

Existem diferentes tipos de arritmias, sendo a fibrilhação auricular (FA), a mais comum. A mesma induz batimentos cardíacos variáveis e frequentemente muito rápidos ou lentos. Com efeito, num cenário de FA, os impulsos elétricos nas aurículas são muito desorganizados e não se encontram sincronizados com os ventrículos. A figura 2.4 ilustra um exemplo de um sinal de ritmo normal e um exemplo de sinal com fibrilhação auricular.

(a) Ritmo normal.



(b) Fibrilhação auricular.

Figura 2.4: Sinais do ECG.

Existem várias abordagens para diagnosticar a FA tais como, exames clínicos, electrocardiograma, monitor de Holter, entre outras. Contudo o ECG é o exame a que se recorre sempre para confirmar o diagnóstico final.

A forma de onda do ECG contém indicações essenciais acerca da condição de saúde do coração. Com base na figura 2.4, podemos concluir, que uma forma de onda do ECG normal é composta por uma onda P, um complexo QRS e uma onda T, por esta mesma ordem. Ao invés, num sinal do ECG de fibrilhação auricular, a onda P não se encontra presente sendo, por conseguinte, substituída por muitas ondas inconsistentes [Hagiwara et al., 2018].

Devido à natureza variável do sinal do ECG, pode ser desafiante interpretá-lo com base em observações efetuadas durante janelas de tempo limitado. Posto isto, são necessários sistemas de diagnóstico auxiliares autónomos para facilitar o diagnóstico da FA através dos sinais do ECG, podendo assim contribuir para um tratamento o mais cedo possível [Hagiwara et al., 2018].

Alguns dos referidos métodos utilizados baseiam-se na representação eletrocardiográfica da FA, sendo esta caracterizada por intervalos RR irregulares e ausência de ondas P. Com base nestas idiossincrasias foram desenvolvidos vários algoritmos para detetar FA a partir da forma de onda do ECG [Chandra et al., 2017]. Esses algoritmos baseiam-se na análise de tempo-frequência do ECG, na avaliação dos intervalos RR, na inferência a curto prazo da variabilidade da frequência cardíaca (VFC) e, ainda, numa análise sequencial para verificar a ausência de ondas P [Billeci et al., 2017].

2.2.3 Ruído

Os sinais de ECG podem ser difíceis de interpretar e tornam-se, portanto, complexos sempre que existam ruídos de alta frequência e, também, ruídos provenientes dos elétrodos ou dos movimentos musculares. A maioria dos equipamentos atuais de ECG já contém filtros integrados para lidar, precisamente, com a maioria dos diferentes tipos de ruído. No entanto, acontece que, quanto mais "limpo" for o sinal, maior será também a quantidade de informações perdidas, isto é, a eliminação excessiva de ruído pode apagar as diferenças e subtilezas entre as diversas doenças, o que torna a correta identificação de cada um impossível. Posto isto, é necessária uma eliminação de ruído controlada para que seja possível visualizar todas as características do sinal da onda, sem perda de informação importante.

2.3 *Machine Learning*

Machine learning é uma área em que ouve um crescimento rápido, com a existência de várias aplicações. Na área da saúde, a classificação de batimentos cardíacos é provavelmente a aplicação mais desenvolvida de *machine learning* no ECG. O *machine learning* pode ser usado para realizar cálculos avançados e complexos.

A ideia desta aprendizagem é que devemos treinar as máquinas e, para isso, damos-lhes acesso a dados, medidas de desempenho e deixar-se-á o algoritmo "aprender", isto é, ajustar de modo iterativo o modelo de representação de conhecimento de modo a que o desempenho seja melhorado. Os dados consistem em duas ou mais classes e são treinados com base em vetores de *features*. Para treinar um modelo de classificação, o método de aprendizagem depende do conjunto de *labels* fornecidos ou não aos vetores de *features* e, com isto, podem ser divididos em aprendizagem supervisionada e não supervisionada, respetivamente.

Em *machine learning*, existem duas entidades principais: o *teacher* e o *learner*. O *teacher* tem o conhecimento necessário para realizar uma determinada tarefa, e o objetivo do *learner*

é aprender o conhecimento para realizar a tarefa [Lampropoulos and Tsihrintzis, 2015].

Podemos considerar, um conjunto $X = (x_1, \ldots, x_n)$ com n amostras, supondo que todas as amostras são extraídas de forma independente e idêntica a partir de uma distribuição comum χ. O objetivo na aprendizagem não supervisionada é encontrar uma estrutura interessante nos dados X. Este problema pode ser o de descobrir grupos de exemplos semelhantes dentro dos dados (agrupamento), determinar a distribuição dos dados que poderiam ter gerado X (estimativa de densidade), entre outros [Bishop, 2006].

O objetivo, no caso da aprendizagem supervisionada, é encontrar um mapeamento de x a y, dado um conjunto de pares de treino (x_i, y_i) [Chapelle et al., 2010]. Os y_i são denominados *labels* das amostras x_i, com $Y = (y_1, \ldots, y_n) \in \mathcal{Y}$ sendo um conjunto de *labels*. Esta tarefa é bem definida, pois um mapeamento pode ser avaliado em termos de seu desempenho preditivo em amostras de teste. Quando os *labels* são contínuos, a tarefa é chamada regressão. Por outro lado, quando *labels* assumem valores num conjunto finito (discreto), a tarefa é chamada classificação.

Por último é mencionado uma aprendizagem que é fundamentalmente diferente da aprendizagem supervisionada. Esta denomina-se de aprendizagem por reforço, neste caso, o *learner* não tem conhecimento a priori do que fazer. Em vez disso, ele deve realizar ações para atingir uma meta, e uma recompensa é dada pelo *teacher* em cada estado de acordo com a meta definida [Sutton and Barto, 1998]. Portanto, o *learner* deve evoluir por meio de tentativa e erro, equilibrando-se entre a exploração de novas ações possíveis e a exploração de seu conhecimento atual.

2.3.1 Redução da dimensão de *features*

Para a maioria dos modelos de *machine learning*, a necessidade de fazer um escalonamento de *features* fornece classificadores mais precisos. Portanto, fazer uma dimensionamento de *features* é uma etapa comum no algoritmo de pré-processamento. O escalonamento de *features* pode ser realizado com métodos diferentes, onde a normalização e a padronização de *features* são dos exemplos mais comuns.

A padronização de *features* é realizada através de uma estimativa da média e da variância para cada uma delas e no vetor de *features*, enquanto, a normalização de *features* dimensiona-as com base nos valores máximos e mínimos de cada uma.

2.3.2 Treino e Teste

O treino do classificador é feito para que este aprenda os padrões dos dados fornecidos, supervisionados ou não. A melhor classificação para estes dados é encontrada através da avaliação da variação dos diferentes modelos de classificação existentes. A validação de um modelo de *machine learning* é feita através de testes a dados que não foram introduzidos no treino.

	Class 1 Predicted	Class 2 Predicted
Class 1 Actual	True Positives	False Negatives
Class 2 Actual	False Positives	True Negatives

Figura 2.5: Matriz de Confusão.

O teste de um modelo de classificação é feita através da previsão dos dados de teste, isto é, dados que contém os *labels* e estes são usados para verificar se o classificador consegue prever o *label* verdadeiro. As previsões destes dados são exibidos numa matriz de confusão, que mostra a relação entre as classes previstas e as atuais, representada na Figura 2.5.

As previsões consistem em Verdadeiro Positivo (TP) ou Verdadeiro Negativo (TN) se as classes corretas foram previstas, e Falso Positivo (FP) e Falso Negativo (FN) se a classe for prevista como a classe errada. Através disto, podemos calcular a *Accuracy* do modelo, que significa a proporção do número total de batimentos positivos e negativos verdadeiros para o número total de batimentos existentes, é dada pela equação 2.1 [Jambukia et al., 2015].

$$ACC = \frac{TP + TN}{TP + TN + FN + FN}. \tag{2.1}$$

A *Sensitivity*, que é a proporção de batimentos positivos verdadeiros em relação ao total de batimentos positivos verdadeiros e falsos negativos; a *Specificity*, que significa a proporção de batimentos negativos verdadeiros em relação ao total de batimentos negativos verdadeiros e positivos; e por último, a *Precision* é a proporção de positivos verdadeiros em relação ao total de positivos previstos; são também, três medidas de desempenho avaliadas através da matriz de confusão. Podem ser calculadas pelas seguintes equações:

$$Sensitivity = \frac{TP}{TP + FN}, \tag{2.2}$$

$$Specificity = \frac{TN}{TN + FP}, \tag{2.3}$$

$$Precision = \frac{TP}{TP + FP}. \tag{2.4}$$

Uma das medidas de desempenho utilizadas baseia-se na análise da Curva ROC (*Receiver operating characteristic*) é representada por um gráfico que ilustra a capacidade de diagnóstico de um sistema classificador binário conforme seu limite de discriminação é variado. A curva ROC compara os valores da taxa de verdadeiro positivo (TPR) contra a taxa de falso positivo (FPR) em várias configurações de limite. A taxa de verdadeiro positivo

também é conhecida como sensibilidade. A taxa de falso positivo também é conhecida como probabilidade de falso alarme e pode ser calculada como (1 - especificidade) [Wikipedia, 2020].

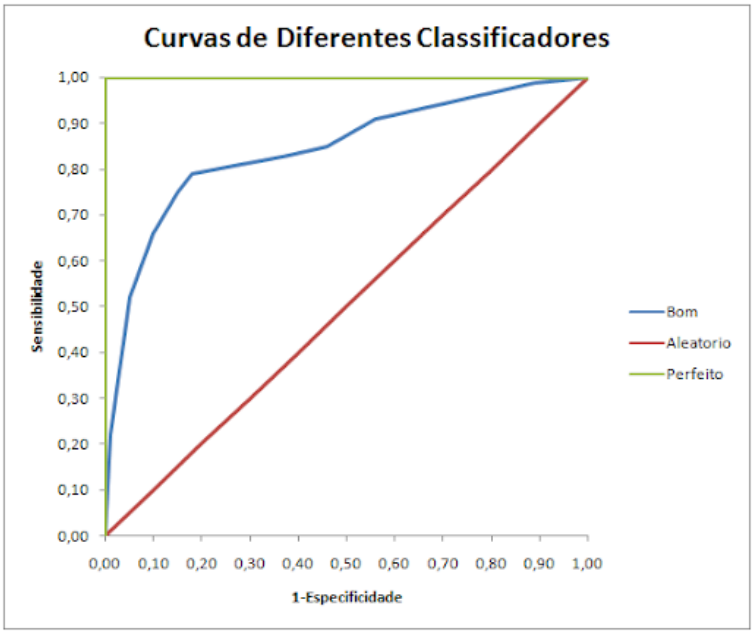


Figura 2.6: Representação da curva ROC

Podemos observar que se as distribuições de probabilidade para deteção e alarme falso forem conhecidas, a curva ROC pode ser construída através da função de distribuição cumulativa (área sobre a distribuição de probabilidade) da probabilidade de deteção no eixo y versus a função de distribuição cumulativa da probabilidade de alarme falso no eixo x.

A área sobre a curva ROC (AUC) é igual à probabilidade de que o modelo classifique um exemplo positivo aleatório mais alto do que um exemplo negativo aleatório.

2.3.3 Parâmetros de Otimização

Existem diversos tipos de classificadores, e cada um requer parâmetros diferentes para produzir os limites de decisão do classificador. Por exemplo, ao configurar o hiperparâmetro *grid search*, o modelo pode ser treinado com parâmetros diferentes, usa o número de valores por dimensão determinado pelo valor de número de divisões da *grid*. Pesquisa numa ordem aleatória, usando amostragem uniforme sem substituição da *grid*. Isto leva, a que a *accuracy* possa ser melhorada consoante os parâmetros fornecidos ao modelo de classificação. Embora a estimativa de hiperparâmetros com *grid search* seja um processo demorado, este pode fornecer melhores modelos de classificação.

O hiperparâmetro *random search*, tal como a *grid search*, faz testes com diferentes parâmetros. No entanto, faz uma pesquisa aleatoriamente entre pontos, onde o número de pontos corresponde ao valor de iterações.

Por último, o *bayesian optimization* tenta minimizar uma função objetivo escalar $f(x)$ para x em um domínio limitado. A função pode ser determinística ou estocástica, o que significa que pode retornar resultados diferentes quando avaliada no mesmo ponto x. Os componentes de x podem ser reais contínuos, inteiros ou categóricos, o que significa um conjunto discreto de nomes.

Capítulo 3

Metodologia

Este capítulo vai apresentar as etapas desenvolvidas na deteção automática da doença cardíaca "fibrilhação auricular'. São apresentados todos os passos realizados, desde a preparação de dados até ao processo final da classificação. Foi usada a biblioteca HCTSA para a extração de *features* e todos os classificadores usados são baseados em *machine learning*. Estes estão disponíveis na *Toolbox Statistics and Machine Learning* do *Matlab*. Posto isto, a Figura 3.1 representa as diferentes etapas de desenvolvimento mencionadas anteriormente, e que serão descritas nos subcapítulos seguintes.

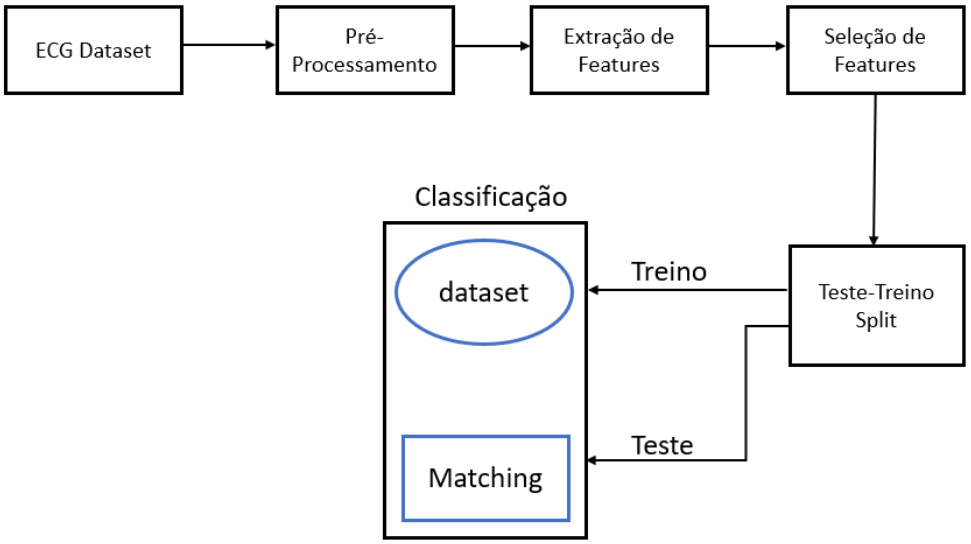
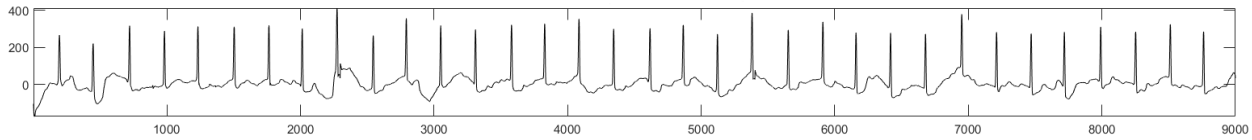


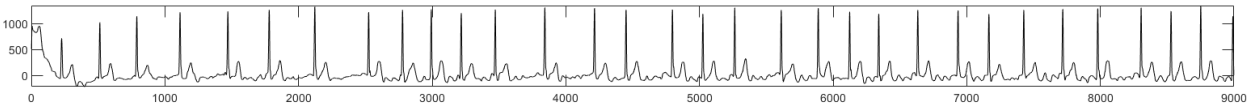
Figura 3.1: Diagrama de blocos.

3.1 Preparação de Dados

Neste estudo, a base de dados de ECGs utilizada para a classificação da fibrilhação auricular foi retirada da *PhysioNet Challenge 2017 Database* [Clifford et al., 2017]. Os dados do ECG são gravados por um dispositivo da *AliveCor* [AliveCor, 2020], todos os registos de



(a) Sinal do ritmo normal.



(b) Sinal de Fibrilhação auricular.

Figura 3.2: Amostra dos sinais normal e FA.

ECG têm uma duração entre 9 e 61 segundos. Este faz o registo do ECG de uma única *lead* (*short single lead ECG*), isto faz aumentar a complexidade do método de deteção da fibrilhação auricular, pois normalmente os sinais do ECG são registados com 12 *leads* por um período mais longo. Os dados foram digitalizados pela *AliveCor* em tempo real a 44,1kHz e resolução de 24 bits por *software* de *demodulation*. Por último, os dados foram armazenados com 300 *samples* por segundo, em ficheiros de 16 bits com um filtro passa-banda de 0,5-40 Hz e um intervalo dinâmico de ± 5 [mV] [Clifford et al., 2017]. O conjunto de dados utilizados contém 8528 gravações de ECG. Destas gravações foram utilizadas 1500 gravações de ECG normais e 500 gravações de fibrilhação auricular, com comprimentos a variar entre 2808 e 18000 amostras, para fins de treino e teste, ignorando outras entradas de dados irrelevantes. Embora a análise pudesse ter sido realizada com um conjunto de dados de séries temporais de comprimento fixo, pode-se mostrar como as operações informativas existentes na biblioteca utilizada podem ser recuperadas mesmo no caso em que os registos das séries temporais são de comprimentos muito diferentes. Isto significa, que as operações conseguem relacionar as séries temporais mesmo estas tendo comprimentos diferentes.

As etiquetas (*labels*) identificam a doença correspondente a cada série temporal, estes são fornecidos por um especialista na área e considero-o como a verdade fundamental no âmbito desta pesquisa.

Sinais de gravação de dois pacientes, um com ritmo normal e outro com fibrilhação auricular, estão representados na Figura 3.2. Podemos observar que as ordenadas representam o número de amostras obtidas. O ECG normal, apesar de algum ruído nos primeiros dez segundos, apresenta uma distância quase constante entre os picos, que representa o intervalo RR. O ECG de FA, por outro lado, apresenta intervalo RR inconstante, que é uma característica significativa doença.

Estes sinais de ECG vão ser analisados como séries temporais, isto é, são medições que foram efectuadas durante um período de tempo. Normalmente, na área da medicina, medidas de entropia, como entropia de amostra, são muito usadas nas análises médicas das series temporais. De seguida, é descrito o método usado para fazer a extração de *features* das

séries temporais.

3.2 Extração de *Features*

Os classificadores de *machine learning* requerem quase sempre a definição de um vetor de *features* para descrever o batimento do ECG. Cada batimento é composto por múltiplas ondas que descrevem diferentes ciclos cardíacos que, por sua vez, podem descrever diferentes tipos de *features*. Para isso, existem *features* que se baseiam na descrição da forma de onda do ECG, designadas *features* morfológicas, estas descrevem os batimentos cardíacos com base nas observações do próprio sinal [Yeap et al., 1990], [Hu et al., 1993]; *features* retiradas do intervalo dos batimementos cardíacos, *features* baseadas na frequência e polinômios de *Hermite* [Lagerholm et al., 2000], entre outros.

Um dos problemas na extração de *features* é a proporção entre a quantidade de dados de treino disponíveis e o número de *features* extraídas, que, por vezes, é muito pequena, isto é, pode ocorrer overfitting.

O processo de extração de *features* é uma parte crucial da aprendizagem supervisionada, onde diversos pontos podem fornecer *features* exclusivas para diferentes classes de sinais de entrada. Com isto, o objetivo da aprendizagem supervisionada é trabalhar com um conjunto de dados de treinamento rotulados, onde existem pares de entrada e saída desejados.

Neste estudo, a abordagem utilizada na extração de *features* permite fazer comparações de diversos dados e métodos científicos, permitindo, assim, organizar os conjuntos de dados das séries temporais automaticamente de acordo com as suas propriedades. Esta abordagem é totalmente automatizada e não usa nenhum conhecimento prévio do domínio sobre as séries temporais ou qualquer informação sobre os pressupostos teóricos subjacentes aos métodos de análise: simplesmente usa o comportamento empírico dos métodos e séries temporais como uma plataforma de comparação.

3.2.1 Highly Comparative Time-Series Analysis (HCTSA)

A extração de *features* é feita usando a biblioteca HCTSA [Fulcher, 2013-2020]. Este método de análise de séries temporais assume uma variedade de formas, desde estatísticas de resumo simples até ao ajuste de modelos estatísticos. Estes modelos são implementados num único algoritmo: uma operação que resume uma entrada de série temporal num único número real. Cada operação, ρ, é um algoritmo que utiliza uma série temporal, $x = (x1, x2 \ldots, xi)$, como entrada e gera um único número real, ou seja, $\rho : \mathbb{R}^N \to \mathbb{R}$. A saída de uma dessas operações vai ser designada por *feature*. Portanto, uma série temporal é designada por x, que é composta por xi (amostra de um instante dentro da uma série temporal), mas x_i também é uma série temporal.

A biblioteca contém mais de 9000 operações que quantifica uma ampla gama de propriedades de séries temporais, incluindo: estatísticas básicas de distribuição (localização,

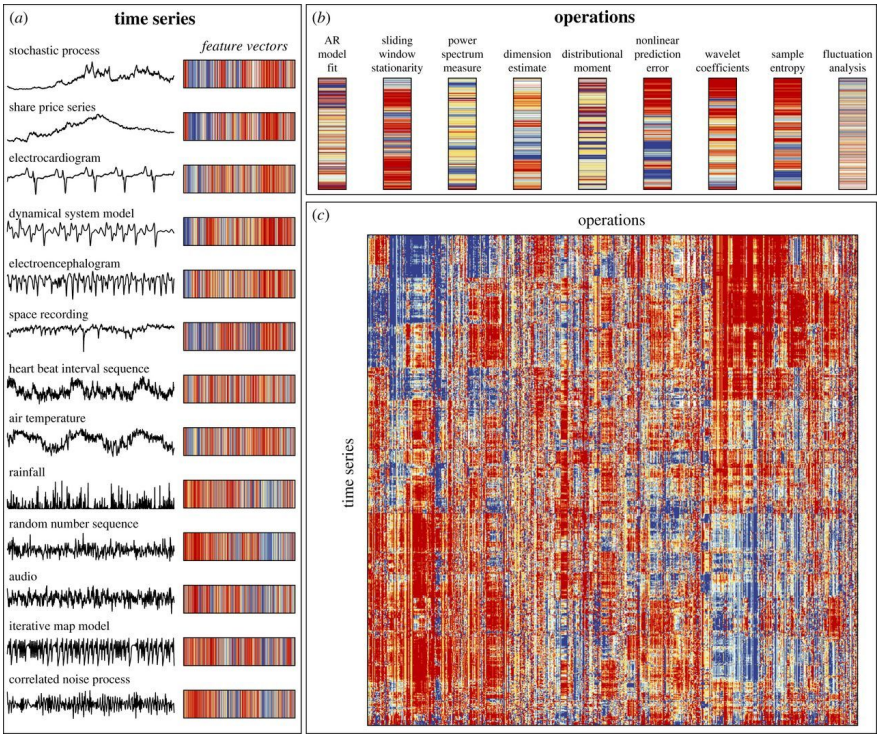


Figura 3.3: Representação de séries temporais e métodos de análise [Fulcher et al., 2013].

propagação, Gaussianas, entre outras); correlações lineares (por exemplo, autocorrelações e características de espetro de potência); estacionárias (*StatAv*, testes de raiz unitária, erros de previsão); informações teóricas e medidas de entropia (informação mútua automática, entropia aproximada e complexidade $Lempel - Ziv$); análise de série temporal não linear (por exemplo, dimensão de correlação, estimativas de expoente de *Lyapunov*); ajustes de modelo linear e não linear (qualidade de ajuste e valores de parâmetro de média móvel autorregressiva, processos gaussianos e modelos de espaço de estados); e outros (por exemplo, métodos *wavelet*, propriedades de redes derivadas de séries temporais, entre outros) [Fulcher, 2020].

A Figura 3.3 é uma representação do modo como a análise de séries temporais é efetuada. Com base na figura 3.3 , de seguida explica-se o processo de introdução das séries temporais na biblioteca hctsa. As séries temporais dos ECG (2000 séries, neste trabalho) vão ser colocadas num ficheiro (de formato ".mat" do Matlab) que vai conter três variáveis: *timeSeriesData*, *labels* e *keywords*. A primeira vai corresponder a uma célula Nx1 (N corresponde ao número de ECGs - 2000) onde cada elemento contém um vetor de valores de uma série temporal (Figura 3.3 a); A segunda é uma célula Nx1 que especifica o nome de cada uma das séries temporais; por último, a terceira vai conter uma palavra-chave que identifica o tipo de cada série temporal. As operações são representadas como vetores de *features* (neste caso 7702 features utilizadas) que contêm as saídas num conjunto de dados de série temporais (Figura 3.3 b). De seguida, é feita a análise do resultado da aplicação de um grande conjunto de operações ao conjunto de séries temporais utilizadas. Este cálculo pode ser visualizado como uma matriz de dados, onde as séries temporais representam as linhas e as operações as colunas, isto é, cada elemento da matriz, D_{ij}, contém a saída de

uma operação, F_j, aplicada a uma série temporal, x_i, de modo que $D_{ij} = F_j(x_i)$. A figura 3.3 c representa este procedimento.

As operações são selecionadas de uma forma completamente automatizada, mas cada operação selecionada fornece uma compreensão do conjunto de dados, contribuindo com uma medida interpretável da diferença estrutural entre as duas classes utilizadas. Por exemplo, os resultados das séries de ritmo normal tendem a ter intervalos entre batimentos mais longos e maior entropia do que séries de fibrilhação auricular.

A matriz utilizada vai conter nas colunas as operações e nas linhas as séries temporais. A matriz final obtida através da análise efetuada é chamada matriz de *features*, contém 7702 *features* (colunas) que foram obtidos através de cada operação efetuada a cada uma das séries temporais (linhas). Posteriormente, faz-se um processamento das *features* calculadas, isto é, muitas das operações efetuadas podem fornecer resultados que não são números reais ou determinada série temporal pode ser inadequada para a operação. Portanto, é necessário a utilização de uma transformação que permita comparar as séries temporais de forma significativa, isto é, ao calcular distâncias entre vetores de *features* de séries temporais, o intervalo de resultados de todas as operações deve ser semelhante, de modo a que seja possível ponderá-las de igual maneira, independentemente dos intervalos dos resultados obtidos. Tendo em conta isto, foi usada uma transformação sigmoidal com *outliers* [Fulcher et al., 2013]:

$$\widehat{f} = \left\{1 + exp\left[-\frac{f - median(f)}{1.35 \times iqr(f)}\right]\right\}^{-1}, \tag{3.1}$$

onde $\widehat{f}$ representa as saídas normalizadas (0 e 1) de uma dada operação em todas as séries temporais; f representa as saídas D_{ij}; $median(f)$ representa a mediana da amostra de f e $iqr(f)$ é o intervalo interquartil. A constante 1.35 é escolhida de maneira a que, para um f com distribuição gaussiana, este seja equivalente à função sigmóide logística [Bishop, 2006]:

$$\widehat{f} = \left\{1 + exp\left[-\frac{f - \mu_f}{\sigma_f}\right]\right\}^{-1} \tag{3.2}$$

que usa a média, μ_f em vez da mediana, e o desvio padrão σ_f, no lugar do intervalo interquartil.

Depois de aplicar a transformação, explicada na equação 3.1, o resultado é linearmente redimensionado para um intervalo de unidade de modo a que cada operação tenha a mesma normalização na faixa de saída: entre 0 e 1. A matriz de *features* resultante vai conter 6679 *features* normalizadas devido a terem sido eliminados os valores das operações que forneceram resultados que não são números reais ou as séries temporais que não são adequadas para a operação. Posteriormente, é feita a seleção das melhores *features* através da *Feast Toolbox* (*Feature Selection Toolbox*), explicada na Secção 3.3.1, com o objetivo de reduzir a matriz obtida anteriormente.

3.3 Seleção de *Features*

Normalmente, um conjunto de dados de grande dimensão é um desafio para o *machine learning*. Algumas das aplicações práticas mais relevantes podem facilmente ter mais de 10000 *features*. Portanto, fazer extração de um elevado número de *features* pode levar a uma carga computacional muito elevada e, por vezes, algumas dessas *features* podem ser irrelevantes para a tarefa em questão ou redundantes no contexto de outras. Por isso, é necessário a existência de um método automático que selecione subconjuntos significativamente menores dessas *features*.

No caso em estudo, foram extraídas inúmeras *features* das séries temporais utilizadas. Sabe-se que para se obter bons resultados é necessário reduzir a quantidade de *features* existentes. Para isso, foi utilizado um algoritmo de seleção de *features*, *Feast Toolbox*, cujo objetivo é reduzir o conjunto de dados obtido através da extração efetuada na Secção 3.2.1.

3.3.1 *Feast Toolbox*

O *Feast*(*Feature Selection Toolbox*) fornece implementações de algoritmos de seleção de *features* comuns com filtros baseados na informação mútua e uma implementação de *RELIEF* [Brown et al., 2012]. Pode-se definir a informação mútua como a diferença entre a entropia e a entropia condicional para um par de variáveis, I(X;Y). A Entropia é designada por $H(x)$, para uma variavél aleatória X e a entropia condicional é definada por $H(X|Y)$.

A entropia de uma variável aleatória X mede a incerteza sobre o estado de uma amostra x de X. A entropia de X é definida em termos da distribuição de probabilidade $p(x)$ sobre os estados de X como mostra a equação 3.3.

$$H(X) = -\sum_{x \in X} p(x) log(p(x)). \tag{3.3}$$

A entropia condicional X em Y, mede a incerteza esperada do estado da amostra x quando Y é conhecido. É calculada a média dos possíveis estados de Y, de modo a que exista uma medida útil quando Y é desconhecido. Podem-se utilizar duas definições equivalentes, em termos da distribuição de probabilidade conjunta $p(x, y)$,

$$H(X|Y) = -\sum_{y \in Y} p(y) H(X|Y = y) \tag{3.4}$$

$$= -\sum_{x \in X} \sum_{y \in Y} p(x, y) log\ p(x|y). \tag{3.5}$$

Portanto, pode-se afirmar que a informação mútua mede a redução média da incerteza no estado de X quando o estado de Y é conhecido e, portanto, existe um aumento da informação. A informação mútua é uma medida simétrica, em que $I(X;Y) = I(Y;X)$, ou seja, a informação ganha sobre X quando Y é conhecido, e é igual à informação ganha sobre

Y quando X é conhecido, representado pela equação 3.6 [Brown et al., 2012].

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X). \tag{3.6}$$

A informação mútua também pode ser expressa pela distribuição conjunta $p(x,y)$ e o produto de ambas as distribuições marginais $p(x)p(y)$, que são definidas da seguinte maneira:

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} p(x,y)log\ \frac{p(x,y)}{p(x)p(y)}. \tag{3.7}$$

Através da equação 3.6 pode-se observar como a informação mútua consegue atingir os seus valores máximos e mínimos com mais facilidade. O valor máximo é o mínimo das duas entropias $H(X)$ e $H(Y)$, ocorre quando o conhecimento de uma variável permite uma previsão perfeita do estado da outra. Em relação à equação 3.7, isto vai acontecer devido a $p(x,y)$ ficar igual a $p(x)$ ou $p(y)$ para todos os valores [Brown et al., 2012].

O *Feast* formula a tarefa de seleção de *features* como um problema de probabilidade condicional, estabelecendo ligações precisas entre as funções de probabilidades e a heurística de seleção de *features* atuais dos critérios de informação mútua. Assume que existe um processo subjacente $p : X \to Y$, do qual temos uma amostra de N observações, onde cada observação é um par (x,y), que consiste num vetor de *features* d-dimensional $x = [x_1,\ldots,x_d]^T$, e uma classe-alvo y, extraída das variáveis aleatórias subjacentes $X = \{X_1,\ldots,X_d\}$ e Y. Além disso, assume que $p(y|x)$ é definido por um subconjunto das *features* d em x, enquanto as restantes *features* são irrelevantes [Brown et al., 2012].

O *Feast* inclui inúmeros métodos de seleção, mas para este estudo foram obtidos resultados para sete desses métodos, que serão descritos a seguir. Para a seleção das *features* é utilizado uma matriz que contém todas as *features* extraídas e um vetor com os *labels* associados.

Os métodos de filtro são definidos por um critério J, também referido como um "índice de relevância" ou critério de "pontuação" que se destina a medir o grau de potencial de uma *feature* ou subconjunto de *features* quando usadas num classificador [Brown et al., 2012]. Para um *label* de classe Y, a pontuação de informações mútuas para uma *feature* X_k é representada pela equação 3.8.

$$J_{mim}(X_k) = I(X_k;Y). \tag{3.8}$$

Este processo, que considera uma pontuação para cada *feature* independentemente das outras, tem sido usada diversas vezes na literatura, por exemplo, [Lewis, 1992]. Designa-se este critério como MIM (*Mutual Information Maximisation*) e é utilizado para classificar as *features* na ordem da pontuação MIM, e selecionar as principais *features* - K, onde K vai ter uma pré-definição de um certo número de *features* [Brown et al., 2012].

Outro critério utilizado foi o MIFS (*Mutual Information Feature Selection*), que tem

como objetivo a "redundância de relevância", isto é, um conjunto de *features* não deve ter uma elevada correlação. Este critério apresenta a seguinte equação:

$$J_{mifs}(X_k) = I(X_k;Y) - \beta \sum_{X_j \in S} I(X_k;X_j), \tag{3.9}$$

onde S representa o conjunto de *features* selecionadas. Para isso, é incluído o termo $I(X_k;X_j)$ que vai garantir a relevância da *feature*. Mas este processo vai introduzir uma penalidade para conseguir impor correlações baixas entre *features* já selecionadas em S. Com isto, verifica-se que se está a selecionar *features* sequencialmente, o que leva à construção iterativa de um subconjunto final de *features*. Por último, temos o parâmetro β que é configurável e deve ser definido experimentalmente. Em experiencias relizadas anteriormente, concluiu-se que $\beta = 0$ é equivalente a $J_{mim}(X_k)$, que seleciona *features* de forma independente, enquanto um valor maior colocará mais ênfase na redução das dependências entre *features* [Brown et al., 2012].

O próximo critério concentra-se no aumento de informações complementares entre *features* e foi proposto por [Yang and Moody, 1999]. O critério JMI (*Joint Mutual Information*) tem uma pontuação para uma *feature* X_k que é:

$$J_{jmi}(X_k) = \sum_{X_j \in S} I(X_k X_j;Y). \tag{3.10}$$

Podemos constatar que a informação entre os alvos e um conjunto de variáveis aleatória $X_k X_j$ é definido pelo emparelhamento de X_k com cada *feature* previamente selecionada. A intenção é a seguinte: se a *feature* candidata é 'complementar' às *features* existentes, devemos incluí-la [Brown et al., 2012].

O CMI (*conditional mutual information*) é outro dos critérios utilizados, este usa a pontuação para uma *feature* X_k como:

$$J_{cmi}(X_k) = I(X_k;Y|S) \tag{3.11}$$

A expressão $I(X_k;Y|S)$ significa que vai devolver a informação mútua entre X e Y, que é condicionada em S.

O critério proposto por [Peng et al., 2005] é também utilizado, que se designa por MRMR (*Minimum-Redundancy Maximum-Relevance*) e é representado pela seguinte equação:

$$J_{mrmr}(X_k) = I(X_k;Y) - \frac{1}{|S|} \sum_{j \in S} I(X_k;X_j). \tag{3.12}$$

Pode concluir-se que o MRMR tem uma crença mais forte nas suposições de independência de pares conforme o conjunto de *features* S cresce, isto é, o MRMR move-se linearmente dentro do espaço conforme o conjunto de *features* S cresce.

O critério que se segue foi proposto por [Fleuret, 2004] e baseia-se na maximização condicional da informação mútua (CMIM),

$$J_{cmim}(X_k) = min_{X_j \in S}[I(X_k; Y|X_j)] \tag{3.13}$$

que pode ser reescrita como,

$$J_{cmim}(X_k) = I(X_k; Y) - max_{X_j \in S}[I(X_k; X_j) - I(X_k; X_j|Y)]. \tag{3.14}$$

O CMIM examina as informações entre uma *feature* e o destino, condicionada a cada *feature* atual. O termo $[I(X_k; X_j) - I(X_k; X_j|Y)]$ é a informação de interação - que pode ser negativa ou positiva.

O último critério utilizado foi o DISR (*Double Input Symmetrical Relevance*), proposto por [Meyer and Bontempi, 2006],

$$J_{disr}(X_k) = \sum_{j \in S} \frac{I(X_k X_j; Y)}{H(X_k X_j Y)}. \tag{3.15}$$

Este critério usa um termo de normalização nas informações mútuas, para compensar a tendência inerente para *features* de alta raridade.

Estes critérios são usados na prática como algoritmos que fazem a seleção das melhores *features*. De cada algoritmo são selecionadas as 300 melhores da matriz de *features* inicial.

Posteriormente é desenvolvido um código em Matlab para fazer a seleção das 40 melhores entre todos os algoritmos utilizados.

3.3.2 Descrição das *Features*

Como foi visto anteriormente, através da *Feast Toolbox* foram selecionadas as 300 melhores e, posteriormente, as 40 melhores *features* para serem utilizadas. Cada uma destas tem uma propriedade associada, como foi referenciado na secção 3.2.1. De seguida, é explicado a que grupo de operação pertence cada uma das 40 *features* como ilustrado na Tabela 3.1.

O primeiro grupo refere-se à Correlação que consiste, em resumir as propriedades básicas da forma como os valores de uma série temporal são correlacionados ao longo do tempo. As *features* deste grupo fazem a análise das estatísticas sobre a função automática de informação mútua para uma série temporal, utilizam a informação mútua automática com adição de ruído e, por último, existe a análise dos componentes principais de uma série temporal em um espaço de incorporação [Fulcher, 2020].

Para além disso, outro dos grupos baseia-se na *Entropia and information theory*, onde é medida a entropia e complexidade das séries temporais. Estas *features* analisam como as propriedades da série temporal mudam com aumentos aleatórios e outras baseiam-se na entropia de multiescala de uma série temporal [Fulcher, 2020].

Tabela 3.1: Grupo de *features* utilizadas neste estudo.

Grupo	***Features***
Correlation	Co_AddNoise IN_AutoMutualInfoStats NL_embed_PCA
Entropia and information theory	EN_Randomize EN_mse
Time-series model fitting and forecasting	MF_FitSubsegments MF_GP_FitAcross MF_GARCHfit MF_GARCHcompare MF_ExpSmoothing MF_armax MF_StateSpace_n4sid FC_Surprise FC_LoopLocalSimple
Stationarity and step detection	SY_SlidingWindow SY_VarRatioTest
Nonlinear time-series analysis and fractal scaling	NL_TSTL_FractalDimensions NL_crptool_fnn NL_MS_nlpe SC_FluctAnal
Others	PP_Compare PH_Walker

Outro dos grupos de análise é o *Time-series model fitting and forecasting* que se refere ao ajuste de modelos e de previsões de séries temporais. As *features* utilizadas neste grupo baseiam-se na modelação, ajuste e comparação de modelos de séries temporais, na robustez dos parâmetros do modelo em diferentes segmentos de uma série temporal e como a previsão local depende do comprimento da janela utilizada [Fulcher, 2020].

As *features* do grupo de *Stationarity and step detection* analisam como as propriedades de uma série temporal mudam ao longo do tempo. Para isso, calculam o teste de razão da

variância para uma série e medidas da janela deslizante de estacionariedade [Fulcher, 2020]. Também é utilizado o grupo da *Nonlinear time-series analysis and fractal scaling* que contém métodos de análise de séries temporais não lineares, onde se inclui dimensões de incorporação e análise de flutuação por uma variedade de métodos. Algumas das *features* utilizadas baseiam-se nas estatísticas de análise de falsos vizinhos próximos ($NL\_crptool\_fnn$) e do erro normalizado de previsão não linear da interpolação de constante *drop-one-out* [Fulcher, 2020].

O último grupo faz a análise de outro tipo de propriedades, tais como valores extremos, gráficos de visibilidade, simulações baseadas na física e dependência de pré-processamentos aplicados a uma série temporal [Fulcher, 2020].

3.4 Classificadores

Neste estudo, foi utilizada a aplicação *Classification Learner*, disponível na *Toolbox Statistics and Machine Learning* do *Matlab*, para fazer a análise das *features* selecionadas na Secção 3.3. Foram efetuados testes em todos os 25 classificadores disponíveis na aplicação e selecionados os que produziram melhor desempenho. Para efetuar estes testes é necessário uma matriz com as *features* selecionadas e outra com os *labels* respetivos.

Após o treino dos classificadores, foram selecionados os 4 melhores classificadores com melhor desempenho para serem utilizados na avaliação do conjunto de dados de teste.

3.4.1 *Decision Tree* (Árvore de Decisão)

A *Decision Tree* é um algoritmo de aprendizagem supervisionada, em que os dados são continuamente divididos de acordo com um determinado parâmetro. As Árvores de Decisão consistem em: *nodes* que testam o valor de um determinado atributo; *branch* que correspondem ao resultado do teste e dizem se se pode ligar ao próximo nó e *leaf nodes* são os terminais que preveem o resultado (representam *labels* de classe ou distribuição de classe). Na figura 3.4 está ilustrado um exemplo deste tipo de classificação.

O classificador de *Decision Tree* encontra a relação entre o que é previsto e o resultado final. O processo inicia-se no nó raiz, onde cada passo vai descendo e selecionando as regras do comando binário seguindo os ramos até ao último nó. Mas a criação de uma *decision tree* segue algumas regras essenciais [Javatpoint, 2011-2018a]:

- O nó é puro: isto ocorre quando os nós contêm observações de uma só classe. É usado o algoritmo *Gini's Diversity Index*;
- Existem menos observações num determinado nó do que os critérios de tamanho de *leaf* mínimo definidos;
- Alcança o número máximo de divisões.

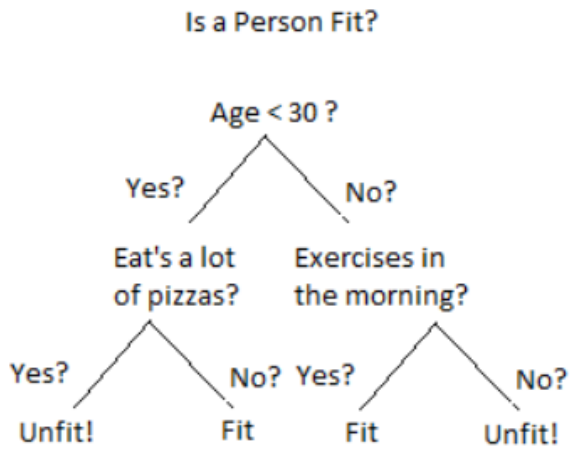


Figura 3.4: Exemplo de uma *decision tree*.

O critério utilizado para o desdobramento foi o *Gini's Diversity Index* que é representado pela seguinte equação:

$$Gini(T) = 1 - \sum_{j} p^2(j). \tag{3.16}$$

onde j representa as classes e $p(j)$ a probabilidade da classe j que segue pelo nó.

Foi utilizado o classificador *Coarse Tree* que é usado para classificar dados de ocupação, é de rápida execução, utiliza pouca memória e é de fácil interpretação. O *Coarse Tree* tem poucas *leafs* com número máximo de divisões situado em quatro [Javatpoint, 2011-2018a].

3.4.2 *Logistic Regression*

A regressão logística é um algoritmo de aprendizagem supervisionada, usado principalmente para problemas de classificação binária, isto é, conjuntos de dados onde $y = 0$ ou 1, e 1 denota a classe padrão. Com a ajuda da regressão logística, obtemos uma classificação categórica que resulta na saída pertencente a uma das duas classes. Embora "regressão" contradiga "classificação", o foco aqui é na palavra "logística" que se refere à função logística que faz a tarefa de classificação neste algoritmo.

A regressão logística tem dois componentes: hipótese e curva sigmóide. Com base nessa hipótese, pode-se derivar a probabilidade resultante do evento. Os dados obtidos a partir da hipótese são então ajustados à função *log* que cria uma curva em formato de S chamada 'sigmóide'. Por meio dessa função *log*, pode-se determinar a categoria à qual pertencem os dados de saída [Javatpoint, 2011-2018b]. A curva sigmóide pode ser visualizada na Figura 3.5.

Podemos escrever a equação logística associada à figura 3.5 como,

$$h(x) = \frac{1}{1 + e^{-x}}. \tag{3.17}$$

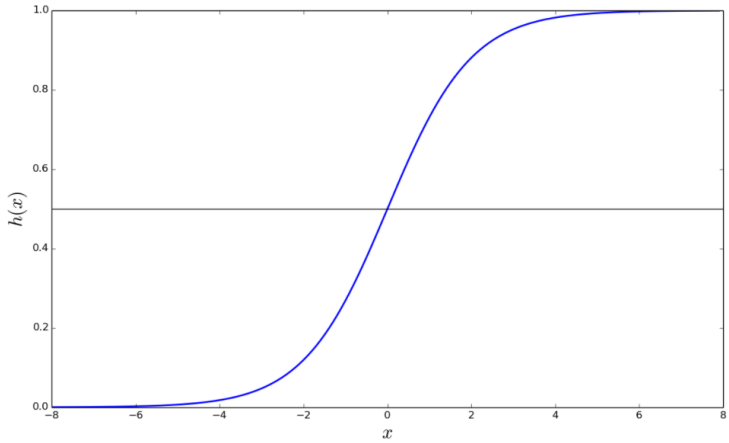


Figura 3.5: Curva sigmóide com formato em S.

Na regressão logística, a saída assume a forma de probabilidade da classe padrão. Por ser uma probabilidade, a saída encontra-se na faixa de $[0, 1]$. A saída, valor y, é criada pelo *log* da transformação do valor x, utilizando a função logística. Um limite é então aplicado para forçar essa probabilidade numa classificação binária [Javatpoint, 2011-2018b].

3.4.3 *Ensemble learning*

Os métodos de *ensemble* são uma técnica de *machine learning* que combinam vários modelos de base para produzir um modelo preditivo ideal. As técnicas *ensemble* procuram as vantagens individuais de cada classificador, combinando-as de forma a obter uma melhor solução. As *Decision Trees* são particularmente adequadas para trabalhar com métodos de *ensemble* porque são rápidas e instáveis [Gashler et al., 2008]. *Decision trees* podem ser instáveis porque pequenas variações nos dados podem resultar no desenvolvimento de uma árvore completamente diferente. Ao usá-las em *ensemble*, esse problema é atenuado.

Boosting e *bagging* são técnicas de modelo de acumulação que recuperam os dados de treino para criar novos modelos para cada amostra desenhada. Neste estudo foram usados dois classificadores baseados no modelo *Boosting ensemble*:

Ensemble Boosted Trees: O *AdaBoost* é um modelo de *ensemble* usado para impulsionar o desempenho das *decision trees* e de outros classificadores, e é baseado em problemas de classificação binária. Este vai aprender com erros anteriores, como, por exemplo, pontos de dados de classificação incorreta. Esta aprendizagem faz aumentar o peso dos pontos de dados classificados incorretamente.

Este método é muito utilizado na classificação binária, cujo o algoritmo treina os *learners* sequencialmente. Para cada *learner* com índice t, o *AdaBoost* calcula o erro de classificação ponderado,

$$\varepsilon_t = \sum_{n=1}^{N} d_n^{(t)} | (y_n \neq h_t(x_n)), \tag{3.18}$$

onde x_n é um vetor de valores preditores para a observação n; y_n é o verdadeiro *label* da

classe; h_t é a previsão do *learner* com índice t e $d_n^{(t)}$ é o peso da observação n na etapa t. De seguida, aumenta os pesos das observações classificadas incorretamente pelo *learner* t; e reduz os pesos das observações classificadas corretamente pelo aluno t. O próximo aluno $t+1$ é então treinado nos dados com pesos atualizados d_n^{t+1}. Após o fim do treino, o *AdaBoost* calcula a previsão de novos dados através da equação seguinte:

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x), \tag{3.19}$$

onde $\alpha_t = \frac{1}{2} log \frac{1-\varepsilon_t}{\varepsilon_t}$ são os pesos das hipóteses fracas do conjunto [The MathWorks, 1994-2020b].

Ensemble RUSBoosted Trees: O *RUSBoosted* é especialmente eficaz na classificação de dados desequilibrados, isto é, uma das classes nos dados de treino tem muito menos elementos do que a outra. O algoritmo usa N, o número de elementos na classe com o menor número de elementos nos dados de treino, como unidade básica para amostragem. Classes com mais elementos são amostradas, ficando apenas N observações de cada classe. Por outras palavras, se existir K classes, então, para cada *learner* fraco no conjunto, este vai escolher um subconjunto dos dados com N observações para cada uma das classes K. O procedimento de *boosting* segue o procedimento de *Adaptive Boosting* para *Multiclass Classification*, o qual vai reponderar e construir o conjunto *ensemble* [The MathWorks, 1994-2020b].

O *Adaptive Boosting* para *Multiclass Classification* é uma extensão do *AdaBoost* para várias classes. Em vez do erro de classificação ponderada, este método usa pseudo-perda ponderada para observações N e classes K tal que,

$$\varepsilon_t = \frac{1}{2} \sum_{n=1}^{N} \sum_{k \neq y_n} d_{n,k}^t (1 - h_t(x_n, y_n) + h_t(x_n, k)), \tag{3.20}$$

onde $h_t(x_n, k)$ é a confiança da predição pelo *learner* na etapa t na classe k, variando entre 0 (nada confiante) e 1 (altamente confiante) [The MathWorks, 1994-2020b].

Interpretar a pseudo-perda é mais difícil do que o erro de classificação, mas a ideia é a mesma. A pseudo-perda pode ser usada como uma medida da precisão da classificação de qualquer *learner* num conjunto.

Capítulo 4

Testes e Análise de Resultados

Foram efetuados diversos estágios até chegar à parte da classificação. Primeiro, foi feita a introdução das séries temporais dos 2000 ECG's e a extração das *features* através da biblioteca HCTSA. De seguida, foi utilizada a *Feast Toolbox* para a seleção das melhores *features*, que serão introduzidas nos classificadores. Para ser feita essa introdução, foi necessário dividir o conjunto de dados final, em conjunto para treino e conjunto para teste. Será avaliada a eficácia da classificação geral, isto é, conjunto de deteção tanto da FA como do ritmo normal, e a eficácia individual na deteção da fibrilhação auricular dos classificadores utilizados.

Os classificadores utilizados são os presentes na aplicação *Classificiation Learner* do MATLAB, que contém 25 classificadores, divididos por diferentes grupos. Esses grupos são classificadores baseados em: *Decision Tree*, *Discriminant Analysis*, *Logistic Regression* , *Naive Bayes*, *SVM*, *KNN*, por último, *Ensemble*.

4.1 Dataset

Para a realização dos testes, foi criado um *dataset* derivado das 2000 séries temporais, normais e fibrilhação auricular. Este *dataset* é divido de forma aleatória em conjunto de treino e conjunto de teste. Vão conter, respetivamente, 1600 e 400 séries temporais, as respetivas *features* e os *labels* correspondentes para cada uma das classes representadas (Normal e FA), que estão contidas numa matriz. Como o objetivo principal é determinar a eficácia do classificador na deteção de Fibrilhação auricular, o ritmo normal é usado para poder ser feito comparações sobre a *sensitivity* na deteção individual da FA.

O conjunto de treino vai ser inserido num classificador. Para isso, é necessário utilizar a parte correspondente aos dados do conjunto de treino, este vai conter as 40 *features* selecionadas anteriormente e os *labels* correspondentes. Foram usadas as 40 melhores, devido a vários testes efetuados e sendo que 40 é o melhor número de *features* para ser utilizado na classificação.

O conjunto de teste vai ser utilizado para a validação do modelo de classificação com melhor eficácia, o qual, representa os dados que contêm os *labels* que vão ser usados para

verificar se o classificador prevê o *label* correto.

4.2 Análise do Modelo de Classificação

Conforme mencionado anteriormente, esta pesquisa visa encontrar o melhor método para a classificação da Fibrilhação auricular. Portanto, diferentes classificadores foram testados a fim de determinar quais resultam na maior eficácia de classificação. Para chegar a uma conclusão válida e selecionar um classificador, todos foram executados para as mesmas condições. Além disso, o desempenho de cada classificador foi medido através da *accuracy* da classificação dos dados de teste, de acordo com a equação 2.1.

Por vezes, a eficácia não reflete o desempenho da classe através de um pequeno número de observações, pois é a proporção de classificações corretas nesta classe que depende do número de observações. Portanto, outros parâmetros também usados, como a *Sensitivity*, *Specificity* e Área sobre a curva ROC (AUC), que são representados na curva ROC(*Receiver Operating Characteristic*) são aplicados na avaliação de desempenho. O parâmetro verdadeiro positivo e falso positivo são duas variáveis presentes na curva ROC. Esta é traçada com base nessas duas variáveis em limiares diferentes, o que ilustra a capacidade de diagnóstico de um classificador binário em todos os limiares de distinção possíveis. AUC é um parâmetro para medir o desempenho do classificador quando o limiar de discriminação é alterado.

O objetivo deste estudo consiste em obter um modelo final com boa classificação. Para isso, é necessário escolher um bom método de validação, para que seja possível avaliar a eficácia prevista dos classificadores selecionados. A validação determina o desempenho do modelo em novos dados, em comparação com os dados de treino e ajuda a escolher o melhor modelo, e por último, também protege contra o *overfitting*. Por isso, é necessário escolher um método de validação antes de treinar qualquer classificador, para que seja possível comparar todos com o mesmo modelo atribuído.

Devido ao conjunto de dados ser pequeno, é importante levar em consideração o efeito do ajuste dos parâmetros da classificação em torno de 10% do total de dados. Embora isso possa derivar em resultados de alta precisão, eles não são um reflexo verdadeiro do desempenho do classificador. Mais especificamente, enquanto certos parâmetros podem aumentar a eficácia da classificação de um pequeno conjunto de dados de teste, o mesmo não é verdade para outros conjuntos. Portanto, a fim de evitar essas armadilhas, uma técnica de *N-fold cross-validation* foi usada para determinar a eficácia da classificação de cada classificador. Esta técnica é normalmente usada quando o tamanho dos dados de teste é limitado, para garantir uma estimativa estável e confiável do desempenho do modelo. A aplicação utilizada para a classificação contém como opção escolher o método *10-fold cross-validation* antes do início do treino do conjunto de dados selecionado.

De seguida, o conjunto de dados de treino já criado e composto por uma matriz com 40 *features* e pelos *labels* associados a cada classe (normal e FA) é colocado na aplicação

Tabela 4.1: Treino - Eficácia dos 4 melhores classificadores.

	Classificadores			
	Coarse Tree	Logistic Regression	Ensemble Boosted Trees	Ensemble RUSBoosted Trees
ACC	89.9%	89.6%	**90.1%**	89.6%

Classification Learner. Posteriormente, vão ser corridos os 25 classificadores existentes e destes foram escolhidos os 4 melhores, como mostra a tabela 4.1.

Pode-se verificar, através da Tabela 4.1, que o método *Ensemble Boosted Tree* apresenta o melhor resultado na classificação final, 90.1%. A Figura (4.1) representa a matriz de confusão e a Figura 4.2 a curva ROC para este classificador.

Na matriz de confusão podemos ver a relação entre as classes verdadeiras (linhas) e as classes previstas (colunas). Como foi usado o método *10-fold cross-validation*, a matriz de confusão é calculada através das previsões das observações realizadas. As células na diagonal correspondem à classe verdadeira e à classe prevista. Se essas células diagonais forem azuis, a classificação feita às observações dessa classe é verdadeira.

Observa-se que o desempenho do classificador para a Taxa Positiva Verdadeira (TPR), que significa proporção de observações classificadas corretamente por classe verdadeira, é de 87.5% para a Fibrilhação auricular e 90.9% para o ritmo normal. Para um Taxa Negativa Falsa (FNR), que significa proporção de observações classificadas incorretamente por classe verdadeira, é de 12.5% para fibrilhação auricular e de 9.1% para ritmo normal.

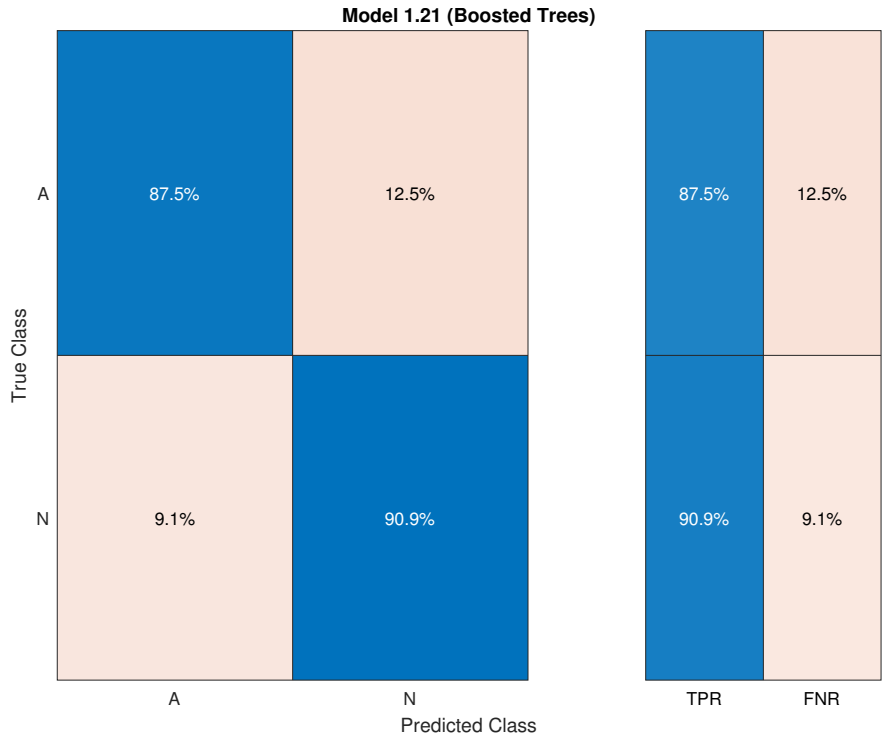
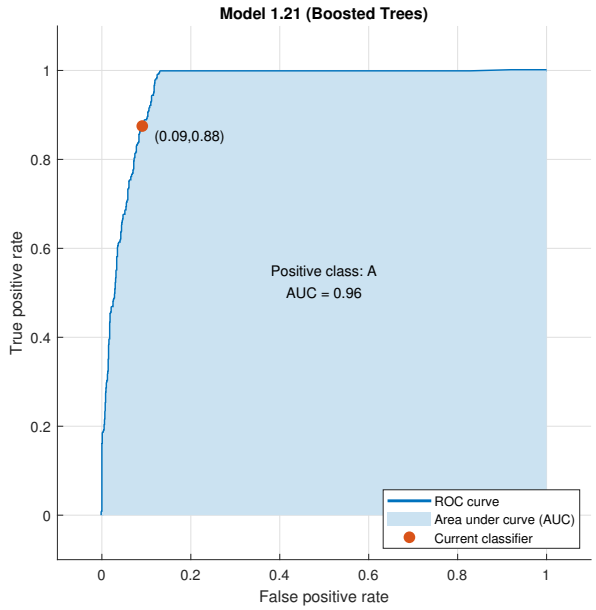


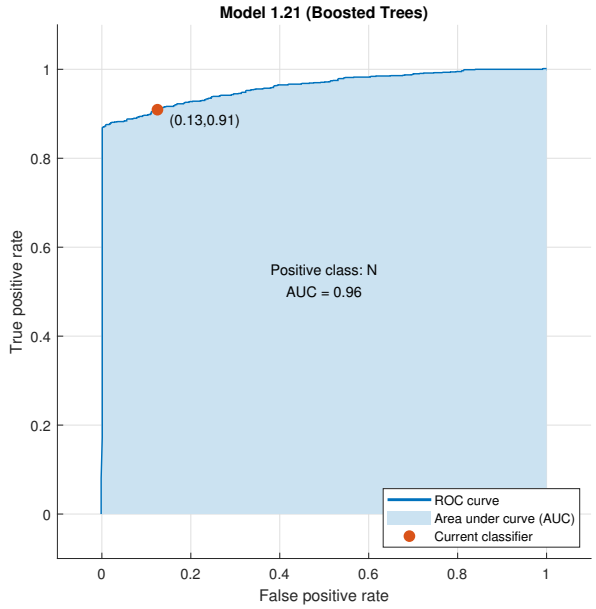
Figura 4.1: Matriz de confusão para o resultado do treino com *Ensemble Boosted Tree*.

Em relação à curva ROC, podemos observar a representação da taxa de verdadeiro positivo *versus* a taxa de falso positivo para o treino do classificador. A marca laranja, presente

no gráfico, corresponde ao desempenho da classe positiva selecionada (A (Fibrilhação auricular) ou N (ritmo Normal)). Essa marca mostra os valores da taxa de falsos positivos (FPR) e da taxa de verdadeiros positivos (TPR). O número de AUC significa a medida da qualidade geral do classificador. Valores maiores de AUC indicam melhor desempenho do classificador. Visualizando o AUC nas curvas ROC representadas podemos observar que têm um valor elevado.



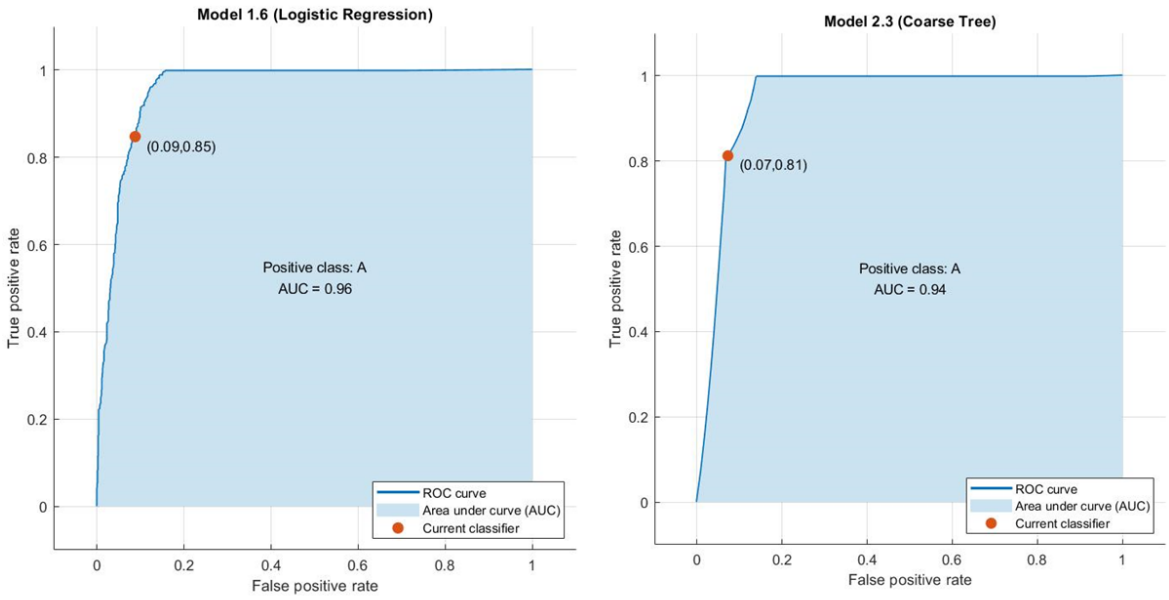
(a) Curva ROC para fibrilhação auricular.



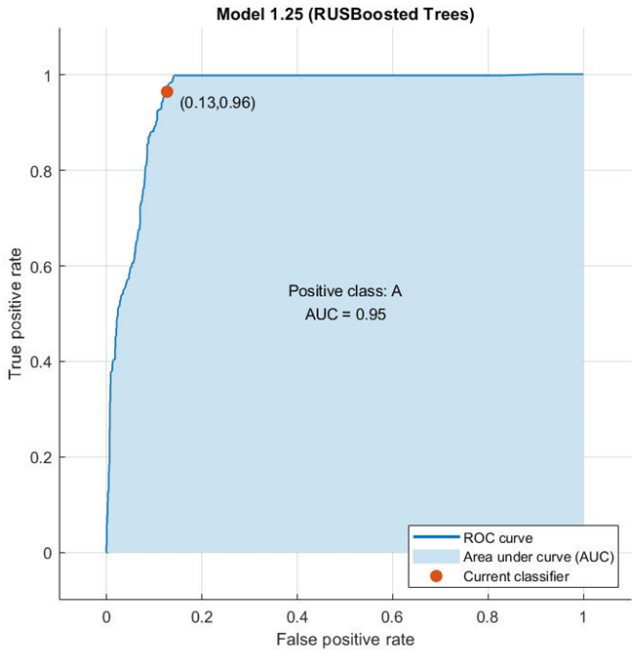
(b) Curva ROC para ritmo normal.

Figura 4.2: Representação dos gráficos das curvas ROC para cada uma das classes.

A figura 4.3 mostra as curvas ROC dos outros classificadores.



(a) Curva ROC para fibrilhação auricular dos métodos *Logistic Regression e Coarse Tree.*



(b) Curva ROC para Fibrilhação auricular para o método*Ensemble RUSBoosted Trees.*

Figura 4.3: Representação dos gráficos das curvas ROC para uma classe.

De seguida, os 4 melhores modelos da classificação do conjunto de treino são exportados e utilizados no conjunto de teste. Os valores da *accuracy* obtidos estão representados na Tabela 4.2. Ao analisar a Tabela 4.2 podemos observar que o método *Ensemble RUSBoosted*

Tabela 4.2: Teste - Eficácia dos 4 melhores classificadores.

	Classificadores			
	Coarse Tree	Logistic Regression	Ensemble Boosted Trees	Ensemble RUSBoosted Trees
ACC	91.25%	91.5%	92.75%	**93%**

Tree foi o que obteve melhor eficácia no conjunto de teste.

Sabemos que a validação de um modelo de classificação é feita através dos dados que contém os *labels* e estes são usados para verificar se o classificador prevê o *label* correto. Uma maneira de verificar esta relação é através da matriz de confusão, que exibe essas previsões dos dados e mostra a relação entre as classes previstas e as atuais. De seguida, da Figura 4.4 à Figura 4.7 são mostradas as matrizes de confusão dos 4 classificadores relativas ao conjunto de teste. Cada matriz de confusão vai conter o desempenho do classificador para a TPR e para a FNR, bem como a percentagem de valores previstos positivos e a taxa falsa de descoberta. A partir destas 4 matrizes de confusão vamos analisar a percentagem de acertos tanto na fibrilhação auricular como do ritmo normal. Estes valores vão estar representados na Tabela 4.3 e, posteriormente, será feita uma análise dos mesmos.

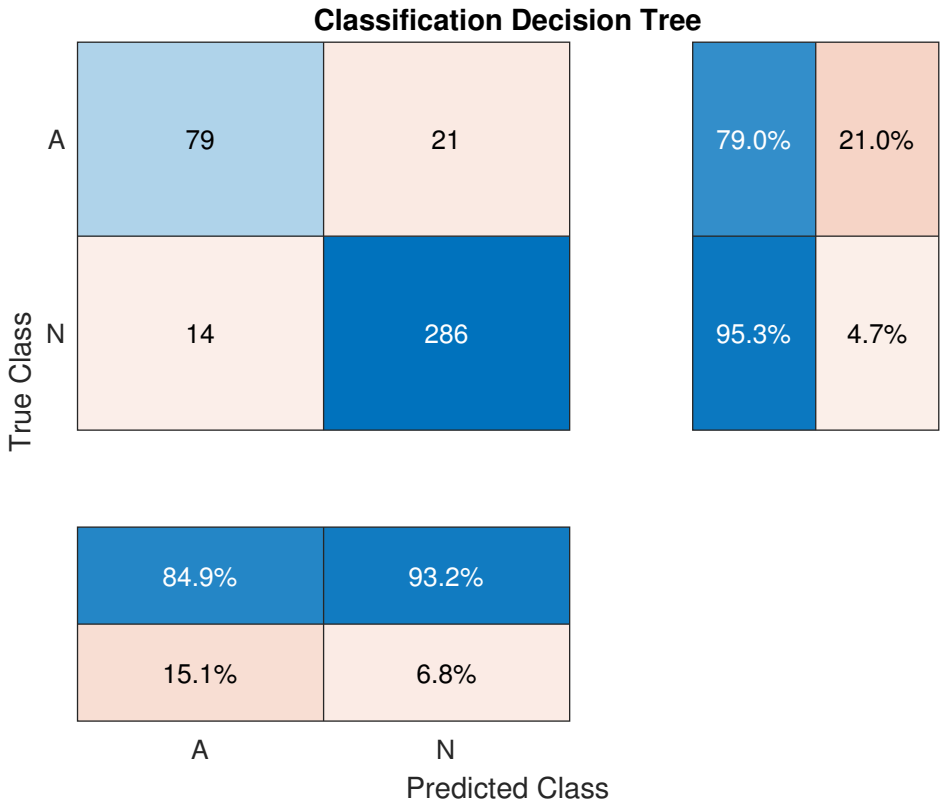


Figura 4.4: Matriz de confusão para o resultado do teste com *Coarse Tree*.

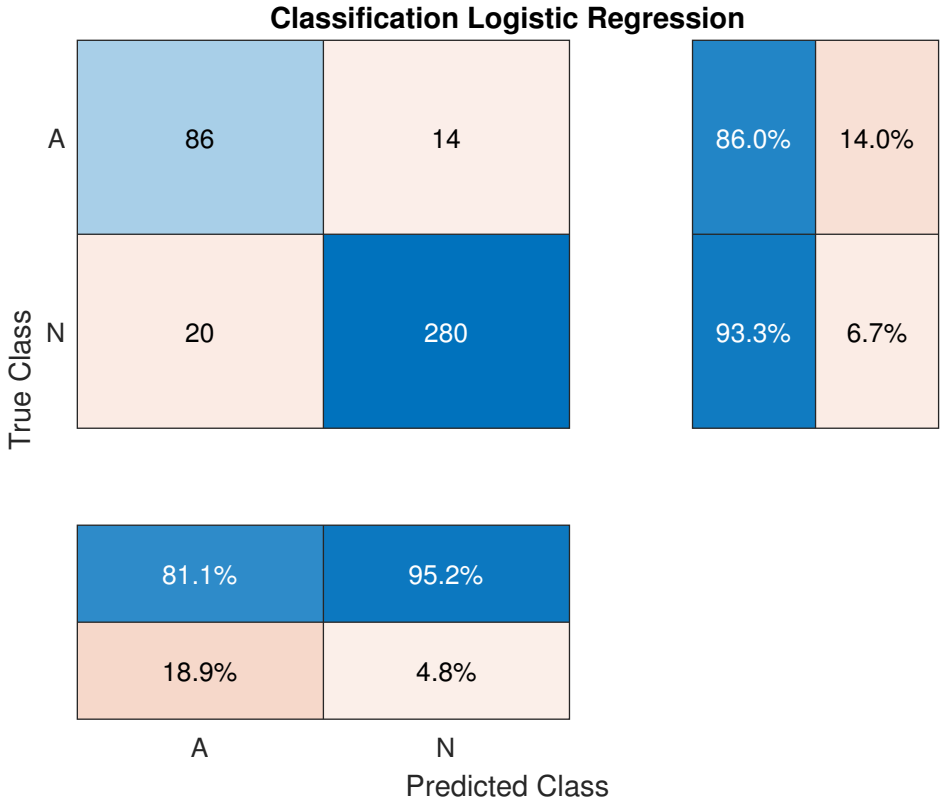


Figura 4.5: Matriz de confusão para o resultado do teste com *Logistic Regression*.

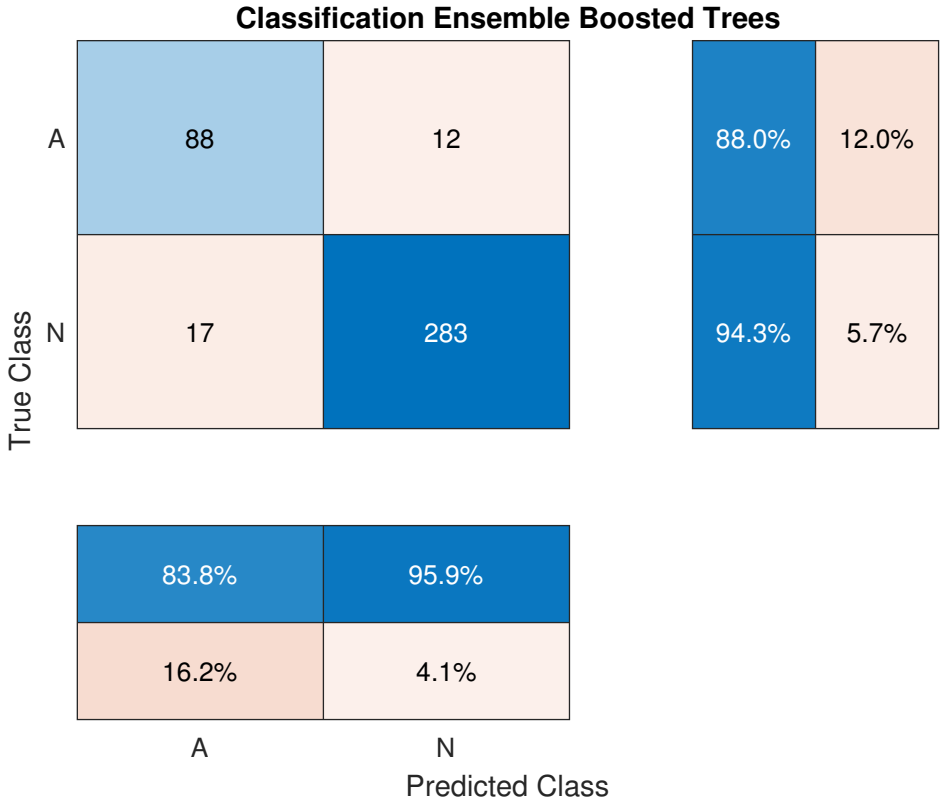


Figura 4.6: Matriz de confusão para o resultado do teste com *Ensemble Boosted Tree*.

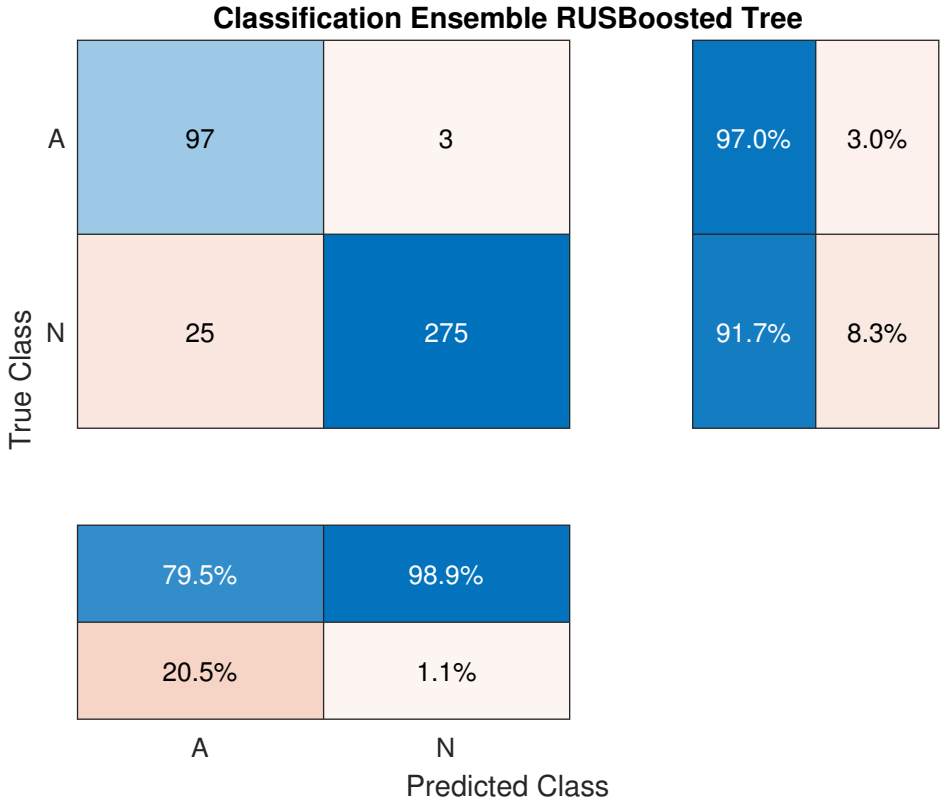


Figura 4.7: Matriz de confusão para o resultado do teste com *Ensemble RUSBoosted Tree*.

Verifica-se através da análise às matrizes de confusão representadas que a deteção da fibrilhação auricular é mais alta no método *Ensemble RUSBoosted Tree*. Das 100 amostras existentes no conjunto de teste, 97 foram identificadas corretamente, enquanto 3 foram classificadas de forma incorreta. A mesma análise pode ser feita para a deteção do ritmo normal, num conjunto de 300 amostras foram identificadas corretamente 275 e 25 são classificadas como falsos alarmes. Através da matriz de confusão da Figura 4.7 pode-se calcular as seguintes métricas:

- *Accuracy* que mostra o número de classes previstas corretamente no total das classes,

  $ACC = \frac{TP+TN}{TP+FP+FN+TN} = \frac{97+275}{97+3+25+275} = 0.93$ que corresponde a uma percentagem de 93%, como já foi mencionado na Tabela 4.2;

- *Precision* é a relação entre a os valores corretamente identificados na classe FA e todos os valores de FA existentes,

  $Precision = \frac{TP}{TP+FP} = \frac{97}{97+25} = 0.795$ que corresponde a uma percentagem de 79.5%;

- *Sensitivity* é a relação entre os valores corretamente identificados numa classe e os valores totais previstos nessa classe específica,

  $Sensitivity = \frac{TP}{TP+FN} = \frac{97}{97+3} = 0.97$ que corresponde a uma percentagem de 97.0% como é mostrado na matriz confusão;

- *Specificity* é a proporção dos verdadeiros negativos identificados corretamente por um teste diagnóstico. Sugere o quão bom é o teste na identificação de condições normais (negativas) e estima a probabilidade de pacientes sem doença poderem ser corretamente excluídos,

  $Specificity = \frac{TN}{TN+FP} = \frac{275}{25+275} = 0.9167$ corresponde a uma percentagem de 91.67% como se pode observar na matriz confusão.

- *false positive rate* (FPR) é a probabilidade de rejeitar falsamente a hipótese nula para um teste particular. A taxa de falsos positivos é calculada como a razão entre o número de eventos negativos categorizados incorretamente como positivos (falsos positivos) e o número total de eventos negativos reais (independentemente da classificação),

  $FPR = \frac{FP}{TN+FP} = \frac{25}{275+25} = 0.09$ corresponde a uma percentagem de 9% como se pode observar na matriz confusão.

Tabela 4.3: *Sensitivity* dos classificadores na deteção de FA e ritmo normal para o conjunto de teste.

Dataset	**Classificadores**			
	Coarse Tree	Logistic Regression	Ensemble Boosted Tree	Ensemble RUSBoosted Tree
FA	79%	86%	88%	**97%**
Normal	**95.3%**	93.3%	94.3%	91.7%

Foi criada também uma Tabela 4.4 com os valores individuais da deteção da FA e N para

Tabela 4.4: *Sensitivity* dos classificadores na deteção de FA e ritmo normal para o conjunto de treino.

Dataset	**Classificadores**			
	Coarse Tree	Logistic Regression	Ensemble Boosted Tree	Ensemble RUSBoosted Tree
FA	81.3%	91%	87.5%	**94.8%**
Normal	**92.8%**	85.5%	90.9%	87.8%

o conjunto de treino, sendo, assim, possível comparar os dois conjuntos. Em termos de classificação individual da fibrilhação auricular, o conjunto de teste apresenta o melhor resultado para o classificador *Ensemble RUSBoosted Tree*, comparando com os outros classificadores a diferença entre a eficácia é notória. No caso do conjunto de treino, continua a ser o mesmo classificador a obter o melhor resultado mas com uma percentagem mais baixa.

Pode-se concluir, através da análise de todos os resultados obtidos, que os métodos de classificação *ensemble* apresentam os melhores resultados em termos de *accuracy* tanto para o conjunto de treino como para o conjunto de teste.

Por último, observando as Tabelas 4.5 e 4.6 podemos constatar que o conjunto de testes obtém melhores resultados nas métricas de desempenho dos classificadores. Em termos de *Precision* e *Specificity* o classificador *Ensemble RUSBoosted Tree* apresenta melhores

Tabela 4.5: Parâmetros de avaliação dos diferentes classificadores para a fibrilhação auricular no conjunto de treino.

FA	**Classificadores**			
	Coarse Tree	Logistic Regression	Ensemble Boosted Tree	Ensemble RUSBoosted Tree
Sensitivity	81.3%	91%	87.5%	**94.8%**
Specificity	93.7%	95.0 %	95.6%	**98%**
Precision	**78.9%**	76.0 %	76.3%	72.1%

Tabela 4.6: Parâmetros de avaliação dos diferentes classificadores para a fibrilhação auricular no conjunto de teste.

FA	**Classificadores**			
	Coarse Tree	Logistic Regression	Ensemble Boosted Tree	Ensemble RUSBoosted Tree
Sensitivity	79%	86%	88%	**97%**
Specificity	**95.3**%	93.3%	94.3%	91.67%
Precision	**84.9%**	81.1%	83.8%	79.5%

resultados, mas em relação à métrica *Sensitivity* o classificador *coarse tree* apresenta o melhor resultado.

É importante referir que um dos passos mais importantes na classificação de doenças cardíacas é a seleção de *features*. Tendo em conta isto, foi criada uma tabela 4.7 com a comparação dos resultados com e sem utilização de um método para fazer a seleção de *features*. Para esta comparação é utilizada a matriz inicial de *features* que continha 6679 *features* e a matriz final já com a seleção de *features* efetuada que contém 40 *features*. Esta comparação é feita para o conjunto de teste.

Tabela 4.7: Resultados para a FA com e sem seleção de *features*.

FA	**Classificadores**			
	Coarse Tree	Logistic Regression	Ensemble Boosted Tree	Ensemble RUSBoosted Tree
Sem seleção	66%	65%	76%	92%
Com seleção	79%	86%	88%	**97%**

Podemos observar que a utilização de um selecionador de *features* é bastante necessário para conseguirmos obter resultados mais precisos. E a existência de um grande número de *features* torna o processo de classificação muito mais lento.

4.3 Comparação do Desempenho dos Métodos de Deteção de FA

A deteção automática e eficaz da FA, usando uma monitorização contínua dos ECG, pode levar a um diagnóstico prematuro e, consequentemente, fornecer mais oportunidades para um tratamento mais eficaz da doença, evitando assim as complicações crónicas da FA.

Devido a estes fatores, diversos algoritmos têm sido devolvidos nos últimos anos, para a deteção automática da FA. Estes algoritmos dependem da ausência de ondas P ou irregularidades RR ou da combinação de ambas as características para detectar episódios de FA. Mas a necessidade de detetar as ondas P e/ou pico R leva a que os desempenhos dos algoritmos estejam dependentes da precisão da etapa de deteção, que, por vezes, pode não funcionar devido à existência de picos ausentes ou classificados erradamente. Por isso, conclui-se que a classificação da FA com alta precisão é um problema desafiador.

Tabela 4.8: Diferentes metodologias de classificação de FA.

Investigadores	*Dataset*	Método	Modelos de classificação	Avaliação dos métodos
Maji et al. (2013)	*MIT-BIH arrhythmia*	Ausência e irregularidades da onda P	classificador supervisionado	*Sensitivity*- 96% ; *Specificity* - 93%
Martin Kropf et al. (2017)	*Physionet*	Ausência da onda P e intervalos RR	*Random Forest* baseados no classifiador *bagged decision tree*	*Accuraccy* - 0.83
A.Muthuchudar and S.Baboo (2013)	*MIT-BIH Arrhythmia*	Intervalos e amplitude QRS; Onda P e T; Intervalos PR e ST	Remoção de Ruído com transformada *Wavelet*; FFN com algoritmo de retropropagação	*Accuracy* – 96%
S. Dash et al.(2009)	*MIT-BIH Atrial Fibrillation* e a *MIT-BIH Arrhythmia*	Intervalos RR		*Specificity* - 95.1% ;*Sensitivity* – 94.4%
Colloca et al.(2013)	*MIT-BIH Arrhythmia* e *MIT-BIH Atrial Fibrillation* e *MIT-BIH Normal Sinus Rhythm*	Localização do ponto R; *Features* baseadas na análise do intervalo RR	SVM otimizado com *grid-search*	*Accuracy* – 85.45% ; *Specificity* - 82.9% ; *Sensitivity* – 100%
Hyun-Woo Kim et al. (2019)	*Load cell sensors and PSL-iECG2* que liga ao arduino uno	*Shannon entropy* através dos intervalo RR e Picos R; RMSSD	KNN, DT, NNs	*Accuracy* - 89.1%
Asgari et al (2015)	*MIT-BIH Atrial Fibrillation*	*Log-energy entropy , peak-to-average power ratio*	*Stationary wavelet trasform*; SVM	*Sensitivity* – 97.0% ; *Specificity* - 97.1% ; *Accuracy* - 97.1%
lJibin Wang et all. (2020)	*MIT-BIH Atrial Fibrillation*	WPT e funções de correlação aleatórios para extração	SVM; KNN; ANN	*Sensitivity* – 98.7% ; *Specificity* - 98.9% ; *Accuracy* - 98.8%
Método Proposto	*PhysioNet Challenge 2017*	HCTSA e *Feast Toolbox* para a seleção das *features*	CT; LR; Ensemble	*Precision* – 79.5% ; *Specificity* - 91.67% ; *Sensitivity* - 97%

A Tabela 4.8 ilustra as diferentes metodologias desenvolvidas na classificação da fibrilhação auricular. Inclui técnicas de pré-processamento, seleção e métodos de extração de *features*, técnicas de classificação, métodos de avaliação e métricas obtidas em cada artigo.

A partir da Tabela 4.8 observa-se que os seguintes artigos [Kim et al., 2019], [Dash et al., 2009], [Colloca et al., 2013], [Maji et al., 2013], [Kropf et al., 2017] e [Muthuchudar and Baboo, 2013] baseiam-se na identificação dos intervalos RR, localização do pico R e/ou onda P e também na análise dos intervalos QRS, ST e PR. Comparando os métodos anteriores com o que foi utilizado por [Asgari et al., 2015], nota-se que este obtém melhor resultados em termos de *Specificity*, *Sensitivity* e *Accuracy*. Isto acontece porque o método de [Asgari et al., 2015] utiliza a alta resolução de frequência de tempo da transformada *wavelet* estacionária e captura a atividade auricular calculando a razão do espectro de potência média de pico em diferentes bandas de frequência, em vez de fazer a deteção do pico R e/ou onda P ou

intervalos RR, ST e PR. Pois estes apresentam algumas limitações quer na eficiência de detecção desses parâmetros, que pode afetar amplamente o desempenho de classificação dos métodos, quer na parte do funcionamento segmentos curtos do ECG, principalmente para alguns esquemas baseados na irregularidade dos intervalos, o que pode levar à perda de segmentos de FA de curto prazo.

Outro dos métodos também representado na Tabela 4.8 é proposto por [Wang et al., 2020], e apresenta o melhor resultado na deteção da FA. Este utiliza no conjunto de dados experimentais o *10-fold cross-validation*, e a estratégia de extração das *features* provou fornecer uma forte capacidade de classificação para detetar o segmento curto do ECG de dez segundos e eliminar a necessidade de detecção de alguns parâmetros-chave, como ondas P e/ou pico R. Um dos pontos importantes deste método é a utilização da correlação entre as séries de coeficiente de *wavelet*, isto é, a regularidade do ritmo normal pode manter uma alta correlação entre os coeficientes *wavelet* correspondentes, enquanto distúrbios auriculares súbitos reduzem significativamente essa correlação. Portanto, pode-se considerar uma referência essencial para os médicos diagnosticarem a FA com mais eficácia e rapidez.

Tabela 4.9: Métodos de deteção de doenças cardíacas.

Investigadores	**Método**	**Modelos de classificação**	**Avaliação dos métodos**
J. Bogatinovski et al. (2019)	Segmentos PR através da biblioteca HCTSA	*AdaBoost* e *Gradient Boosting*	*Sensitivity* (VEB) - 0.76 e *Accuracy* (VEB) - 0.97 ; *Sensitivity* (SVEB) - 0.01 e *Accuracy* (SVEB) - 0.96
De Chazal et al.(2004)	- Intervalo RR; Intervalo de batimentos; Segmentos do ECG; Extração através de *features* morfológicas do ECG	Discriminador linear	*Sensitivity* (VEB) - 77.7% ; *Sensitivity* (SVEB) - 75.9%

A Tabela 4.9 mostra dois métodos de deteção de doenças cardíacas diferentes da fibrilhação auricular. Os dois métodos fazem a deteção dos batimentos do tipo ectópico supraventricular (SVEB) e ectópico ventricular (VEB). São mencionados devido ao método utilizado para a extração de *features*.

Como se pode observar, o método do [Bogatinovski et al., 2019] utiliza a mesma biblioteca que a utilizada nesta dissertação. O método de extração pertencem ao domínio das *features* de séries temporais globais derivados de várias áreas técnicas e científicas. Quanto ao método de [De Chazal et al., 2004], baseia-se em *features* de caráter morfológico e algumas são retiradas do intervalo RR. As *features* extraídas foram divididas em 8 grupos para examinar a performance de cada uma delas através do classificador.

O método implementado nesta dissertação utiliza a biblioteca HCTSA para fazer a extração de *features* das séries temporais. O que diferencia este métodos dos apresentados na Tabela 4.8 é o facto de não usar um método de pré-processamento para detetar intervalos, segmentos ou ondas. Isto torna-se uma vantagem pois uma deteção errada destes pode levar, por vezes, a resultados finais enganosos.

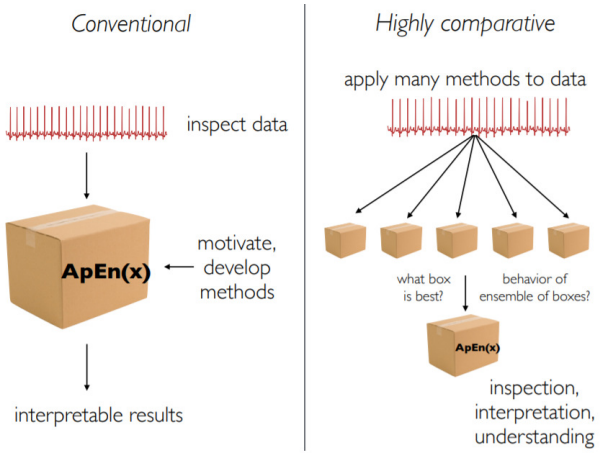


Figura 4.8: Comparação do método convencional vs o método HCTSA.

A Figura 4.8 mostra um exemplo da diferença entre o HCTSA e os métodos normalmente utilizados para a extração de *features*. Nos método designados por "convencionais" é necessário fazer uma análise da série temporal e dos diferentes parâmetros da onda e, posteriormente ser feita a extração das *features*. Por outro lado, o HCTSA utiliza normalmente a série temporal completa e extrai todas as *features* da mesma, através do conjunto de operações existente nesta biblioteca. Também é possível fazer a normalização e comparações entre as *features* extraídas.

Capítulo 5

Conclusão e Trabalho Futuro

5.1 Conclusão

A presente dissertação propõe um método de deteção da FA através de extração massiva de *features* e da utilização de classificadores baseados em *machine learning*. Este método elimina a necessidade de deteção da onda P e/ou pico R; uma etapa de pré-processamento exigida por muitos dos algoritmos existentes.

Com base nos resultados desta pesquisa, pode-se concluir que o método utilizado para o reconhecimento da fibrilhação auricular é capaz de classificar corretamente uma elevada percentagem de ECG analisados. Além disso, os dados suportam o uso de diferentes classificadores como método de classificação para reconhecimento da fibrilhação auricular. Obtendo assim um resultado de *sensitivity* de 97% para o conjunto de teste através do classificador *Ensemble RUSBoosted Tree*. Estes resultados foram baseados num pequeno conjunto de dados (2000ECGs) e, como tal, os dados não refletem de forma justa a confiabilidade desse sistema como um aplicativo do mundo real. No entanto, enquanto o procedimento de treino é estendido, o sistema deve ser aplicável a uma gama mais ampla de ECG.

A biblioteca HCTSA demonstra a sua utilidade para a extração de *features* através da sua ampla gama de operações de análise de séries temporais altamente comparativa. Os resultados mostram que a utilização de uma grande base de algoritmos disponíveis de diferentes áreas é bastante eficaz, e retira a necessidade de haver processos para determinar as diferentes *features* a extrair.

5.2 Trabalho Futuro

Diferentes métodos de classificação e extração de *features* podem ser implementados como trabalhos futuros. Um dos aspetos a melhorar seria a inclusão de um maior número de dados de ECGs e a introdução de novos tipos de arritmias. Com isto, poderia observar-se como funcionaria a extração de *features* através da biblioteca HCTSA e a posterior classificação das mesmas.

Outro possível desenvolvimento é a utilização do *deep learning*,mais propriamente, das CNN(Convolutional Neural Network) pois tem a capacidade de "aprender", automaticamente, as *features* de médio e alto nível a partir de imagens não treinadas. Portanto, usando imagens de ECGs, primeiro criar imagens 2-D (espectrogramas), de seguida,fazer o aumento de dados e a extração das *features* dos dados (usando o modelo CNN) e fazer a classificação com base nas *features* extraídas. Desenvolver um estudo computacional com vista à classificação dessas imagens seria um possível trabalho futuro.

Bibliografia

[Academic and Center, 2011] A. C. Academic and R. M. Center. arritmias. `https://www.saudecuf.pt/mais-saude/doencas-a-z/arritmias`, 2011. (Cited on page 2).

[Acharya et al., 2007] R. Acharya, J. Suri, and J. Spaan. *Advances in cardiac signal processing*. Springer Verlag, 2007. (Cited on page 11).

[AliveCor, 2020] I. AliveCor. Alive cor. `https://www.alivecor.com/`, 2020. (2 citations in pages 7, and 17).

[Apple, 2018] I. Apple. Apple watch series 4. `https://www.apple.com/ca/apple-watch-series-4/health`, 2018. (Cited on page 7).

[Asgari et al., 2015] S. Asgari, A. Mehrnia, and M. Moussavi. Automatic detection of atrial fibrillation using stationary wavelet transform and support vector machine. *Computers in Biology and Medicine*, 60:132 – 142, 2015. doi: https://doi.org/10.1016/j.compbiomed.2015.03.005. URL `http://www.sciencedirect.com/science/article/pii/S0010482515000839`. (Cited on page 41).

[Billeci et al., 2017] L. Billeci, F. Chiarugi, M. Costi, D. Lombardi, and M. Varanini. Detection of AF and other rhythms using RR Variability and ECG spectral measures. *Computing in Cardiology*, 44:1–4, 2017. (Cited on page 12).

[Bishop, 2006] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*, chapter 1.5.5. Springer, 2006. (2 citations in pages 13, and 21).

[Bogatinovski et al., 2019] J. Bogatinovski, D. Kocev, and A. Rashkovska. Feature extraction for heartbeat classification in single-lead ecg. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 320–325, 2019. (Cited on page 42).

[Brown et al., 2012] G. Brown, A. Pocock, M.-J. Zhao, and M. Lujan. Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection. *Machine Learning Research*, pages 27–66, 2012. (3 citations in pages 22, 23, and 24).

[Carre Technologies, 2012] I. Carre Technologies. Hexoskin smart shirt. `https://www.hexoskin.com/`, 2012. (Cited on page 7).

[Chandra et al., 2017] B. S. Chandra, C. S. Sastry, S. Jana, and S. Patidar. Atrial fibrillation detection using convolutional neural networks. *Computing in Cardiology*, 44:1–4, 2017. (Cited on page 12).

[Chapelle et al., 2010] O. Chapelle, B. Schlkopf, and A. Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010. ISBN 0262514125, 9780262514125. (Cited on page 13).

[Clifford et al., 2017] G. D. Clifford, C. Liu, B. Moody, L. H. Lehman, I. Silva, Q. Li, A. E. Johnson, and R. G. Mark. AF classification from a short single lead ECG recording: The PhysioNet/computing in cardiology challenge 2017. *Computing in Cardiology*, 44: 1–4, 2017. (3 citations in pages 3, 17, and 18).

[Colloca et al., 2013] R. Colloca, A. E. Johnson, L. Mainardi, and G. D. Clifford. A support vector machine approach for reliable detection of atrial fibrillation events. In *Computing in Cardiology 2013*, pages 1047–1050, 2013. (Cited on page 41).

[Dallali et al., 2011] A. Dallali, A. Kachouri, and M. Samet. A classification of cardiac arrhythmia using wt, hrv, and fuzzy c-means clustering. *Signal Processing: An International Journal (SPJI)*, Volume (5):101–108, 01 2011. (Cited on page 2).

[Dash et al., 2009] S. Dash, K. H. Chon, S. Lu, and E. A. Raeder. Automatic real time detection of atrial fibrillation. *Annals of Biomedical Engineering*, 2009. URL `https://doi.org/10.1007/s10439-009-9740-z`. (Cited on page 41).

[Davey and Sharman, 2018] P. Davey and D. Sharman. The electrocardiogram. *Medicine (United Kingdom)*, 46(8):443–452, 2018. doi: 10.1016/j.mpmed.2018.05.004. (3 citations in pages xi, 6, and 7).

[De Chazal et al., 2004] P. De Chazal, M. O'Dwyer, and R. B. Reilly. Automatic classification of heartbeats using ECG morphology and heartbeat interval features. *IEEE Transactions on Biomedical Engineering*, 51(7):1196–1206, 2004. (3 citations in pages 1, 8, and 42).

[E. et al., 2016] T. E., A. El-Sayed, and S. R. A Survey on Classification of ECG Signal Study. *Communications on Applied Electronics*, 6(5):11–16, 2016. (Cited on page 9).

[F.Baltazar, 2009] R. F.Baltazar. *Basic and Bedside Electrocardiography*. Lippincott Williams & Wilkins, 2009. (4 citations in pages xi, 8, 9, and 10).

[Firoozabadi et al., 2018] R. Firoozabadi, R. E. Gregg, and S. Babaeizadeh. P-wave Analysis in Atrial Fibrillation Detection Using a Neural Network Clustering Algorithm. *Computing in Cardiology*, 2018-September, 2018. (Cited on page 2).

[Fleuret, 2004] F. Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, December 2004. ISSN 1532-4435. (Cited on page 25).

[Fulcher, 2013-2020] B. Fulcher. hctsa: highly comparative time-series analysis. `https://github.com/benfulcher/hctsa`, 2013-2020. (Cited on page 19).

[Fulcher, 2020] B. Fulcher. manual outlines the steps required to set up and implement highly comparative time-series analysis. `https://hctsa-users.gitbook.io/hctsa-manual/`, 2020. (4 citations in pages 20, 25, 26, and 27).

[Fulcher et al., 2013] B. Fulcher, M. Little, and N. Jones. Highly comparative time-series analysis: the empirical structure of time series and their methods. *Journal of The Royal Society Interface*, 10(83):20130048, 2013. doi: 10.1098/rsif.2013.0048. URL `https://royalsocietypublishing.org/doi/abs/10.1098/rsif.2013.0048`. (3 citations in pages xi, 20, and 21).

[Gashler et al., 2008] M. Gashler, C. Giraud-Carrier, and T. Martinez. Decision tree ensemble: Small heterogeneous is better than large homogeneous. In *2008 Seventh International Conference on Machine Learning and Applications*, pages 900–905, 2008. (Cited on page 29).

[Gordon Betts et al., 2013] J. Gordon Betts, P. Desaix, E. Johnson, J. E. Johnson, O. Korol, D. Kruse, B. Poe, J. A. Wise, M. Womble, and K. A. Young. *Anatomy & Physiology*. OpenStax College, Rice University, 2013, 2013. (Cited on page 5).

[Hagiwara et al., 2018] Y. Hagiwara, H. Fujita, S. L. Oh, J. H. Tan, R. S. Tan, E. J. Ciaccio, and U. R. Acharya. Computer-aided diagnosis of atrial fibrillation based on ECG Signals: A review. *Information Sciences*, 467:99–114, 2018. URL `https://doi.org/10.1016/j.ins.2018.07.063`. (2 citations in pages 11, and 12).

[Hu et al., 1993] Y. H. Hu, W. J. Tompkins, J. L. Urrusti, and V. X. Afonso. Applications of artificial neural networks for ecg signal detection and classification. *Journal of electrocardiology*, 26 Suppl:66–73, 1993. (Cited on page 19).

[Jambukia et al., 2015] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati. Classification of ECG signals using machine learning techniques: A survey. *Conference Proceeding - 2015 International Conference on Advances in Computer Engineering and Applications, ICACEA 2015*, pages 714–721, 2015. (4 citations in pages xiii, 8, 9, and 14).

[Javatpoint, 2011-2018a] S. Javatpoint. Decision tree classification algorithm. `https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm`, 2011-2018a. (2 citations in pages 27, and 28).

[Javatpoint, 2011-2018b] S. Javatpoint. Logistic regression in machine learning. `https://www.javatpoint.com/logistic-regression-in-machine-learning`, 2011-2018b. (2 citations in pages 28, and 29).

[Khan, 2004] E. Khan. Clinical skills: the physiological basis and interpretation of the ECG. *British journal of nursing (Mark Allen Publishing)*, 13(8):440–446, 2004. doi: 10.12968/bjon.2004.13.8.12778. (Cited on page 7).

[Kim et al., 2019] H. Kim, K. Lee, C. Moon, and Y. Nam. Comparative analysis of machine learning algorithms along with classifiers for af detection using a scale. In *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 427–429, 2019. (Cited on page 41).

[Kropf et al., 2017] M. Kropf, D. Hayn, and G. Schreier. ECG classification based on time and frequency domain features using random forests. *Computing in Cardiology*, 44:1–4, 2017. (2 citations in pages 2, and 41).

[Lagerholm and Peterson, 2000] M. Lagerholm and G. Peterson. Clustering ECG complexes using hermite functions and self-organizing maps. *IEEE Transactions on Biomedical Engineering*, 47(7):838–848, 2000. (No citations).

[Lagerholm et al., 2000] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo. Clustering ecg complexes using hermite functions and self-organizing maps. *IEEE Transactions on Biomedical Engineering*, 47(7):838–848, 2000. doi: 10.1109/10.846677. (Cited on page 19).

[Lampropoulos and Tsihrintzis, 2015] A. S. Lampropoulos and G. A. Tsihrintzis. *Machine Learning Paradigms: Applications in Recommender Systems*. Springer Publishing Company, Incorporated, 2015. ISBN 3319191349, 9783319191348. (Cited on page 13).

[Lewis, 1992] D. D. Lewis. Feature selection and feature extraction for text categorization. In *In Proceedings of Speech and Natural Language Workshop*, pages 212–217. Morgan Kaufmann, 1992. (Cited on page 23).

[LLC, 2018] R. R. LLC. Clinical electrocardiography and ecg interpretation. `https://ecgwaves.com/topic/introduction-electrocardiography-ecg-book/`, 2018. (2 citations in pages 5, and 6).

[Lyon et al., 2018] A. Lyon, A. Mincholé, J. P. Martínez, P. Laguna, and B. Rodriguez. Computational techniques for ECG analysis and interpretation in light of their contribution to medical advances. *Journal of The Royal Society Interface*, 15, jan 2018. (2 citations in pages 1, and 2).

[Maji et al., 2013] U. Maji, M. Mitra, and S. Pal. Automatic detection of atrial fibrillation using empirical mode decomposition and statistical approach. *Procedia Technology*, 10: 45 – 52, 2013. doi: https://doi.org/10.1016/j.protcy.2013.12.335. URL `http://www.sciencedirect.com/science/article/pii/S2212017313004891`. (Cited on page 41).

[Meyer and Bontempi, 2006] P. Meyer and G. Bontempi. On the use of variable complementarity for feature selection in cancer classification. In *Applications of Evolutionary Computing*, volume 3907, pages 91–102. Springer Berlin / Heidelberg, 2006. URL `http://dx.doi.org/10.1007/11732242_9`. (Cited on page 25).

[Muthuchudar and Baboo, 2013] A. Muthuchudar and S. Baboo. A study of the processes involved in ecg signal analysis. *International Journal of Scientific*, 3(3), 2013. (Cited on page 41).

[Obermeyer and Emanuel, 2016] Z. Obermeyer and E. J. Emanuel. Predicting the future—big data, machine learning, and clinical medicine. *New England Journal of Medicine*, 375, 2016. (Cited on page 1).

[Omron Healthcare, 2020] A. Omron Healthcare. Omron ecg monitor hcg-801. `https://zenicor.com/zenicor-ekg/`, 2020. (Cited on page 7).

[Park et al., 2006] C. Park, P. H. Chou, Y. Bai, R. Matthews, and A. Hibbs. An ultra-wearable, wireless, low power ecg monitoring system. In *2006 IEEE Biomedical Circuits and Systems Conference*, pages 241–244, 2006. doi: 10.1109/BIOCAS.2006.4600353. (Cited on page 7).

[Peng et al., 2005] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, pages 1226–1238, 2005. (Cited on page 24).

[Potter, 2011] L. Potter. Understanding an ecg. `https://geekymedics.com/understanding-an-ecg/`, 2011. (Cited on page 6).

[Qardio, 2018] I. Qardio. Qardiomd. `https://www.getqardio.com/qardiomd-ecg/`, 2018. (Cited on page 7).

[Sampson and McGrath, 2015] M. Sampson and A. McGrath. Understanding the ecg. part 1: Anatomy and physiology. *British Journal of Cardiac Nursing*, 10(11):548–554, 2015. (Cited on page 6).

[Sutton and Barto, 1998] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT press Cambridge, 1998. (Cited on page 13).

[Thaler, 2012] M. S. Thaler. *The only EKG book you'll ever need.* Wolters Kluwer/Lippincott Williams & Wilkins, 7 edition, 2012. (Cited on page 6).

[The MathWorks, 1994-2020a] I. The MathWorks. Matlab. `https://la.mathworks.com/products/matlab.html`, 1994-2020a. (Cited on page 3).

[The MathWorks, 1994-2020b] I. The MathWorks. Ensemble algorithms. `https://la.mathworks.com/help/stats/ensemble-algorithms.html`, 1994-2020b. (Cited on page 30).

[Vafaie et al., 2014] M. H. Vafaie, M. Ataei, and H. R. Koofigar. Heart diseases prediction based on ECG signals' classification using a genetic-fuzzy system and dynamical model of ECG signals. *Biomedical Signal Processing and Control*, 14(November):291–296, 2014. (Cited on page 8).

[Valentin Fuster, 2010] A. C. Valentin Fuster. *Committee on Preventing the Global Epidemic of Cardiovascular Disease : Meeting the Challenges in Developing Countries Board on Global Health Valentín Fuster and Bridget B . Kelly , Editors.* Washington (DC): National Academies Press (EUA) ; 2010, 2010. (Cited on page 1).

[Valerie C. Scanlon, 2007] T. S. Valerie C. Scanlon. *Essentials of Anatomy and Physiology.* F.A. Davis Company, 2007. (2 citations in pages xi, and 6).

[Wang et al., 2020] J. Wang, P. Wang, and S. Wang. Automated detection of atrial fibrillation in ecg signals based on wavelet packet transform and correlation function of random process. *Biomedical Signal Processing and Control*, 55:101662, 2020. doi: https://doi.org/10.1016/j.bspc.2019.101662. URL `http://www.sciencedirect.com/science/article/pii/S1746809419302435`. (Cited on page 42).

[Wikipedia, 2020] Wikipedia. Receiver operating characteristic. `https://en.wikipedia.org/wiki/Receiver_operating_characteristic`, 2020. (Cited on page 15).

[Yang and Moody, 1999] H. Yang and J. Moody. Feature selection based on joint mutual information. In *Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis*, pages 22–25, 1999. (Cited on page 24).

[Yeap et al., 1990] T. H. Yeap, F. Johnson, and M. Rachniowski. ECG beat classification by a neural network. *Proceedings of the Annual Conference on Engineering in Medicine and Biology*, 12(pt 3):1457–1458, 1990. (Cited on page 19).

[Zipes et al., 2018] P. Zipes, D.P.and Libby, R. Bonow, D. Mann, and G. Tomaselli. *Braunwald's Heart Disease E-Book: A Textbook of Cardiovascular Medicine.* Elsevier Health Sciences; Philadelphia, PA, USA: 2018, 2018. (Cited on page 1).