Classificação Antecipada de Séries Temporais

por

Liliana Borges Emílio

Dissertação de Mestrado em Modelação, Análise de Dados e Sistemas de Apoio à Decisão

Orientada por

Professor Doutor João Manuel Portela da Gama

Professor Doutor Pedro Pereira Rodrigues

Faculdade de Economia

Universidade do Porto

2014

Nota Biográfica

Liliana Borges Emílio nasceu a 25 de Março de 1990, fez a sua formação básica e secundária em Vila Real. Licenciou-se em Gestão pela Faculdade de Economia do Porto em 2011, iniciando o Mestrado em Análise de Dados e Sistemas de Apoio à Decisão no mesmo ano.

A sua atividade profissional iniciou-se na Qmetrics, onde realizou um estágio profissional como Técnica de Estatística. Atualmente é Consultora de Business Intelligence na Novabase.

Agradecimentos

Em primeiro lugar queria agradecer aos meus pais por todo o acompanhamento ao longo destes anos.

Queria também deixar o meu agradecimento aos orientadores Professor Doutor João Gama e Professor Doutor Pedro Rodrigues pelo incentivo e todas as dicas valiosas para a conclusão deste trabalho.

Ao Pedro por estar comigo em todos os momentos.

A todos os meus amigos e colegas de trabalho que sempre demostraram o seu apoio.

Ao André Reis por todo o planeamento e incentivo para o término desta etapa.

Resumo

Atualmente os métodos de Extração de Conhecimento de Dados auxiliam cada vez mais o processo de tomada de decisão nos mais diversos domínios, sejam eles financeiros, meteorológicos, económicos ou em cuidados médicos. Decisões acertadas, muitas vezes, são difíceis de tomar principalmente em condições de elevado *stress*, como é o caso de um trabalho de parto onde a diferença entre uma boa ou má decisão médica pode ditar a vida ou a morte.

A classificação antecipada de séries temporais visa prever o desfecho de um determinado acontecimento antes de este acontecer, sendo tal possível através da análise das primeiras amostras de informação da série temporal a classificar. Uma vez que na sua grande maioria os dados biomédicos podem ser tratados como séries temporais, é de grande importância a aplicação deste tipo de técnicas em áreas como a medicina.

Ao longo deste trabalho são aplicadas técnicas de classificação antecipada a frequências cardíacas fetais recolhidas no trabalho de parto, com o objetivo de identificar o instante de tempo em que é possível prever o desfecho do parto antes de este terminar.

Conclui-se que analisando os primeiros 727 valores de frequências cárdicas fetais é possível identificar um parto com mau desfecho.

Palavras chave: Séries Temporais; Classificação; Classificação Antecipada de Séries Temporais; Frequência Cardíaca Fetal.

Abstract

Nowadays, more and more Data Mining is helping decision making process in such diverse fields as financial, meteorological, economic or even health care. Often, it isn't easy to make the right decision, especially when under stressful conditions, e.g. during labor, where the right decision means life over death.

Early classification on time series aims to predict a result of an event even before it could happen. This is done by analyzing the first data samples of the time series to consider. Since most of the biomedical data can be treated as time series, these kinds of technics are of great relevance in such fields as medicine.

During this essay, early classification techniques are applied in fetal heart rates collected during active phase of labor, it aims to identify an instance of time from which is possible to predict the labor success or failure before it happens.

Analyzing first 727 fetal heart rate values, we conclude that it is possible classify a labor in a failure.

Keywords: Time Series; Classification; Early Classification on Time Series; Fetal Heart Rate.

Índice

Índice de Figuras

Índice de Tabelas

Capítulo 1 - Introdução

1.1 Motivação

A Extração de Conhecimento de Dados, do inglês *Data Mining*, é um instrumento valioso para a previsão de determinados acontecimentos mediante os acontecimentos históricos, por exemplo, com base na análise histórica de dados de um cliente bancário é possível prever se o cliente em causa vai pagar o empréstimo que agora requer, minimizando assim potenciais imparidades para a entidade bancária. Com base no histórico de compras associado a um cartão de crédito pode-se prever se uma transação é, ou não é, fraudulenta. As tendências de mercado são também possíveis de aferir através da análise de comportamentos de consumidores (Han e Kamber, 2006)

Todavia, no campo médico a consequência de uma má decisão pode ter implicações complexas nomeadamente a perda de uma vida humana, por este motivo, quanto melhores forem os métodos de auxílio ao diagnóstico clínico melhor será a deteção de potenciais problemas, sendo possível intervir atempadamente. De facto é fundamental perceber o instante em que se pode e deve intervir, assim como perceber qual é o *trigger* para determinado acontecimento, que seja bom ou mau.

O campo médico é exemplo de uma área onde vastas quantidades de séries temporais são geradas. Segundo dados obtidos no estudo de Pitts *et al*., 2008, no ano de 2006 das cerca de 119,2 milhões de visitas aos serviços de emergência de hospitais dos Estados Unidos da América, 25% tiveram como auxílio ao diagnóstico sinais cardíacos, que podem ser analisados como séries temporais (Buza *et al*., 2011). Também os atuais dispositivos de monitorização médica são capazes de recolher e transmitir massivas quantidades de dados (Hosseinkhah *et al*., 2009). Segundo dados da *American Academy of Family Physicians* (AAFP), em 2002 cerca de 85% dos quatro milhões de nascimentos nos Estados Unidos da América foram monitorizados.

A análise dos dados obtidos ao longo do trabalho de parto é sem dúvida de grande utilidade na medida em que pode ajudar a descobrir e compreender padrões que evidenciem o desfecho de um parto. Desta forma seria possível providenciar informação em tempo útil aos clínicos podendo, em caso de previsão de um mau desfecho, salvar a vida de uma criança, ou em caso de previsão de um bom desfecho seria evitada uma cesariana desnecessária.

A aplicação de técnicas de classificação antecipada, do inglês *early classification*, pode dar um contributo bastante valioso à previsão do desfecho de um determinado parto, uma vez que pela análise da informação inicial, recolhida durante monitorização eletrónica fetal podem ser encontrados padrões que caracterizem os partos com bom ou mau desfecho. Sempre que o comportamento de uma variável monitorizada coincidisse com os padrões encontrados seria emitido um alerta aos clínicos sobre o desfecho final do parto antes de ocorrer algum incidente fatal para o feto.

Na verdade, a monitorização fetal tem sido uma preocupação e um tema atual, por exemplo, no ano de 2013 o site Physionet lançou um desafio relacionado com a análise de eletrocardiogramas fetais. Em 2014, o grupo privado de investimento New World Angels fez um investimento de 2,1 milhões de dólares para o desenvolvimento e comercialização de um sistema de monitorização materno e fetal (Fonte Reuters). Existem cada vez mais dispositivos móveis de monitorização fetal móvel, sendo exemplo disso o Monica AN24, que possibilita a monitorização do bem estar fetal em casa. Existem também cada vez mais aplicações móveis relacionadas com bem estar fetal, por exemplo a aplicação *Baby Scope App*, *Fetal Doppler UnbornHeart*, *Fetal Heart Rate Monitor App*, entre outras.

1.2 Objetivos

No presente trabalho pretende-se analisar frequências cardíacas recolhidas durante o trabalho de parto com o intuito de prever eventuais riscos associados ao parto.

Tanto quanto o nosso conhecimento, é a primeira vez que técnicas de classificação antecipada serão aplicadas em dados de frequências cardíacas fetais, onde o resultado da classificação poderá fazer a diferença entre a vida e a morte.

Além da apresentação de modelos existentes sobre o tema Classificação Antecipada, assim como as suas principais vantagens e limitações, o objetivo primário será encontrar séries temporais através das quais é possível realizar uma classificação pela análise dos primeiros valores de frequência cardíaca fetal, assim como encontrar o instante de tempo a partir do qual a classificação pode ser realizada.

1.3 Organização

O trabalho apresentado tem a seguinte organização: no segundo capítulo, denominado Trabalho Relacionado, é apresentado o Processo de Conhecimento em Base de Dados, nomeadamente quais os passos que devem ser tomados para que seja possível extrair conhecimentos de determinados dados. No mesmo capítulo são, também, expostos diversos conceitos relacionados com séries temporais, que servem de base para a compreensão do terceiro capítulo, intitulado A Classificação Antecipada em Séries Temporais. Neste capítulo são apresentados os modelos de Classificação Antecipada existentes na literatura. Por sua vez, no quarto capítulo é apresentado um estudo de caso, onde nos dados em análise é aplicado um modelo de Classificação Antecipada de séries temporais, apresentado no capítulo três, com variantes específicas e dirigidas à análise em causa. A linguagem R foi a escolhida para a elaboração de todo o quarto capítulo. Por fim, no capítulo cinco são apresentadas as principais conclusões e sugestões de trabalho futuro.

Capítulo 2 - Trabalho Relacionado

2.1 Descoberta de Conhecimento em Base de Dados

Ao longo das últimas décadas verificou-se um aumento da quantidade de dados armazenados, provenientes das mais diversas áreas, devido essencialmente à evolução das tecnologias de informação, nomeadamente o cada vez menor custo de armazenamento de dados. A crescente dimensão das bases de dados são uma evidência dessa mesma capacidade, no entanto o tratamento, análise e o acesso simples e rápido à informação constante, tanto de forma direta como através de técnicas de análise de dados, nessas mesmas bases de dados, não acompanhou esta mesma tendência de crescimento, o que originou a necessidade de criação de mecanismos que permitam extrair o máximo de respostas úteis, rápidas e válidas sobre os dados armazenados. De forma a colmatar as limitações humanas, que não conseguem analisar atempadamente toda a informação contida nos repositórios de dados, vários campos de pesquisa, desde a Estatística à Ciência de Computadores, deram o seu contributo para melhorar o processo de procura (Norton, 1999*)*. A interseção de várias áreas de interesse, como Aprendizagem Automática, do inglês *Machine Learning*, Base de Dados, Estatística, Inteligência Artificial, Visualização de Dados, resultou no Processo de Descoberta de Conhecimento em Base de Dados, do ingês *Knowledge Discovery in Databases* (KDD) (Fayyad *et al*., 1996).

O KDD é um processo, não trivial, que tem como objetivo a identificação de padrões, úteis e compreensíveis de dados (Fayyad, 1997). Este processo concentra-se, igualmente, na forma como os dados são armazenados, na forma como se pode ter acesso aos mesmos, preocupa-se com a eficiência dos algoritmos, assim como de que forma os resultados podem ser interpretados e como a relação homem-máquina pode ser apoiada (Fayyad *et al*., 1996).

O termo Extração de Conhecimento de dados, do inglês *Data Mining* é, muitas vezes, usado como sinónimo de KDD, no entanto, o *Data Mining* é o núcleo do processo de

KDD, sendo um dos passos do KDD (Fayyad, 1997)(Maimon e Rokach, 2010). Entende-se como *Data Mining* o conjunto de métodos e técnicas, que exploraram e analisam dados, de forma automática ou semiautomática, no sentido de encontrar os padrões, associações e tendências úteis, escondidos nos dados (Tufféry, 2011).

No total, são nove os passos que constituem o processo de KDD que são em seguida numerados e detalhados, assim como estão visíveis na figura 1 (Maimon e Rokach, 2010) (Fayyad *et al*, 1996):

1. Adquirir conhecimento relevante e perceber os objetivos do problema em causa;
2. Identificar os dados, ou conjunto de dados, sobre os quais as pesquisas deverão incidir;
3. Pré-processamento e limpeza dos dados – remoção de ruído e *outliers*, assim como perceber qual a estratégia para lidar com valores em falta;
4. Transformar os dados – utilização de métodos para reduzir a dimensionalidade ou transformação dos atributos (discretização);
5. Escolha da tarefa de *Data Mining* a utilizar – Classificação, Regressão, Sumarização;
6. Escolha do algoritmo de *Data Mining*;
7. Aplicação do algoritmo escolhido;
8. Avaliação do desempenho do algoritmo;
9. Uso do conhecimento adquirido.

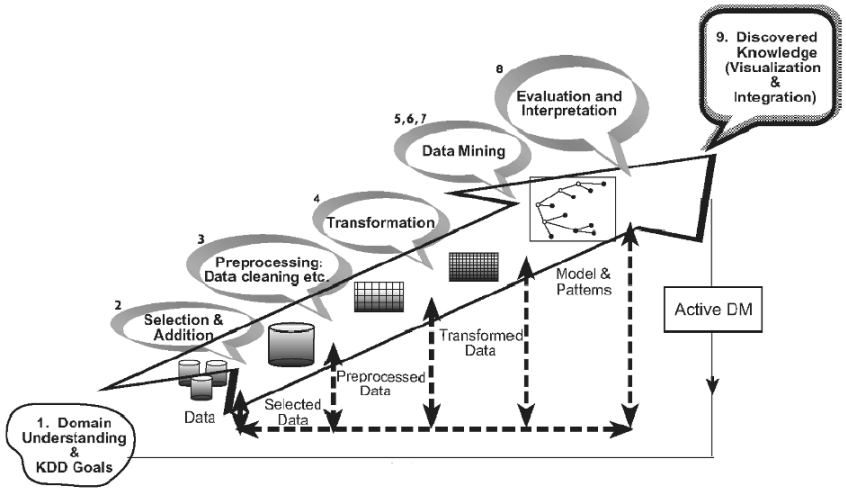

Figura 1 - Processo de KDD (Maimon e Rokach, 2010)

2.2 Extração de Conhecimento de Dados

As técnicas de Extração de Conhecimento de Dados, do inglês *Data Mining*, podem ser usadas consoante o propósito em causa, assim estas podem dividir-se em técnicas: orientadas à verificação ou orientadas à descoberta. A primeira divisão diz respeito à avaliação de uma hipótese colocada por uma fonte externa, tratando-se de um método mais estatístico, que inclui testes como por exemplo testes de hipóteses e ANOVA. A segunda divisão relaciona-se com a descoberta automática de padrões nos dados (Maimon e Rokach, 2010).

As técnicas orientadas à descoberta, do inglês *discovery-oriented*, podem ser subdivididas em Descrição, que consiste na apresentação dos padrões de forma interpretável e Previsão, cujo objetivo é a previsão do valor de uma variável denominada alvo, dependente ou endógena associada a um objeto, como função de um certo número de outras variáveis denominadas controlo, independentes ou exógena, associadas a esse mesmo objeto. Se a variável alvo for do tipo qualitativo, trata-se de um problema de Classificação, se pelo contrário, o valor da variável dependente for do tipo numéricos, trata-se de uma Regressão (Fayyad *et al*., 1996)(Tufféry, 2011).

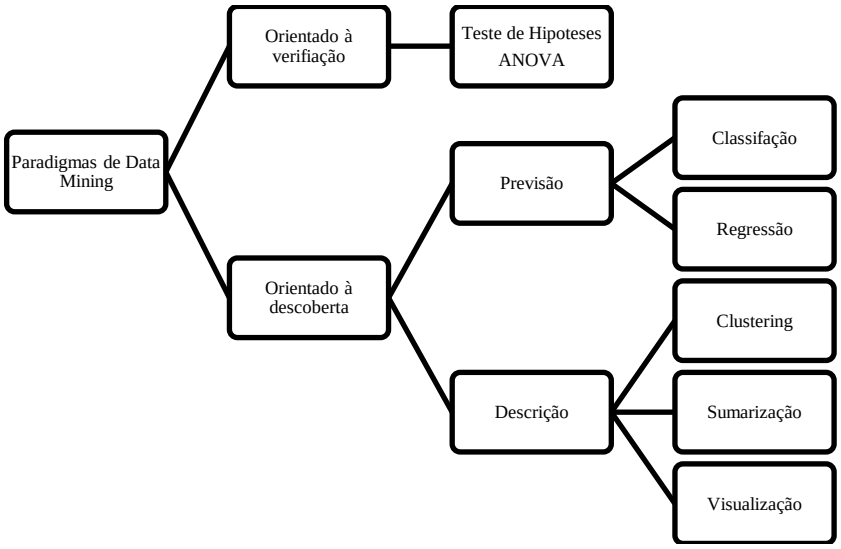

Figura 2 - Paradigma de Data Mining (Maimon e Rokach, 2010)

2.2.1 Classificação

A Classificação é uma das tarefas mais comuns feitas pelo ser humano, uma vez que para compreender e comunicar sobre o mundo que o rodeia é necessário categorizar e classificar, sendo exemplo disso a identificação dos cães pelas suas raças (Berry e Linoff, 2004). O objetivo desta tarefa é encontrar um modelo, também denominado como classificador, que distinga e descreva as classes a que cada objeto pertence, isto é, de acordo com as propriedades que cada objeto possui o modelo atribui uma classe e nunca atribui mais que uma ou nenhuma (Chen *et al*., 1996) (Bramer, 2007) (Han e Kamber, 2006).

A Classificação pode ser decomposta em duas fases, a fase de aprendizagem e a fase de teste. Na primeira fase, também conhecida por fase de treino ou indutiva, é construído um classificador analisando um conjunto de dados de treino e a respetiva classe associada. Na segunda fase, reconhecida igualmente por dedutiva, o classificador obtido na fase anterior é verificado através dos dados de teste. Assim consegue-se perceber qual o melhor modelo obtido na fase de treino através de medidas de desempenho, do inglês *accuracy* (Han e Kramber, 2006)(Tufféry, 2011). A figura três é ilustrativa deste processo.

O desempenho da tarefa de classificação é calculado através dos dados de teste, comparando a classe que o modelo atribuiu a um objeto de teste com a classe real desse mesmo objeto (Han e Kramber, 2006).

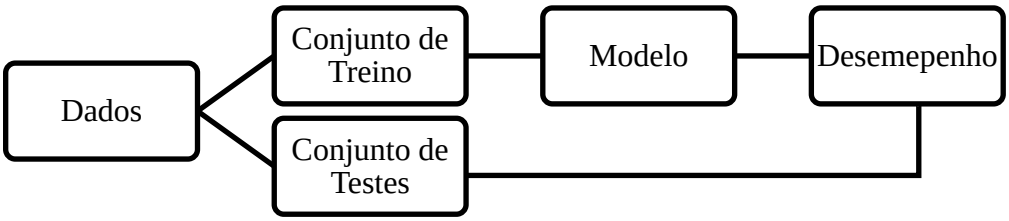

Figura 3 - Método de avaliação da acurácia (Han e Kamber, 2006)

A divisão de dados entre dados de treino e teste é feita através de métodos de amostragem, tais como, *holdout,* amostragem aleatória, validação cruzada, *bootstrap* (Han e Kramber, 2006).

No método *holdout* dois terços dos dados iniciais são alocados aos dados de treino, sendo o remanescente alocado aos dados de teste. A amostragem aleatória é uma variante do método *holdout,* sendo que o método *holdout* é repetido *k* vezes e o desempenho do classificador é dada pela média de cada uma das *k* iterações (Han e Kramber, 2006).

Na validação cruzada é escolhido o número de partições, do inglês *folds*, que se pretendem para os dados, sendo em seguida estes divididos em k-*folds*: D_1, D_2, D_3, … D_k. Nas primeiras iterações os dados de treinos são compostos por D_2 até D_k, sendo D_1 a partição para testes, na segunda iteração D_1 faz parte dos dados de treino, enquanto D_2 faz parte dos de teste (Han e Kramber, 2006). Vários estudos, referentes a diferentes base de dados, mostraram que dez é o número indicado de *folds* para obter a melhor taxa de acerto (Witten *et al*., 2011).

A subjacente ao *bootstrap* é obtenção de amostras com reposição a partir da base de dados, isto é, para uma base de dados com n instâncias são retiradas n amostras, que podem ser repetidas, formando outra base de dados com n instâncias (Witten *et al*., 2011). Nesta segunda base de dados existirão elementos repetidos o que fará com que outros elementos não constem na mesma, assim, os elementos não pertencentes à segunda base de dados, irão integrar o conjunto de dados de teste (Witten *et al*., 2011). O desempenho do classificador, representado por ACC, é calculado da seguinte forma (Han e Kramber, 2006):

$$\text{ACC}\ (M) = \sum_{i=1}^{k} (0{,}623 \text{ X ACC } (M_{\text{i conj teste}}) + 0{,}368 \text{ X ACC } (M_{\text{i conj treino}})$$

Todavia, a mera avaliação da taxa de acerto do classificador pode não ser suficiente, ou seja, uma taxa de acerto de 90% pode não ser aceitável em certos cenários, uma vez que o custo associado a determinadas decisões é variável, ou seja, o custo de não detetar um derrame de petróleo, com potenciais consequências ambientais desastrosas, é superior a

detetar um falso derrame (Han e Kramber, 2006) (Witten *et al*., 2011). Por este motivo é importante construir uma matriz onde são identificados todos os casos possíveis de classificação e as respetivas ocorrências, por exemplo número de falsos alarmes, alertas não acionados, desta forma será possível compreender em que medida um classificador reconhece as diferentes classes (Han e Kramber, 2006).

Uma matriz de confusão é uma tabela de tamanho m x m, onde m representa o número de classes e cada elemento representado na matriz de confusão refere-se a um exemplo dos dados de teste. Atentando à tabela 1, na célula denominada VP (verdadeiros positivos) é identificado o número de exemplos da classe C1 que o classificador classificou como C1, o mesmo se passa na célula VN (verdadeiros negativos) para a classe C2, idealmente todas os exemplos de testes deveriam estar nesta diagonal. Por vezes o classificador falha a classificação, atribuindo a exemplos da classe C1 a classe C2, para esse caso estamos perante falsos negativos, FN. O último caso relaciona-se com o facto de o classificador atribuir a classe C1 a exemplos da série C2, esse caso é relativo a falsos positivos, FP (Han e Kramber, 2006)(Witten *et al*., 2011).

Tabela 1 - Matriz de Confusão

		Classe prevista pelo classificador	
		C1	C2
Classe Real	C1	VP	FN
	C2	FP	VN

Como dito existem certos custos que não podem ser negligenciados, desta forma são utilizados outros avaliadores para um dado classificador, a especificidade e sensibilidade. A especificidade representa a taxa de acerto na classe negativa e a sensibilidade a taxa de reconhecimento na classe positiva e calculam-se da seguinte forma (Han e Kramber, 2006):

$$\text{Especificidade} = \frac{VN}{FP+VN} \quad (2.1)$$

$$\text{Sensibilidade} = \frac{VP}{VP+FN} \quad (2.2)$$

2.2.1.1 Classificador K Vizinhos Mais Próximos

Uma vez definidos os métodos de divisão dos dados em dados de treino e teste, assim como já mencionadas as principais medidas de avaliação de um modelo, importa descrever o modelo de classificação que será utilizado e bastante mencionado ao longo deste trabalho.

O classificador K-Vizinhos Mais Próximos, do inglês K-*Nearest-Neighbor* (k-NN), é um algoritmo que classifica uma determinada observação com base na classe mais comum dos seus K vizinhos mais próximos, tomando-se como vizinho mais próximo o que apresenta menor distância (An, 2009).

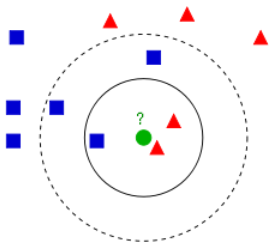

Figura 4 - Algoritmo 1-NN, Fonte: Wikipedia

Cada exemplo é descrito por n atributos que são armazenados, sendo o processo de aprendizagem diferido no tempo, ou seja, apenas se procede à classificação quando existe uma nova observação a classificar. A proximidade entre duas observações é calculada com base na distância entre os diversos atributos que caracterizam essas mesmas observações (An, 2009) (Han e Kramber, 2006):

$$dist(X_1, X_2)=\sqrt{\sum_{i=1}^{n}(x_{i1}-x_{i2})^2} \quad (2.3)$$

Atentando à figura 4 acima e assumindo apenas um vizinho mais próximo, K = 1, se o elemento a classificar for a bola verde nas classes azul ou vermelho, então será calculada a distância entre a bola verde e cada um dos restantes elementos através da fórmula 2.3. A classe atribuída à bola verde será a classe do elemento mais próximo. Visualmente nota-se que o triângulo vermelho é o que possuí menor distância, logo a classe atribuída pelo vizinho mais próximo da bola verde é a classe vermelha.

Assumindo três vizinhos mais próximos, o modelo 3-NN iria verificar quais os três objetos mais próximos através da aplicação da equação 2.3. Pela análise da figura nota-se que dois vizinhos mais próximos são da classe vermelha, enquanto um é da classe azul, neste caso a classe atribuída seria a classe vermelha, uma vez que está em maioria nos vizinhos mais próximos.

2.3 Extração de Conhecimento de Dados Temporais

O *Data Mining* pode ser categorizado de acordo com o tipo de dados que se pretende analisar, isto é, se os dados em análise forem financeiros está-se perante *Data Mining* Financeiro, se os dados em análise forem dados biológicos está diante *Data Mining* Biológico, se os dados forem séries temporais está-se perante *Data Mining* Temporal (Han, 2009).

O *Data Mining* Temporal tem ganho maior notoriedade nas últimas décadas (Mitsa, 2009), o que se encontra intimamente ligado com o facto da grande parte das bases de dados possuir referências temporais (Jensen e Snodgrass, 2009). Esta categoria concentra-se em aplicar técnicas de *Data Mining* a um conjunto de dados sequenciais, isto é, dados que se encontram ordenados de acordo com um índice, que geralmente se trata de um índice temporal, permitindo desta forma proceder à análise dos dados de acordo com a sua evolução temporal (Laxman e Sastry, 2006). Segundo Mitsa, 2009 é possível dividir o *Data Mining* Temporal em três tipos:

- Séries temporais – representam valores reais medidos em espaço de tempo regulares;
- Sequências Temporais – os valores que representam podem ou não ser regulares;
- Dados Temporais Semânticos, do ingês *semantic temporal data* – definidos em contexto, por exemplo “sénior” e “adulto”.

Como dito, o presente estudo debruça-se sobre um conjunto de séries temporais, sendo estas a base para a análise a levar a cabo, pelo que as restantes categorias mencionadas não serão alvo de desenvolvimento.

2.3.1 Séries Temporais

Nas últimas décadas verificou-se um crescente interesse em aplicações de extração de conhecimento tendo como base as séries temporais devido, essencialmente, à velocidade a que as mesmas são geradas, às diversas áreas que originam séries temporais e também devido à sua fácil obtenção, através de aplicações científicas e financeiras, temperaturas diárias, total de vendas semanal, entre outros (Ding *et al.*, 2008) (Fu, 2011).

Uma série temporal é uma sequência de números reais, onde cada número representa um valor a cada instante de tempo (Chan e Fu, 1999). Utilizando a notação de Xing *et al*., 2009, uma série temporal pode ser representada por "s", cujo comprimento total é L. Portanto s[i] é o valor da série temporal no instante de tempo i, sendo que $1 \leq i \leq L$.

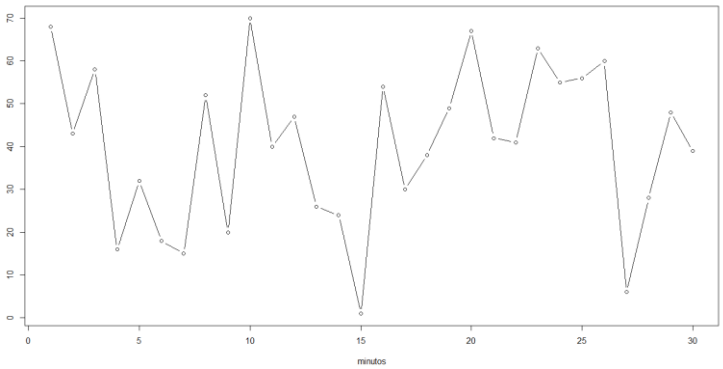

Figura 5 - Exemplo de uma série temporal

Na figura cinco é apresentada a medição de uma variável em minutos, cujo comprimento é 30, utilizando a notação de Xing *et al*., 2009 L=30. Também é possível verificar que no décimo minuto, i=10, a variável assume o valor 70, sendo representado por: s[10] = 70.

2.4 Classificação de Séries Temporais

A análise de séries temporais é um problema antigo da Estatística, cuja principal aplicação é a previsão meteorológica ou financeira, contudo as tarefas de classificação e correspondência, do inglês *matching*, de séries temporais são mais recentes. Nestas tarefas os algoritmos tradicionais de *machine learning* não têm a mesma eficácia, devido às características típicas de séries temporais como alta dimensionalidade, correlação e ruído nos dados (Keogh e Kasetty, 2002) (Laxman e Sastry, 2006). Na chamada classificação tradicional a ordem dos atributos é irrelevante, assim como a interação entre as variáveis, pelo que, o classificador considera-os independentes das suas posições relativas, descurando o facto de que em dados temporais a ordem temporal é crucial para encontrar as melhores características discriminativas entre os exemplos a classificar (Bagnall *et al*., 2012). Além disso, nem sempre o processo de definição das classes é uma tarefa fácil, porque não dispensa o conhecimento de um perito, sendo um exemplo claro desta necessidade os dados da bolsa (Nanopoulos *et al*., 2001). Outro entrave bastante comum à definição de classes é o facto de uma classe encontrar-se bem definida e a outra não ter uma estrutura específica (Keogh e Pazzani, 1998).

O problema de classificação de séries temporais, utilizando a notação de Xing *et al*., 2009, consiste em a partir de um conjunto de dados de treino, representado por T, aprender um classificador, representado por C, tal que a cada série temporal do grupo de treino, $t \in T$, pertença a uma classe, $t.c \in C$. O classificador deve prever a classe de qualquer série temporal, s, sendo que C(s) é a classe atribuída à série s. O desempenho do classificador é avaliado através de um conjunto de dados teste, denominado T', isto é, cada série temporal $t' \in T'$ pertence a uma classe conhecida, representada por $t'.c \in C$. O desempenho do classificador C é avaliado pela coincidência entre a classe gerada pelo classificador e a classe a que t' de facto pertence (Xing *et al*., 2009). A fórmula de cálculo é retirada do trabalho de Xing *et al*., 2009 e pode ver-se na página seguinte:

$$\text{Desempenho}(C,T') = \frac{|\{C(t')=t'.c \mid t' \in T'\}|}{|T'|} \quad (2.4)$$

Contudo, para que a análise de séries temporais seja eficiente e eficaz são necessários métodos de representação das séries, capazes de reduzir a dimensionalidade, assim como medidas de similaridade (Ding *et al*., 2008).

2.4.1 Técnicas de Redução de Dimensionalidade em Séries Temporais

A alta dimensionalidade, resultado do elevado número de atributos ou conjunto de atributos, constitui um sério obstáculo à eficiência da maioria dos algoritmos de *Data Mining*. A diminuição da dimensionalidade tem variadas vantagens, passando a numerar (Chizi e Maimon, 2010):

1. Redução do custo de aprendizagem;
2. Melhoraria do desempenho do modelo;
3. Redução as dimensões irrelevantes;
4. Redução dimensões redundantes.

As séries temporais são na sua maior parte caracterizadas pela sua elevada dimensionalidade, sendo que o tratamento de uma grande quantidade de informação pode ser custoso em termos de processamento e armazenamento, por esse motivo os métodos de redução de dimensionalidade são desejáveis (Ding *et al*., 2008). Contudo, estes métodos devem preservar as características fundamentais do conjunto de dados em análise. Uma das características que deverá ser preservada é a de "menor limite", do inglês *lower bounding*, isto é, se duas séries temporais são semelhantes no espaço original, então também terão que o ser no espaço transformado (Mitsa, 2009).

Os métodos de representação podem ser divididos em quatro grupos (Mitsa, 2009):

1. Baseado no modelo;

2. Adaptados aos dados*;*
3. Não adaptados aos dados;
4. Ditada pelos dados.

Apenas será alvo de nota o primeiro grupo, uma vez que este será o utilizado ao longo do trabalho. Neste primeiro grupo encaixam-se os métodos como a Transformada Discreta de Fourrier (DFT), Transformada Discreta de Ondas, do inglês Discrete Wavelet Transform (DWT) e a Agregação por Aproximação em Partes, do inglês Piecewise Aggregate Approximation (PAA) (Mitsa, 2009). Apenas o último método mencionado será abordado uma vez que é o único a ser utilizado neste trabalho.

2.4.1.1 Agregação por Aproximação em Partes

A Agregação por Aproximação em Partes é um método de redução de dimensionalidade proposto por Keogh *et al*., 2000, sendo este método mais simples que a DFT e DWT (Mitsa, 2009). O PAA divide as séries temporais em w segmentos de igual comprimento, após a divisão em w segmentos, o valor que cada segmento assume é correspondente ao valor médio dos valores pertencente ao segmento em causa (Mitsa, 2009). Na figura 5, em baixo, mostra a série original X após a mesma ser dividida em oito segmentos obteve-se a série X'.

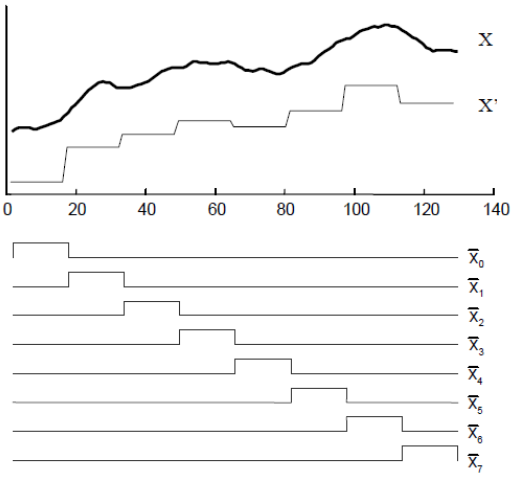

Figura 6 - Método Agregação por Aproximação em Partes (Keogh et al., 2001)

Existem dois casos para os quais a PAA não tem qualquer impacto, o primeiro tem a ver o número de segmentos definidos ser coincidente com o tamanho da série: w=L, a série transformada seria igual à série original. O segundo caso é relativo à existência de apenas um segmento, o que faz com que a série transformada seja uma média da série temporal original (Keogh *et al*., 2001).

2.4.2 Medidas de Similaridade em Séries Temporais

As medidas de similaridade entre as séries temporais devem ser meticulosamente definida para que seja refletida a di(semelhança) entre as séries em comparação, quanto maior a sua distância, menor a sua semelhança e vice-versa (Ding *et al*., 2008) (Milanović e Stamenković, 2011). O cálculo de similaridade entre duas séries temporais é uma rotina importante nas aplicações de *Data Mining*, nomeadamente na Classificação, uma vez que os valores de distância encontrados para as séries temporais em análise revelam se existe alguma similaridade no comportamento das mesmas (Ratanamahatana e Keogh, 2005) (Milanović e Stamenković, 2011).

Segundo Mista, 2009, os três tipos de medidas de similaridade mais utilizados são distância baseada em similaridade, do inglês *distance-based similarity*, do segundo grupo fazem parte as medidas *dynamic time warping*, por fim, no terceiro tipo são é enquadrada a distância maior subsequência em comum do inglês, *longest common subsequence*. Apenas será feita menção dos dois primeiros tipos de medidas de similaridade, uma vez que serão as medidas exploradas no trabalho apresentado.

Do primeiro grupo faz parte a distância Euclidiana (Mitsa, 2009). A distância Euclidiana é fácil de implementar e surpreendentemente é muito competitiva face as demais medidas, especialmente quando existe um grande conjunto de dados de treino (Ding *et al*., 2008). Por outro lado, a sua utilidade é limitada uma vez que as séries a comparar têm de ter a mesma escala e comprimento e não podem existir *gaps* entre as mesmas, além disto a distância Euclidiana é suscetível a ruído nos dados e variações no

eixo temporal (Mitsa, 2009). A fórmula de cálculo pode ser definida, com base em Xing *et al*, 2009, da seguinte forma:

$$Dist(s, s') = \sqrt{\sum_{i=1}^{L} (s[i]-s'[i])^2} \quad (2.5)$$

No sentido de mitigar as desvantagens da distância Euclidiana, Berndt e Clifford, 1994 propuseram uma técnica chamada *dynamic time warping* (DTW), cujo objetivo é o alinhamento de as séries temporais de forma que a distância entre elas fosse minimizada. Assim, considerando Q e C duas séries temporais de comprimento p e m, respectivamente, visíveis na figura 7A, alinhamento destas duas séries temporais é dado pela construção de uma matriz p x m onde o elemento (i^{th},j^{th}) corresponde à distância entre dois pontos q_i e c_j, como representado na figura 7B (Xi *et al*., 2006).

$$Q= q_1,q_2, \ldots, q_i,\ldots,q_p$$

$$C= c_1,c_2, \ldots, c_j, \ldots, c_m$$

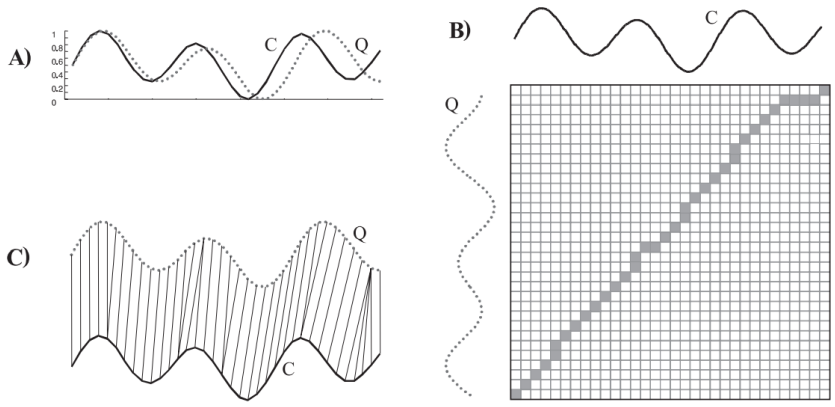

Figura 7 - Exemplo de Alinhamento através de DTW (Keogh e Ratanamahatana, 2005)

A distância Euclidiana é, geralmente, utilizada para calcular a distância entre os pontos, como pode ser observado na figura 7C (Fu, 2011). O conjunto dos elementos (i,j) contíguos formam um caminho "torto", do inglês *warping path*, representado por W, tal como mostra a figura 7B a cinzento. Para ser considerado *warping path* terão de ser satisfeitos alguns critérios, nomeadamente que o "caminho" comece e termine na

diagonal, isto é, $w_1=(1,1)$ e $w_k=(p,m)$, tal como denotam o primeiro e o último quadrados pintados na figura 7B. Em segundo lugar, o caminho, do inglês o *path*, terá de ser contínuo, seja $w_k=(a,b)$ e $w_{k-1}=(a',b')$, então $a-a' \leq 1$, assim como $b-b' \leq 1$, limitando o caminho para células adjacentes. O terceiro critério diz respeito à monotonicidade, tomando o exemplo anterior w_k e w_{k-1}, $a-a' \geq 1$ e $b-b' \geq 1$ (Xi *et al*., 2006).

Através dos critérios definidos podem ser encontrados vários *warpings paths*, contudo o caminho com interesse é o que minimiza o custo acumulado obtido através da equação:

$$DTW\left(C_i,Q_j\right)=dist\left(C_i,Q_j\right)+\min\begin{Bmatrix} DTW\left(C_i,Q_{j-1}\right) \\ DTW\left(C_{i-1},Q_j\right) \\ DTW\left(C_{i-1},Q_{j-1}\right) \end{Bmatrix}$$

Na descoberta do melhor *path* pode ser utilizada uma "janela com tamanho definido para procura", tradução aproximada do conceito inglês *warping window*, representada por r e definido em função do tamanho das séries em causa, quanto maior o r menor será a velocidade do DTW.

Comparativamente com a distância Euclidiana o DTW exige um grande esforço computacional (Xi *et al*., 2006). Na figura 8, abaixo, são mostradas duas séries temporais e a forma como a distância entre elas é calculada, na imagem da esquerda é mostrada a distância Euclidiana, na imagem da direita o métodos de cálculo da distância DTW. As linhas representam os pontos para os quais é utilizada a fórmula de distância mostrada na equação 2.5.

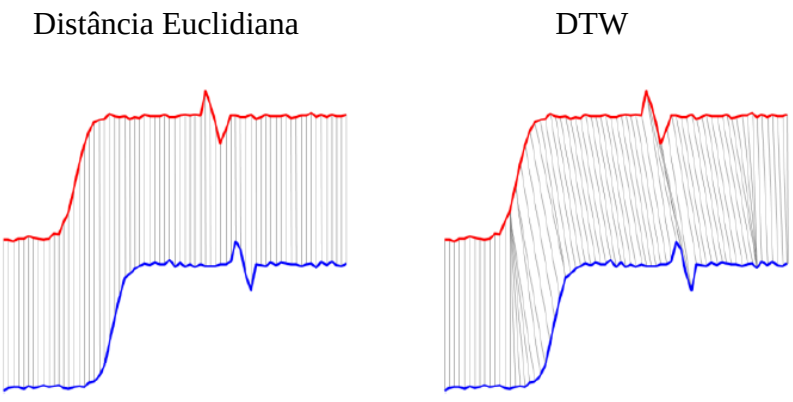

Figura 8 - Cálculo das medidas de similaridade (Ratanamahatana e Keog, 2004).

Capítulo 3 – A Classificação Antecipada em Séries Temporais

A grande maioria dos métodos existentes de classificação de séries temporais extrai padrões de séries completas, sem dar atenção às que se manifestam numa fase inicial, preocupando-se apenas em melhorar a taxa de acerto da classificação (Xing *et al*., 2011), (Xing *et al*., 2008). Todavia, existem campos que tendem a ser sensíveis ao mais fator tempo, veja-se o campo médico, seria útil olhar exclusivamente para os dados iniciais de um paciente e identificar, antes de acontecer, um futuro colapso fatal, ou então, identificar se um tratamento já surtiu efeito e o paciente, sem risco para a sua saúde, pode ser retirado do mesmo (Ghalwash *et al*., 2012).

Através da Classificação Antecipada, do inglês *Early Classification*, é expectável que um dado classificador retorne o instante de tempo a partir do qual possível efetuar uma classificação, sendo o desempenho do classificador associado a esse instante de tempo comparável ao desempenho de um classificador que analise a série temporal completa (Xing *et al*., 2009). Por fim, os resultados desta classificação devem ser interpretáveis, uma vez que a falta de resultados interpretáveis leva a que os peritos no domínio em estudo fiquem reticentes em adotar tal abordagem (Xing *et al*., 2011).

3.1 Modelos de Classificação Antecipada

A Classificação Antecipada não foi sistematicamente estudada (Xing *et al*., 2008). Os primeiros autores que mencionaram o termo "*Early Classification*" foram Rodríguez e Alonso em 2002. Estes apresentaram uma forma de classificação de dados sem os mesmos estarem completos. Isto seria possível através de intervalos de tempo, do inglês *literal*, e junção de classificadores fracos e fortes, do inglês *boosting*. Inicialmente cada *literal* tem o mesmo peso, contudo ao longo das iterações, o peso vai sendo ajustado consoante a classificação, aumentando se for bem classificado pelo classificador ou diminuindo, no caso oposto (Rodriguez e Alonso, 2002).

Mais tarde, Brégon *et al*., 2006 introduziram um sistema que classifica tão cedo quanto possível as falhas de um processo contínuo, a esse sistema chamaram Raciocínio Baseado em Casos, do inglês *Case Based Reasoning* (CBR). Esta metodologia foi definida em quatro passos:

- Recuperação;
- Reutilização;
- Revisão;
- Retenção.

O bom funcionamento deste sistema depende da escolha de um algoritmo de recuperação e de uma medida de similaridade, a escolha dos autores recaiu sobre o algoritmo K-Vizinhos Mais Próximos e sobre a medida *Dynamic Time Warping* (DTW), dado que esta demonstrou melhores resultados (Brégon *et al*., 2006).

No modelo CBR destaca-se o facto de utilizar informação armazenada para a classificação, por conseguinte, o modelo pode incorporar novos casos no sistema a qualquer instante sem recalcular todas as regras de classificação, além disso pode ser usado em séries temporais de diferentes tamanhos, sem que tenha sido treinado para esse tamanho em particular (Brégon *et al*., 2006).

Contudo, os métodos anteriormente propostos apenas se focam em fazer previsões com base na informação parcial, sem a preocupação de qual o menor prefixo que garante uma previsão segura mediante uma determinada confiança (Xing *et al*., 2012). De forma a solucionar a anterior desvantagem, Xing *et al*., 2008 propuseram dois métodos para a classificação de sequências o mais cedo possível:

- Regras de Classificação Sequencial;
- Árvores de Decisão Sequenciais Generalizadas.

No sentido de compreender os modelos que seguidamente são apresentados é necessário definir alguns conceitos. Para que a Classificação Antecipada seja bem sucedida é necessário um classificador que seja serial, isto é, que examine as séries temporais da direita para a esquerda e através dessa análise faça uma previsão da classe a que as séries pertencem com determinada confiança (Xing et al., 2008).

Formalmente, tomando L como o comprimento total de uma sequência, para uma sequência $s=a_1 \ldots a_L$, diz se que a sequência $s' = a_1 \ldots a_{l'}$ é um prefixo de s, tal que, $1 \leq l' \leq L$. Um classificador diz-se serial quando a classe atribuída a uma sequência no instante l_0 se mantém inalterada em instantes seguintes e é exatamente igual à classificação feita para o comprimento total da sequência, ou seja, $C(s[1, l_0])=C(s[1, l_0+1])=C(s[1, l_0+k])=C(s)$, onde $K \geq 0$ e l_0 inteiro e positivo (Xing et al., 2008). O comprimento l_0 ,através do qual se faz a previsão, é denominado custo da previsão e representa-se $Custo(C,s)=l_0$ (Xing *et al*., 2008).

3.1.1 Modelo de Regras de Classificação Sequencial

O modelo Regras de Classificação Sequencial, do inglês *Sequential Classification Rule* (SCR), tem como principal objetivo encontrar regras a partir de um conjunto de dados de treino que sejam representativas de uma determinada classe. Cada regra é composta por uma ou mais características, sendo que uma característica é definida como uma pequena sequência que pode aparecer mais que uma vez numa sequência e é representada por *f* (Xing *et al*., 2008).

Cada regra terá associado um suporte e uma confiança respeitantes à percentagem de sequências na base de dados que confirmam determinada regra e à taxa de acerto de uma regra na classe de uma determinada sequência, respetivamente. No sentido de construir regras para a classificação é necessário encontrar características que constituam essas mesmas regras, para tal existem dois métodos (Xing *et al*., 2008):

- Utilidade
- Características Top-k

No método de seleção por utilidade são definidos requisitos para que uma característica seja útil e consequentemente possa ser empregue em classificação antecipada. O primeiro requisito exige que a característica em causa seja relativamente frequente, uma vez que a frequência nos dados de treino é um indício que será frequente nos dados

futuros. A segunda diz respeito à capacidade discriminativa das características, se essa capacidade for elevada melhor será o desempenho do classificador. Por último, as características com maior interesse são as que se revelam mais cedo (Xing *et al*., 2008).

Inicialmente, no método de utilidade é medido o grau de aleatoriedade da base de dados sequencial em análise, representado pelo grau de entropia, onde p_c é a probabilidade de uma sequência s pertencer à classe c, como demonstra a equação 3.1. Em seguida para cada uma das características, é verificada a respetiva frequência e antecipação como indica a equação 3.2. Por fim, com base nos valores obtidos em 3.1 e 3.2, é verificado ganho de informação de cada uma das características pesado pela antecipação das mesmas, utilizando a equação 3.3 (Xing *et al*., 2008):

$$E(SDB)= - \sum_{c \in C} p_c \log p_c \quad (3.1)$$

$$wsup_{SDB}(f)= \frac{\sum_{f \sqsubseteq s,\, s \in SDB} \frac{1}{||minprefix\ (s,f)||}}{||SBD||} \quad (3.2)$$

$$U(f)=(E(SDB)-E(SDB)_f)^w \, wsup_{SDB}(f) \quad (3.3)$$

Pela análise das equações 3.2 e 3.3 pode-se concluir que quanto menor o prefixo de uma qualquer característica, maior será a sua utilidade. Desta forma, o fator tempo tem um papel decisivo na escolha das características mais indicadas para a previsão antecipada. Analisando atentamente a equação 3.3 existe uma variável "w", definida pelo utilizador, que potencia o ganho de informação em detrimento da antecipação da característica em caso de "w" assumir valores superiores a um (Xing *et al*., 2008).

Contudo, o método de utilidade apresentado pode não ser eficaz uma vez que exige ao utilizador a definição de determinados parâmetros com impacto na extração de características, o que pode culminar num número reduzido de características para a construção de um classificador, ou até em excesso de características. De forma a solucionar esta desvantagem é apresentado método Características Top-k. Este método tem como objetivo encontrar as k características com maior utilidade, começando a sua procura nas características de tamanho um (Xing *et al*., 2008).

A figura 9 é ilustrativa de todo o processo de seleção de características. Como referido, são consideradas todas as sequências de tamanho um e calculada a utilidade de cada uma. As k'< k características com maior utilidade são introduzidas num grupo chamado Semente.

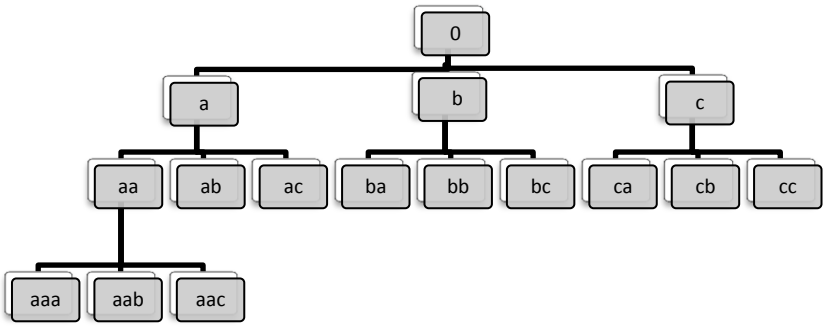

Figura 9 - Método das Característica Top-k (Xing *et al*., 2008)

Para as sequências do grupo Semente é calculada a utilidade dos seus descendentes. Pela análise da figura 9 conclui-se que a característica de tamanho 2 "aa" é descendente da característica de tamanho um "a". Se a utilidade de "aa" for menor que "a", para-se a procura de características nesse ponto, caso contrário o processo é continuado calculando a utilidade de "aaa", descendente de "aa" (Xing *et al*., 2008).

À medida que utilidades mais altas são encontradas nos descendentes, o grupo Semente é confrontado, ou seja, este grupo contém as K características com maior utilidade encontradas, contudo nesse grupo há uma característica à qual pertence a menor utilidade, que define a barreira de valores de utilidade do grupo Semente. Assim, se a menor utilidade do grupo Semente for também menor que a utilidade de um descendente, o descendente é adicionado ao grupo Semente, o valor que define a barreira de utilidade do grupo é atualizado e a característica com menor utilidade é removida do grupo. O processo é repetido até não restar nenhuma característica ou até quando deixar de se verificar uma superioridade na utilidade dos descendentes (Xing *et al*., 2008).

Uma vez encontradas e selecionadas as melhores características, podem ser construídas as regras de classificação. A construção de regras é feita através de uma árvore, onde a raiz está vazia, nos nós imediatamente abaixo encontram-se as características de

tamanho um e nos níveis mais abaixo encontram-se os descendentes, à semelhança da figura 9. Geralmente um nó representa um conjunto de características através das quais é possível classificar, $R_1: f_1 \rightarrow \ldots \rightarrow f_l \Rightarrow c$. Um nó descendente de R_1 é representado por $R_2: f_1 \rightarrow \ldots \rightarrow f_l \rightarrow f_{l+1} \Rightarrow c$ (Xing *et al*., 2008).

As primeiras regras alvo de análise são aquelas cujo custo é previsão é menor, nos nós respeitantes a essas regras são verificadas as classes dominantes, desta forma é possível concluir que $f_1 \Rightarrow c$. O processo é repetido para todos os nós, optando-se por escolher primeiramente as regras com menor custo de previsão e com uma confiança maior que um determinado limite (Xing *et al*., 2008).

Para as sequências do conjunto de treino que não são classificadas com as regras existentes, todo o processos é repetido tem como base apenas essas sequências. De novo se extraem e selecionam e característica como exposto anteriormente (Xing *et al*., 2008).

3.1.2 Modelo Árvores de Decisão Sequenciais Generalizadas

O modelo de Árvores de Decisão Sequencias Generalizadas, do inglês *Generalized Sequential Decision Tree* (GSDT) é um dos métodos de Classificação Antecipada proposto por Xing *et al*., 2008. O objetivo deste método é a construção de um conjunto de características *A* que muito provavelmente estão presentes nas sequências a classificar (Xing *et al*., 2008).

O raciocínio deste método é, também, semelhante ao de uma árvore de decisão. Na raiz encontram-se as características escolhidas para integrar um conjunto *A*, estas características devem dividir os dados de treino em grupos, o grupo das sequências cobertas pela característica um, pela característica dois, pela característica três e assim sucessivamente. Uma dada sequência não é exclusiva de um grupo, podendo existir sequências em dois ou mais grupos, e por conseguinte a robustez do classificador GSDT é aumentada. Para evitar falhas de classificação, o modelo guarda a classe maioritária,

assim quando uma sequência a classificar não é coberta por nenhuma regra, é atribuída a classe guardada (Xing *et al*., 2008).

O método de extração e seleção de características Top-k apresentado na seção anterior é o indicado para a construção do conjunto A.

Contudo, métodos anteriormente expostos, SDR e GSDT, exigem que exista uma discretização dos dados em análise, o que pode resultar numa discretização errada de uma dada série ou em perda de informações importantes (Xing *et al*., 2009)

3.1.3 Modelo de Classificação Antecipada com algoritmo 1-Vizinho Mais Próximo

O modelo de Classificação Antecipada utilizando 1-Vizinho Mais Próximo, tradução do inglês para 1-*Nearest Neighbor Early Classification*, é um modelo que estende o clássico classificador 1-NN para o domínio da Classificação Antecipada através de algumas adaptações (Xing *et al*.,2009).

Como expectável, os dados são divididos em dados de treino e teste, sendo que para os dados de treino é aprendido todo o modelo e retirada toda a informação útil para a classificação dos dados de teste (Xing *et al*.,2009).

O primeiro passo deste modelo é a identificação do comprimento total das séries temporais presentes nos dados de treino, este comprimento total é definido como L (Xing *et al*.,2009). Em seguida para esse comprimento são calculados os vizinhos mais próximos revertidos, do inglês *reverse nearest neighbor*, de cada uma das séries temporais do conjunto de dados de treino (Xing *et al*.,2009). Os *reverse nearest neighbor* (RNN) dizem respeito às séries temporais que tratam uma série do conjunto de treino como sua vizinha mais próxima, estes para o tamanho L são representados como $RNN^{L}(t)$ (Xing *et al*.,2009).

Em seguida a estabilidade dos RNN encontrados em L é explorada para instantes de tempo anteriores a L, começando por verificar os RNN para instante L-1, L-2 e assim sucessivamente, até o conjunto encontrado no instante L sofrer alterações para um qualquer L-i (Xing *et al*.,2009). Formalmente, pretende-se encontrar l<L tal que $RNN^{l}(t) = RNN^{l+1}(t) = \cdots = RNN^{L}(t)$, se esta igualdade se verificar então pode-se concluir que a série temporal t pode ser usada para fazer uma classificação antecipada no instante de tempo *l* (Xing *et al*.,2009).

O passo seguinte deste modelo está relacionado com o cálculo de comprimentos mínimos de previsão, do inglês *minimum prediction length* (MPL), de todas as séries temporais no conjunto de dados de treino. O MPL representa o instante de tempo a partir do qual é possível a série em causa fazer uma classificação antecipada (Xing *et al*.,2009). Contudo, o comprimento mínimo de previsão de uma dada série temporal representado por MLP(t)= k (k≤l≤L) tem de satisfazer determinados requisitos para que possa classificar antecipadamente, nomeadamente: $RNN^{l}(t) = RNN^{L}(t) \neq 0$ e $RNN^{k-1}(t) \neq RNN^{L}(t)$ (Xing *et al*.,2009). Para o caso em que no comprimento L nenhuma série temporal trate uma série t como sua vizinha mais próxima, tem-se que o MPL de t é L, representado por MPL(t)=L (Xing *et al*.,2009). O prefixo mínimo de previsão, do inglês *minimum prediction prefix*, é obtido através de MPP(t)=t[1,MPL(t)] (Xing *et al*.,2009).

Uma vez obtidos todos os MPL, pode-se proceder à classificação das séries temporais de teste. Desta forma, para cada uma série temporal t' no instante de tempo i verifica-se qual o seu vizinho mais próximo, representado por $NN^{i}(t')$, se a série vizinha mais próxima tiver um MPL igual ou inferior a i, pode-se efetuar uma classificação antecipada, o classificador irá atribuir a classe da série vizinha mais próxima (Xing *et al*.,2009). Pelo contrário, se o MPL da série vizinha mais próxima for superior a i, não é possível classificar e consequentemente será necessário esperar por mais valores de t' adiando a classificação (Xing *et al*.,2009).

Os pontos fracos reconhecidos, pelos autores, a este método prendem-se com o facto do mesmo exigir que as séries nos dados de treino sejam estáveis após o MPL, condicionando a possibilidade de encontrar prefixos mais pequenos. A segunda

desvantagem está relacionada com a possibilidade o ajuste aos dados de treino, do inglês *overfitting* (Xing *et al*., 2009). O *overfitting* é um fenómeno que se verifica quando o algoritmo de aprendizagem se adapta aos dados de forma tal, que as variações aleatórias nesses dados são tidas no modelo como significantes (Klawonn e Rehm, 2009).

3.1.4 Modelo de Classificação Antecipada de Séries Temporais

O modelo de Classificação Antecipada de Séries Temporais, do inglês *Early Classification on Time Series* (ECTS), foi proposto com o objetivo de mitigar as principais desvantagens do modelo de Classificação Antecipada utilizando 1- Vizinho Mais Próximo, nomeadamente o *overfitting* e a rigidez do mesmo devido à estabilidade dos RNN que é exigida (Xing *et al*., 2009). Este novo modelo é também uma extensão do método 1-NN *Early Classification*, criando agrupamentos, do inglês *clusters*, através dos dados de treino utilizando o método de Agrupamento Multicamadas Hierárquico pela distância mínima, do inglês Si*ngle Link Multilayer Hierarchical Clustering* (MLHC). Após serem identificados todos os *clusters* é calculado o MPL de cada um (Xing *et al*., 2009).

Os primeiros passos deste método são muito semelhantes aos passos do modelo 1-NN *Early Classification*, para cada série temporal do conjunto de dados de treino são calculados os MPL assim como os respetivos vizinhos mais próximos (Xing *et al*., 2009). Em seguida é aplicado o método MLHC no espaço R^L, sendo agregadas no mesmo *cluster* as séries temporais com menor distância entre si e na iteração seguinte os *cluster* que sejam mutuamente vizinhos mais próximos (Xing *et al*., 2009). Dois *clusters* dizem-se mutuamente vizinhos mais próximos se ambos se tratam como vizinhos mais próximos (Xing *et al*., 2009). De cada vez que um novo *cluster* é formado é calculado o novo MPL, sendo que o *minum prediction length* dos *clusters* obtido através da satisfação dos seguintes requisitos: MPL(s) = k se para qualquer $l \geq k$, $RNN^l(S) = RNN^L(S)$, se S é 1-NN consistente e se para l=k-1 não é possível satisfazer

o primeira ou o segunda requisito enunciado (Xing *et al*., 2009). Um *cluster* é definido como 1-NN consistente se cada uma das séries temporais que o constituí tem como vizinha mais próxima uma série temporal pertencente a esse mesmo *cluster* (Xing *et al*., 2009). O processo termina quando o MPL de cada uma das séries temporais é menor que todos os *clusters* que contêm uma determinada série temporal (Xing *et al*., 2009).

Existem alguns casos especiais neste método que merecem relevo, por exemplo quando é formado um cluster não puro, isto é, quando as séries temporais que constituem um *cluster* têm classes diferentes, o *cluster* em causa não é agregado com mais nenhum *cluster*, terminando a iteração nesse momento para esse *cluster* (Xing *et al*., 2009).

3.1.5 Modelo Relaxado de Classificação Antecipado de Séries Temporais

O modelo Relaxado de Classificação Antecipada de Séries Temporais, do inglês *Early Classification on Time Series Relaxed* (ECTS *relaxed*), tem como base o modelo ECTS anteriormente apresentado (Xing *et al*., 2012). Como visto, o modelo ECTS exige que os RNN sejam estáveis a partir de um determinado instante, todavia esta estabilidade é corrompida por séries temporais mal classificadas, tipicamente, estas séries temporais estão muito próximas da fronteira de decisão e consequentemente aumentam o comprimento mínimo de previsão (Xing *et al*., 2012). As séries temporais pertencentes ao conjunto de treino são mal classificadas são representadas por T_{miss} (Xing *et al*., 2012).

O modelo ECTS relaxado tem em conta o facto de que séries temporais mal classificadas podem esconder padrões discriminativos importantes, assim sendo este modelo ignora a instabilidade provocadas por T_{miss} (Xing *et al*., 2012). Formalmente, para um conjunto de treino T de *full length* L e para um cluster discriminativo S, o MPL(S)=k, se $RNN^{l}(S) \cap (T - T_{miss}) = RNN^{L}(S) \cap (T - T_{miss})$, se no espaço R^{l} o cluster S é 1-NN consistente e se no espaço l=k-1 alguma das condições não for satisfeita (Xing *et al*., 2012).

Na fase de treino do classificador, é expectável que o MPL do ECTS relaxado seja menor do que o obtido pelo modelo ECTS, assim como é expectável que a classificação seja igual em ambos os modelos (Xing *et al*., 2012).

3.1.6 Modelo de Classificação Antecipada através de Formas Discriminativas

A maioria dos métodos existentes de classificação antecipada consegue classificar séries temporais com uma boa taxa de acerto, contudo não extraem nenhuma característica ou padrão que explique a opção do classificador por determinada classe (Xing *et al*., 2011). O modelo *Early Distinctive Shapelet Classification* (EDSC) pretende colmatar as falhas nos modelos até agora apresentados, na medida em que integra formas, do inglês, *shapelets* no classificador (Xing *et al*., 2011).

Uma *shapelet* é uma subsequência de uma série temporal, que pode representar todos os exemplos de uma classe (Ye e Keogh 2009). Uma *shapelet* perfeita será aquela que representa todas as séries temporais de uma classe e não representa nenhuma série temporal das classes restantes (Xing *et al*., 2011). Utilizando a nomenclatura de Xing *et al*.,2011 uma *shapelet* é definida como f=(s,δ,c), onde s é uma série temporal, δ é um limiar para a distância e c a classe alvo.

O modelo EDSC funciona em dois passos, no primeiro são extraídas todas as subsequências entre um tamanho máximo e mínimo, enquanto no segundo passo são selecionadas as melhores subsequências encontradas no primeiro passo (Xing *et al*., 2011).

Para extrair todas as subsequências possíveis dos dados de treino é necessário definir um tamanho mínimo e máximo, sendo extraídas todas as sequências entre o tamanho mínimo e máximo definido (Xing *et al*., 2011). Neste primeiro passo também é importante o cálculo do limiar de distância, para tal podem ser utilizados os métodos:

Best Match Distance, Kernel Density Estimation e Chebyshev Inequality (Xing *et al*., 2011).

No método Best Match Distance (BMD) é encontrada a menor distância entre uma *shapelet* local e cada uma das séries de treino para a classe alvo, obtidas as distâncias estas são ordenadas em ordem ascendente formando uma lista de BMD (Xing *et al*., 2011). Em seguida, para as maiores distâncias encontradas anteriormente são verificadas as *shapelets* que falham na classificação da classe alvo. Através das distâncias que falham a classificação é encontrado um limite de distância que separa as classes, sendo a precisão calculada com base no limite. O limite de distância utilizado é aquele que maximiza a precisão (Xing *et al*., 2011).

O segundo método, Kernel Density Estimation aplica a densidade de kernel às listas de BMD com o objetivo de calcular as funções densidade probabilidade da classe alvo e não alvo, de forma a que a densidade probabilidade de pertencer à classe alvo seja superior ao limiar de distância (Xing *et al*., 2011).

O último método de extração de *shapelets* é o método da Desigualdade de Chebyshev (CHE), este método concentra-se na lista BMD da classe não alvo, tratando estes valores como exemplos de uma amostra aleatórias calculando a sua média e variância (Xing *et al*., 2011). Após o cálculo destas medidas calcula o intervalo para o qual a classe não alvo apresenta pouca probabilidade de aparecer (Xing *et al*., 2011). O limiar de distância é dado pelo δ máximo encontrado e decorrente da fórmula $\delta=\max\{\text{Média}(V_{f,\bar{c}})\text{-k*Var}(V_{f,\bar{c}}),0\}$ (Xing *et al*., 2011).

Uma vez encontradas todas as *shapelets* possíveis através dos métodos apresentados, é necessário selecionar as que demonstram melhor poder discriminativo e antecipação, do inglês *earliness* (Xing *et al*., 2011). A distância antecipada, do inglês *early match distance* (EML), vem responder a essa necessidade, devido ao facto de ser uma medida que incorpora a *earliness* de uma *shapelet* para a classificação de séries temporais, sendo calculada através da fórmula (Xing *et al*., 2011):

$$EML(f,t)= \min_{len(s)\leq i\leq len(t)} dist(t[i\text{-}len(s)+1,i],s] \leq \delta$$

Até ao momento de seleção de *shapelets* é necessário calcular mais duas medidas: a revocação pesada (Revocação W) e a utilidade, em seguida são apresentadas as respetivas fórmulas de cálculo (Xing *et al*., 2011):

$$\text{Revocação W} = \frac{1}{||T_{\bar{c}}||} \sum_{t \in T} \frac{1}{\sqrt[\alpha]{\text{EML}(f,t)}}$$

$$\text{Utilidade (f)} = \frac{2 \text{ x Precisão (f) x Revocação W (f)}}{\text{Precisão (f)+Revocação W (f)}}$$

Obtida a utilidade de cada uma das séries temporais extraídas é necessário proceder à sua ordenação, sendo a *shapelet* com maior utilidade, denominada f1, comparada com os dados treino, diz-se que uma *shapelet* cobre uma série temporal dos dados de treino se a distância for menor que um certo limiar e se a classe da *shapelet* for a mesma que a série temporal a classificar (Xing *et al*., 2011). Então, todas as séries temporais nos dados de treino cobertas pela *shapelet* de maior utilidade são marcadas, mantendo-se as restantes *shapelets* e séries temporais não cobertas (Xing *et al*., 2011). Iterativamente a *shapelet* com segunda maior utilidade é aplicada nas séries temporais não cobertas por f1, tendo que cobrir pelo menos uma dessas séries temporais (Xing *et al*., 2011). O processa continua por ordem decrescente de utilidade até serem selecionadas um determinado número de *shapelets* ou uma percentagem de séries temporais de treino cobertas (Xing *et al*., 2011). As *shapelets* selecionadas podem ser utilizadas imediatamente para classificação (Xing *et al*., 2011).

3.1.7 Modelo de Classificação Antecipada de Séries Temporais Multivariadas

O modelo de Classificação Antecipada de Séries Temporais Multivariadas, do inglês *Early Classification of Multivariate Time Series* (ECMTS), é um modelo de classificação antecipada de séries temporais que generaliza o conceito de *shapelets*

locais para um conceito multivariado (Ghalwash e Obradovic, 2012). Uma *shapelet* multivariada consiste em múltiplos segmentos em que cada um desses segmentos representa uma dimensão (Ghalwash e Obradovic, 2012). Formalmente uma *shapelet* multivariada é representada por $f=(s,l,\Delta,c_f)$, sendo constituída por uma sequência s de comprimento *l* pertencente a uma classe c, sendo c_f a classe objetivo (Ghalwash e Obradovic, 2012). O elemento $\Delta= (\delta^1,\delta^2, \ldots, \delta^N)$ diz respeito aos limiares de distância que têm de ser cumpridos para que no processo de classificação se opte pela classe c_f (Ghalwash e Obradovic, 2012). Exemplificando, a distância entre uma *shapelet* multivariada de N dimensões e uma série temporal de N dimensões é dada pela equação:

$$dist(s,T)=[dist(s^1,T^1), dist(s^2,T^2),\ldots, dist(s^N,T^N)]$$

Para que a série a classificar seja classificada como pertencente à classe c_f é necessário que as várias distâncias da série temporal T sejam inferiores ao limiar de distância definidos para essa dimensão (Ghalwash e Obradovic, 2012).

$$\forall(T_i,c_f) \implies dist(s^j,T_i^j) \leq \delta^j \ \forall_j=1,..N$$

Em linha com os restantes modelos apresentados, as séries temporais em análise são divididas em dados de treino e dados de teste.

Em primeiro lugar para os dados de treino extraem-se todas as *shapelets* de tamanho a l a L, assumindo que todos os segmentos começam na mesma posição (Ghalwash e Obradovic, 2012). Em seguida, calculam-se as distâncias entre as *shapelets* encontradas e todas as séries temporais dessa dimensão, estas distâncias integrarão uma matriz N x M, N relativo ao número de dimensões e M relativo ao número de séries temporais (Ghalwash e Obradovic, 2012). Os passos descritos anteriormente são repetidos para todas as dimensões de todas séries temporais, o que leva a que exista uma grande número de *shapelets*, por este motivo é necessário atribuir uma pontuação às *shapelets* que reflita a sua antecipação e discriminação entre as classes (Ghalwash e Obradovic, 2012). Assim será necessário calcular os limiares de distância obtidos através de dois pontos consecutivos resultantes do ordenamento da matriz, sendo escolhido aquele que

maximiza o ganho de informação pesado proposto pelos autores Ghalwash e Obradovic, 2012, esta medida inclui a antecipação de uma *shapelet* e é calculado da seguinte forma:

$$IG=Entropia-\frac{M_L}{M}-E_L-\frac{M_R}{M}-E_R$$

A variável M_L diz respeito ao número de séries temporais cuja distância a uma determinada shapelet é menor que o limiar de distância, enquanto M_R reflete o número de séries temporais com distância superior ao limiar de distância (Ghalwash e Obradovic, 2012). A variável E_L está relacionada com o cálculo de entropia para as séries cuja distância a uma determinada *shapelet* é menor que um dado limiar (Ghalwash e Obradovic, 2012).

Após o cálculo dos limiares de distância, feita a escolha das *shapelets* que são discriminativas e se revelam mais cedo que as demais, pode-se avaliar o desempenho das mesmas através dos dados de teste. Desta forma, são examinados *l* pontos das séries temporais de teste e chamadas as melhores *shapelets* de comprimento *l* para efetuarem a sua classificação. Se a *shapelet* em causa cobrir a série temporal então é feita a classificação atribuindo classe da *shapelet*. Se não for possível para o comprimento l classificar, aguarda-se por mais pontos (Ghalwash e Obradovic, 2012). Pode dar-se que para o último ponto da série temporal nenhuma classe seja atribuída, nesse caso este método rotula a série temporal como um exemplo não classificado (Ghalwash e Obradovic, 2012).

3.1.8 Modelo Híbrido de Classificação Antecipada

O modelo Híbrido de Classificação Antecipada é um modelo combina o modelo Hidden Markov (HMM) com o poder discriminativo do modelo Máquinas Vetor Suporte, do inglês *Support Vector Machine* (SVM), e que é capaz de classificar antecipadamente, com performance competitiva séries temporais multivariadas (Ghalwash *et al*., 2012). A figura 10, na página seguinte, é ilustrativa dos vários passos deste modelo.

O primeiro passo deste modelo é a divisão dos dados em análise em dados de treino e dados de teste, sendo que os dados de teste definidos nesta fase permanecem intocáveis até ao final do processo (Ghalwash *et al*., 2012). Em seguida, através dos dados de treino é definida uma porção de dados que serve para treino e outra de teste, nomeadas dados de treino para treino e dados de treino para teste (Ghalwash *et al*., 2012). Para os dados de treino para treino são treinados os modelos de HM para comprimento específico e para comprimento diferente do contido no conjunto de treino para treino, representado pela seta com número um na figura 10 (Ghalwash *et al*., 2012). No treino em comprimento específico são extraídos todos os segmentos das séries temporais que variam de 0 e *l*, avançando em seguida um ponto na posição inicial de forma a extrair todos os padrões possíveis (Ghalwash et al., 2012). No treino para comprimento diferente, o processo anterior é repetido desde o instante l até um instante *l*+k, onde k não deve exceder 50% da série temporal (Ghalwash et *al*., 2012).

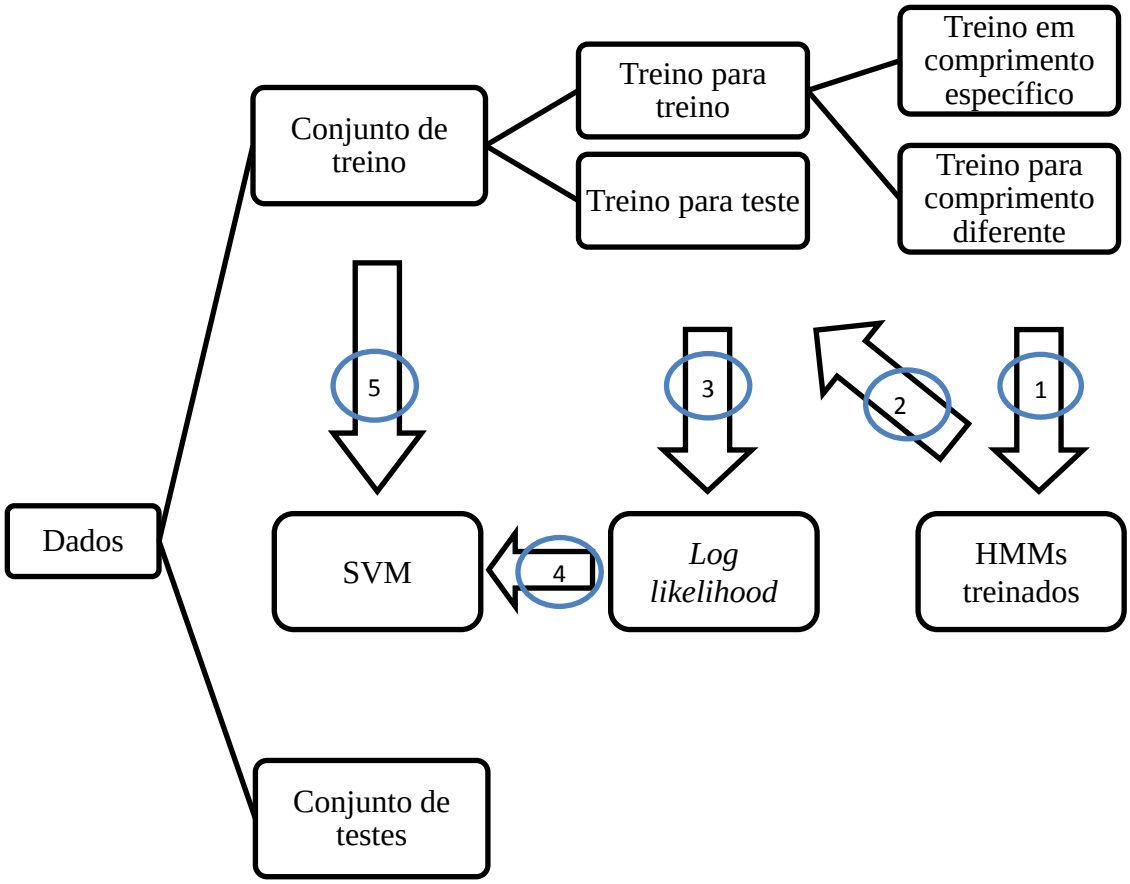

Figura 10 - O modelo Híbrido de Classificação Antecipada

Após conclusão deste processo, no conjunto de dados de treino para testes são aplicados todos os modelos HM treinados anteriormente para obter os respetivas máximas verosimilhanças, do inglês *log likelihoods*, das séries, que servirão de *input* para o modelo SVM (Ghalwash *et al*., 2012). Pormenorizadamente, a primeira sequência obtida através dos dados de treino para treino a ser chamada é a que tem menor comprimento, se esse tamanho *l*, então para cada um dos segmentos dos dados de treino para teste são analisados *l* pontos e os N modelos HM treinados geram os *log likelihoods* (Ghalwash *et al*., 2012).

Todo o processo até este ponto é repetido cinco vezes para obter 5 validações cruzadas e uma vez feitas as repetições desejadas, os *log likelihoods* obtidos serão os *inputs* do modelo, número quatro da figura 10 (Ghalwash *et al*., 2012). Por fim, os parâmetros do modelo SVM obtidos serão otimizados usando a validação cruzada anterior, após otimização, pode-se terinar o modelo no conjunto total dos dados de treino, número cinco da figura 10.

Na fase de teste serão usados os dados de teste iniciais e para cada uma dessas séries temporais pertencentes aos dados de testes serão analisados *l* pontos, calculando os log likelihoods desse tamanho através de todos os modelos HMM (Ghalwash *et al*., 2012). Como dito na fase de treino, os log likelihoods obtidos vão ser os input do modelo SVM (Ghalwash *et al*., 2012). Em seguida, através das probabilidades obtidas pelo SVM, nomeadamente a probabilidade de um exemplo x pertencer às classes y_1 e y_2 : $P(y_1|x)$ e $P(y_2|x)$, calcula-se a diferença $P(y_1|x) - P(y_2|x)$ concluindo a que classe o exemplo x pertence, isto é, se a diferença for superior é zero é mais provável que o exemplo x pertença à classe y_1, se menor que zero será mais provável a classe y_2 (Ghalwash *et al*., 2012). A probabilidade deve ser superior a um parâmetro previamente definido para que se faça uma previsão confiante, caso contrário espera-se por mais pontos da série temporal a classificar (Ghalwash *et al*., 2012).

3.1.9 Modelo de Classificação Antecipada de Séries Temporais com Garantia de Confiança

O modelo de Classificação Antecipada de Séries Temporais com Garantia de Confiança, do inglês, *Early Time Series Classification with Reliable Guarantee*, é um método que garante que a classe atribuída a uma série temporal incompleta é com grande probabilidade igual à classe atribuída pelo classificador quando toda a série temporal está disponível, podendo ser aplicado a problemas de uma ou mais classes (Anderson *et al*., 2012).

Para que este método seja bem sucedido é necessário uma função de classificação do exemplo x, representado por $\hat{g}(x)$ e um conjunto A englobem a probabilidade da classe atribuída no instante i ser igual à atribuída no instante L, i<L, a probabilidade é representada por τ (Anderson *et al*., 2012). As variáveis anteriormente definidas deve cumprir a regra de decisão:

$$\hat{g}(x)=g \text{ para todo } x \in A \text{ e para algum conj. A tal que } P(X \in A | Z=z) \geq \tau$$

Atentando à equação acima, são necessários três passos antes de obter a função de classificação, o calculo da densidade condicional P(x|z), o conjunto A, cálculo da probabilidade $P(X \in A | Z=z)$ e verificar se é superior a τ (Anderson *et al*., 2012).

O conjunto A pode ser construído através dos métodos Chebyshev, Naive Bayes Quadratic e Naive Bayes Box, podendo visualiza-se na figura 11 as diferenças dos métodos citados (Anderson *et al*., 2012).

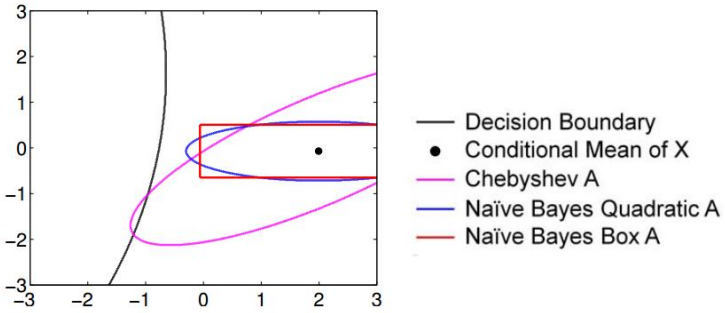

Figura 11 - Métodos para construção do conjunto A, retirado de Anderson *et al*., 2012

No seu estudo Anderson *et al*., 2012 optaram pelo método Naive Bayes Quadratic devido ao facto de ser uniformemente melhor e do método de Chebyshev ser demasiado conservativo. Apenas é atribuída uma classe ao exemplo a classificar se o conjunto A não coincidir com a fronteira de decisão.

Este método pode ser usado em problemas de duas ou mais classes. Sejam o conjunto de classes G={1,2} , $f_1(x)$ e $f_2(x)$ representam as funções discriminativas das classes definida em G. Calculando a diferença $f(x)=f_2(x)-f_1(x)$ pode-se obter:

$$\hat{g}(z)=\begin{cases} 1, & \max_{x\in A} f(x)\leq 0 \\ 2, & \min_{x\in A} f(x)>0 \\ \text{nenhuma decisão}, & \text{caso contrário} \end{cases}$$

No método destinado a problemas multi classe, todas as classes, denominadas por G, são tomadas como candidatas, escolhendo quaisquer dois pares de classe e verifica-se $\min_{x\in A} f_c(x)- f_h(x)\geq 0$. Se a diferença for superior a zero, diz-se que h domina c, caso contrário é verificada a desigualdade $\min_{x\in A} f_h(x)- f_c(x)\geq 0$, se for superior a zero, diz-se que a classe c é dominante e h dominada. Se o valor obtido não for superior a zero as classes são deixadas de fora. O processo termina quando existir uma classe dominante ou quando não for possível classificar.

$$\hat{g}(z)=\begin{cases} c, & \min_{x\in A} f_c(x)- f_h(x)\geq 0 \text{ para todos } c\neq h \\ \text{nenhuma decisão}, & \text{caso contrário} \end{cases}$$

No seu trabalho os autores referem a importância do pré processamento dos dados, nomeadamente a redução da dimensionalidade. Pretende-se encontrar um espaço, mais pequeno que o atual, onde as classes se encontram bem separadas, através da redução da dimensionalidade linear. Assim, seria diminuído o impacto do ruído e de características não discriminativas dos dados, que levaria a uma acurácia maior. Usaram Local discriminative Gaussian porque discrimina bem os dados multi classe, é rápido e tem um bom desempenho mesmo quando há pouco exemplos de treino e a são de elevada dimensionalidade.

3.1.10 Modelo de Classificação com Opção de Rejeição

O modelo de Classificação com Opção de Rejeição, do inglês *Classifier With a Reject Option* (CWRO), tem como principal objetivo a classificação antecipada em tempo real. O foco deste modelo advém do facto de todos os modelos até agora apresentados terem como pressuposto que os dados em análise estão armazenados em disco e são processados *offline* e consequentemente o tempo de classificação pode aumentar. A antecipação deste modelo da classificação deve-se ao facto do algoritmo efetuar a classificação com base de numa porção da informação temporal disponível (Hatami e Chira, 2013).

Este modelo de classificação antecipada utiliza dois algoritmos para efetuar a classificação de uma qualquer série temporal, utilizando o seu grau de concordância para aceitar ou rejeitar uma dada classe, como é visível na figura 12 (Hatami e Chira, 2013).

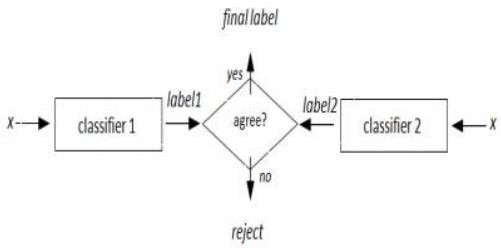

Figura 12 - Modelo de Classificação Antecipada com Opção de Rejeição (Hatami e Chira, 2013)

O modelo obtido calcula a probabilidade de uma série temporal pertencer a uma qualquer classe, se esta percentagem for inferior a um dado limite, espera-se por mais informação, consequentemente é aumentado o custo de classificação. O processo continua até se atingir um determinado patamar de confiança ou o custo de obter mais informação for demasiado elevado. Os autores definem o custo de uma decisão errada como 1 e o custo de adiar a decisão maior que zero e menor que 0,5 (Hatami e Chira, 2013).

Capítulo 4 - Estudo de Caso

4.1 O Problema

No presente capítulo pretende-se classificar frequências cardíacas fetais recolhidas antes do nascimento de um bebé, utilizando técnicas de classificação antecipada com o objetivo de verificar a existência de algum tipo de comportamento na fase inicial da recolha da frequência cardíaca fetal que ajude a prever o desfecho do parto.

A base de dados em estudo é composta por vinte e oito frequências cardíacas fetais e a cada uma delas está associada uma classe, bem ou mal. Utilizando a terminologia de Xing *et al*., 2009 o conjunto de classes possíveis define-se como: C={Bem, Mal}. Dado que um sinal de frequência cardíaca fetal pode ser considerado como uma série temporal, a sua análise será formulada como um problema de classificação de séries temporais. Posto isto, serão aplicadas algumas técnicas de *Data Mining* Temporal apresentadas no capítulo anterior, para dar resposta às questões:

É possível classificar o desfecho de um parto antes de se ter acesso à informação completa? Se sim, qual é esse instante de tempo? Qual é o melhor método para a classificação antecipadas das séries temporais?

4.2 Dados do Estudo de Caso

As frequências cardíacas fetais sobre as quais a análise vai incidir foram recolhidas na tese de doutoramento de Maria Antónia Costa em 2010, intitulada "*Development and Evaluation of a Combination of Computer Analysis of Cardiotocography and Electrocardiography for Intrapartum Fetal Monitoring"*. Estas foram recolhidas durante o trabalho de parto, na fase ativa, a mães grávidas de um feto com, pelo menos, mais de trinta e seis semanas de gestação e ao feto é desconhecido qualquer tipo de má formação (Costa, 2010). Do total das frequências cardíacas fetais dezoito tiveram um desfecho positivo e os restantes dez mau desfecho.

Como referido no segundo capítulo, os primeiros passos do processo de KDD são relativos à identificação e compreensão da natureza dos dados. Seguindo esta linha de raciocínio, na próxima subseção parte é feita uma breve descrição acerca de *Data Mining* em dados médicos, enumerando principais vantagens e limitações, assim como uma descrição da cardiotocografia, visto que é a partir deste método que se obtém as frequências cardíacas fetias sobre as quais a análise apresentada se irá debruçar.

4.2.1 Extração de Conhecimento de Dados Médicos

A aplicação de técnicas de análise de dados e de D*ata Mining* foram extensamente exploradas e utilizadas em áreas como a indústria e negócios, contudo quando comparado com dados médicos a sua aplicação foi menor (Hosseinkhah *et al*., 2009). Apesar dos dados médicos serem os mais observados, são também o tipo de dados biológicos mais difíceis de explorar e analisar (Cios e Moore, 2002). Aprender um classificador para este tipo de dados é de extrema utilidade tanto para a monitorização de pacientes, como previsão de resultados de um tratamento e, até mesmo, suporte à decisão médica (Batal *et al*., 2013).

As bases de dados médicas possuem particularidades específicas que devem ser tomadas em conta no momento da sua análise, nomeadamente a heterogeneidade dos dados, o volume, a complexidade, assim como restrições éticas e legais relacionadas com a sua utilização (Cios e Moore, 2002). Como principais limitações a apontar à análise dados médicos (Koh e Tan, 2005):

- Existem poucos dados disponíveis ao público em geral, estes ficam confinados a clínicas e laboratórios;
- Nas bases de dados disponíveis faltam de dados, existem dados errados e inconsistentes, assim como dados que provêm de fontes e formatos diferentes;
- O sucesso da aplicação de técnicas *Data Mining* em dados médicos depende do conhecimento na área;

- O desenvolvimento de aplicações médicas requere investimento, tempo e esforço.

Apesar de todas as limitações enumeradas, os métodos de *Data Mining* podem dar o seu contributo no sentido de melhorar a qualidade dos tratamentos, aumentar a taxa de sobrevivência, assim como no processo de tomada de decisão, seja através da descoberta de padrões ou tendências (Mao *et al*., 2012)(Koh e Tan, 2005). Atualmente as decisões clínicas são, maioritariamente, tomadas com base na experiência e intuição dos médicos em detrimento do conhecimento que se pode obter nas bases de dados (El-Sappagh *et al*., 2013). Um estudo do Instituto de Medicina estimou que 44.000 a 98.000 de americanos morrem todos os anos devido a erros médicos, originando um prejuízo, com base um estudo feito em 1999, de pelo menos 17 biliões de dólares (Hauskrecht *et al*., 2013).

4.2.2 Cardiotocografia

A cardiotocografia é um método que possibilita aferir o bem estar fetal através da contínua monitorização da frequência cardíaca fetal e de contrações uterinas (Luzetti *et al*., 1999). O bem estar fetal resulta do normal funcionamento da transferência do sangue materno para a placenta e, através do funcionamento próprio desta, da transferência do oxigénio presente no sangue materno para o sangue fetal (Neilson, 2013).

O trabalho de parto é uma situação potencialmente ameaçadora ao bem estar fetal, visto que fortes contrações uterinas param o fluxo de sangue materno para a placenta comprometendo a oxigenação fetal (Neilson, 2013). A hipóxia, resultante da falta de oxigenação, representa uma boa parte dos partos mal sucedidos, todavia mais de 50% dos partos com mau desfecho são causados pelo não reconhecimento de padrões de frequência cardíaca fetal (Chinnasamy *et al*.,2013). Na realidade, ao longo do trabalho de parto existem variadas mudanças no traçado da CTG que não são claras e que podem, ou não, acarretar ameaças à vida fetal. Passadas algumas décadas desde a

introdução da CTG nas rotinas clínicas, a capacidade preditiva dos algoritmos de *Data Mining* ainda não é suficiente (Luzetti *et al*., 1999) (Chinnasamy *et al*.,2013).

Apesar de não existirem números concretos do número de mortes feitas no parto devido à má oxigenação, sabe-se que em 2002, segundo dados Direção de Serviços de Informação e Análise da Direção-Geral da Saúde em 2013, morreram 192 fetos devido a hipóxia.

A figura 13 é ilustrativa de processo de extração de frequência cardíaca fetal. Esta pode ser feita de forma externa, através de um sensor colocado no abdómen, como mostra a figura, ou de forma interna através de um elétrodo colocado na cabeça de um bebé (Alfirevic *et al*.,2006).

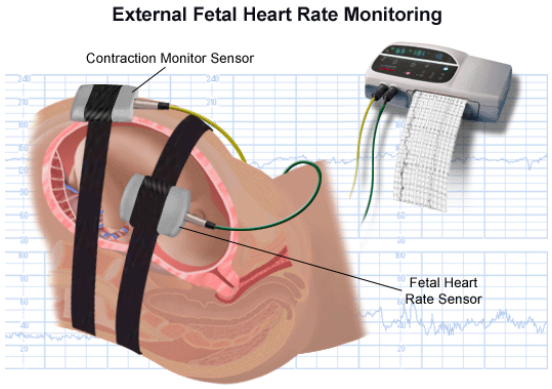

Figura 13 - Método de recolha de CTG, Fonte: Johns Hopkins Medicine

4.3 Modelo em Estudo

Como mencionado, os dados disponíveis são relativos a frequências cardíacas fetais, sendo o objetivo primário da presente dissertação aplicar técnicas de *early classification* de forma a encontrar um instante de tempo em que é possível prever o desfecho de um parto de forma proveitosa. Se num instante inicial de um parto for possível prever que este terá um mau desfecho, pode-se intervir no sentido de evitar o mau desfecho previsto.

O modelo escolhido para aplicar aos dados foi o modelo de Classificação Antecipada utilizando 1-Vizinho mais Próximo proposto por Xing *et al*., 2009, devido à sua simplicidade e bons resultados como visto na revisão de literatura. Em seguida é apresentado o pseudo-código a aplicar na base de frequências cardíacas fetais.

Algoritmo 1: Classificação Antecipada RNN

```
1   L<-full length
2     for each t_i ϵ T
3       compute RNN^L(t_i), i =1:28
4         if  RNN^L(t_i) = 0
5           remove t_i from T
6     end for
7     for each=(L-1):1
8       for each t ϵ T
9           repeat compute RNN^l(t)
10          until  RNN^L(t)≠ RNN^l(t)
11      end for
12    end for
13  return l, RNN^L(t), RNN^l(t)
```

Atentando no primeiro ponto do algoritmo um é de notar que, L assume o valor referente ao comprimento total das séries temporais e será o tamanho de referência. No ponto dois e três, para cada uma das séries temporais pertencentes a um conjunto de treino são calculadas as séries temporais que as tratam como vizinhas mais próximas, se, de acordo com o ponto quatro e cinco, nenhuma série temporal a tratar como vizinha mais próxima e por conseguinte o conjunto de *reverse nearest neighbor* retornar vazio, a série é removida do conjunto de treino.

O ponto cinco não é mencionado explicitamente no trabalho de Xing *et al*., 2009, contudo por uma questão de rapidez do algoritmo e menor custo computacional optou-se pela remoção de séries sem *reverse nearest neighbor*. Além disso como dito na revisão bibliográfica, o algoritmo em causa tem como base a estabilidade dos RNN, ou seja, para ser possível efetuar uma classificação antecipada é exigido que o conjunto de RNN se mantenha constante em instantes L-1, L-2, comparativamente com o instante L, o que implica a existência de RNN em L.

Uma vez removidas, de T, as séries cujo RNN é igual a zero, para as restantes séries é verificada a existência de RNN para instantes de tempo L-1, L-2, L-i, dado i=1:L-1, e assim sucessivamente até se verificar uma alteração relativamente ao conjunto de RNN encontrado em L. Nesse momento o algoritmo devolve o instante de tempo *l* em que a série temporal t deixou de ter como RNN os mesmo que em L, assim como o conjunto de RNN em L e em *l*.

Através da informação que o algoritmo devolve podem-se tirar conclusões relativamente aos instantes a partir dos quais se podem fazer previsões.

4.4 Aplicações do Modelo em Estudo

Nesta subseção irão ser apresentadas as experiências realizadas com o intuito de responder às questões formuladas no início deste capítulo. Numa primeira fase não irá ser feito qualquer tipo de pré processamento uma vez que se irá utilizar o classificador 1-NN com distância Euclidiana e como indicaram Xing *et al*., 2009, o método a aplicar não requere discretização. Como medida de dissemelhança serão utilizadas as distância Euclidiana e *Dynamic Time Warping*, uma vez que não há nenhuma evidência clara que uma seja superior à outra, enquanto uma dada medida de similaridade é mais eficiente em certo tipo de dados, noutra base de dados pode ser a distância com pior desempenho (Wang *et al*., 2013). Por isso apenas se vão tirar conclusões relativamente às medidas de similaridade após serem aplicadas nos dados em causa.

O comprimento das séries temporais em estudo é variável oscilando entre 8.180 e 189.056 pontos. No entanto, para aplicação de medidas de similaridade como a distância Euclidiana é necessário que todas as séries em análise tenham o mesmo comprimento (Mitsa, 2009). Desta forma, o tamanho de todas as séries foi reduzido à série mais pequena, denominada bem9, cujo comprimento é de 8180 pontos. As séries com comprimento maior foram cortadas pelo final, ou seja, para a série bem1 como comprimento de 13.994 pontos apenas serão alvo de análise os últimos 8180 pontos.

Utilizando a terminologia de Xing *et al*., 2009, o *full length* será assumido como 8180, desta forma representa-se L = 8180 e o *full lengthspace* é R^{8180}.

4.4.1 Aplicação do Modelo 1-NN com Distância Euclidiana

Na primeira experiência realizada utilizou-se a distância Euclidiana e calculou-se as vizinhas mais próximas das séries temporais disponíveis na base de dados para o comprimento L=8180. Desta forma, foi possível construir a tabela 2, onde na primeira coluna, denominada base, são identificadas todas as séries disponíveis na base de dados e na segunda coluna, NN sinónimo de *nearest neighbor*, está identificada a série mais próxima da série base.

Tabela 2 - Vizinhos mais próximos L=8180

base	NN	Base	NN
bem1	mal3	**bem15**	bem11
bem2	mal3	**bem16**	bem11
bem3	mal3	**bem17**	mal3
bem4	mal5	**bem18**	bem11
bem5	bem11	**mal1**	mal3
bem6	bem8	**mal2**	mal3
bem7	bem12	**mal3**	bem2
bem8	mal3	**mal4**	mal5
bem9	bem11	**mal5**	mal4
bem10	bem6	**mal6**	bem13
bem11	bem13	**mal7**	mal10
bem12	bem8	**mal8**	bem11
bem13	bem11	**mal9**	bem11
bem14	bem11	**mal10**	bem15

Pela análise da tabela 2 conclui-se que a série mais próxima da série bem13 é a série bem11, de facto, analisando a matriz de dissemelhança presente no Anexo 7.1, resultado do cálculo da distância Euclidiana entre todas as séries temporais disponíveis, a série com menor distância a bem13 é bem11 com 2.959.519. Pela análise da figura 14 são

visualmente percetíveis as diferenças e semelhanças entre as séries bem11 a vermelho e bem13 a preto.

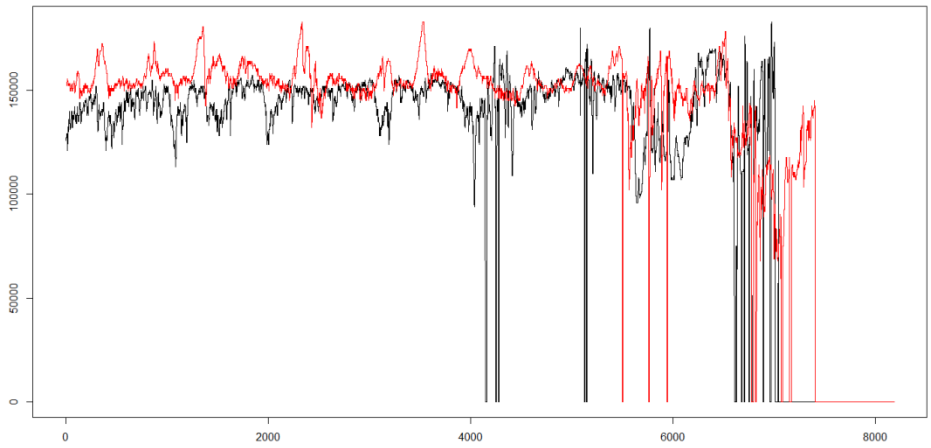

Figura 14 - Séries vizinhas mais próximas

O segundo passo será avaliar o desempenho do classificador empregue nesta experiência para os dados em análise, ainda com L=8180. A base de dados foi dividida em dados de treino, T, e dados de teste, T', o método de amostragem utilizado nesta experiência foi a amostragem aleatória. Foram feitas cem amostragens aleatórias e calculado o desempenho do classificador para as mesmas, no final, através da média dos cem desempenhos, obteve-se uma taxa de aproximadamente 64,5%.

Uma das amostras aleatórias ditou como pertencente aos dados de teste o conjunto T'={bem2, bem7, bem10, bem14, mal7, mal8}. O objetivo será classificar cada uma das séries $t' \in T'$ com base nas séries de treino. Utilizando os resultados da classificação construiu-se a tabela 3, a primeira coluna, t', representa a série do conjunto de teste que se pretende classificar, na segunda coluna, $C(t')$, encontra-se a classe atribuída pelo classificador, por sua vez, na terceira coluna encontra-se verdadeira classe da série t'.

Tabela 3 - Classificação de séries de testes

t'	$C(t')$	$t'.c$
bem2	Mal	Bem
bem7	Bem	Bem
bem10	Bem	Bem
bem14	Bem	Bem
mal7	Mal	Mal
mal8	Bem	Mal

Pela análise da tabela conclui-se que a classe atribuída pelo classificador 1-NN à série bem2 é a classe mal, uma vez que nos dados de treino a classe da série vizinha mais próxima, utilizando a distância Euclidiana, é a classe mal, consultando a tabela 2 verifica-se que a série mais próxima é a série mal3. Contudo a verdadeira classe desta série é a classe bem, como mostra a terceira coluna da tabela.

Através da análise da tabela 3 é possível construir uma matriz de confusão, onde são identificadas o número de séries que pertencem à classe Bem e que de facto foram classificadas como pertencentes a essa classe, são também identificadas o número de séries da classe Bem identificadas como pertencentes à classe mal, assim como a mesma análise é feita para a classe Mal. Como visto na tabela 3, três séries temporais pertencentes à classe Bem são classificadas como pertencentes à série bem, então essas três séries vão integrar o grupo de verdadeiros positivos. Apenas uma série temporal pertencente à classe Mal é classificada como pertencente a essa classe, assim apenas um registo será integrado no grupo de verdadeiros positivos. No entanto duas séries foram mal classificadas, a série bem2 pertencente à classe Bem foi classificada como pertencente à classe mal, assim esta integrará o grupo de falsos negativos, por seu lado, a série mal8 foi classificada como pertencente à classe Bem, esta classificação faz com que a série integre o grupo de falsos positivos.

Tabela 4 - Matriz de Confusão Classificação Antecipada 1-NN

		Previsto	
		Bem	Mal
Real	Bem	3	1
	Mal	1	1

Como mencionado no capítulo dois, muitas vezes o desempenho do classificador não é suficiente para avaliar o mesmo, exemplo disso é o caso em estudo, ou seja, é aceitável um alerta de mau desfecho quando na verdade o parto vai correr bem, mas não se pode tolerar que o classificador preveja um desfecho positivo quando na realidade terá um mau desfecho, como é exemplo a série mal8. O custo associado ao último caso

reportado é muito superior ao primeiro, logo será sempre preferível um classificador com elevada taxa de acerto para a classe Mal.

As medidas de especificidade e sensibilidade podem ser, também, calculadas através da matriz de confusão. De acordo com as equações (2.1) e (2.2) definidas no segundo capítulo, as taxas de especificidade, ou taxa de acerto para a classe Mal, e a sensibilidade, taxa de acerto para a classe Bem, de acordo com a tabela 4 assumem valores de 50% e 75%, respetivamente. No caso genérico, resultado das médias das cem amostragens aleatórias realizadas, obteve-se uma taxa de especificidade de 45% e uma taxa de sensibilidade de 74,5%. Podendo concluir-se que o modelo utilizado reconhece melhor a classe Bem do que a classe Mal, acabando por ser um ponto desfavorável, dado que o custo de falhar esta classe é muito superior.

Após calcular as medidas de avaliação de desempenho do modelo para L=8180, pode-se passar para a segunda parte do modelo, descrita no Algoritmo um. A primeira parte relaciona-se com o cálculo dos *reverse nearest neighbor* para cada uma das séries na base de dados. Pela análise da tabela 2 podemos retirar essa informação começando por questionar "Quem é a série vizinha mais próxima série bem7?". Respondendo à questão, verifica-se que a série bem7 trata como sua vizinha mais próxima a série bem12, contudo a série bem12 não trata a série bem7 como sua vizinha mais próxima, pois bem7 ter bem12 como sua vizinha não implica que bem7 seja a vizinha mais próxima de bem12, o que vai de encontro ao formulado por Korn e Muthukrishnan, 2000. Assim, pela análise da coluna NN da tabela 2 verificam-se as séries cujo *reverse nearest neighbor* é diferente de zero, exemplificando, a série bem7 não está presente na segunda coluna da tabela 2, o que significa que nenhuma série temporal a trata como vizinha mais próxima. Utilizando a nomenclatura de Xing *et al*., 2009: RNN^{8180}(bem7) = 0.

Agindo de acordo com os pontos três e quatro do Algoritmo 1, a série bem7 é retirada do conjunto T uma vez que o conjunto *de reverse nearest neighbor* é zero. Voltando a analisar a tabela 2 verificamos que as séries bem2, bem6, bem8, bem11, bem12, bem13, bem15, mal3, mal4, mal5 e mal10 se mantêm para análise dos restantes instantes de tempo uma vez que se encontram na coluna NN dessa tabela, consequentemente se conclui que os respetivos *reverse nearest neighbor* são diferentes de zero. À semelhança

de bem7, as restantes séries são eliminadas dado que nenhuma série as trata como vizinhas mais próximas, explicando a sua ausência da coluna NN.

Da execução dos pontos oito a dez do Algoritmo 1, para a série temporal bem11 no instante obtém-se um RNN^{8180}(bem11) ={bem5, bem9, bem13, bem14, bem 15, bem16, bem18, mal8, mal9}, em seguida calcula-se o conjunto de *reverse nearest neighbor* de bem11 no espaço 8179, verificando se o conjunto se manteve igual quando comparado com o conjunto de RNN do espaço 8180, se sim volta-se a calcular o conjunto reverse *nearest neighbor* para 8178 e assim sucessivamente até se verificar uma alteração nesse conjunto, como descrito no ponto dez do algoritmo.

Tabela 5 - Vizinhos mais próximos ao longo do tempo

	8180	**8179**	**8178**	**8177**	**[…]**	**7981**	**7980**	**7979**
bem5	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
bem9	bem11	bem11	bem11	bem11	bem11	bem11	bem11	mal3
bem13	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
bem14	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
bem15	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
bem16	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
bem18	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
mal8	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11
mal9	bem11	bem11	bem11	bem11	bem11	bem11	bem11	bem11

Pela análise da tabela 5 confirma-se que as séries temporais que tratam bem11 como sua vizinha mais próxima mantêm-se entre os instantes de tempo 7980 a 8180, havendo no instante 7979 uma alteração notada a amarelo. Nesse instante a série bem11 é destituída de vizinha mais próxima da série bem9, passando a ser mal3 a vizinha mais próxima.

Desta forma utilizando a nomenclatura de Xing *et al.*,2009 tem-se que $RNN^{8180}(bem11) = RNN^{8179}(bem11) = \ldots = RNN^{7980}(bem18)$ como mostra a tabela cinco, que $RNN^{8180}(bem11) = RNN^{7980}(bem11) \neq 0$ e que $RNN^{8180}(bem11) \neq RNN^{7979}(bem11)$. Assim, conclui-se que o comprimento mínimo de previsão referente à série temporal bem11 é de 7980, este comprimento representa, também, o instante a partir do qual se pode usar bem11 para fazer classificações antecipadas. Utilizando novamente a nomenclatura de Xing *et al.*,2009: MPL(bem11) = 7980.

Tendo em consideração que os pontos das séries temporais chegam em ordem ascendente é de importância fulcral perceber, para todas as séries temporais, os instantes de tempo a partir dos quais se pode fazer uma classificação. Se existir uma série temporal para classificar no instante de tempo i, será calculado o vizinho mais próximo dessa série para o instante i, apenas lhe será atribuída a classe do vizinho mais próximo no instante i se o *minimum prediction length* do vizinho mais próximo for maior ou igual ao instante de tempo i (Xing *et al*., 2009).

Dada a importância do *minimum prediction length* para a classificação antecipada de séries temporais, é apresentado o valor deste indicador chave para cada uma das séries presente na base de dados, obtendo a tabela 6 abaixo.

Tabela 6 - Comprimento Mínimo de Previsão

Base	MPL	Base	MPL
bem1	8180	**bem15**	6505
bem2	7419	**bem16**	8180
bem3	8180	**bem17**	8180
bem4	8180	**bem18**	8180
bem5	8180	**mal1**	8180
bem6	7963	**mal2**	8180
bem7	8180	**mal3**	7934
bem8	7803	**mal4**	6234
bem9	8180	**mal5**	7313
bem10	8180	**mal6**	8180
bem11	7980	**mal7**	8180
bem12	7431	**mal8**	8180
bem13	7427	**mal9**	8180
bem14	8180	**mal10**	727

Como visto no exemplo anterior o MPL de bem11 é 7980, por isso na coluna correspondente à série bem11 vê-se esse valor. Atentando aos MPL das restantes série, nota-se que algumas séries, como bem1, apontam para o valor de 8180. Este valor é atribuído às séries que não são tratadas como vizinhas mais próximas de uma outra série.

Continuando a análise da tabela seis, é possível visualizar um MPL salientado a amarelo, este corresponde ao instante mais antecipado a partir do qual é possível

realizar uma previsão. Dado que é de uma série temporal da classe Mal, chama-se a atenção para o potencial desta série. De notar, se no instante 727 existir alguma série temporal, representada por t600, que trate mal10 como sua vizinha mais próxima, esta é possível classificar com a classe Mal. Este é um caso em que deve existir uma intervenção médica, que pode salvar uma vida, uma vez que t600 muito provavelmente terá o mesmo desfecho de mal10. Na figura 15 abaixo é percetível o comportamento geral da série ao longo do tempo, a vermelho encontra-se notado o comportamento inicial da série, que leva a crer que esta terá mau desfecho. Na figura 16 pode-se visualizar de forma mais aproximada o comportamento inicial desde o instante 1 até ao instante 727.

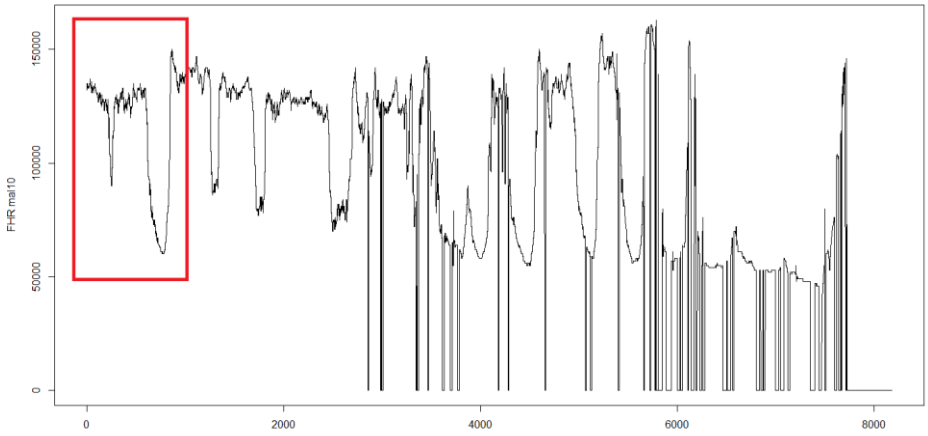

Figura 15 - Frequência cardíaca fetal da série temporal bem10

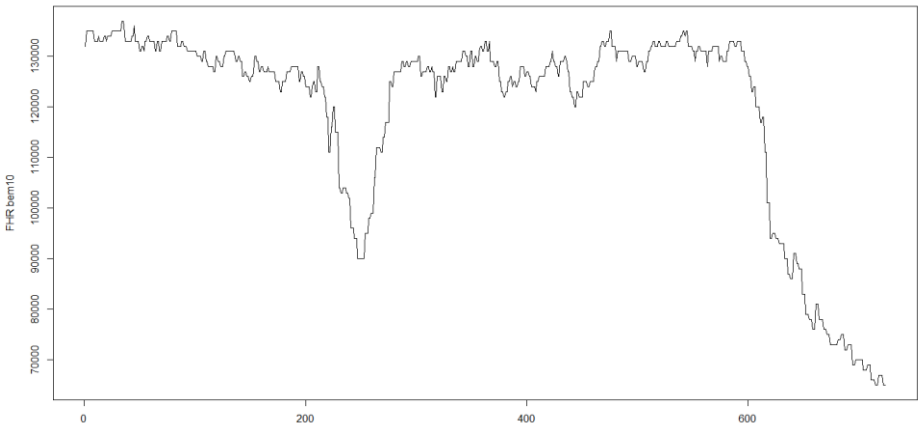

Figura 16 - Frequência cardíaca fetal da série bem 10 entre os instantes 1 a 727

Também pode acontecer que t600 no instante 727 trate mal4 como sua vizinha mais próxima, neste caso não se pode classificar t600, sendo necessário esperar pela chegada de mais pontos. A cada instante de tempo seguinte a 727 é verificada a série mais próxima da série t600, se o MPL da série vizinha mais próxima for superior ou igual ao instante de tempo é possível classificar.

No instante de tempo 7.000 existem três séries temporais através das quais é possível efetuar uma classificação, bem15, mal4 e mal10, se t600 tratar alguma das séries enumeradas como vizinha mais próxima então é atribuída a classe da série mais próxima. Caso nenhuma das séries enumerada seja a vizinha mais próxima, então é necessário esperar por mais pontos, no limite apenas pode ser possível classificar uma série temporal no instante 8180, por exemplo se a série t600 tratar bem1 como sua vizinha mais próxima desde o instante de tempo 1 até 8180.

4.4.2 Aplicação do Modelo 1-NN com Distância Dynamic Time Warping

Na segunda experiência pretende-se repetir os testes realizados na experiência 4.4.1, mas com diferente medida de similaridade, nesta experiência será utilizada o *Dynamic Time Warping*. O primeiro passo será calcular, através deste método, os vizinhos mais próximos das séries em análise, o desempenho do classificador 1-NN, a respetiva sensibilidade e especificidade para o comprimento total das séries temporais L=8180.

Assim, alinhando cada uma das séries temporais com as restantes séries, calculou-se a distância entre as mesmas obtendo-se uma matriz de similaridade disponível no Anexo 7.2. As séries bem6 e bem10 através da distância Euclidiana distavam 5.196.554, através da distância DTW distam 11.988. Esta diferença no valor das distâncias vai-se verificar para todas as frequências cardíacas fetais, uma vez que a distância DTW é sempre menor ou igual à distância Euclidiana devido à minimização da distância acumulada através no *warping path*.

Um exemplo de alinhamento entre séries temporais pode ser visualizado na figura abaixo.

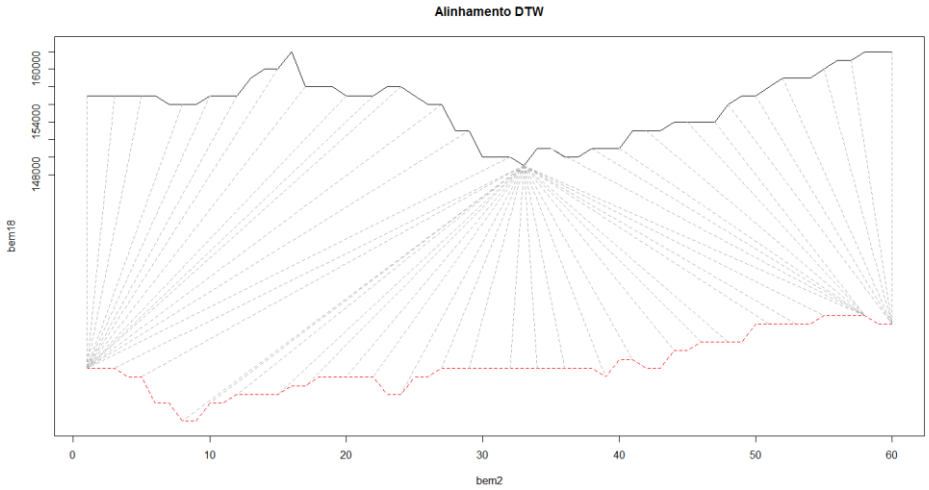

Figura 17 - Alinhamento DTW

Através da aplicação do classificador 1-NN DTW obteve-se a tabela 7. Esta tabela é composta pelas mesmas colunas da experiência anterior, relembre-se a tabela2, na primeira coluna são identificadas todas as séries temporais presentes na base de dados, na segunda coluna encontra-se a série vizinha mais próxima da série identificada na primeira coluna. Pela análise do quadro nota-se que a série bem1 utilizando a distância DTW trata como sua vizinha mais próxima a série bem4.

Tabela 7 - Vizinhos mais próximos DTW

base	NN
bem1	bem5
bem2	mal3
bem3	mal1
bem4	mal5
bem5	mal1
bem6	mal3
bem7	bem8
bem8	bem13
bem9	mal3
bem10	bem9
bem11	bem14
bem12	bem13
bem13	mal3
bem14	bem11

base	NN
bem15	bem13
bem16	bem9
bem17	bem8
bem18	bem11
mal1	mal2
mal2	bem3
mal3	mal1
mal4	bem6
mal5	bem4
mal6	bem9
mal7	bem5
mal8	mal4
mal9	mal8
mal10	bem8

Novamente, utilizando a amostragem aleatórias são divididos cem vezes os dados em dados de teste e treino, em seguida para cada uma das cem amostras aleatórias é calculado o desempenho do classificador, assim como a especificidade e sensibilidade associadas. Por fim, calculando a média das cem repetições obtém-se um desempenho de cerca de 54,1%, a taxa de acerto na classe Mal foi de 39,3% e na classe Bem foi de 62%. Esta taxa de acerto fica bastante aquém da taxa de sucesso esperada, uma vez que a grande parte da bibliografia disponível menciona uma superioridade no desempenho de um classificador quando usado a distância DTW (Ratanamahatana e Keogh, 2004).

4.4.3 Aplicação do Modelo 1-NN utilizando a distância Euclidiana e Agregação por Aproximação em Partes

Como visto no segundo capítulo, uma das grandes vantagens da aplicação de técnicas de redução de dimensionalidade em séries é a melhoria do desempenho dos classificadores. Na prossecução do melhor de desempenho para o classificador 1-NN, foi aplicada a técnica Agregação por Aproximação em Partes (PAA). A escolha recaiu sobre esta técnica devido à sua fácil e compreensão implementação.

Definiu-se que cada série temporal pode ser representada por vinte segmentos, cada um deles com quatrocentos e nove pontos e para cada um dos vinte segmentos foi calculada a média dos respetivos pontos. De notar que se mantiveram os zeros que constituem as séries temporais, dando o seu contributo para o cálculo da média. O valor dos segmentos é igual à média dos pontos que os compõe, ou seja, o primeiro segmento de uma qualquer série será um valor constante que corresponde à média dos primeiros quatrocentos e vinte e nove pontos dessa mesma série. Atentando na figura 18 vemos a série bem11 original e na figura 19 a série bem11 transformada.

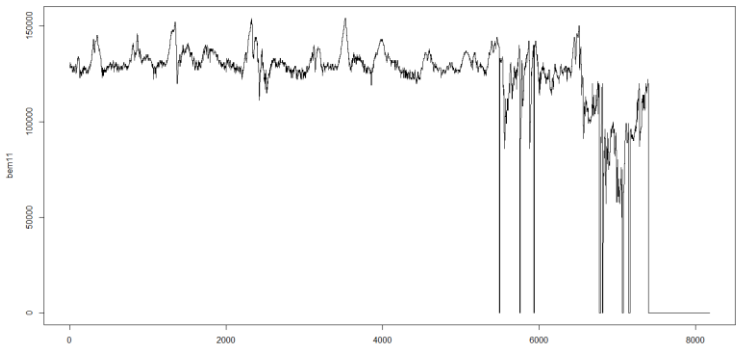

Figura 18 - Série temporal bem11

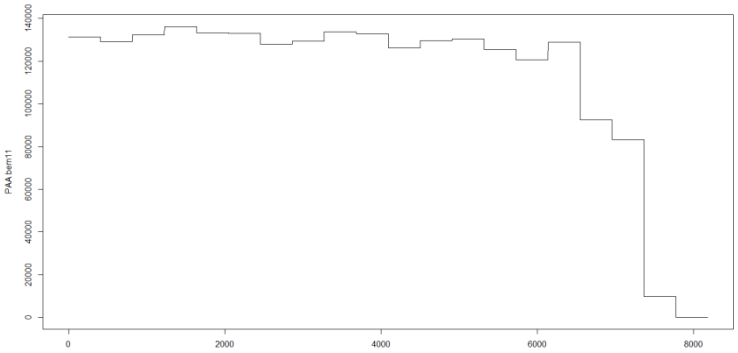

Figura 19 – Série temporal bem 11 transformada pelo método PAA

Em linha com as experiências anteriores, foram criadas cem amostragens aleatórias, os dados considerados como treino foram transformados utilizando usando o método PAA, enquanto as séries de teste se mantiveram no formato original sem qualquer transformação, uma vez que em casos reais os pontos chegam em ordem ascendente e não é possível fazer a média de quatrocentos e nove pontos quando apenas temos trezentos.

Antes de avançar para a análise do desempenho do classificador é importante analisar o quadro dos vizinhos mais próximos tal e qual as experiências anteriores. Na primeira coluna encontram-se todas as séries temporais de bem1 a mal10, estas não têm qualquer tipo de transformação, uma vez que o objetivo é tratar cada uma delas como pertencentes aos dados de teste. Na segunda coluna encontra-se a série transformada considerada série próxima da série identificada na primeira coluna. Exemplificando, a série bem1 a classificar, então esta fará parte do conjunto de teste e as restantes séries

do conjunto de treino, por isso às séries que não bem1 será aplicado o método PAA. Depois de transformadas as séries, pode-se calcular a distância, neste caso Euclidiana, entre a série bem1 não transformada e as restantes transformadas.

Tabela 8 – Vizinhos mais Próximos PAA, L=8180

Base	NN	Base	NN
bem1	mal1	**bem15**	bem6
bem2	mal3	**bem16**	mal8
bem3	bem17	**bem17**	bem3
bem4	bem14	**bem18**	bem11
bem5	bem7	**mal1**	mal3
bem6	bem10	**mal2**	bem17
bem7	bem5	**mal3**	bem2
bem8	bem6	**mal4**	mal5
bem9	bem10	**mal5**	mal4
bem10	bem6	**mal6**	mal4
bem11	mal9	**mal7**	mal10
bem12	bem5	**mal8**	bem9
bem13	mal6	**mal9**	bem11
bem14	mal5	**mal10**	bem14

Observando a tabela 8 conclui-se que a série bem1 trata como vizinha mais próxima mal1.

Sendo o foco desta experiência a melhoria da taxa de acerto e da especificidade, importa verificar se houve alguma melhoria nas medidas de desempenho do classificador 1-NN-PAA face ao classificador 1-NN com distância Euclidiana. Novamente, foram criadas cem amostragens aleatórias e calculada a média dos desempenhos de cada uma dessas amostras, assim obteve-se uma taxa de acerto de aproximadamente 59%. Todavia média a taxa de acerto obtida por método não foi superior à taxa de acerto da primeira experiência que rondou os 64,5%. Quanto às demais medidas de desempenho, a sensibilidade situou-se em 68,1% e a especificidade rondou os 40,4%.

4.4.4 Aplicação do Modelo 1-NN após Substituição de Zeros

No sentido de melhorar o desempenho do classificador 1-*Nearest Neighbor*, os testes reportados nas experiências 4.4.1, 4.4.2 e 4.4.3 foram repetidos após a substituição de todos os zeros que constituem cada uma das séries temporais.

A cada uma das séries temporais retirou-se os zeros e, com os valores restantes, foi calculada a média. Em seguida substituíram-se os zeros que constituem cada série pela média obtida anteriormente. Após a conclusão da substituição, foram feitas cem amostragens aleatórias de séries temporais em treino e teste.

Repetindo-se todo o processo aplicado em 4.4.1 através do classificador 1-NN utilizando distância Euclidiana, a taxa de desempenho obtida foi cerca de 59,2%, a especificidade rondou os 34,5 e a taxa de sensibilidade 72,3%. De facto não houve melhoria em nenhum dos três indicadores.

Igualmente, foi repetido o processo 4.4.2 utilizando o classificador 1-NN e a distância *Dynamic Time Warping* para o conjunto de séries temporais em análise. Da média das cem amostragens aleatórias obteve-se uma taxa de acerto de 50%, sendo a taxa de especificidade 22,3% e a sensibilidade de 67,3%.

No terceiro teste, foi aplicado o método Agregação por Aproximação em Partes obtendo, à semelhança da experiência 4.2.3, vinte segmentos de quatrocentos e nove pontos casa. O desempenho do classificador, resultante da média das cem amostragens aleatórias, rondou a casa dos 55,3%, a taxa de acerto na classe Bem rondou os 80,3%, por fim a taxa de acerto na classe Mal rondou os 45,5%. Nesta experiência deve ser salientado que se atingiu a maior taxa de acerto na classe Bem.

Capítulo 5 - Conclusões e Trabalho Futuro

O trabalho desenvolvido teve como principal foco a classificação antecipada de séries temporais, do inglês *early classification on time series*, sendo demonstrado ao longo do mesmo as suas vantagens e limitações, assim como a sua aplicabilidade em diversas áreas, culminando com a análise desta metodologia num caso de estudo de domínio médico.

O caso de estudo apresentado é relativo à aplicação de técnicas de classificação antecipada de séries temporais em frequências cardíacas fetais (FHR), dado que este sinal pode ser considerado uma série temporal e, também, dada a sensibilidade dos dados em análise ao fator tempo. A cada uma das frequências cardíacas fetais que integrar a base de dados em análise está associada uma classe relativa ao desfecho do parto, Bom ou Mau.

Como mencionado, de acordo com a bibliografia analisada, este trata-se de um estudo pioneiro de aplicação de técnicas de classificação antecipadas em séries relacionadas com frequência cardíaca fetal. No domínio médico o fator tempo é crucial, podendo vir a fazer a diferença entre a vida e a morte, pelo que, neste trabalho pretendeu-se encontrar um instante de tempo em que é possível classificar uma frequência cardíaca fetal sem diminuição drástica no desempenho do classificador. Nesse sentido optou-se pela aplicação do modelo de Classificação Antecipada 1-Vizinho mais Próximo proposto por Xing *et al*., 2009 devido sobretudo aos bons resultados, facilidade de compreensão e implementação.

O modelo aplicado teve algumas variantes, o primeiro modelo utilizou a distância Euclidiana para verificar os vizinhos mais próximos e *reverse nearest neighbors*. No segundo modelo utilizou-se a distância *dynamic time warping* e, na terceira experiência, aplicou-se o método de redução de dimensionalidade *piecewise aggregate approximation*. Por fim, os zeros de cada uma das séries temporais foram substituídos pela média de cada uma destas, repetindo-se as mesmas experiências. Em termos de

desempenho verificou-se que o primeiro modelo aplicado, Classificação Antecipada 1-NN com zeros, teve melhor desempenho.

Para o modelo com melhor desempenho em termos de taxa de acerto foi explorada a estabilidade dos *reverse nearest neighbor*, que segundo a bibliografia devolvem o instante de tempo em que é possível classificar séries temporais.

Os resultados encontrados mostraram que a partir do instante 727 é possível classificar qualquer série temporal que trate mal10 como sua vizinha mais próxima e partir do instante 6234 qualquer série que trate mal4 como sua vizinha mais próxima. Estes dois instantes encontrados são bastante positivos, principalmente o instante 727 associado à frequência cardíaca fetal mal10, uma vez que numa fase inicial é identificada uma situação de risco, dando margem manobra às equipas médicas para intervirem evitando uma morte fetal.

Desta forma foi possível responder ao objetivo primário deste trabalho, uma vez que é exequível classificar novas séries temporais antes de se ter acesso a todas as frequências cardíacas fetais. É assim viável classificar uma série temporal quando são recebidas 727 frequências cardíacas fetais.

Apesar de tudo, a taxa de acerto não foi tão boa como o desejável, situando-se na casa dos 64,5%, podendo dever-se a diversos fatores internos específicos de cada paciente, que não foram levados em conta neste estudo, visto que a mesma condição pode demorar diferente tempo a manifestar-se (Ghalwash *et al*., 2012). Outro fator prende-se com o sexo do bebé, tal como é mencionado na bibliografia o sexo do bebé pode ter influência no batimento cardíaco, sendo o sexo feminino o que melhor se adapta às situações decisivas durante o parto (Bernardes *et al*., 2009).

A técnica de redução de dimensionalidade, PAA, utilizada neste trabalho poderia ser explorada no tocante ao número de segmentos escolhidos, assim como a janela de procura da medida de similaridade DTW. Também, a aplicação de técnicas mais complexas de redução de dimensionalidade poderia dar um contributo positivo à taxa de desempenho do classificador.

Um modelo multivariado de classificação antecipada de séries temporais poderia ter melhor desempenho, na medida em que considera mais que uma variável ajuda a prever o desfecho do parto, como por exemplo as contrações uterinas ou até informação sobre o eletrocardiograma fetal, também outros condicionantes maternos podiam ser incorporados no modelo.

Posto isto, apesar de pouco explorada a classificação antecipada tem um enorme potencial pela sua aplicabilidade em diversas áreas, assim como pelos seus impactos a nível económicos e sócias. De facto saber em antemão o que irá acontecer daqui a uns minutos ou horas traz a possibilidade de uma melhor preparação para um qualquer acontecimento, ou até a diminuição de ansiedade face ao desconhecido.

Capítulo 6 - Bibliografia

An, A. (2009), "Classification Methods", in Wang, J. (editors), Encyclopedia of Data Warehousing and Mining, 2ª edição, pp. 196-201, IGI GLOBAL.

Alfirevic, Z., D. Devane, D., e G. M. Gyte (2006), "Continuous Cardiotocography (CTG) as a Form of Electronic Fetal Monitoring (EFM) for Fetal Assessment during Labour", *Cochrane Database Syst Rev*, Vol. 3, Nº 3.

Anderson, H.S, N. Parrish e M. R. Gupta (2012), "Early Time-Series Classification with Reliability Guarantee", *Technical Report SAND2012-6961 Sandia National Laboratories*.

Bagnall, L., L.M. Davis, J. Hills e J. Lines (2012), "Transformation Based Ensembles for Time Series Classification", *Proceedings SDM*, pp. 307-318.

Batal, I., H. Valizadegan, G.F. Cooper e M. Hauskrecht (2013), "A Temporal Pattern Mining Approach for Classifying Electronic Health Record Data", *ACM Transactions on Intelligent Systems and Technology*, Vol. 4, Nº 4:63.

Bernardes, J., H. Gonçalves, D. Ayres-de-Campos e A. Pa. Rocha (2009), "Sex Differences in Linear and Complex Fetal Heart Rate Dynamics of Normal and Acidemic Fetuses in the Minutes Preceding Delivery", *Journal of perinatal medicine*, Vol. 37, Nº 2, pp.168-176.

Berndt, D. J., e J. Clifford (1994), "Using Dynamic Time Warping to Find Patterns in Time Series", *KDD workshop* , Vol. 10, Nº. 16, pp. 359-370.

Berry, M. J. A. e G. S. Linoff (2004), *Data Mining Techniques for Marketing, Sales and Customer Relationship Manager*, Wiley.

Bramer, M. (2007), *Principles of Data Mining*, Springer.

Bregón, A., M.A. Simon, J.J. Rodriguez, C. Alonso, B. Pulido e I. Moro (2006), "Early Fault Classification in Dynamic Systems using Case-Based Reasoning", in Marín *et* al (editors), Current Topics in Artificial Intelligence, Vol. 4177, pp. 211-220, Springer.

Buza, K., A. Nanopulos, L. Schmidt-Thieme e J. Koller (2011), "Fast Classification of Electrocardiograph Signal via Instance Selection", *Proceedings - 2011 1st IEEE International Conference on Healthcare Informatics, Imaging and Systems Biology*, pp. 9-16.

Chan, K., e W. Fu (1999), "Efficient Time Series Matching by Wavelets", *Proceedings of the 15th IEEE International Conference on Data Engineering*, pp. 126-133.

Chen, M., S., Han, J., e Yu, P. S. (1996), "Data mining: An overview from a database perspective", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 8, Nº 6, pp. 866–883.

Chinnasamy, S., C. Muthusamy, e G. Gopal (2013), "An Outlier Based Bi-Level Neural Network Classification System for Improved Classification of Cardiotocogram Data", *Life Science Journal*, Vol. 10, Nº 1, pp 244-251.

Chizi B., O. Maimon (2010), "Dimension Reduction and Feature Selection", in Maimon O. e L. Rokach (editors), Data Mining and Knowledge Discovery Handbook, 2ª edição, pp. 83-100, Springer.

Cios, K.J. e G.W. Moore (2002), "Uniqueness of Medical Data Mining", *Artificial Intelligence in Medicine*, Vol. 26, Nº 1-2, pp. 1-24.

Costa, M.A.M.N. (2010), "Development and Evaluation of a Combination of Computer Analysis of Cardiotocografy and Electrocardiography for Intrapartum Fetal Monitoring, Faculdade de Medicina da Universidade do Porto, http://hdl.handle.net/10216/26571, acedido em 12 de Agosto de 2014.

Ding H., G. Trajcevski, P. Scheuermann, X. Wang e E. Keogh (2008), "Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures", *Proceedings of the VLDB Endowment*, Vol. 1, Nº 2, pp. 1542-1552.

Direção de Serviços de Informação e Análise (2013), "Estudo Comparativo do Número de Óbitos e Causas de Morte da Mortalidade Infantil e Suas Componentes (2009-2012)", Direção-Geral da Saúde, versão 2.

El-Sappagh, S. H., S. El-Masri, A. M. Riad e M. Elmogy (2013), "Data Mining and Knowledge Discovery: Applications, Techniques, Challenges and Process Models in Healthcare", *Journal of Engineering Research and Applications*, Vol.3, Nº3, pp. 900-906.

Fayyad, U.M, G. Piatetsky-Shapiro e P. Smyth (1996), "From Data Mining to Knowledge Discovery in Databases", *Al Magazine*, Vol.17, Nº 3, pp. 37-54.

Fayyad, U.M. (1997), "Data Mining and Knowledge Discovery in Databases: Implications for Scientific Databases", *Proceedings of the Ninth International Conference on Scientific and Statistical Database Management*, pp. 2-11.

Fu, T. (2011), "A Review on Time Series Data Mining", *Engineering Applications of Artificial Intelligence*, Vol. 24, Nº. 1, pp. 164-181.

Ghalwash, M.F., D. Ramljak, e Z. Obradovic (2012), "Early Classification of Multivariate Time Series using a Hybrid HMM/SVM Model", *Bioinformatics and Biomedicine IEEE International Conference*, pp.1-6.

Ghalwash, M.F. e Z. Obradovic (2012), " Early Classification of Multivariate Temporal Observations by Extraction of Interpretable Shapelets", *BMC Bioinformatics*, Vol.13.

Han, J. (2009), "Data Mining", in Liu, L. e M. Tamer Ozsu (editors.), Encyclopedia of Database Systems, pp. 595-598, Springer.

Han, J, M. Kamber (2006), *Data Mining Concepts and Techniques*, Morgan Kaufmann.

Hatami N. e C. Chira (2013), "Classifiers With a Reject Option for Early Time-Series Classification", *Proceedings of the IEEE Symposium on Computational Intelligence and Ensemble Learning*, pp. 9-16.

Hauskrecht, M., I. Batal, M. Valko, S. Visweswaran, G .F. Cooper e G. Clermont (2013), "Outlier Detection for Patient Monitoring and Alerting", *Journal of Biomedical Informatics*, Vol. 46, Nº1, pp. 47-55.

Hosseinkhah, F., H. Ashktorab, R. Veen e M. M. Owrang O. (2009), "Challenges in Data Mining on Medical Databases", *in* J. Erickson (editors), Database Technologies: Concepts, Methodologies, Tools, and Applications, Vol.1-4, pp. 1393-1404, IGI Global.

Keogh, E. e M. Pazzani (1998), "An Enhanced Representation of Time Series Which Allows Fast and Accurate Classification, Clustering and Relevance Feedback", *Proceeding of the 4 th Int'l Conference on Knowledge Discovery and Data Mining*, Vol 98, pp. 239-241.

Keogh, E., K. Chakrabarti, M. Pazzani e S. Mehrotra (2001), "Dimensionality Reduction for Fast Similarity Search in Large Time Series Database", *Knowledge and Information Systems*, Vol. 3, Nº 3, pp.263-286.

Keogh, E. e S. Kasetty (2002), "On The Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration", *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp 102-111.

Klawonn F. e F. Rehm (2009), "Cluster Analysis for Outlier Detection", in J. Wang (editors), Encyclopedia of data warehousing and mining, Vol2, pp. 214-218, IGI GLOBAL.

Koh, H. C. e G. Tan (2005),"Data Mining Applications in Healthcare", *Journal of healthcare information management*, Vol. 19, Nº 2, pp. 64-72.

Korn, F. e S. Muthukrishnan (2000) "Influence Sets Based on Reverse Nearest Neighbor Queries", *ACM SIGMOD Record*, Vol. 29, Nº 2, pp. 201-212.

Jensen, C.S., R.T. Snodgrass (2009), "Temporal Data Models", in Liu, L. e M. Tamer Ozsu (editors.), Encyclopedia of Database Systems pp. 2952-2956, Springer.

Johns Hopkins Medicine, "External Fetal Heart Rate Monitoring", http://www.hopkinsmedicine.org/healthlibrary/GetImage.aspx?ImageId=126116 acedido em 15 de Setembro de 2014

Laxman, L. e P. Sastry (2006), "A Survey of Temporal Data Mining", *Sadhana*, Vol.31, Nº 2, pp.173-198.

Luzietti, R., R. Erkkola, U. Hasbargen, L. A. Mattsson, J. M. Thoulon, e K. G. Rosén (1999), "European Community Multi-Center Trial "Fetal ECG Analysis During Labor": ST plus CTG Analysis, *Journal of perinatal medicine*, Vol. 27, Nº 6, pp. 431-440.

Maimon, O. e L. Rokach (2010), "Introduction to Knowledge Discovery and Data Mining", in O. Maimon and L. Rokach (editors), Data Mining and Knowledge Discovery Handbook, pp.1-15, Springer.

Mao, Y., W. Chen, Y. Chen, C. Lu, M. Kollef, e T. Bailey (2012), "An Integrated Data Mining Approach to Real-Time clinical Monitoring and Deterioration Warning", *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.1140-1148.

Milanović M. e M. Stamenković (2011), "Data Mining in Time Series", *EkonomskiHorizonti - Faculty of Economics, University of Kragujevac*, Vol. 13, Nº 1, pp. 5-25.

Mitsa, T. (2009), *Temporal Data Mining*, Chapman & Hall/CRC.

Nanopoulos, A., R. Alcock, e Y. Manolopoulos (2001), "Feature-based Classification of Time-series Data", in Mastorakis N. e S. D. Nikikipoulos (editors), Information processing and technology, pp. 49-61.

Neilson, JP (2013), "Fetal Electrocardiogram (ECG) for Fetal Monitoring During Labour", *Cochrane Database Syst Rev*, 5.

Norton, M. J. (1999), "Knowledge discovery in databases", in Qin, J. e M.J Norton (editors), Knowledge Discovery in Bibliographic Databases, Vol. 48, Nº 1, pp. 9-21, Library Trends.

Pitts, S. R., R. W. Niska, J. Xu, e C. W. Burt (2008), "National hospital ambulatory medical care survey: 2006 emergency department summary", *National Health Statistics Reports*, Vol. 7, Nº 7, pp. 1-38.

Ratanamahatana, C. A. e E. Keogh (2004), "Making Time-Series Classification More Accurate Using Learned Constraints", *Proceedings of SIAM International Conference on Data Mining*, pp. 11-24.

Ratanamahatana, C. A. e E. Keogh (2005), "Three Myths about Dynamic Time Warping", *Proceedings of SIAM International Conference on Data Mining (SDM '05), Newport Beach*, CA, April 21-23, pp. 506-510 .

Reuteurs, "New World Angels Leads $2.1 Million Investment in OBMedical to for Development of the Laborview™ Maternal-Fetal Monitoring Sensor System", http://www.reuters.com/article/2014/07/29/fl-obmedical-new-world-a-idUSnBw295122a+100+BSW2014072, acedido em 12 de Agosto de 2014.

Rodríguez, J. J., C. J. Alonso (2002), "Boosting Interval-based Literals: Variable Length and Early Classification", *ECAI'02 Workshop on Knowledge Discovery from (Spatio-) Temporal Data*, pp. 51–62.

Tufféry, S. (2011), *Data Mining and Statistics for Decision Making*, Wiley.

Ye, Y. e E. Keogh (2009), Time Series Shapeletes: A New Primitive for Data Mining, *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 947-956.

Xi, X., E. Keogh, C. Shelton e L. Wi (2006), "Fast Time Series Classification Using Numerosity Reduction", *Proceedings of the 23rd International Conference on Machine Learning*, pp.1033-140.

Xing, Z., J. Pei, G. Dong, e P. S. Yu (2008), "Mining Sequence Classifiers for Early Prediction", *Proceedings of the 2008 SIAM International Conference on Data Mining*, pp. 644– 655.

Xing, Z., J. Pei e P.S. Yu (2009), "Early Classification on Time Series: A Nearest Neighbor Approach", *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pp. 1297-1302.

Xing, Z., J. Pei, P. Yu e K. Wang (2011), "Extracting interpretable features for early classification on time series", *Proceedings of SDM*, Vol. 11, pp. 247-258.

Xing, Z., J. Pei, and P. S. Yu (2012), "Early Classification on Time Series", *Knowledge and Information Systems: An International Journal*, Vol. 31, Nº 1, pp. 105-127.

Wang, X., A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann e E. Keogh, (2013), "Experimental Comparison of Representation Methods and Distance Measures for Time Series Data", *Data Mining and Knowledge Discovery*, Vol. 26, Nº 2, pp. 275-309.

Wikipedia, "K-Nearest Neighbors Algorithm", http://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm, acedido em 29 de Agosto de 2014.

Witten, I. H., E. Frank e M. A. Hall (2011), *Data Mining Pratical Machine Learning Tools Techniques*, Morgan Kaufman.

Capítulo 7 – Anexos

7.1 Matriz de Dissemelhança 1-NN Distância Euclidiana, L=8180

	bem1	bem2	bem3	bem4	bem5	bem6	bem7	bem8	bem9	bem10	bem11	bem12	bem13	bem14
bem1		6.151.528	5.973.726	8.135.649	7.102.673	7.047.613	6.769.300	6.588.594	7.717.953	8.015.844	6.777.333	6.322.010	7.692.695	8.113.825
bem2	6.151.528		4.941.080	7.655.092	5.735.982	4.857.232	5.749.106	4.336.540	5.266.400	5.949.275	5.411.378	5.333.126	5.926.772	7.445.077
bem3	5.973.726	4.941.080		7.574.718	6.252.243	5.727.460	5.961.886	5.052.280	6.413.433	6.904.035	5.453.243	5.395.498	6.286.570	6.957.449
bem4	8.135.649	7.655.092	7.574.718		6.648.489	6.502.776	7.393.565	6.709.100	6.882.853	7.729.437	4.823.104	7.055.978	5.701.292	4.713.248
bem5	7.102.673	5.735.982	6.252.243	6.648.489		4.904.340	5.604.024	4.816.298	5.638.627	6.407.782	4.491.993	4.546.713	5.384.128	6.195.537
bem6	7.047.613	4.857.232	5.727.460	6.502.776	4.904.340		5.473.617	3.903.012	4.545.497	5.196.554	3.992.905	4.609.152	4.737.441	6.161.548
bem7	6.769.300	5.749.106	5.961.886	7.393.565	5.604.024	5.473.617		5.291.870	6.418.040	6.705.824	4.995.790	4.825.869	5.512.226	6.826.469
bem8	6.588.594	4.336.540	5.052.280	6.709.100	4.816.298	3.903.012	5.291.870		4.679.920	5.276.838	4.384.245	4.189.082	5.307.474	6.491.616
bem9	7.717.953	5.266.400	6.413.433	6.882.853	5.638.627	4.545.497	6.418.040	4.679.920		5.348.765	4.499.620	5.401.588	4.944.479	6.524.322
bem10	8.015.844	5.949.275	6.904.035	7.729.437	6.407.782	5.196.554	6.705.824	5.276.838	5.348.765		5.654.342	6.247.866	6.120.303	7.117.324
bem11	6.777.333	5.411.378	5.453.243	4.823.104	4.491.993	3.992.905	4.995.790	4.384.245	4.499.620	5.654.342		4.318.265	2.959.519	4.078.102
bem12	6.322.010	5.333.126	5.395.498	7.055.978	4.546.713	4.609.152	4.825.869	4.189.082	5.401.588	6.247.866	4.318.265		5.467.155	6.485.919
bem13	7.692.695	5.926.772	6.286.570	5.701.292	5.384.128	4.737.441	5.512.226	5.307.474	4.944.479	6.120.303	2.959.519	5.467.155		4.632.814
bem14	8.113.825	7.445.077	6.957.449	4.713.248	6.195.537	6.161.548	6.826.469	6.491.616	6.524.322	7.117.324	4.078.102	6.485.919	4.632.814	
bem15	6.801.532	4.920.875	5.412.442	6.516.598	5.000.641	4.202.523	5.060.022	4.092.951	5.091.517	5.800.917	3.822.631	4.426.515	4.726.774	5.988.599
bem16	8.148.450	6.247.177	6.848.092	6.332.922	5.694.421	5.088.769	6.817.324	5.307.818	5.409.229	6.478.372	4.186.002	5.751.630	4.936.904	5.754.691
bem17	6.017.477	5.080.820	4.889.201	7.045.103	5.872.392	5.261.317	5.424.583	4.602.812	5.896.991	6.447.506	4.969.932	5.190.682	5.735.704	6.560.719
bem18	6.605.186	5.598.060	5.655.453	5.444.925	4.844.091	4.614.324	5.407.237	4.630.531	5.016.707	6.069.367	3.370.438	4.541.876	4.441.528	4.886.173
ma1	6.064.850	4.726.510	5.288.974	7.502.463	6.068.757	5.654.930	6.204.320	5.093.983	6.088.385	6.469.680	5.825.310	5.901.840	6.508.638	7.424.246
mal2	6.491.157	5.597.704	5.396.390	7.107.039	6.258.895	5.808.937	6.054.272	5.390.690	6.738.375	6.915.526	5.501.823	5.828.966	6.336.529	6.799.198
mal3	5.699.626	3.115.325	4.565.305	7.229.531	5.459.513	4.412.205	5.289.579	3.854.198	4.922.265	5.595.309	4.853.645	4.891.095	5.440.235	6.943.856
mal4	8.167.783	6.588.516	6.770.290	5.295.592	5.682.992	5.119.388	6.320.223	5.545.947	5.307.569	6.236.541	3.720.721	6.111.708	4.124.831	4.572.429
mal5	8.299.858	7.016.807	7.086.217	4.539.453	6.124.652	5.545.168	6.903.288	5.990.500	5.646.849	6.650.693	3.948.788	6.757.473	4.187.471	4.126.247
mal6	8.329.761	6.653.685	7.001.754	6.680.715	6.460.161	5.735.008	6.771.361	6.437.750	5.746.310	6.860.233	4.468.641	6.672.629	4.222.062	5.573.635
mal7	9.259.137	9.674.096	8.772.923	7.607.220	8.560.086	8.864.292	9.038.482	8.748.938	9.565.605	9.757.366	7.390.432	8.809.838	8.199.354	6.828.231
mal8	7.857.179	5.568.437	6.166.399	6.926.085	5.260.764	4.615.215	6.036.542	4.779.520	4.744.667	5.762.714	4.254.436	5.214.978	4.593.077	6.047.203
mal9	7.947.650	7.438.385	7.344.483	6.260.283	6.471.358	6.147.778	7.012.081	6.234.221	6.648.001	7.400.887	4.896.400	5.954.732	5.793.192	5.674.114
mal10	7.492.149	7.449.976	6.995.191	5.318.769	6.116.783	6.495.281	6.987.407	6.347.759	6.945.459	7.382.828	4.743.992	6.471.169	5.732.709	4.557.714

	bem15	Bem16	bem17	bem18	ma1	mal2	mal3	mal4	mal5	mal6	mal7	mal8	mal9	mal10
bem1	6.801.532	8.148.450	6.017.477	6.605.186	6.064.850	6.491.157	5.699.626	8.167.783	8.299.858	8.329.761	9.259.137	7.857.179	7.947.650	7.492.149
bem2	4.920.875	6.247.177	5.080.820	5.598.060	4.726.510	5.597.704	3.115.325	6.588.516	7.016.807	6.653.685	9.674.096	5.568.437	7.438.385	7.449.976
bem3	5.412.442	6.848.092	4.889.201	5.655.453	5.288.974	5.396.390	4.565.305	6.770.290	7.086.217	7.001.754	8.772.923	6.166.399	7.344.483	6.995.191
bem4	6.516.598	6.332.922	7.045.103	5.444.925	7.502.463	7.107.039	7.229.531	5.295.592	4.539.453	6.680.715	7.607.220	6.926.085	6.260.283	5.318.769
bem5	5.000.641	5.694.421	5.872.392	4.844.091	6.068.757	6.258.895	5.459.513	5.682.992	6.124.652	6.460.161	8.560.086	5.260.764	6.471.358	6.116.783
bem6	4.202.523	5.088.769	5.261.317	4.614.324	5.654.930	5.808.937	4.412.205	5.119.388	5.545.168	5.735.008	8.864.292	4.615.215	6.147.778	6.495.281
bem7	5.060.022	6.817.324	5.424.583	5.407.237	6.204.320	6.054.272	5.289.579	6.320.223	6.903.288	6.771.361	9.038.482	6.036.542	7.012.081	6.987.407
bem8	4.092.951	5.307.818	4.602.812	4.630.531	5.093.983	5.390.690	3.854.198	5.545.947	5.990.500	6.437.750	8.748.938	4.779.520	6.234.221	6.347.759
bem9	5.091.517	5.409.229	5.896.991	5.016.707	6.088.385	6.738.375	4.922.265	5.307.569	5.646.849	5.746.310	9.565.605	4.744.667	6.648.001	6.945.459
bem10	5.800.917	6.478.372	6.447.506	6.069.367	6.469.680	6.915.526	5.595.309	6.236.541	6.650.693	6.860.233	9.757.366	5.762.714	7.400.887	7.382.828
bem11	3.822.631	4.186.002	4.969.932	3.370.438	5.825.310	5.501.823	4.853.645	3.720.721	3.948.788	4.468.641	7.390.432	4.254.436	4.896.400	4.743.992
bem12	4.426.515	5.751.630	5.190.682	4.541.876	5.901.840	5.828.966	4.891.095	6.111.708	6.757.473	6.672.629	8.809.838	5.214.978	5.954.732	6.471.169
bem13	4.726.774	4.936.904	5.735.704	4.441.528	6.508.638	6.336.529	5.440.235	4.124.831	4.187.471	4.222.062	8.199.354	4.593.077	5.793.192	5.732.709
bem14	5.988.599	5.754.691	6.560.719	4.886.173	7.424.246	6.799.198	6.943.856	4.572.429	4.126.247	5.573.635	6.828.231	6.047.203	5.674.114	4.557.714
bem15		5.065.151	4.961.091	4.494.384	5.653.300	5.519.509	4.501.220	5.194.418	5.675.659	5.770.499	8.401.462	5.049.175	6.060.370	6.149.813
bem16	5.065.151		6.204.457	5.061.102	6.794.227	6.897.278	5.829.964	4.991.317	5.135.777	5.909.246	8.752.405	5.287.451	6.180.886	6.252.026
bem17	4.961.091	6.204.457		5.085.000	5.537.987	5.358.569	4.383.503	6.236.193	6.523.203	6.812.036	8.705.425	5.956.806	6.900.830	6.685.053
bem18	4.494.384	5.061.102	5.085.000		5.854.204	5.700.858	5.163.244	4.573.216	4.848.477	5.250.471	7.788.532	4.950.774	5.513.579	5.042.153
ma1	5.653.300	6.794.227	5.537.987	5.854.204		5.621.201	4.421.854	6.517.400	6.811.316	7.391.052	8.834.804	6.318.702	7.313.963	6.929.428
mal2	5.519.509	6.897.278	5.358.569	5.700.858	5.621.201		5.002.992	6.507.892	6.696.445	7.190.148	8.640.261	6.651.909	7.346.781	6.765.283
mal3	4.501.220	5.829.964	4.383.503	5.163.244	4.421.854	5.002.992		5.959.656	6.421.566	6.377.156	9.330.123	5.227.852	6.824.735	7.009.273
mal4	5.194.418	4.991.317	6.236.193	4.573.216	6.517.400	6.507.892	5.959.656		3.588.076	5.213.330	7.991.391	5.032.988	5.701.643	5.183.017
mal5	5.675.659	5.135.777	6.523.203	4.848.477	6.811.316	6.696.445	6.421.566	3.588.076		5.332.650	7.530.281	5.482.296	5.859.682	4.909.815
mal6	5.770.499	5.909.246	6.812.036	5.250.471	7.391.052	7.190.148	6.377.156	5.213.330	5.332.650		8.942.604	5.638.673	6.638.067	6.515.501
mal7	8.401.462	8.752.405	8.705.425	7.788.532	8.834.804	8.640.261	9.330.123	7.991.391	7.530.281	8.942.604		9.032.947	8.558.762	6.312.163
mal8	5.049.175	5.287.451	5.956.806	4.950.774	6.318.702	6.651.909	5.227.852	5.032.988	5.482.296	5.638.673	9.032.947		6.143.459	6.577.673
mal9	6.060.370	6.180.886	6.900.830	5.513.579	7.313.963	7.346.781	6.824.735	5.701.643	5.859.682	6.638.067	8.558.762	6.143.459		5.998.464
mal10	6.149.813	6.252.026	6.685.053	5.042.153	6.929.428	6.765.283	7.009.273	5.183.017	4.909.815	6.515.501	6.312.163	6.577.673	5.998.464	

7.2 Matriz de Dissemelhança 1-NN Distância DTW, L=8180

	bem1	bem2	bem3	bem4	bem5	bem6	bem7	bem8	bem9	bem10	bem11	bem12	bem13	bem14
bem1	0	12.894	10.604	11.573	10.126	12.143	13.579	11.034	10.541	12.628	12.342	11.193	10.146	10.478
bem2	12.894	0	18.126	26.235	18.118	15.748	21.812	16.463	17.041	18.993	22.812	20.905	20.020	27.485
bem3	10.604	18.126	0	11.494	8.989	11.432	13.496	11.167	12.689	14.511	11.389	11.876	10.508	9.989
bem4	11.573	26.235	11.494	0	10.561	10.913	11.954	10.231	10.160	13.163	9.614	11.797	9.646	9.311
bem5	10.126	18.118	8.989	10.561	0	9.774	11.651	10.112	9.703	11.990	11.536	11.341	9.193	9.980
bem6	12.143	15.748	11.432	10.913	9.774	0	12.244	9.424	9.188	11.988	10.401	10.866	8.035	10.414
bem7	13.579	21.812	13.496	11.954	11.651	12.244	0	10.528	12.192	14.031	12.709	11.421	11.242	11.252
bem8	11.034	16.463	11.167	10.231	10.112	9.424	10.528	0	11.191	12.980	9.785	10.164	8.615	8.865
bem9	10.541	17.041	12.689	10.160	9.703	9.188	12.192	11.191	0	10.595	13.637	11.087	10.024	13.069
bem10	12.628	18.993	14.511	13.163	11.990	11.988	14.031	12.980	10.595	0	14.639	13.713	11.656	13.149
bem11	12.342	22.812	11.389	9.614	11.536	10.401	12.709	9.785	13.637	14.639	0	10.155	8.210	8.005
bem12	11.193	20.905	11.876	11.797	11.341	10.866	11.421	10.164	11.087	13.713	10.155	0	9.058	10.106
bem13	10.146	20.020	10.508	9.646	9.193	8.035	11.242	8.615	10.024	11.656	8.210	9.058	0	8.352
bem14	10.478	27.485	9.989	9.311	9.980	10.414	11.252	8.865	13.069	13.149	8.005	10.106	8.352	0
bem15	12.326	18.239	12.515	11.622	11.282	10.850	11.307	9.651	11.411	13.070	10.339	10.488	9.320	10.317
bem16	10.800	19.781	11.100	9.722	9.441	9.368	11.489	10.537	8.497	11.357	11.800	11.421	9.271	10.248
bem17	12.607	19.306	12.946	14.523	13.092	12.623	12.713	11.132	13.335	15.560	14.063	13.240	13.182	13.189
bem18	10.837	21.384	9.563	9.897	10.094	11.494	12.263	10.955	13.261	15.677	8.688	11.189	10.325	8.780
ma1	10.552	15.173	8.359	11.922	8.688	9.293	13.079	10.557	10.771	12.169	11.641	11.106	8.478	11.332
mal2	11.948	19.036	8.556	13.891	11.711	13.666	14.492	12.609	16.169	16.434	12.441	13.903	11.945	12.119
mal3	11.028	11.023	9.618	11.450	9.428	7.717	13.498	9.961	8.392	11.559	10.172	10.779	7.644	10.115
mal4	16.130	24.941	16.076	11.584	12.839	9.289	12.687	10.882	10.067	14.165	11.596	12.338	10.511	11.795
mal5	12.272	24.412	12.714	8.500	11.512	10.443	12.424	11.138	10.675	14.451	9.973	11.325	9.691	10.245
mal6	12.302	22.750	13.261	11.230	11.307	10.473	12.518	11.583	10.072	10.983	13.391	12.452	10.559	12.350
mal7	13.643	30.633	13.626	13.487	13.012	14.945	13.785	13.982	14.539	15.674	15.387	15.738	14.158	13.292
mal8	12.967	19.522	13.903	11.486	11.334	10.099	12.397	11.023	10.169	11.808	13.109	12.359	10.285	12.202
mal9	14.283	25.761	15.627	13.464	13.572	13.849	13.234	12.566	13.191	14.531	15.600	14.986	13.950	13.421
mal10	14.977	28.694	14.610	13.529	14.660	15.807	14.444	13.010	16.891	18.706	11.521	13.731	13.941	11.788

	bem15	Bem16	bem17	bem18	ma1	mal2	mal3	mal4	mal5	mal6	mal7	mal8	mal9	mal10
bem1	12.326	10.800	12.607	10.837	10.552	11.948	11.028	16.130	12.272	12.302	13.643	12.967	14.283	14.977
bem2	18.239	19.781	19.306	21.384	15.173	19.036	11.023	24.941	24.412	22.750	30.633	19.522	25.761	28.694
bem3	12.515	11.100	12.946	9.563	8.359	8.556	9.618	16.076	12.714	13.261	13.626	13.903	15.627	14.610
bem4	11.622	9.722	14.523	9.897	11.922	13.891	11.450	11.584	8.500	11.230	13.487	11.486	13.464	13.529
bem5	11.282	9.441	13.092	10.094	8.688	11.711	9.428	12.839	11.512	11.307	13.012	11.334	13.572	14.660
bem6	10.850	9.368	12.623	11.494	9.293	13.666	7.717	9.289	10.443	10.473	14.945	10.099	13.849	15.807
bem7	11.307	11.489	12.713	12.263	13.079	14.492	13.498	12.687	12.424	12.518	13.785	12.397	13.234	14.444
bem8	9.651	10.537	11.132	10.955	10.557	12.609	9.961	10.882	11.138	11.583	13.982	11.023	12.566	13.010
bem9	11.411	8.497	13.335	13.261	10.771	16.169	8.392	10.067	10.675	10.072	14.539	10.169	13.191	16.891
bem10	13.070	11.357	15.560	15.677	12.169	16.434	11.559	14.165	14.451	10.983	15.674	11.808	14.531	18.706
bem11	10.339	11.800	14.063	8.688	11.641	12.441	10.172	11.596	9.973	13.391	15.387	13.109	15.600	11.521
bem12	10.488	11.421	13.240	11.189	11.106	13.903	10.779	12.338	11.325	12.452	15.738	12.359	14.986	13.731
bem13	9.320	9.271	13.182	10.325	8.478	11.945	7.644	10.511	9.691	10.559	14.158	10.285	13.950	13.941
bem14	10.317	10.248	13.189	8.780	11.332	12.119	10.115	11.795	10.245	12.350	13.292	12.202	13.421	11.788
bem15	0	11.269	12.171	11.804	11.794	13.453	10.750	11.328	10.885	12.302	14.552	10.936	12.591	12.251
bem16	11.269	0	13.211	11.922	9.607	13.689	8.781	11.108	10.323	10.812	13.360	11.287	12.961	14.230
bem17	12.171	13.211	0	13.204	12.176	14.073	11.938	14.314	14.665	15.055	15.746	14.122	15.019	14.452
bem18	11.804	11.922	13.204	0	11.413	11.834	11.618	12.876	9.477	14.239	13.549	13.780	14.576	11.458
ma1	11.794	9.607	12.176	11.413	0	10.828	6.792	15.664	12.550	11.471	13.837	12.469	15.344	14.981
mal2	13.453	13.689	14.073	11.834	10.828	0	11.604	17.327	15.616	14.649	13.801	16.141	16.410	15.440
mal3	10.750	8.781	11.938	11.618	6.792	11.604	0	14.144	10.963	10.778	14.249	11.400	15.055	15.034
mal4	11.328	11.108	14.314	12.876	15.664	17.327	14.144	0	9.522	10.839	16.738	9.561	14.301	15.649
mal5	10.885	10.323	14.665	9.477	12.550	15.616	10.963	9.522	0	12.449	14.736	11.118	13.677	12.726
mal6	12.302	10.812	15.055	14.239	11.471	14.649	10.778	10.839	12.449	0	14.569	10.970	13.896	17.241
mal7	14.552	13.360	15.746	13.549	13.837	13.801	14.249	16.738	14.736	14.569	0	15.030	13.980	13.969
mal8	10.936	11.287	14.122	13.780	12.469	16.141	11.400	9.561	11.118	10.970	15.030	0	11.911	15.720
mal9	12.591	12.961	15.019	14.576	15.344	16.410	15.055	14.301	13.677	13.896	13.980	11.911	0	15.147
mal10	12.251	14.230	14.452	11.458	14.981	15.440	15.034	15.649	12.726	17.241	13.969	15.720	15.147	0