CLASSIFICAÇÃO DE SÉRIES TEMPORAIS VIA DIVERGENTE ENTRE DENSIDADES DE PROBABILIDADE NO ESPAÇO DE FASES

André Santos Teixeira de Carvalho

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientadores: Carlos Eduardo Pedreira
Carlos Eduardo Ribeiro de Mello

Rio de Janeiro
Novembro de 2016

CLASSIFICAÇÃO DE SÉRIES TEMPORAIS VIA DIVERGENTE ENTRE DENSIDADES DE PROBABILIDADE NO ESPAÇO DE FASES

André Santos Teixeira de Carvalho

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Carlos Eduardo Pedreira, Ph.D.

Prof. Geraldo Bonorino Xexéo, D.Sc.

Prof. Carlos Eduardo Ribeiro de Mello, D.Sc.

Prof. Eduardo Fonseca Mendes, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
NOVEMBRO DE 2016

Santos Teixeira de Carvalho, André

Classificação de séries temporais via divergente entre densidades de probabilidade no espaço de fases/André Santos Teixeira de Carvalho. – Rio de Janeiro: UFRJ/COPPE, 2016.

X, 52 p.: il.; 29, 7cm.

Orientadores: Carlos Eduardo Pedreira

Carlos Eduardo Ribeiro de Mello

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2016.

Referências Bibliográficas: p. 48 – 52.

1. Séries Temporais. 2. Classificação. I. Pedreira, Carlos Eduardo *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Agradecimentos

À minha família, pelo apoio durante todos os anos e em todos os momentos. Sou muito grato especialmente à minha mãe Almira, meu pai Sebastião e minha tia Carmen, pela educação que me foi dada. Andreia, Ana Paula, Otavio e Júlia pela amizade, união e compreensão com meu estresse elevado durante os últimos tempos.

À minha namorada, Leticia Brugger, pelo apoio dado nesses anos de mestrado e pelo companheirismo e amor compartilhados durante todos os anos de namoro sem os quais eu não teria conseguido terminar este trabalho. Essa conquista também é sua.

Aos meus melhores amigos, Thiago Cacicedo e Yuri Bastos, pela amizade e momentos de descontração compartilhados. Aos todos os amigos que fiz na UFRJ, em especial Douglas Paranhos, Luan Garrido e Thais Vianna, espero que nossas amizades perdurem para o resto de minha vida.

Aos colegas e ex-colegas de trabalho da Coppetec, Hotel Urbano e Globo.com pela compreensão nos momentos em que estive ausente e pelo apoio dado para meu crescimento profissional.

Aos meus orientadores Pedreira e Cadu, pelos conselhos, discussões e por todo o incentivo para que este trabalho fosse concluído com qualidade.

Aos membros da banca, por terem aceitado contribuir com críticas e observações sobre este trabalho.

A todos os professores e educadores responsáveis por minha formação sem os quais não teria chegado tão longe. O trabalho de vocês foi fundamental e serei eternamente grato.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

CLASSIFICAÇÃO DE SÉRIES TEMPORAIS VIA DIVERGENTE ENTRE DENSIDADES DE PROBABILIDADE NO ESPAÇO DE FASES

André Santos Teixeira de Carvalho

Novembro/2016

Orientadores: Carlos Eduardo Pedreira
Carlos Eduardo Ribeiro de Mello

Programa: Engenharia de Sistemas e Computação

Séries temporais estão presentes nas mais diversas áreas de pesquisa, como eletrocardiogramas, reconhecimento de voz e escrita. A classificação de séries temporais é uma tarefa que tem atraído a atenção de pesquisadores de diversas áreas do conhecimento e tem como objetivo a detecção de padrões que discriminem as séries em um conjunto predefinido de classes. Diversas técnicas para classificação de séries temporais estão disponíveis na literatura. Muitas dessas abordagens definem métricas de distância entre as séries e as utilizam em classificadores clássicos, como o k-vizinhos mais próximos, enquanto que outras constroem modelos estatísticos a partir dos dados.

Nesta dissertação, é apresentado um novo método de classificação de séries temporais. A proposta utiliza a técnica *time delay embedding* para reconstruir o espaço de estados das séries temporais e supõe que estes são amostras de populações com distribuições desconhecidas. Essas distribuições são estimadas através de um método não paramétrico e uma medida de divergente entre funções de densidade de probabilidade é utilizada para calcular a distância entre as mesmas e as classifica de acordo com seus vizinhos.

Para avaliar a efetividade do método, dois experimentos foram conduzidos. O primeiro analisa o impacto dos parâmetros do método no resultado da classificação e na métrica de distância entre dois conjuntos de dados diferentes. O segundo avalia os resultados obtidos em diversos conjuntos de dados e os compara com outros métodos da literatura. Os resultados indicam que a proposta se apresenta como uma alternativa promissora, obtendo resultados comparáveis aos melhores métodos da literatura.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

TIME SERIES CLASSIFICATION VIA DIVERGENCE BETWEEN PROBABILITY DENSITIES IN PHASE SPACE

André Santos Teixeira de Carvalho

November/2016

Advisors: Carlos Eduardo Pedreira
Carlos Eduardo Ribeiro de Mello

Department: Systems Engineering and Computer Science

Time series are present in several areas of research, such as electrocardiograms, voice and handwritting recognition. Time series classification has been attracting the attention of researchers from different areas of knowledge. It aims at detecting patterns that discriminate against these time series in a pre-defined set of classes. Several techniques for time series classification are available in the literature. Many of these approaches are based on metrics between time series to apply them on classic classifiers such as the k-nearest neightboors, whereas others build statistical models on the data.

In this work, we propose a new method for time series classification. The proposal employs the technique of time delay embedding to reconstruct the time series state space assuming these are samples *iid* from the unknown population corresponding distributions. These distributions are then estimated by nonparametric methods and a divergence measure between them is taken as the distance between the correspondng time series to classify them according to their neightbors.

To evaluate the effectiveness of the proposal, two experiments were conducted. The first analyzes the impact of the method parameters on the classification and on the metric in two different data sets. The second evaluates the results obtained in several data sets and compares them to other literature methods. The results indicate that the proposed method seems to be a promising alternative, achieving results comparable to the current state of the art.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Nos últimos anos, o aumento na capacidade e a redução dos custos de armazenamento de dados vem facilitando a retenção de dados históricos por tempo indeterminado. Esse fato, aliado a outros avanços tecnológicos relacionados à chamada Internet das Coisas, como o desenvolvimento de sensores e dispositivos móveis, vem contribuindo para um aumento crescente de aplicações envolvendo a coleta, análise e extração de conhecimento a partir de dados temporais.

Uma série temporal, como o nome sugere, é uma coleção de observações realizadas, geralmente em intervalos de tempo fixos, ao longo do tempo[1]. Podem ter aplicações nas mais diversas áreas, como, por exemplo, eletrocardiogramas, o movimento dos preços no mercado de ações, sequências de proteínas, espectrogramas de alimentos e no reconhecimento de fala[2].

Existem diferentes tarefas relacionadas à Mineração de Dados de séries temporais, cada uma com objetivos e aplicações distintas. No caso dos preços de ações na bolsa de valores, por exemplo, estamos interessados na **previsão** de valores futuros a partir dos dados passados. Outro exemplo são os eletrocardiogramas, onde há preocupação com **detecção de anomalias** para fins de diagnóstico[2][3].

Nesta dissertação, tratamos especificamente a tarefa de **classificação** de séries temporais. Essa tarefa tem atraído a atenção de pesquisadores de diversas áreas do conhecimento, onde deseja-se classificar uma série temporal em uma de duas ou várias classes predefinidas. A classificação de batidas de coração para fins de diagnóstico de anomalias é um exemplo de classificação de séries temporais bastante comum na literatura[4][5][6][7][3]. Nesse exemplo, uma batida de coração é comparada a batidas que foram anotadas previamente por médicos especialistas.

A classificação de séries temporais não é uma tarefa simples, pois características como tamanho, alinhamento e ruído exercem grande influência no classificador treinado. Dessa forma, um algoritmo de classificação precisa ser robusto em relação a essas características, além de ser capaz de detectar padrões locais e globais das séries.

Ao longo dos anos, alguns métodos foram desenvolvidos a fim de atender esses requisitos. Dentre as abordagens encontradas podemos citar o classificador baseado em vizinhos próximos utilizando uma métrica de distância capaz de corrigir o desalinhamento entre as séries, chamado de *Dynamic Time Warping*[8], bem como modelos baseados nos *Hidden Markov Models*[9]. A maioria desses métodos utiliza modelos e/ou distâncias baseadas diretamente no conjunto de dados das séries temporais.

Dentre os métodos que buscam mapear as séries temporais em um espaço multidimensional, destacam-se aqueles que utilizam técnicas para representar o espaço de fases da série. Na literatura, encontram-se métodos como o desenvolvido em [10], onde modelos estatísticos são construídos para representar cada uma das classes e métodos como [11], no qual os autores extraem *features* a partir deste espaço. Ou seja, métodos que buscam definir métricas de similaridade entre as representações das séries no espaço de fases não foram explorados pela literatura.

Como a representação no espaço de fases pode possuir um poder discriminatório maior do que as presentes apenas no domínio das frequências[12], analisar e propor métricas de distância entre séries temporais representadas no espaço de fases mostra-se um caminho de pesquisa interessante, com oportunidades de melhoria dos resultados do estado da arte e características desafiadoras.

1.1 Contribuições

Neste trabalho é proposto um método de classificação de séries temporais baseado no divergente entre as funções de densidade de probabilidade estimadas a partir de cada amostra (série). Dessa forma, cada série temporal é representada por sua função de densidade de probabilidade (estimada a partir de sua representação no espaço de fases), caracterizando um novo espaço de representação das séries, onde modelos de classificação são posteriormente aplicados. Essa representação funcional tende a ser mais complexa e portanto captura melhorar a relação entre duas séries. Considerando técnicas de estimação de densidades de probabilidade não paramétricas, o método proposto permite o treinamento e obtenção de modelos eficientes sobretudo para séries temporais complexas.

A proposta foi comparada com outros métodos clássicos da literatura em diversos conjuntos de dados diferentes e seus resultados foram discutidos. Por fim, uma análise da influência dos parâmetros que constituem o método foi conduzida em dois conjuntos de dados diferentes e seu comportamento foi discutido.

Dessa forma, as principais contribuições deste trabalho são:

1. Proposta e implementação de um novo método de classificação de séries tem-

porais, combinando teorias clássicas de sistemas dinâmicos e medidas de divergentes entre funções de densidade de probabilidade, a fim de definir uma relação de distância entre séries capaz de detectar padrões locais e globais. O método obteve resultados comparáveis aos outros métodos da literatura (capítulo 3).

2. Análise qualitativa e quantitativa da influência dos parâmetros do método com relação à performance de classificação e ao comportamento da métrica de divergente (seção 4.2).

3. Comparação do método proposto com métodos clássicos da literatura, como o vizinho mais próximo com distância euclidiana e *dynamic time warping*, em diversos conjuntos de dados de séries temporais de diversas áreas (seção 4.3).

1.2 Organização

A presente dissertação está organizada da seguinte forma: o capítulo 2 apresenta as teorias utilizadas pelo método proposto e faz uma revisão sobre algumas abordagens comuns encontradas na literatura para a tarefa de classificação de séries temporais. Em seguida, o capítulo 3 apresenta uma descrição do método proposto. Neste capítulo, as motivações e ideias gerais do método são apresentadas, seguidas de uma descrição detalhada de cada uma das etapas e técnicas envolvidas no método.

Uma série de experimentos é conduzida no capítulo 4 com o objetivo de analisar o comportamento do método em diversos conjuntos de dados diferentes. Além disso, os resultados obtidos são comparados a outros métodos da literatura.

Por fim, no capítulo 5 a conclusão deste trabalho é apresentada em conjunto com considerações sobre os objetivos alcançados e possíveis pontos de melhoria por meio de trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

Neste capítulo serão apresentados os conceitos básicos utilizados nesta dissertação, além de uma revisão bibliográfica dos trabalhos relacionados ao método proposto.

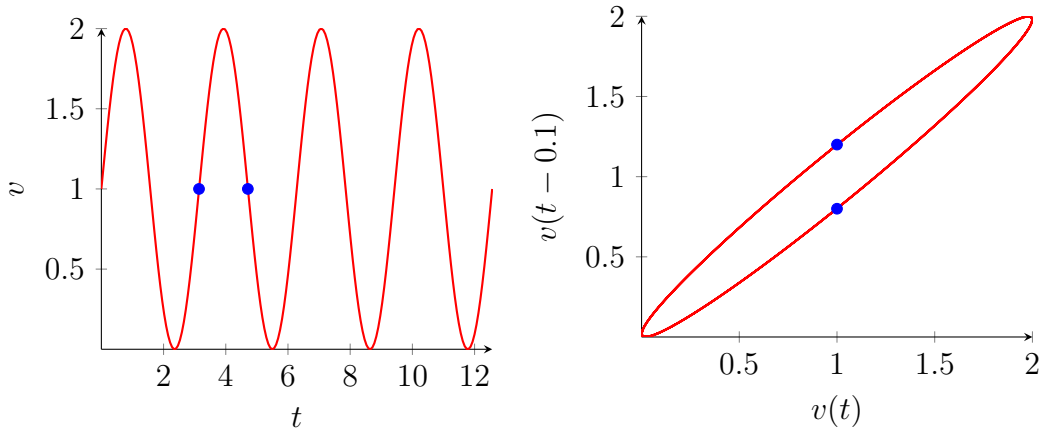
Na seção 2.1 apresenta-se o espaço de fases e a técnica utilizada para sua construção a partir de séries temporais. Em seguida, na seção 2.2 discute-se as técnicas de estimativa de funções de densidade de probabilidade e na seção 2.3 são discutidas as métricas de similaridade entre estas funções. Na seção 2.4 o problema de classificação de séries temporais é apresentado e discutido em conjunto com alguns dos métodos presentes na literatura. A seção 2.5 é dedicada aos métodos de classificação de séries temporais baseados no espaço de fase reconstruído e, por fim, na seção 2.6 conclui-se a revisão.

2.1 Espaço de Fases

Um sistema dinâmico consiste no conjunto de estados possíveis, combinado com uma regra que determina o estado presente em termos de estados passados[13]. Uma série temporal pode ser vista como medidas feitas neste sistema ao longo do tempo. Por exemplo, podemos medir a velocidade de um projétil ao longo do tempo (figura 2.1a). Neste caso, apenas o valor de $v(t)$ não é capaz de descrever por completo o estado do sistema em determinado instante, visto que, com $v(t) = 1$, por exemplo, a velocidade do projétil pode estar crescente ou decrescente.

O Teorema de Takens[14] prova que o espaço de estados de um sistema desconhecido pode ser reconstruído por meio do processo conhecido como *embedding*. Quando este processo é efetuado de maneira correta, o Teorema garante que o espaço reconstruído possui dinâmicas topologicamente idênticas ao sistema original e a esse espaço dá-se o nome de *Reconstructed Phase Space*(RPS).

Pode-se construir o RPS a partir de uma série temporal utilizando um processo chamado de *time-delay embedding*[15]. Seja $x = x_n, n = 1 \dots N$ uma série temporal



(a) (b)

Figura 2.1: Exemplo da técnica *time-delay embedding*

discreta. A matriz de reconstrução do espaço de fase com dimensão d e atraso τ, no *time-delay embedding* é definida como:

$$X = \begin{bmatrix} \mathbf{x_{1+(d-1)\tau}} \\ \mathbf{x_{2+(d-1)\tau}} \\ \vdots \\ \mathbf{x_N} \end{bmatrix} = \begin{pmatrix} x_{1+(d-1)\tau} & \dots & x_{1+\tau} & x_1 \\ x_{2+(d-1)\tau} & \dots & x_{2+\tau} & x_2 \\ \vdots & \vdots & \vdots & \vdots \\ x_N & \dots & x_{N-(d-2)\tau} & x_{N-(d-1)\tau} \end{pmatrix}. \tag{2.1}$$

Cada linha da matriz 2.1 é um vetor que representa um único ponto no espaço:

$$\mathbf{x_n} = \begin{bmatrix} x_{n-(d-1)\tau} & \dots & x_{n-\tau} & x_n \end{bmatrix}, \tag{2.2}$$

onde $\mathbf{n} = (1 + d(d-1))\tau \dots N$. Cada um desses vetores constitui um ponto neste espaço de fase reconstruído[10]. Por exemplo, o RPS da série temporal discreta $T_1 = [1, 2, 3, 3, 3, 4, 5, 6]$ com $d = 3$ e $\tau = 2$ é representado pela matriz:

$$X = \begin{bmatrix} \mathbf{x_5} \\ \mathbf{x_6} \\ \mathbf{x_7} \\ \mathbf{x_8} \end{bmatrix} = \begin{pmatrix} x_5 & x_3 & x_1 \\ x_6 & x_4 & x_2 \\ x_7 & x_5 & x_3 \\ x_8 & x_6 & x_4 \end{pmatrix} = \begin{pmatrix} 3 & 3 & 1 \\ 4 & 3 & 2 \\ 5 & 3 & 3 \\ 6 & 4 & 3 \end{pmatrix}. \tag{2.3}$$

Podemos, então, utilizar a técnica do *time-delay embedding* para reproduzir o estado do sistema da figura 2.1a utilizando duas dimensões. Na figura 2.1b podemos visualizar a trajetória do sistema gerado a partir do *time-delay embedding* com $d = 2$ e $\tau = 0.1$, a esta visualização dá-se o nome *delay plot*[13]. Dessa forma, estamos descrevendo o estado do sistema utilizando não apenas $v(t)$, mas também $v(t-0.1)$, mapeando todos os estados do sistemas para pontos diferentes. No caso de $v(t) = 1$, por exemplo, conseguimos diferenciar o estado de aceleração positiva ($v(t-1) < v(t)$)

do caso em que esta é negativa ($v(t-1) > v(t)$) pois são os dois pontos $(1, x)$ nos lados opostos do ciclo representado em 2.1b.

Uma função que mapeia de maneira 1-para-1 os pontos de uma representação compacta (neste caso, uma série temporal) para um espaço $\mathbb{R}^d$ é chamada de um *embedding* do conjunto e d é chamado de *embedding dimension*. No exemplo da figura 2.1, $d = 2$ é uma *embedding dimension*, porque representa tal mapeamento.

Nem sempre é possível obter esse mapeamento com $d = 2$. Segundo o Teorema de Takens[14], se a dimensão do espaço original é m, então é necessário uma dimensão $d > m * 2$, no *time-delay embedding*, para que o espaço reconstruído represente, de maneira completa, a topologia do espaço original. É possível obter um mapeamento suficientemente bom, mesmo com $d < m*2$[12], e que, dependendo da complexidade do sistema original, m pode ser infinita[13].

Como o RPS possui as mesmas dinâmicas que o espaço de estados original, essa representação pode ser utilizada tanto na descrição, quanto na obtenção de informações relacionadas a ele. Dessa forma, a técnica do *time-delay embedding* pode ser utilizada para extração de informações da série temporal para fins de classificação destas, pois esta representação pode possuir um poder discriminatório maior do que as presentes apenas no domínio das frequências[12]. Na seção 2.4.3 alguns modelos que utilizam a representação em RPS e o *time-delay embedding* para a classificação de séries temporais serão apresentados[6][16].

2.1.1 Escolha dos parâmetros do RPS

Existem algumas heurísticas para a estimação dos parâmetros para a construção do RPS, d e τ. O parâmetro d pode ser obtido a partir da técnica de falsos vizinhos [6]. Falsos vizinhos são pontos no espaço de fase de dimensão d que são próximos entre si, porém não são próximos no espaço de dimensão $d + 1$. Considerando novamente a série temporal discreta $T_1 = [1, 2, 3, 3, 3, 4, 5, 6]$, por exemplo, podemos construir sua representação com $d = 2$ e $\tau = 2$:

$$X = \begin{pmatrix} 3 & 1 \\ 3 & 2 \\ 4 & 3 \\ 5 & 3 \\ 6 & 4 \end{pmatrix}. \tag{2.4}$$

Neste espaço, a distância euclidiana D_2 entre os pontos $(3, 1)$ e $(3, 2)$ é $D_2 = 1$. Na representação com $d = 3$ (equação 2.3), uma nova dimensão é adicionada a cada um desses pontos: 3 no primeiro (resultando no ponto $(3, 3, 1)$) e 4 no segundo (resultando no ponto $(4, 3, 2)$), ou seja, a distância entre os valores da nova dimensão

$D_3 = 1$. O método dos falsos vizinhos estes pontos serão considerados falsos vizinhos se:

$$\frac{D_3}{D_2} > R_{tol}, \tag{2.5}$$

onde R_{tol} é um limiar fixo. O número de pontos que são falsos vizinhos indica se uma dimensão mais alta deveria ser utilizada, em [6], por exemplo, 0.001 é o limite no percentual de falsos vizinhos usados na seleção do parâmetro.

O método de Cao [17], também utilizado na literatura, é uma alteração no método de falsos vizinhos no qual não se faz mais necessário a determinação de um limiar.

Dois métodos existentes para a seleção do parâmetro τ são baseados na informação da autocorrelação[1] e na informação mútua[18].

Dada um atraso τ, a autocorrelação entre X_t e $X_{t-\tau}$ é calculada como:

$$C(X_t, X_{t-\tau}) = \frac{E(X_t X_{t-\tau}) - E(X_t)^2}{E([X_t - E(X_t)]^2)}. \tag{2.6}$$

O método escolhe o τ que corresponde a primeira vez que a autocorrelação torna-se positiva[1]. A informação mútua indica a quantidade de informação conhecida sobre o valor de $X_{t+\tau}$ dado que se conhece o valor de X_τ. Para seu cálculo, um histograma com *bins* de tamanho fixo é construído e a informação mútua é computada por:

$$M(X_t, X_{t-\tau}) = \sum_{i,j} p_{ij}(\tau) \ln \frac{p_{ij}(\tau)}{p_i p_j}, \tag{2.7}$$

onde p_i é a probabilidade do ponto X_t estar no i-ésimo *bin* e $p_{ij}(\tau)$ é a probabilidade condicional de X_t estar no i-ésimo *bin* no tempo t e no j-ésimo *bin* no tempo $t + \tau$[18]. O primeiro mínimo da função de informação mútua foi definido como um bom critério para a escolha do atraso na construção do RPS, inclusive sendo superior ao método da autocorrelação[12], pois indicaria o τ em que o acrescimento de informação é máximo[1].

2.2 Estimativa de densidades de probabilidade

Os métodos de estimação de densidade de probabilidade podem ser divididos em duas categorias: paramétricos e não paramétricos. Na estimativa paramétrica, assume-se que a densidade segue um modelo paramétrico já conhecido, dessa forma, a tarefa se resume a estimar os parâmetros de tal modelo, usando, por exemplo, o *maximum likelihood*. Os métodos não paramétricos, por outro lado, partem do princípio de que não é sabida nenhuma informação sobre a densidade *a priori*, o que

é muito comum em situações práticas[19].

Existem diversos métodos não paramétricos para a estimativa da densidade de probabilidade a partir de um conjunto de dados. A **Janela de Parzen**, também conhecida como *Kernel Density Estimation*(KDE)[20], é um dos mais utilizados na literatura[21]. O método estima de maneira empírica a função de densidade de probabilidade de acordo com a densidade local de cada uma das observações.

Para um dado conjunto de observações $x_1, x_2, ..., x_N$ obtidas a partir de uma função de densidade de probabilidade f desconhecida, o KDE provê a seguinte estimativa de densidade para uma observação x:

$$\hat{f}(x) = \frac{1}{N} \sum_{t=1}^{N} K_h(x, x_t), \tag{2.8}$$

onde K_h é uma função simétrica que integra para 1 ao longo do domínio. Esta é conhecida como função de Kernel, ou apenas, Kernel e h é chamado de largura de banda. Dentre os diversos *Kernels* utilizados na literatura, um dos mais comuns é o Gaussiano:

$$K(x, x_t) = e^{-\frac{||x-x_t||^2}{2\sigma^2}}, \tag{2.9}$$

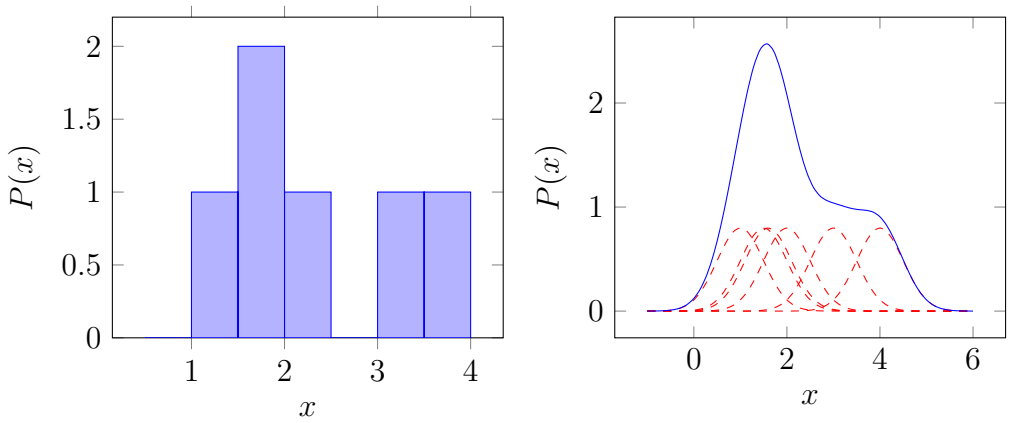
onde σ, o desvio padrão da Gaussiana, é a largura de banda do *kernel*.

A Janela de Parzen é uma generalização de histogramas, onde são utilizados *bins* infinitesimais e, com isso, dependendo da escolha da função de *kernel* é possível obter uma aproximação muito mais suave e contínua. Por exemplo, a figura 2.2 compara ambos os métodos ao estimar uma função de densidade de probabilidade a partir de 6 pontos. A utilização do *kernel* Gaussiano na Janela da Parzen corresponde a centralizar uma gaussiana com desvio padrão σ em cada um dos pontos e calcular sua função de densidade de probabilidade. Após isso, essas funções são combinadas em uma só. Analisando a referida figura, fica clara a superioridade do KDE em obter uma aproximação mais suave em relação aos histogramas.

2.2.1 O problema da seleção da largura de banda

A largura de banda h é um parâmetro determinante na qualidade da aproximação obtida pela Janela de Parzen. É sabido, inclusive, que este é um parâmetro mais importante do que a própria escolha de *kernel* a ser utilizado[19]. Um h muito grande pode falhar em capturar todas as características da função de densidade de probabilidade em questão enquanto que uma largura de banda h muito pequena pode obter uma aproximação pouco suave, muito suscetível a sofrer com *overfitting*.

Um exemplo de como o *overfitting* pode ocorrer está ilustrado na figura 2.3, na qual foi plotada as estimativas obtidas pelo KDE para diversos valores de h



(a) Histograma

(b) KDE

Figura 2.2: Comparação entre histogramas e Janela de Parzen

e tamanho de amostra $N = 1000$. Comparando a estimativa com $h = 0.01$ e $h = 0.1$ vemos que a segunda obtém uma curva muito mais suave e, portanto, menos suscetível a ruído nos dados. Porém, ao analisar a curva para $h = 0.5$ podemos perceber que a suavidade passou a ser um problema e a mesma não é capaz de estimar a função de densidade de probabilidade original com qualidade.

Dessa forma, para obter uma estimativa de qualidade, faz-se necessária uma escolha de largura de banda adequada para a densidade que se deseja estimar. Existem diversos métodos presentes na literatura que tentam, a partir dos dados, escolher uma largura de banda suficientemente boa. Em geral, para contabilizar a performance de dada largura estes métodos utilizam uma das duas medidas de erro: o *Integrated Squared Error*(ISE) ou o *Mean Integrated Squared Error*(MISE).

Os diversos métodos da literatura para a escolha da largura de banda podem ser divididos em três categorias: métodos baseados em validação cruzada, métodos plugáveis e métodos mistos[22].

Validação Cruzada

Em geral, os métodos de validação cruzada escolhem uma largura de banda h que minimiza o ISE entre a densidade de probabilidade original f e a densidade de probabilidade estimada pelos dados $\hat{f}_h$[22]:

$$\begin{aligned} ISE(f, \hat{f}_h) &= \int_{-\infty}^{\infty} [f(x) - \hat{f}_h(x)]^2 dx \\ &= \int_{-\infty}^{\infty} f^2(x)dx - 2\int_{-\infty}^{\infty} f(x)\hat{f}_h(x)dx + \int_{-\infty}^{\infty} \hat{f}_h{}^2(x)dx. \end{aligned} \quad (2.10)$$

Analisando a equação 2.10, nota-se que o primeiro termo não depende de h e,

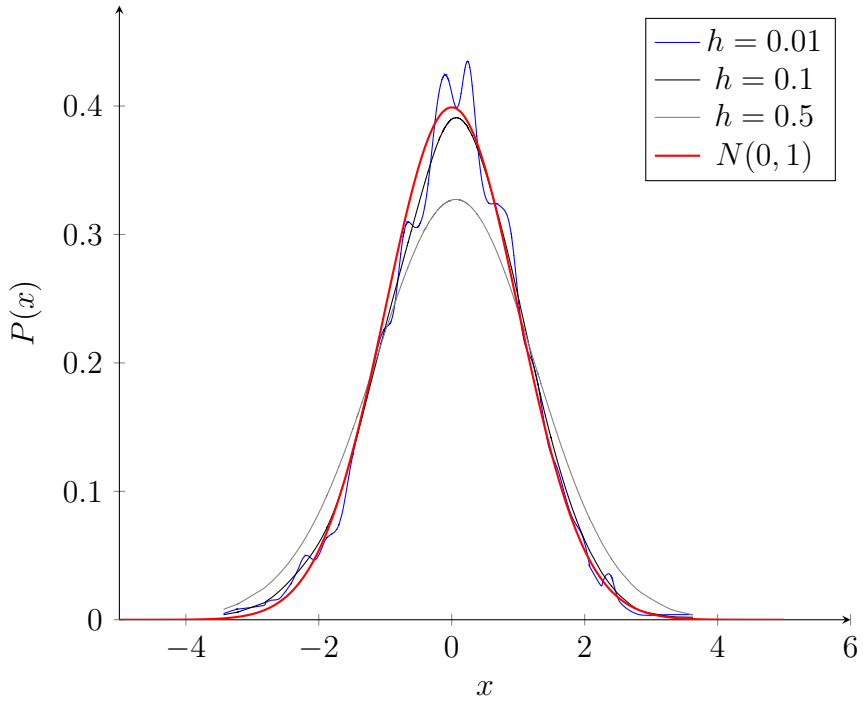


Figura 2.3: Estimativa da pdf de uma gaussiana utilizando o KDE

portanto, pode ser removido da equação de minimização:

$$CV(h) = -2\int_{-\infty}^{\infty} f(x)\hat{f}_h(x)dx + \int_{-\infty}^{\infty} \hat{f}_h{}^2(x)dx. \tag{2.11}$$

Utilizando o KDE e o teorema de convolução das Gaussianas podemos substituir a segunda integral por um somatório, obtendo:

$$CV(h) = -2\int_{-\infty}^{\infty} f(x)\hat{f}_h(x)dx + \frac{1}{N^2}\sum_{i=1,j=1}^{N,N} K_{2h}(x_i, x_j). \tag{2.12}$$

Dessa forma, os métodos de validação cruzada diferem entre si na forma em que calculam o primeiro termo da equação 2.12[22].

A abordagem clássica, conhecida como **validação cruzada de mínimos quadrados**[23], observa o fato de que $\int_{-\infty}^{\infty} f(x)\hat{f}_h(x)dx$ é o valor esperado de $\hat{f}_h(X)$ onde a esperança é calculada em respeito a uma variável independente X e estima este valor por:

$$E[\hat{f}_h(X)] = \frac{1}{N}\sum_{i=1}^{N} \hat{f}_{h,-i}(X_i), \tag{2.13}$$

onde $\hat{f}_{h,-i}$ é o *leave-one-out estimator*, dado por:

$$\hat{f}_{h,-i}(x) = \frac{1}{N-1}\sum_{j=1,i\neq j}^{N} K_h(x, X_j). \tag{2.14}$$

Dessa forma, o método de validação cruzada clássico minimiza a seguinte função em respeito a h:

$$CV(h) = \frac{1}{N^2} \sum_{i=1,j=1}^{N,N} K_{2h}(x_i, x_j) - \frac{2}{N(N-1)} \sum_{i=1,j\neq i}^{N,N} K_h(x, x_j). \tag{2.15}$$

Métodos plugáveis

Os métodos plugáveis utilizam o erro médio quadrático integrado, $MISE(h)$, como função de objetivo a ser minimizada na busca pela largura de banda. O $MISE(h)$ é dado pela equação:

$$MISE(h) = MISE\left[\hat{f}_h(x)\right] = \int_{-\infty}^{\infty} MSE\left[\hat{f}_h(x)\right] dx, \tag{2.16}$$

onde,

$$MSE\left[\hat{f}_h(x)\right] = \frac{h^4}{4}\mu_2(K)^2||f^{''}(x)||_2^2 + \frac{1}{nh}||K||_2^2 + o\left(\frac{1}{nh}\right) + o(h^4), \tag{2.17}$$

de forma que, $\mu_2(K) = \int_{-\infty}^{\infty} s^2K(s)ds$, $||K||_2^2 = \int_{-\infty}^{\infty} K^2(s)ds$ e $o(x)$ é um termo de ordem x. Diferenciando a equação anterior no que se refere a h, obtemos a seguinte equação para h_{opt} que minimiza o erro médio quadrático integrado:

$$h_{opt} = ||K||_2^{2/5} \left(||f^{''}||_2^2[\mu_2(k)]^2 n\right)^{-1/5}. \tag{2.18}$$

Dessa forma, os métodos plugáveis tem por objetivo estimar $||f^{''}||_2^2$ a fim de encontrar a largura de banda ótima h_{opt}.

A **regra de Silverman**[24] supõe que f é uma distribuição normal, com média μ e variância σ^2. Dessa forma:

$$||f^{''}||_2^2 = \sigma^{-5} \int_{-\infty}^{\infty} \left[\omega^{''}(x)\right]^2 dx = \sigma^{-5}\frac{3}{8\sqrt{\pi}} \approx 0.212\sigma^{-5}, \tag{2.19}$$

onde $\omega(x)$ é a função de densidade de probabilidade de uma distribuição normal padrão. Substituindo o desvio padrão σ por um estimador $\hat{\sigma}$:

$$\hat{\sigma} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}, \tag{2.20}$$

obtendo então, a seguinte largura de banda:

$$h_S = \frac{4\hat{\sigma}^5}{3n}^{1/5} \approx 1.06\hat{\sigma}n^{-1/5}. \tag{2.21}$$

Tabela 2.1: Distâncias entre densidades de probabilidade

Nome	**Definição**
Distância de Chernoff[26]	$-\log(\int_X p_1^{\alpha_2}(x)p_2^{\alpha_1}(x)dx)$
Distância de Bhattacharyya[27]	$-\log(\int_X [p_1^{\alpha_2}(x)p_2^{\alpha_1}(x)]^{1/2}dx)$
Distância de Matusita[28]	$(\int_X [\sqrt{p_1(x)} - \sqrt{p_2(x)}]^2 dx)^{1/2}$
Divergente de Kullback-Leiber[29]	$\int_X p_1(x) \log \frac{p_1(x)}{p_2(x)} dx$
Distância de KL simétrica[29]	$\int_X [p_1(x) - p_2(x)] \log \frac{p_1(x)}{p_2(x)} dx$
Distância de Patrick-Fisher[30]	$(\int_X [p_1(x)\pi_1 - p_2(x)\pi_2]^2 dx)^{1/2}$
Distância de Lissack-Fu[31]	$\int_X [p_1(x)\pi_1 - p_2(x)\pi_2]^{\alpha_1} [p_1(x)\pi_1 - p_2(x)\pi_2]^{\alpha_2} dx$
Distância de Kolmogorov[32]	$\int_X [p_1(x)\pi_1 - p_2(x)\pi_2] dx$
Integrated Squared Error[25]	$\int_X [p_1(x) - p_2(x)]^2 dx$

A regra de Silverman faz uma forte suposição sobre a distribuição f. Caso a real distribuição dos dados seja uma normal, h_S irá obter uma boa estimativa da distribuição original. Em geral, h_S tende a obter uma aproximação satisfatória caso f seja unimodal, ligeiramente simétrica e sem cauda longa[22].

Métodos mistos

Os métodos que utilizam validação cruzada são criticados pelos autores por terem a tendência de serem menos suaves e por sofrerem uma alta variabilidade em amostras muito grandes. Ao mesmo tempo, os métodos plugáveis são mais estáveis, porém tendem a ser muito suaves[22]. Os métodos mistos tentam combinar estimadores de ambas as categorias a fim de balancear suas características positivas.

Um exemplo de método misto são aqueles que fazem misturas de larguras de banda, por meio de combinações multiplicativas[22]. Por exemplo, seja h_{CV} a largura obtida por um método baseado em validação cruzada e h_P uma largura obtida por um método plugável, pode-se calcular obter um h_F como $h_F = (h_{CV}^{\alpha} h_P^{\beta})^{\frac{1}{\alpha+\beta}}$.

2.3 Similaridade entre densidades de probabilidade

Existem diversas métricas de distância entre distribuições de probabilidade, conhecidas como *divergences* (divergentes). Essas medidas são chamadas de pseudo-métricas, uma vez que não satisfazem todos os axiomas das métricas, como simetria e inequalidade do triângulo[25].

Na tabela 2.1 estão listadas algumas das medidas de distância/divergente entre duas funções de densidade de probabilidade p_1 e p_2. Calcular essas medidas não é trivial para o caso genérico. Contudo, apenas para algumas famílias de densidade, como no caso das gaussianas, por exemplo, é possível obter formulas analíticas

fechadas[33]. Apesar disso, no caso do *Integrated Squared Error*(ISE), é possível obter uma formula analítica para o seu cálculo a partir de densidades estimadas pelo KDE.

O ISE calcula a divergente entre duas funções de densidade de probabilidade, p e q, como a área total abaixo da função que representa a diferença quadrática entre elas[25]. Este é dado pela fórmula:

$$\begin{aligned} ISE(p,q) &= \int_{-\infty}^{\infty} [p(x) - q(x)]^2 dx \\ &= \int_{-\infty}^{\infty} p^2(x)dx - 2\int_{-\infty}^{\infty} p(x)q(x)dx + \int_{-\infty}^{\infty} q^2(x)dx. \end{aligned} \quad (2.22)$$

É fácil notar que, a medida que p se aproxima de q, $ISE(p,q)$ se aproxima de 0. Conforme visto na seção 2.2, pode-se substituir p e q pelas estimativas obtidas pelo KDE a partir das amostras P e Q de tamanhos N_1 e N_2, respectivamente. Portanto, tem-se que:

$$\hat{p}(x) = \frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i) \quad (2.23)$$

e

$$\hat{q}(x) = \frac{1}{N_2}\sum_{j=1}^{N_2} G_{h_2}(x, x_j), \quad (2.24)$$

onde G_h é o *kernel* gaussiano, obtendo-se assim:

$$\begin{aligned} \widehat{ISE}(p,q) = \int_{-\infty}^{\infty} [\frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i)]^2 dx \\ - 2\int_{-\infty}^{\infty} [\frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i)][\frac{1}{N_2}\sum_{j=1}^{N_2} G_{h_2}(x, x_j)]dx + \\ \int_{-\infty}^{\infty} [\frac{1}{N_2}\sum_{j=1}^{N_2} G_{h_2}(x, x_j)]^2 dx, \end{aligned} \quad (2.25)$$

que pode ser reescrita como:

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i'=1}^{N_1,N_1}\int_{-\infty}^{\infty} G_{h_1}(x,x_i)G_{h_1}(x,x_{i'})dx$$
$$- 2\frac{1}{N_1}\frac{1}{N_2}\sum_{i,j=1}^{N_1,N_2}\int_{-\infty}^{\infty} G_{h_1}(x,x_i)G_{h_2}(x,x_j)dx$$
$$+ \frac{1}{N_2^2}\sum_{j,j'=1}^{N_2,N_2}\int_{-\infty}^{\infty} G_{h_2}(x,x_j)G_{h_2}(x,x_{j'})dx. \quad (2.26)$$

O teorema de convolução das Gaussianas diz que:

$$\int_{-\infty}^{\infty} G_{h_1}(x,x_t)G_{h_1}(x,x_i)dx = G_{h_1+h_2}(x_t,x_i). \quad (2.27)$$

Logo,

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i'=1}^{N_1,N_1} G_{2h_1}G_{h_1}(x_i,x_{i'})dx$$
$$- 2\frac{1}{N_1}\frac{1}{N_2}\sum_{i,j=1}^{N_1,N_2} G_{h_1+h_2}(x_i,x_j)dx$$
$$+ \frac{1}{N_2^2}\sum_{j,j'=1}^{N_2,N_2} G_{2h_2}(x_j,x'_j)dx. \quad (2.28)$$

Dessa forma, a equação 2.28 é uma expressão analítica fechada que permite o cálculo do ISE inteiramente a partir das observações em P e Q.

2.4 Classificação de séries temporais

Problemas de diversas áreas podem ser modelados como problemas de classificação de séries temporais, tais como: reconhecimento de fala, detecção de movimentos e detecção de arritmias em eletrocardiogramas. Estes são alguns dos exemplos comumente encontrados.

Alta dimensionalidade, alta correlação entre as *features* e a grande quantidade de ruído são algumas das características que tornam as séries temporais desafiadoras aos algoritmos clássicos de classificação. Essa dificuldade as torna um problema interessante que tem sido explorado pela literatura. Porém, grande parte das contribuições tem sido voltadas para a definição de medidas de similaridade entre as séries temporais que seriam utilizadas internamente em algoritmos já consolidados[34].

Os métodos de classificação de séries temporais podem ser divididos em 3 categorias distintas: métodos baseados em *features*, métodos baseado em distâncias entre as séries e métodos baseados em modelos[35]. As subseções a seguir fazem uma breve revisão sobre essas categorias.

2.4.1 Métodos baseados em *features*

Métodos convencionais de classificação, como Redes Neurais e Árvores de Decisão, são construídos para a classificação a partir de vetores de *features*. Portanto para que estes sejam utilizados na classificação de séries temporais é necessário definir técnicas para efetuar o mapeamento das séries temporais para o espaço de *features*.

A etapa que difere os métodos dessa categorias dos demais é a etapa de extração de features. Algumas técnicas utilizam os coeficientes de diversas transformadas como *features* dos classificadores. A *Discrete Fourier Transform*(DFT) permite a extração de *features* do domínio das frequências, enquanto que a *Discrete Wavelet Transform*(DWT) permite a análise de informações de ambos os domínios, tempo e frequência[36][5], por exemplo.

Um dos métodos baseados em *features* consiste na determinação de subsequências que possuem o maior poder discriminatório entre as classes do problema. Os *shapelets*, como são chamados em [37], são encontrados a partir do conjunto de treinamento, maximizam o ganho de informação e podem ser utilizados a fim de construir uma árvore de decisão para ser utilizada na classificação de uma série temporal desconhecida.

Alguns métodos baseados em *features* são construídos para problemas específicos. Um desses casos ocorre para a classificação de batidas de coração extraídas de eletrocardiogramas[3][5]. Nestes, *features* com interpretação médica, como a distância entre as diversas fases que compõem uma batida, a frequência cardíaca e outras, são extraídas a partir das séries temporais e utilizadas nos mais diversos métodos de classificação, como Redes Neurais e *Support Vector Machines*[3].

Como nesses métodos as séries temporais são representadas por suas *features* é possível que haja uma perda de informação que afete na performance do classificador; por tal motivo, é necessário analisar, caso a caso, e escolher as melhores *features*.

2.4.2 Métodos baseados em medidas de similaridade

Os métodos baseados em medidas de similaridade consistem na definição de uma medida de similaridade, $D(s_1, s_2)$, entre duas séries temporais s_1 e s_2. Essa medida é então utilizada em algum método de classificação existente, como o *k-nearest neightbours* ou K-vizinhos próximos (KNN). No KNN, dada uma série s que se deseja classificar, calcula-se a distância dela em relação a todas as séries do conjunto

de treinamento T. A série s será classificada como a classe dominante no conjunto das k séries mais próximas de s. O KNN é extremamente sensível à distância D utilizada[35] e, dessa forma, é necessária uma escolha criteriosa de métrica.

É interessante que a métrica escolhida seja robusta a algumas transformações usualmente presentes em séries temporais: escala (alterações na amplitude da série), *warping* (modificações temporais na série), ruído e *outliers*[2].

Uma revisão detalhada das medidas de similaridade entre séries temporais foi feita em [38]. Algumas das distâncias mais utilizadas como função de similaridade entre séries temporais são apresentadas a seguir.

Dentre as medidas de distância existentes, a **distância euclidiana** é uma das mais utilizadas na literatura[34], apesar de não ser robusta às diversas transformações mencionadas na seção anterior. Na distância euclidiana, uma série temporal de tamanho n é vista como um ponto no espaço R^n. Além de necessitar de pré-processamentos para se tornar menos suscetível a transformações na série[39], a distância euclidiana requer que ambas as séries possuam o mesmo tamanho.

Apesar dessas desvantagens, diversas das medidas de similaridade propostas especificamente para as séries temporais falham em superar um classificador baseado em vizinho mais próximo com distância euclidiana em dois conjuntos de dados sintéticos largamente utilizados pela literatura[34].

O ***Dynamic Time Warping*** (DTW)[8] é uma técnica originalmente aplicada na classificação de reconhecimento de fala que utiliza programação dinâmica a fim de alinhar duas séries temporais de forma que a medida de distância entre estas seja minimizada. O objetivo é mitigar o problema de desalinhamento entre as sequências. Um classificador baseado em 1-vizinho mais próximo (1-NN), utilizando o DTW, é difícil de ser superado por outros algoritmos de classificação[40][41].

Com o objetivo de alinhar duas sequências, $Q = q_1, q_2 \ldots, q_i, \ldots, q_p$ e $C = c_1, c_2, \ldots, c_j, \ldots, c_m$, o DTW constrói uma matriz p por m onde o elemento (i, j) representa a distância $d(q_i, c_j)$ entre os pontos q_i e c_j. Utiliza-se a programação dinâmica para encontrar um conjunto contínuo de elementos da matriz que definem o mapeamento entre Q e C e obedecem algumas restrições: 1) o caminho deve começar no elemento do canto inferior esquerdo e terminar no último elemento do canto superior direito (condição de contorno); 2) deve ser composto por elementos adjacentes da matriz (condição de continuidade); por fim, 3) seja $w_k = (a, b)$ o ponto imediatamente seguinte a $w_(k-1) = (a', b')$ então $a - a' \leq 0$ e $b - b' \leq 0$ (condição de monotonicidade)[40].

Dentre todos os caminhos possíveis, o DTW está interessado em obter aquele que minimiza as distâncias entre as séries. É esperado, de maneira intuitiva, que o caminho ótimo não esteja muito distante da diagonal da matriz[40]. Por esse motivo, algumas variações a fim de melhorar o tempo computacional do algoritmo foram fei-

tas para limitar o espaço de busca do caminho ótimo. O algoritmo *cDTW*[40] define um parâmetro w que determina uma condição adicional ao problema: $d(q_i, c_j) = \infty$, se $|i - j| > w$ e, com isso, limita a busca a caminhos que estão próximos da diagonal da matriz. Além de melhorar a performance computacional, o cDTW tem obtido resultados superiores a versão original do DTW. Em [7], o cDTW é comparado ao DTW em três conjuntos de dados de eletrocardiogramas e obteve resultados superiores em todos eles.

Outras abordagens foram propostas para reduzir os requisitos de tempo e espaço do DTW. O *FastDTW*[42] evita a abordagem força bruta da programação dinâmica começando a busca pelo melhor caminho em uma versão de baixa resolução da série temporal, aumentando a resolução até encontrar um caminho que atenda a série temporal na resolução original. O *SparseDTW*[43] tem como objetivo reduzir os requisitos relacionados ao espaço, $O(nm)$ do algoritmo original, utilizando a similaridade entre as séries ao seu favor. Em [44], a série temporal tem sua dimensionalidade reduzida por meio do particionamento da série em quadros e, em cada quadro, a série é representada pela média de seus valores daquele quadro, então, essa nova série reduzida é utilizada no PDTW, uma versão do DTW desenvolvida para trabalhar com essa representação.

Em [45], uma extensão do DTW é proposta, chamada de *Weighted dynamic time warping* (WDTW), que penaliza as distâncias de pontos em relação a diferença de posição entre os pontos nas duas sequências, de forma a diminuir a distorção causada por *outliers*. Ainda, uma forma sistemática de determinar os pesos a serem atribuídos é apresentada. O método é comparado com variações do DTW e outras métricas de similaridade entre séries temporais em conjuntos de dados disponíveis no *UCR Time Series Data Mining Archive*[41], obtendo resultados superiores na maioria dos conjuntos testados. Sua performance em algoritmos de clusterização também é apresentada e discutida.

2.4.3 Métodos baseados em modelos

Os métodos baseados em modelos assumem que as séries de uma classe são geradas por um modelo M[35]. Dada uma classe de séries, M modela a distribuição de probabilidade das séries naquela classe. Na etapa de treinamento, os parâmetros dos modelos são ajustados e na fase de classificação, a série é associada a classe com maior *likelihood*. Algumas abordagens clássicas baseadas em modelos são os *Hidden Markov Models*(HMM)[9] e os modelos ARMA[46].

Alguns métodos baseados em modelos utilizam o espaço de fases reconstruído para treinarem o modelo a ser utilizado na classificação. Estes métodos são brevemente descritos na próxima seção com outros métodos que utilizam esta repre-

sentação para a extração de *features*.

2.5 Classificação no espaço de fases reconstruído

Nesta seção, é feita uma breve revisão sobre trabalhos na área de classificação de séries temporais que utilizam o RPS para representação das séries. Na literatura foram encontrados métodos que, segundo a divisão apresentada na última seção, se enquadram nas categorias de modelos e *features*. Por outro lado, não foram encontrados métodos baseados nas distâncias entre as representações no RPS.

Em [10], os autores desenvolveram um modelo de mistura de gaussianas(GMM) das séries temporais no RPS e utilizaram um classificador Bayesiano para determinar a classe de cada uma das séries a partir dos modelos. Os parâmetros utilizados para a construção do espaço pela técnica do *time-delay embedding*, d e τ, são determinados pelas técnicas dos falsos vizinhos e do primeiro mínimo da função de informação mútua, respectivamente, heurísticas apresentadas na seção 2.1.1.

Um modelo de mistura de gaussianas(GMM) é definido como:

$$p(x) = \sum_{m=1}^{M} w_m N(x; \mu_m, \Sigma_m), \tag{2.29}$$

onde M é o número de misturas, $N(x; \mu_m, \Sigma_m)$ é uma normal com média μ_m e matriz de covariância Σ_m e w_m é o peso na mistura, onde $\sum w_m = 1$. O número de misturas é o único parâmetro do método desenvolvido em [10] e está relacionado diretamente com a distribuição das classes no RPS. Os parâmetros do GMM são estimados pelo algoritmo de *Expectation-Maximization*(EM)[47].

Após cada classe ter seu modelo GMM treinado pelo processo anterior, uma nova série temporal é classificada utilizando o *Bayesian maximum likelihood classifier*. Nele, os *likelihoods* são calculados da seguinte fórmula:

$$p(\mathbf{X}|c_i) = \prod_{n=1+(d-1)\tau}^{N} p(x_n|c_i), \tag{2.30}$$

onde $\mathbf{X}$ é uma matriz dada pelo RPS de dimensão d e atraso τ da série, x_n é um ponto no RPS, $p(x_n|c_i)$ é a probabilidade de x_n dado a i-ésima classe, calculada por 2.29. A série é então classificada como a classe que obtiver o maior *likelihood*. Os resultados deste método foram avaliados em três conjuntos de dados diferentes e comparados com um método baseado em redes neurais, obtendo resultados superiores a este[10].

Com o objetivo de classificar diferentes tipos de arritmias a partir de exames de ECG, [6] compara 3 metodologias de classificação: 1) GMM (a mesma apresentada

em [10]), 2) baseada em *bins* e 3) uma abordagem que utiliza o RPS como entrada para uma *time delay neural network*(TDNN).

Uma TDNN[48] é uma Rede Neural Artificial que tem como principal objetivo a manipulação de dados sequenciais, como séries temporais. Em [6], uma Rede Neural com duas camadas escondidas é utilizada, o número de entradas é igual à dimensão do RPS e existe apenas um neurônio na camada de saída. Para cada classe uma rede neural é treinada utilizando as d primeiras dimensões como entrada e utilizando estas para prever o valor da $(d+1)$-ésima dimensão.

A abordagem baseada em *bins* divide o espaço do RPS em pequenas células e a quantidade de pontos em cada uma delas é utilizada para calcular a densidade daquela célula. De forma a melhorar a estimativa da função de densidade de probabilidade, os autores utilizaram tamanhos de *bins* não uniformes no espaço, levando em consideração a quantidade de pontos próximos. Nesta abordagem, a classificação também é feita por meio de uma classificador Bayesiano.

Em todos os casos, os parâmetros do RPS são selecionados a partir da performance de classificação[6].

Essas abordagens já haviam sido apresentadas em [16] onde foram aplicadas no problema de classificação de arritmias e de reconhecimento de fala, onde os resultados indicaram que o método é capaz de discriminar as classes. Contudo, há a ressalva de que estes não serão superiores a outros métodos quando a fase do sinal original não for importante para a discriminação entre as classes.

Em [4], os autores definem regiões específicas no RPS que contém diferentes tipos de arritmias encontradas em exames de ECG. Essas regiões são utilizadas na classificação de batidas, obtendo bons resultados em um *dataset* de classificação de arritmias utilizando apenas uma árvore de decisão simples para determinar em qual área do RPS uma batida se encontra. Em [49] *features* são extraídas a partir da representação das séries temporais no RPS e utilizadas no reconhecimento de fala a partir de um *Hidden Markov Model* utilizando misturas de gaussianas.

Por fim, em [11] os autores extraem *features* a partir da série temporal representada pela distância de cada um dos pontos no RPS à origem. Ou seja, após mapear a série para um espaço multidimensional, um novo mapeamento é feito para o espaço unidimensional. Neste espaço, os autores extraem *features* utilizando transformadas e utilizam essas *features* na classificação das séries.

2.6 Conclusões

Conforme apresentado na seção 2.5, a representação das séries temporais no RPS, através da técnica *time-delay embedding*, já foi utilizada na literatura de classificação de séries temporais. Contudo, sua utilização está associada, na grande maioria dos

casos, à construção de modelos estatísticos a fim de representar cada uma das classes ou na extração de *features* posteriormente utilizadas em classificadores.

Conforme visto na seção 2.4, métodos baseados em distâncias entre séries temporais são bastante populares na literatura, obtendo resultados difíceis de serem superados. Apesar disso, não foram encontrados métodos que definem métricas de distâncias entre séries temporais no RPS, visando utilizar o maior poder descriminatório presente neste espaço.

O arcabouço apresentado nas seções 2.2 e 2.3 provê o ferramental necessário a fim de se calcular a distância, ou divergente, entre modelos estimados de forma não paramétrica. Essa abordagem pode ser aplicada a funções de densidade estimadas a partir das séries temporais no RPS, construindo, então, um classificador baseado nas distâncias entre essas funcionais.

Capítulo 3

Método Proposto

Neste capítulo, o método de classificação de séries temporais proposto no trabalho é apresentado da seguinte forma: na seção 3.1 as motivações, ideias gerais e raciocínio por trás da criação do método são discutidos; na seção 3.2 a fase de treinamento do método é detalhada e os algoritmos utilizados são explicados. Após, na seção 3.3 é explicada a fase de classificação do método, responsável por classificar uma série desconhecida e, finalmente, na seção 3.4 são apresentadas as conclusões do capítulo.

3.1 Proposta Geral

A ideia principal do método consiste na utilização de uma métrica de distância entre séries temporais a partir da hipótese de que cada série é uma amostra de uma população desconhecida. Utilizando o arcabouço apresentado no capítulo 2, as funções de densidade de probabilidade das distribuições dessas populações podem ser estimadas a partir dos dados das séries e, utilizando uma medida de divergente entre funções de densidade, suas distâncias podem ser aferidas. Por fim, essas distâncias serão utilizadas como critério de classificação pelo método proposto.

A motivação para esta modelagem reside na capacidade de modelar não-linearidades por meio do uso de estatísticas de maior ordem, obtida pela estimativa da função de densidade de probabilidade por meio de métodos como o KDE[33]. Dessa forma, é esperado que a métrica desenvolvida considere a forma completa da distribuição desconhecida, responsável por dar origem a cada uma das amostras. Com isso, espera-se obter um método robusto capaz de detectar padrões tanto locais quanto globais das séries, característica interessante para medidas de distâncias voltadas para esse tipo de dado[35].

Deseja-se definir $D(X, Y)$, uma função que representa a dissimilaridade entre amostras X e Y. Seja X e Y duas amostras aleatórias de distribuições com funções de densidade de probabilidade pdf_X e pdf_Y, pode-se utilizar uma medida de divergente $Dv(pdf_X, pdf_Y)$, apresentadas na seção 2.3, para calcular o divergente entre

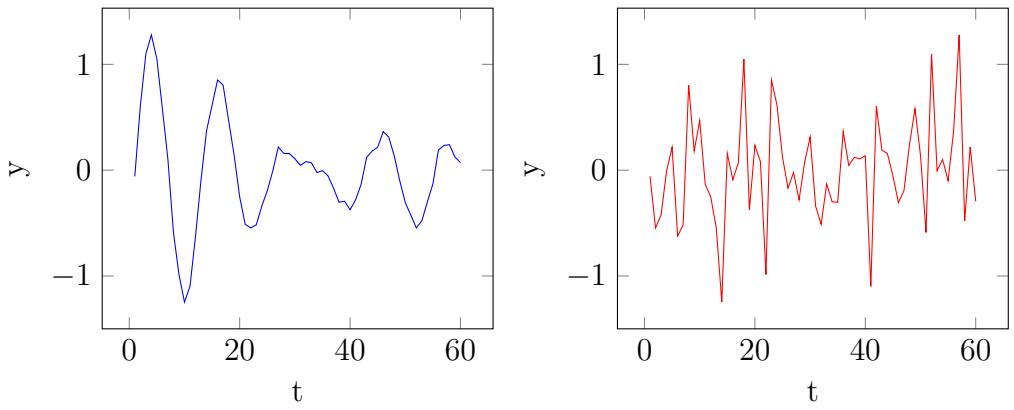


Figura 3.1: Séries temporais construídas como permutação de um mesmo conjunto de dados

as distribuições e assim utiliza-lo para representar a dissimilaridade entre as duas amostras.

Como, no caso geral, conhecemos apenas as amostras e não suas funções de densidade de probabilidade é necessário utilizar uma estimativa das mesmas. Para isso, propõe-se utilizar um dos métodos disponíveis na literatura, como o *Kernel Density Estimation*, apresentado na seção 2.2, para estimar as funções de densidade de probabilidade pdf_X e pdf_Y a partir das amostras X e Y. Desta forma, a partir das estimativas $\widehat{pdf_X}$ e $\widehat{pdf_Y}$, finalmente, pode-se definir $D(X,Y) = Dv(\widehat{pdf_X}, \widehat{pdf_Y})$.

Supondo que X e Y representam séries temporais, onde $X = x_1, x_2, \ldots x_n$ e $Y = y_1, y_2, \ldots y_m$, nossa proposta consiste em construir uma métrica de dissimilaridade $D(X,Y)$ baseada nas funções de densidade de probabilidade correspondentes.

O problema desta abordagem é que ao estimar a densidade de probabilidade a partir dos dados de X e Y, a ordenação temporal entre eles é perdida e, portanto, séries bem diferentes poderiam ter funções de densidade de probabilidade bem similares. Na figura 3.1, por exemplo, podemos visualizar duas séries temporais construídas a partir do mesmo conjunto de dados. Ao estimar as densidades de probabilidade utilizando estes dados são obtidas densidades idênticas, ilustradas na figura 3.2.

Nesse contexto, é necessário utilizar uma modelagem que seja capaz de levar em consideração as características temporais das séries. Conforme visto na seção 2.1, o RPS pode ser utilizado a fim de representar a topologia de um sistema dinâmico, como uma série temporal. Com isso, o método proposto utiliza a técnica de *time delay embedding*, vista na seção 2.1, a fim de representar o espaço de estados rps_X e rps_Y das séries.

Supondo que rps_X e rps_Y são amostras aleatórias geradas a partir de duas funções de densidade de probabilidade pdf_{rpsX} e pdf_{rpsY}, podemos utilizar o mesmo

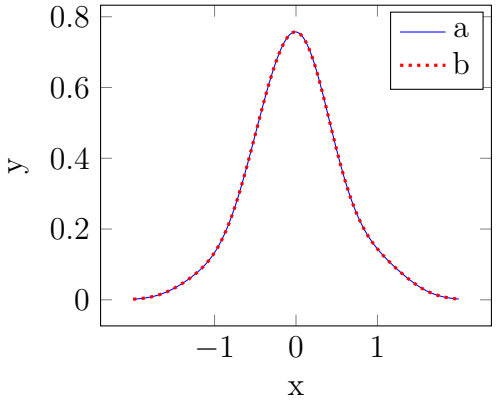


Figura 3.2: Estimativa das densidades das séries da figura 3.1

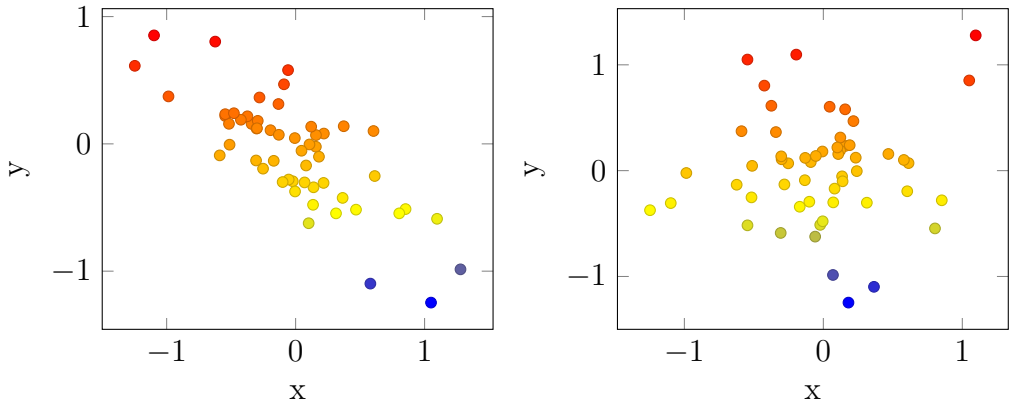


Figura 3.3: Representação das estimativas das densidades das séries da figura 3.1 no RPS

raciocínio aplicado anteriormente para obter a métrica de distância a ser utilizada entre as séries X e Y: $D(X,Y) = Dv(pdf_{rpsX}, pdf_{rpsY}) = Dv(\widehat{pdf_{rpsX}}, \widehat{pdf_{rpsY}})$, onde $\widehat{pdf_{rpsX}}$ e $\widehat{pdf_{rpsY}}$ são as funções de densidade de probabilidade estimadas a partir das amostras rps_X e rps_Y, respectivamente.

Na figura 3.3 um RPS de parâmetros arbitrários foi construído e os dados neste espaço foram utilizados para estimar as densidades das duas séries presentes na figura 3.1, onde os pontos mais vermelhos indicam maior densidade e pontos mais azuis apresentam menor densidade. Nesta representação já é possível visualizar uma diferença entre ambas as séries, ao contrário do obtido pela estimativa da densidade na figura 3.2, uma maior diferenciação é esperada por meio da seleção adequada dos parâmetros do RPS.

Qualquer classificador baseado nas distâncias entre as instâncias a serem classificadas e as instâncias de treinamento, como o k-vizinhos mais próximos, pode ser utilizado pelo método, a fim de classificar uma série temporal desconhecida utilizando a métrica definida.

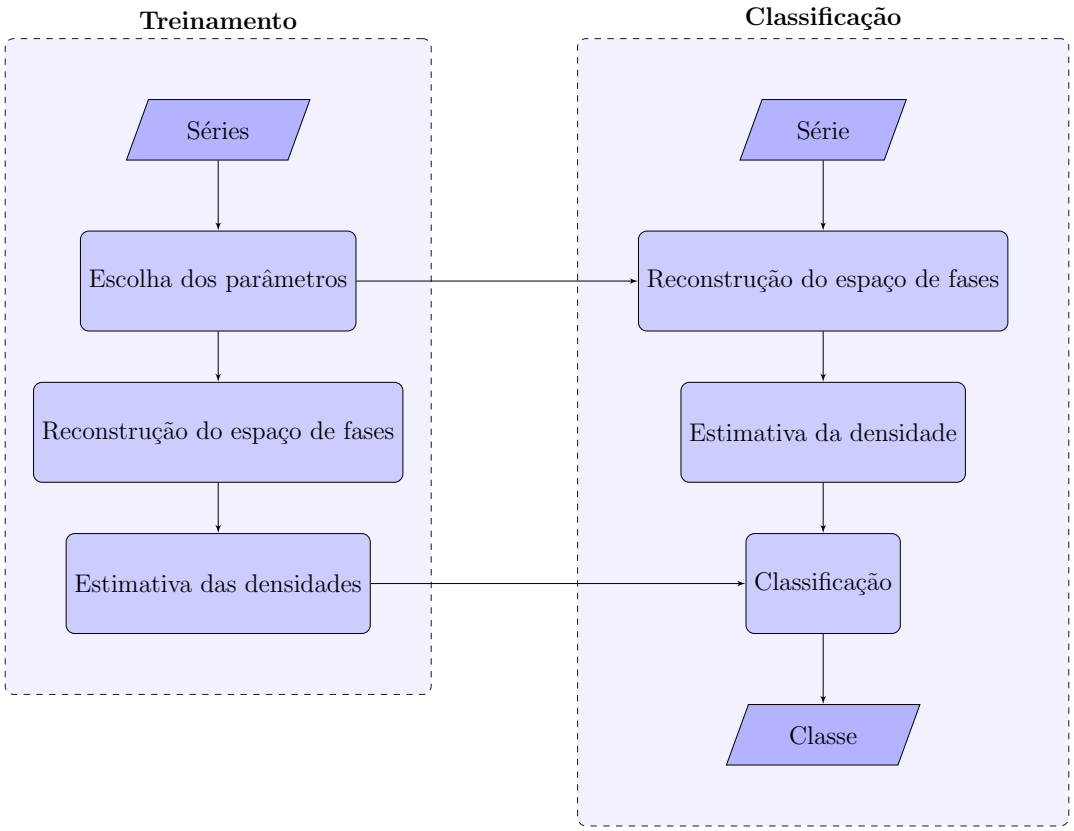


Figura 3.4: Etapas do método proposto

As duas fases presentes no método estão ilustradas no esquema da figura 3.4. A coluna **treinamento** exibe as etapas presentes na fase de treinamento do método:

1. Escolha dos parâmetros - Etapa na qual são estimados os parâmetros da reconstrução do espaço de fases (seção 3.2.1);

2. Reconstrução do espaço de fases - Nesta etapa os espaços de fases de cada uma das séries temporais do conjunto de treinamento são reconstruídos a partir da técnica *time-delay embedding* (seção 3.2.2);

3. Estimativa das densidades - Etapa responsável por estimar as funções de densidade de probabilidade de cada uma das séries temporais no espaço de fases (seção 3.2.3).

Da mesma forma, na coluna **classificação** estão ilustradas as etapas da fase de classificação do método:

1. Reconstrução do espaço de fase - O espaço de fases da série a ser classificada é reconstruído utilizando os mesmos parâmetros obtidos na fase de treinamento (seção 3.3.1);

2. Estimativa da densidade - Responsável por estimar as funções de densidade de probabilidade da série (seção 3.3.2);

3. Classificação - Etapa na qual ocorre a classificação da série temporal (seção 3.3.3).

Cada uma dessas fases (treinamento e classificação) e suas etapas são detalhadas nas próximas seções.

3.2 Treinamento

A fase de treinamento do método tem como objetivo a escolha dos parâmetros de reconstrução do espaço de fases e a obtenção da estimativa das funções de densidade de probabilidade de cada uma das séries presentes no conjunto de treinamento. Cada etapa desta fase é detalhada nas seções a seguir.

3.2.1 Escolha dos Parâmetros

Nesta etapa os algoritmos 1 e 2, os mesmos algoritmos utilizados por [10], são utilizados para a determinação dos parâmetros que serão utilizados para a construção do RPS das séries do conjunto de treinamento.

Algoritmo 1 Selecionar atraso

função SELECIONAATRASO($series$)
  para serie em $series$ **faça**
    $mif \leftarrow MIF(serie)$
    $atraso \leftarrow primeiroMin(mif)$
  fim para
retorna $moda(atraso)$
fim função

No algoritmo 1, a função MIF retorna um vetor contendo os valores da função de informação mútua, de acordo com a equação 2.7, para um intervalo de valores de atraso. Em seguida, a função $primeiroMin$ seleciona o primeiro mínimo deste vetor. Isso é feito para todas as séries do conjunto de treinamento e o valor escolhido é a moda dos atrasos selecionados para cada série.

O algoritmo 2 implementa o método dos falsos vizinhos, apresentado no capítulo 2, para a seleção da dimensão adequada para cada uma das séries. O percentual de falsos vizinhos é calculado variando a dimensão entre 1 e 50 e escolhe-se a primeira cujo percentual de falsos vizinhos é menor que um limiar (definido como 0.001 nesta implementação). A função *rps* constrói a representação da série no espaço multi-dimensional e a função *distancia* calcula a distância euclidiana entre dois pontos. Após selecionar a dimensão adequada para cada série, a dimensão a ser utilizada pelo método é dada pela fórmula $media(dim) + 2 * variancia(dim)$, garantindo que

Algoritmo 2 Selecionar dimensão

função SELECIONADIM($series, atraso$)
  para serie em $series$ **faça**
    $dim[serie] \leftarrow 50$
    para $d \leftarrow 1 \ldots 50$ **faça**
      $falsosVizinhos \leftarrow 0$
      $s_d \leftarrow rps(serie, d, atraso)$
      para ponto p_d em s_d **faça**
        $vizinho_d \leftarrow$ ponto em s_d mais próximo de p_d, excluindo p_d
        $dist_d \leftarrow distancia(p_d, vizinho_d)$
        $p_{d+1} \leftarrow p_d$ com mais uma dimensão
        $vizinho_{d+1} \leftarrow vizinho_d$ com mais uma dimensão
        $dist_{d+1} \leftarrow distancia(p_{d+1}, vizinho_{d+1})$
        se $dist_{d+1} - dist_d \geq limiar$ **então**
          $falsosVizinhos \leftarrow falsosVizinhos + 1$
        fim se
      fim para
      se percentual de falsosVizinhos ≤ 0.001 **então**
        $dim[serie] \leftarrow d$
        Sai do loop
      fim se
    fim para
  fim para
retorna $media(dim) + 2 * variancia(dim)$
fim função

esta seja grande o suficiente para uma representação adequada da maioria das séries no RPS[14].

3.2.2 Reconstrução dos espaços de fases

Neste passo, os parâmetros obtidos pelos algoritmos da etapa anterior são utilizados para a construção do RPS utilizando o método *time delay embedding*, apresentado na seção 2.1. Nesta representação, cada série passa a ser representada por uma matriz $t - ((d-1) * \tau) \times d$ onde τ é o atraso e d é a dimensão selecionados na etapa anterior.

Com esta transformação é esperado que seja possível reconstruir o espaço de estados original do sistema responsável da série temporal e, com isso, revelar padrões temporais que serão utilizados para a estimativa da função de densidade de probabilidade da série temporal na próxima etapa.

3.2.3 Estimativa das funções de densidade de probabilidade

Nesta etapa, as funções de densidade de probabilidade de cada uma das séries do conjunto de treinamento são estimadas por meio do *Kernel Density Estimation* (KDE) a partir da matriz que representa o RPS da série. A largura de banda h utilizada no KDE pode ser estimada por um dos métodos apresentados na seção 2.2.1 ou escolhida empiricamente.

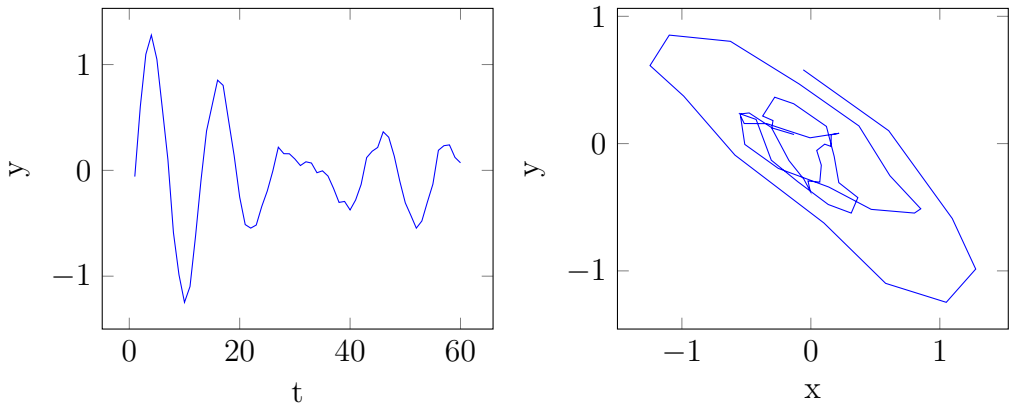
Cada linha da matriz é considerada como um ponto em um espaço d-dimensional e, assim, o KDE estima a densidade de probabilidade centrando uma gaussiana d-dimensional, com desvio padrão h, em cada um dos $t - ((d-1) * \tau)$ pontos neste espaço e somando suas densidades.

Outros métodos não paramétricos, como o *k-nearest neightbor density estimation*[24] e paramétricos, como Misturas de Gaussianas, poderiam ser utilizados para a estimativa da densidade de probabilidade. O KDE foi escolhido pois é o método não paramétrico mais popular na literatura[24] e, além disso, com uma quantidade suficiente de amostras, converge para qualquer função de densidade[50].

Após a conclusão da estimativa das funções de densidade de probabilidade de todas as séries, cada série possuirá um modelo (função de densidade de probabilidade) associado e o método está apto a classificar uma série temporal desconhecida.

3.3 Classificação

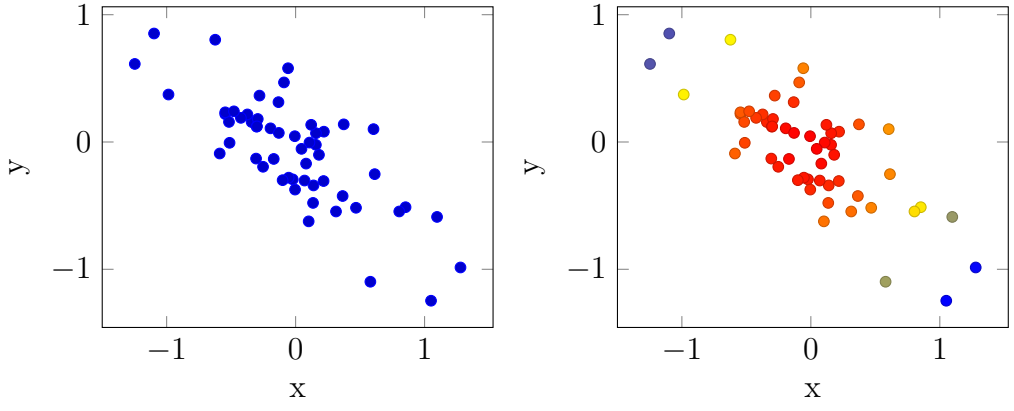
Esta fase é responsável pela classificação de novas séries temporais utilizando como referência as densidades e os parâmetros obtidos pelo método na etapa de treinamento. Cada etapa desta fase será detalhada nas próximas seções.



(a) Série original

(b) RPS $d = 2$, $\tau = 5$

Figura 3.5: Reconstrução do espaço de fases para classificação



(a) RPS $d = 2$, $\tau = 5$

(b) RPS com densidade estimada

Figura 3.6: Estimativa da função de densidade de probabilidade para classificação

3.3.1 Reconstrução do espaço de fases

Nesta etapa o método *time delay embedding* é novamente utilizado para a construção do RPS a partir da série não classificada. Os parâmetros d e τ a serem utilizados no método são aqueles obtidos durante a fase de treinamento.

A figura 3.5 ilustra o resultado desta etapa em um exemplo de série temporal supondo que a dimensão obtida pelo treinamento é $d = 2$ e o atraso é $\tau = 5$. É possível visualizar na figura 3.5b o *delay plot* do sistema, que representa a dinâmica do sistema ao longo do tempo, ou seja, o RPS.

3.3.2 Estimativa da função de densidade de probabilidade

Este passo é responsável por estimar a função de densidade de probabilidade do RPS obtido pela etapa anterior. Assim como no treinamento, o KDE é utilizado para obter a estimativa.

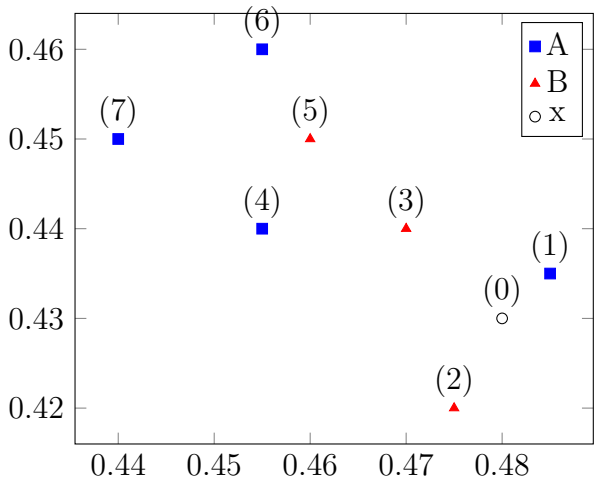


Figura 3.7: Exemplo de classificação com KNN

Conforme pode ser visto na figura 3.6a, a matriz que representa o RPS, obtida na etapa anterior, não é mais interpretada como a trajetória do sistema ao longo do tempo, como na figura 3.5b. Cada linha da matriz é analisada como um ponto em um espaço multidimensional e utilizada para a estimativa da densidade, que pode ser visualizada na figura 3.6b, onde os pontos mais vermelhos são os mais densos enquanto que os azuis são os menos densos.

3.3.3 Classificação

Neste estágio, qualquer classificador pode ser utilizado para classificar a série de acordo com sua distância das séries do conjunto de treinamento. Na implementação feita neste trabalho, foi utilizado um classificador k-vizinhos próximos e seu funcionamento está descrito no algoritmo 3.

Algoritmo 3 Classificação por K-vizinhos mais próximos

função KNN(x, Y, k)
  calcular a distância de x a cada um dos itens em Y
  $vizinhos \leftarrow k$ itens em Y mais próximos de x
retorna classe majoritária dos $vizinhos$
fim função

A função $KNN(x, Y, k)$ classifica a instância x de acordo com a classe majoritária das k instâncias mais próximas de x de acordo com a medida de distância D, onde as instâncias a serem utilizadas são as presentes no conjunto Y.

Por exemplo, na figura 3.7, deseja-se classificar a instância x entre as classes A ou B utilizando o método do KNN e a distância euclidiana. Os pontos estão numerados de acordo com a distância euclidiana a x. A classe escolhida para x depende do valor do parâmetro k, ou seja, da quantidade de vizinhos a ser considerada. Por exemplo,

com $k = 1$, x é classificado como A, com $k = 2$ ocorre um empate entre as duas classes, com $k = 3$ a classe escolhida é a B e assim por diante.

No método proposto, a distância D é o *Integrated Squared Error*, a instância x é a função de densidade de probabilidade estimada pelo KDE utilizando a representação RPS da série temporal e Y é o conjunto de funções de densidade de probabilidade estimadas pelo KDE em cada uma das séries temporais do conjunto de treinamento.

Além de ser simétrico, não negativo e apenas igualar a zero quando ambas as funções de densidade de probabilidade são iguais, o ISE foi escolhido como medida de divergente dada a existência de uma formula analítica fechada (equação 2.28), conforme apresentado na seção 2.3. Outros métodos onde tal formula pode ser obtida, como o divergente Cauchy-Schwarz[19], podem ser alternativas ao ISE.

3.4 Conclusões

Neste capítulo um novo método de classificação de séries temporais foi proposto, apresentado e as etapas envolvidas tanto no treinamento quanto em sua utilização foram detalhadas. O método supõe que o espaço de estados reconstruído (RPS) de cada uma das séries é uma amostra aleatória de uma distribuição desconhecida. A partir disso, um método de estimativa de densidade de probabilidade (KDE) é utilizado e uma medida de divergente entre funções de densidade (ISE) é utilizada para representar a distância entre as séries originais. Essas distâncias são utilizadas pelo classificador como critério de classificação de uma série desconhecida por meio do método de k-vizinhos próximos.

Capítulo 4

Resultados e Discussões

Com o objetivo de avaliar, tanto qualitativamente, quanto quantitativamente, o método proposto, diversos experimentos foram conduzidos sobre uma seleção variada de conjuntos de dados de classificação de séries temporais disponíveis na literatura. Esses experimentos e seus resultados obtidos estão apresentados neste capítulo, organizado da seguinte forma: na seção 4.1, são apresentados os conjuntos de dados utilizados nos experimentos; na seção 4.2, os parâmetros que compõem o método proposto são analisados experimentalmente. Em seguida, os resultados de classificação do método proposto são apresentados, discutidos e comparados com outros métodos da literatura na seção 4.3. Por fim, as conclusões dos experimentos são apresentadas na seção 4.4.

4.1 Conjuntos de dados

Conforme apresentado anteriormente, a classificação de séries temporais tem aplicações nas mais diferentes áreas. Dessa forma, faz-se necessário avaliar e analisar o comportamento do método proposto em dados de diferentes tipos de aplicações.

Para os experimentos conduzidos neste capítulo, foram selecionados 11 conjuntos de dados do *UCR Time Series Classification Archive*[41], um repositório de *datasets* de classificação de séries temporais. Dentre os conjuntos disponibilizados, estão incluídos dados das mais diversas áreas, como eletrocardiogramas, reconhecimento facial, reconhecimento de fala e outros.

A tabela 4.1 contém a lista dos conjuntos escolhidos e suas características principais: quantidade de classes, tamanho do conjunto de treinamento/teste e tamanho das séries.

Uma breve descrição de cada um dos conjuntos de dados é feita a seguir:

Synthetic Control[51] é um conjunto de dados gerado sinteticamente onde cada uma das classes corresponde a um padrão normalmente encontrando no domínio de controle de processos estatísticos.

Tabela 4.1: Conjuntos de dados selecionados para os experimentos

Nome	**Classes**	**Treinamento**	**Teste**	**Tamanho**
Synthetic Control	6	300	300	60
Gun-Point	2	50	150	150
CBF	3	30	900	128
Trace	4	100	100	275
Face (four)	4	24	88	350
Lightning-2	2	60	61	637
Lightning-7	7	70	73	319
ECG	2	100	100	96
Beef	5	30	30	470
Coffee	2	28	28	286
Olive Oil	4	30	30	570

Gun-Point[52] é do domínio de vigilância de vídeo. Todas as 200 instâncias foram criadas a partir de um ator e uma atriz e correspondem à posição, no eixo X, da mão direita do sujeito ao longo do tempo enquanto realizam um de dois tipos de movimentos diferentes.

Trace[53] é um *dataset* sintético que simula a instrumentação de falhas em uma usina nuclear, onde cada classe corresponde a diferentes tipos de eventos anômalos que ocorrem durante o funcionamento das usinas.

Face (four)[52] é do domínio de classificação de faces e consiste no ângulo do perímetro do rosto durante diferentes expressões faciais de quatro indivíduos diferentes.

CBF[54] é um *dataset* sintético onde cada uma das três classes é gerada pelas seguintes funções:

$$\begin{aligned} c(t) &= (6+\eta) * X_{[a,b]}(t) + \epsilon(t) \\ b(t) &= (6+\eta) * X_{[a,b]}(t) * (t-a)/(b-a) + \epsilon(t) \\ f(t) &= (6+\eta) * X_{[a,b]}(t) * (b-a)/(b-t) + \epsilon(t) \end{aligned} \qquad X_{[a,b]} = \begin{cases} 0 & t < a \\ 1 & a \leq t \leq b, \\ 0 & t > b \end{cases}$$

onde η e $\epsilon(t)$ são gerados por uma distribuição normal padrão, a é um inteiro sorteado no intervalo [16,32], uniforme, e $(b-a)$ é um inteiro no intervalo [32,96], também sorteado de maneira uniforme.

Lightning-2 e Ligthning-7[55] contém espectrogramas de relâmpagos obtidos através de instrumentos VHF de diferentes características físicas, agrupados em 7 classes diferentes (lightning-7) ou 2 classes diferentes (lightning-2).

ECG[56] contém as medidas da atividade elétrica do coração, obtidas a partir de eletrodos posicionados em diferentes partes do corpo. Cada série temporal corresponde a uma batida de coração, que foi classificada, por especialistas, como

normal ou anormal.

Beef, Coffee e OliveOil[57] são compostos por espectrogramas de alimentos e são utilizados para classificar tipos de comidas. Cada classe do *Beef* corresponde a um nível de contaminação, as classes do *Coffee* correspondem a variações do café arábico e robusta e, para o *OliveOil*, cada classe corresponde a um azeite extra virgem de diferentes países.

Todas as séries dos *datasets* já são normalizadas utilizando o *z-score*, ou seja, possuem média zero e desvio padrão 1. Cada conjunto é dividido em 2 subconjuntos: treino e teste; a fim de facilitar a comparação dos resultados obtidos por outros métodos publicados na literatura, essa divisão foi respeitada.

Os conjuntos da tabela 4.1 estão entre os mais utilizados na literatura, fato que determinou sua escolha para este trabalho. Além disso, estas são de diferentes áreas e possuem características bastante diversas: número de classes variando entre 2 e 7, tamanho de amostras de treinamento/teste variando de 24 à 300 e tamanho de séries variando de 60 à 637. Dessa forma, esperamos que o método proposto seja avaliado em uma diversidade satisfatória de problemas de classificação de séries temporais.

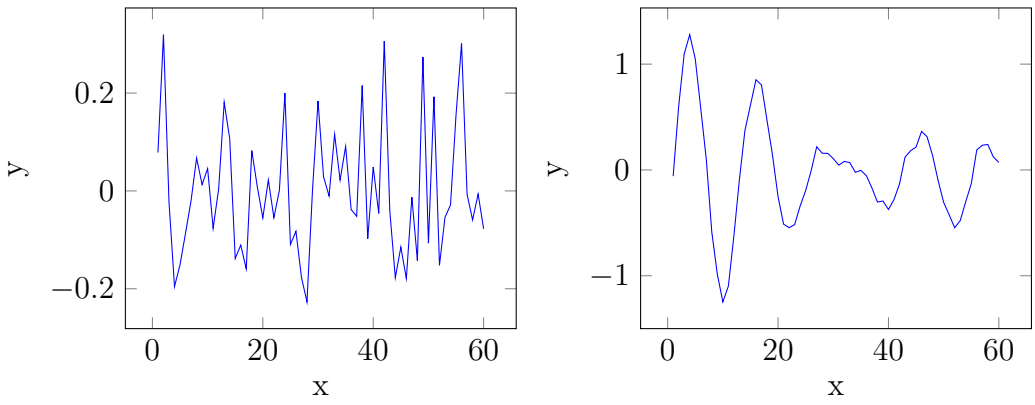
4.2 Análise do comportamento dos parâmetros

Nesta seção, o comportamento dos diversos parâmetros que compõem o método proposto é estudado em dois dos conjuntos de dados apresentados na seção 4.1: o *Synthetic Control* e o *ECG*.

O primeiro, *Synthetic Control* é um conjunto de dados gerado sinteticamente em [51] e possui 6 classes com características bastante distintas. As séries temporais que representam a média de cada uma dessas classes, no conjunto de treinamento, podem ser visualizadas na figura 4.1.

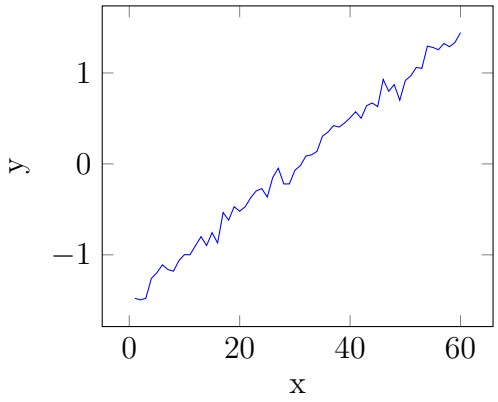
Podemos construir, para fins de visualização, a representação das classes do *Synthetic Control* no *Reproducing Phase Space*(RPS) com $d = 2$. A figura 4.2 consiste na representação de cada uma das instâncias presentes na figura 4.1 neste espaço, utilizando $\tau = 1$, onde as cores representam a densidade dos pontos(vermelho, mais denso, azul, menos denso), calculada pelo método *Kernel Density Estimation*(KDE) com largura de banda $h = 0.1$. Analisando as nuvens de pontos, na figura 4.2, fica claro que essa representação não conseguiu expor as características das séries originais: as classes C e D, por exemplo, foram representadas por nuvens de pontos extremamente parecidas visualmente, enquanto que as suas séries originais são claramente diferentes.

Ainda no espaço de duas dimensões, podemos visualizar o efeito obtido ao representar os pontos com um valor de τ um pouco maior. A figura 4.3 consiste nas mesmas classes, agora em um espaço construído com $\tau = 5$. Neste espaço, as classes

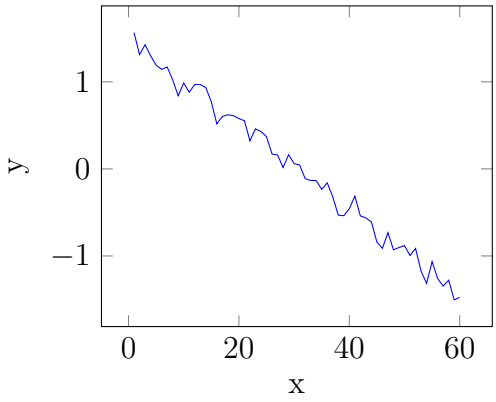


(a) Classe A

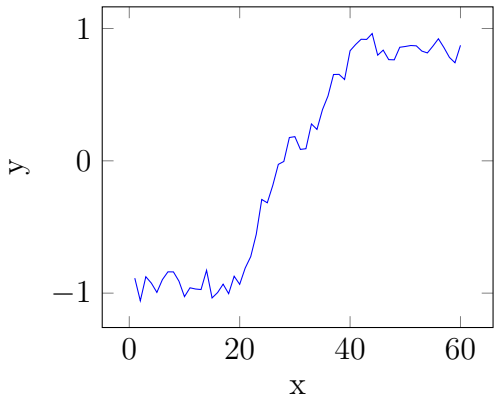
(b) Classe B



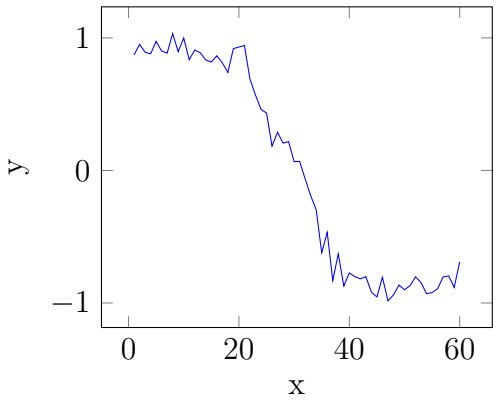
(c) Classe C



(d) Classe D

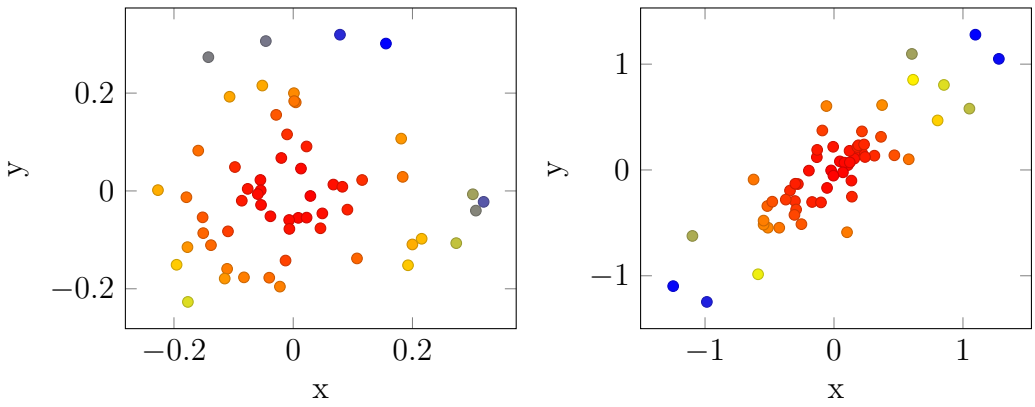


(e) Classe E



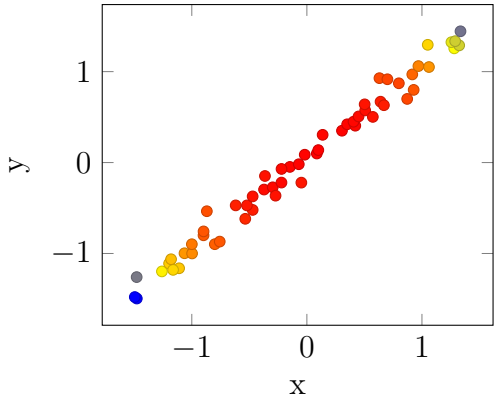
(f) Classe F

Figura 4.1: Classes do conjunto de dados *Synthetic Control*

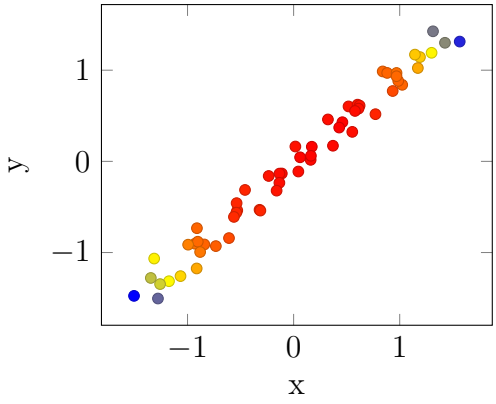


(a) Class A

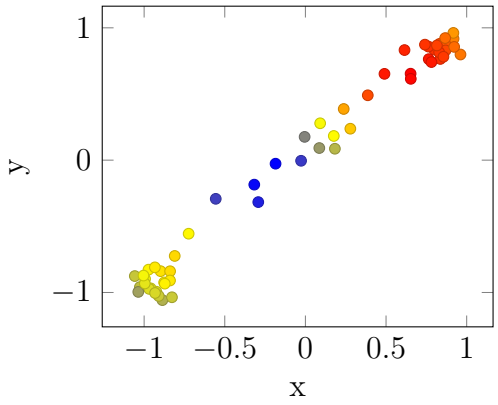
(b) Classe B



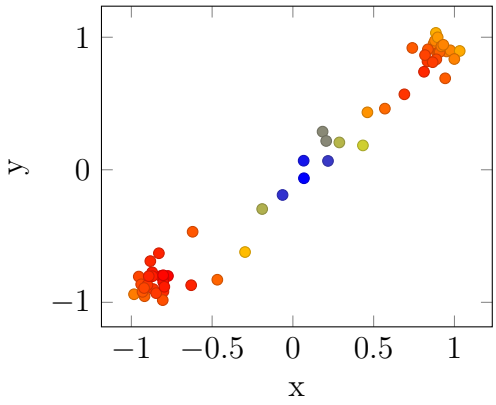
(c) Classe C



(d) Classe D



(e) Classe E



(f) Classe F

Figura 4.2: Representação das classes no RPS utilizando $d = 2$ e $\tau = 1$

A, B, D e E agora são visualmente diferentes de todas as outras, enquanto que as classes C e D permanecem extremamente parecidas.

A representação das séries C e D, no espaço de dimensão 2, é bastante similar, apesar de suas séries serem bastante diferentes na representação original. Isso ocorre, pois, ao analisar os pontos que compõem o RPS apenas pela posição dos mesmos, perdemos algumas informações sobre a ordem em que o espaço foi construído. Por exemplo, seja T_1 a série temporal crescente $T_1 = 1, 2, 3, \ldots, n$ e T_2 a série temporal decrescente $T_2 = n, n-1, n-2, \ldots, 1$. A matriz formada pelos vetores que compõe o RPS, com $d = 2$ e $\tau = 1$ de cada uma das séries é:

$$T_1 = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ \ldots & \ldots \\ n-1 & n \end{bmatrix}, \qquad T_2 = \begin{bmatrix} n & n-1 \\ \ldots & \ldots \\ 3 & 2 \\ 2 & 1 \end{bmatrix},$$

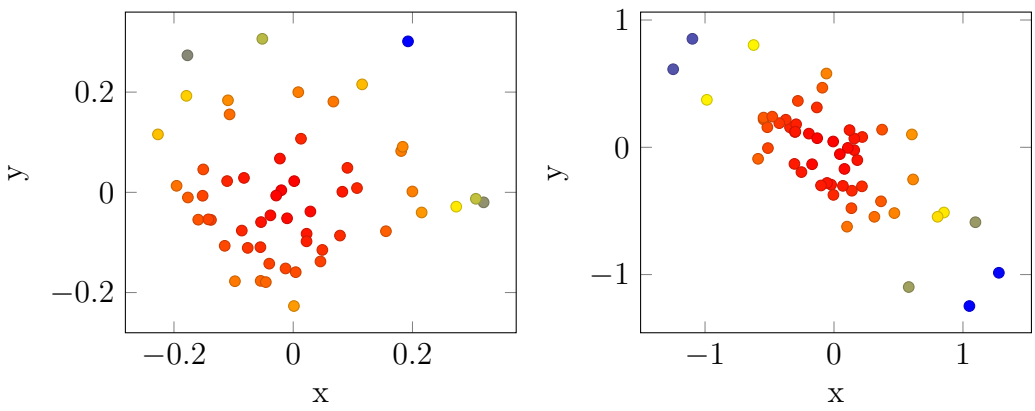
que resultam em nuvens de pontos bastante similares, apesar de serem originadas de séries bem diferentes. Esse é um caso bem particular, mas que pode ser observado nas classes C e D e, de forma menos intensa, nas séries E e F. Espera-se que com o aumento da dimensão d utilizada a representação seja capaz de capturar as informações necessárias a fim de diferenciar as duas classes.

As matrizes de confusão obtidas a partir da classificação das instâncias de treinamento utilizando o método proposto estão apresentadas na tabela 4.2. A classificação das instâncias de teste foi feita pelo método do vizinho mais próximo, onde a distância utilizada foi o *Integrated Squared Error*(ISE) e a largura de banda utilizada foi $h = 0.1$ (a mesma utilizada para representar as densidades dos pontos nos gráficos 4.2 e 4.3).

Na tabela 4.3 nota-se uma grande confusão entre as classes C e D e as classes E e F, como previsto pela análise da representação dessas séries no espaço obtido com parâmetros $d = 2$ e $t = 1$. Ao representar as classes no espaço com $t = 3$, percebe-se uma melhora significativa, através da diminuição da confusão entre as classes, representada na tabela 4.4. Apesar disso, ainda é possível identificar um grau de confusão entre as classes E e F. Já com $d = 8$, tabela 4.5 essa confusão foi praticamente resolvida.

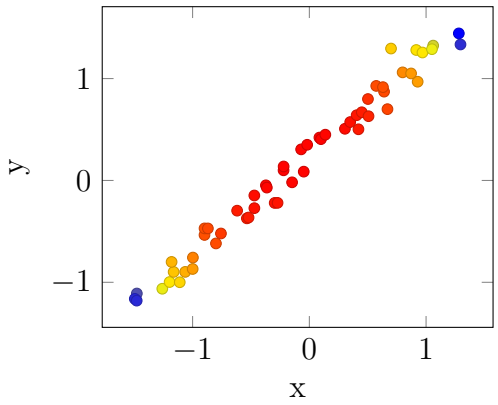
4.2.1 Análise da influência da dimensão no cálculo do ISE

É interessante entender o impacto causado pela alteração na dimensão do RPS em relação a métrica do ISE e também da acurácia de classificação. Não podemos fazer uso da análise visual apresentada na seção anterior, pois assim estaríamos limitados

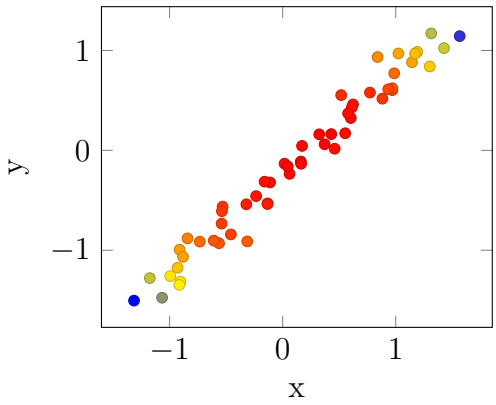


(a) Classe A

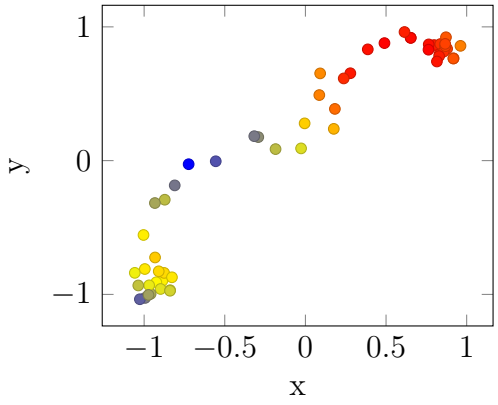
(b) Classe B



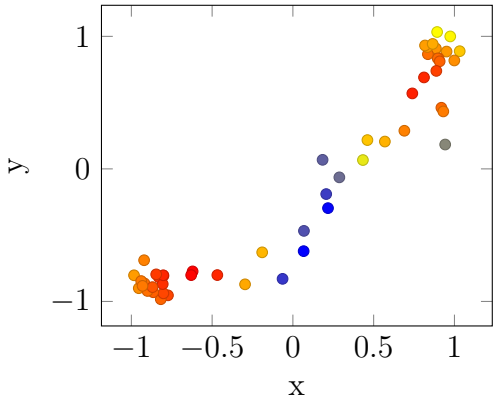
(c) Classe C



(d) Classe D



(e) Classe E



(f) Classe F

Figura 4.3: Representação das classes no RPS utilizando $d = 2$ e $\tau = 5$

Tabela 4.2: Matrizes de confusão *Synthetic Control*

Tabela 4.3: $d = 2$, $\tau = 1$

	A	**B**	**C**	**D**	**E**	**F**
A	49	0	0	1	0	0
B	0	38	4	5	2	1
C	0	4	21	23	1	1
D	0	12	13	22	1	2
E	0	2	2	5	24	17
F	0	4	4	4	18	20

Tabela 4.4: $d = 2$, $\tau = 3$

	A	**B**	**C**	**D**	**E**	**F**
A	46	4	0	0	0	0
B	0	50	0	0	0	0
C	3	0	45	0	2	0
D	1	0	0	45	0	4
E	0	0	2	0	38	10
F	3	0	1	3	4	39

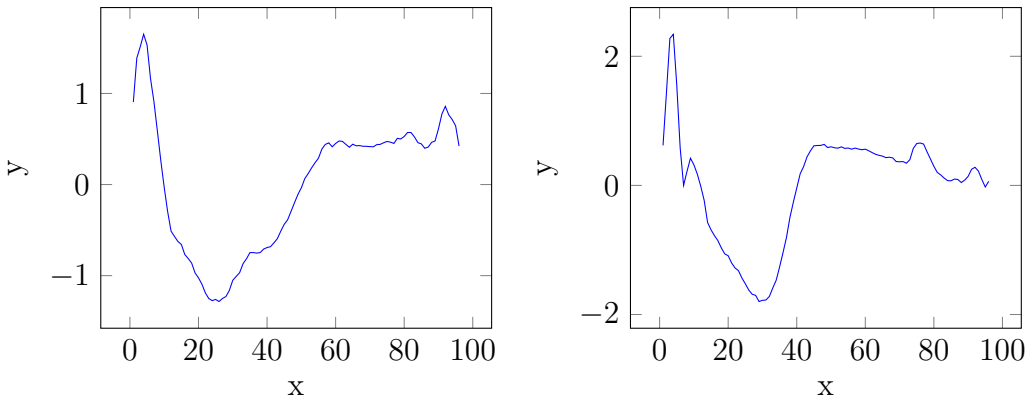
Tabela 4.5: $d = 8$, $\tau = 1$

	A	**B**	**C**	**D**	**E**	**F**
A	50	0	0	0	0	0
B	0	50	0	0	0	0
C	0	0	50	0	0	0
D	0	0	0	49	0	1
E	0	0	2	0	48	0
F	0	0	0	0	1	49

a $d \leq 3$. Porém, podemos analisar a variação do valor do ISE entre as classes a fim de entender a influência aplicada pela dimensão. Neste experimento utilizaremos o conjunto de dados ECG pois o mesmo possui apenas duas classes, facilitando a visualização dos resultados.

O conjunto de dados ECG representa um problema de classificação onde deseja-se classificar batidas de coração (representadas por segmentos obtidos de um exame de ECG) em batidas normais e batidas anômalas. A figura 4.4 exibe um exemplo de segmento de cada uma dessas classes.

A figura 4.5 está representado o efeito do aumento da dimensionalidade do RPS na distância ISE entre as classes do ECG, para valores fixos de $\tau = 1$ e $h = 0.1$. Os



(a) Classe 1 (b) Classe 2

Figura 4.4: Classes do conjunto de dados *ECG*

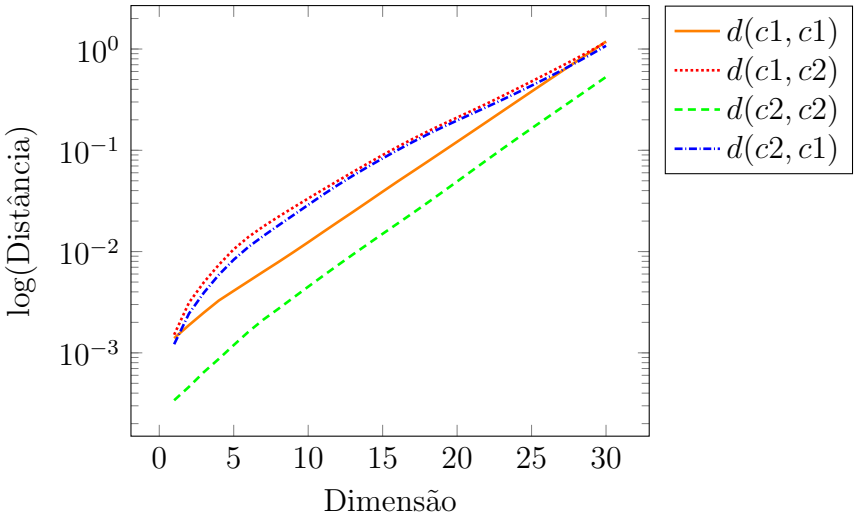


Figura 4.5: Efeito da escolha da dimensão no ISE entre classes do ECG

valores do ISE foram calculados entre a função de densidade de probabilidade das instâncias de treinamento e as de teste.

Para $d < 5$, as distâncias entre classes diferentes e a mesma classe são bem próximas, o que indica que as diferenças entre as classes não estão sendo bem representadas no espaço. Conforme aumenta-se a dimensão, todas as distâncias aumentam, porém, as distâncias entre classes diferentes aumentam com maior ênfase. Nesse ponto, com $10 < d < 20$, a diferença entre as distâncias de uma mesma classe e as distâncias de classes diferentes é maximizada. Assim, espera-se que com uma dimensão nessa ordem a acurácia da classificação seja também maximizada. Para dimensões $d > 25$, todos os ISE's passam a crescer de forma mais acelerada: isso ocorre pois estamos mantendo o valor de h fixo, onde $h = 0.1$ e conforme aumentamos a dimensão estamos entrando em espaços cada vez mais esparsos. Para essas dimensões, a distância entre a classe C1 de treinamento e C1 de teste, torna-se maior do que a distância entre classes diferentes, o que indica que a acurácia da classificação neste espaço tende a ser inferior.

A figura 4.6 exibe a variação da acurácia da classificação para $d's$ no intervalo utilizado na figura 4.5, confirmando os intervalos onde se esperava a melhor performance de classificação. A acurácia é calculada como a razão das instâncias corretamente classificadas sobre o total de instâncias, ou seja:

$$\text{acurácia} = \frac{\text{instâncias classificadas corretamente}}{\text{total de instâncias classificadas}} \tag{4.1}$$

Um importante efeito do aumento da dimensão, além do aumento da dimensionalidade por si só, é a diminuição no tamanho das amostras obtidas. Uma série temporal de tamanho t é representada em um RPS com dimensão d e atraso τ, por N vetores de dimensão d, onde $N = t - ((d-1) * \tau)$. Dessa forma, ao aumentarmos

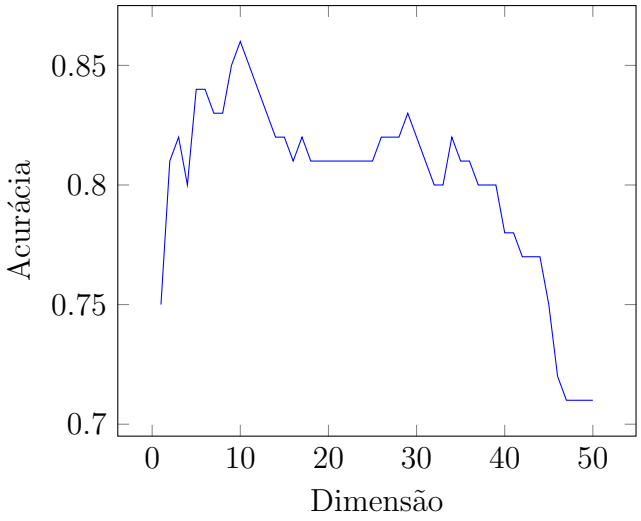


Figura 4.6: Efeito da escolha da dimensão na acurácia

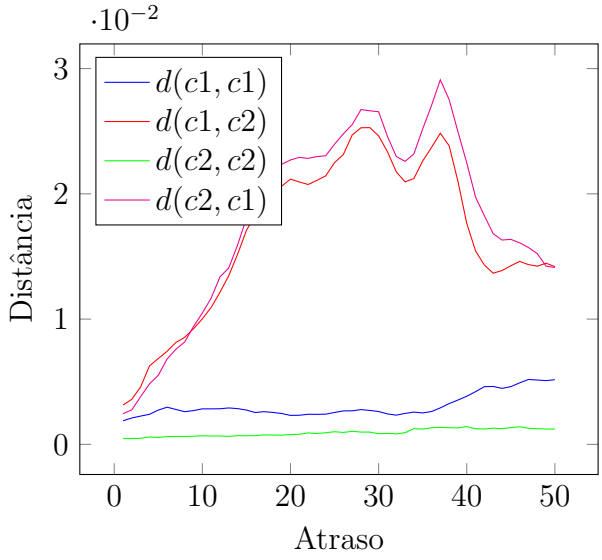


Figura 4.7: Efeito da escolha do atraso no ISE entre classes do ECG

a dimensão d, e/ou o atraso τ, aumentamos a dificuldade em se obter uma boa estimativa da densidade de probabilidade das amostras no RPS; efeito indesejável, que pode ocasionar em uma queda na performance da classificação.

4.2.2 Análise da influência do atraso no cálculo do ISE

Na figura 4.7 podemos visualizar o efeito causado pelo atraso τ na métrica ISE entre as classes. Assim como no caso da dimensão, a distância entre as classes diferentes é diretamente influenciada pela escolha do τ, porém, ao contrário do que ocorre com a dimensão, a distância entre as mesmas classes varia pouco. A diferença máxima entre as distâncias de uma mesma classe e as distâncias entre classes diferentes, neste *dataset*, ocorre com τ próximo de 30.

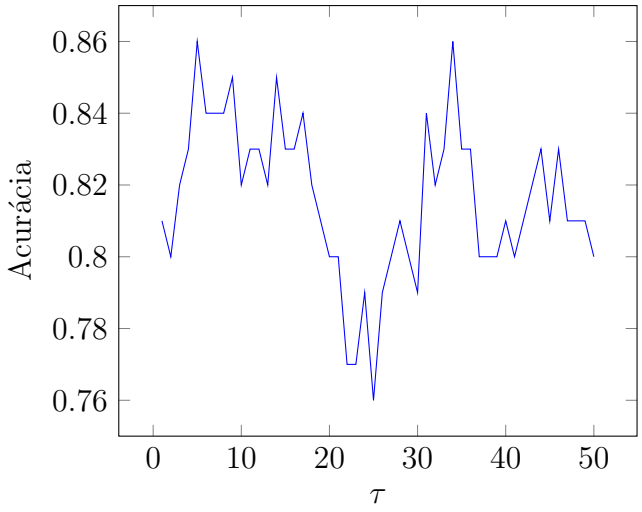


Figura 4.8: Efeito da escolha da τ na acurácia

Apesar disso, as distâncias estão duas ordens de grandeza menores do que as obtidas ao variarmos a dimensão. Ou seja, o atraso possui uma influência, ao menos no caso do ECG, muito menor na métrica do ISE. Isso fica evidente ao analisarmos a figura 4.8, onde podemos visualizar o comportamento da acurácia (calculada utilizando a equação 4.1) de classificação ao variarmos o τ.

Nota-se que não existe uma tendência, como ocorre no caso da dimensão e, além disso, a faixa de variação da acurácia é menor, algo esperado devido ao comportamento das distâncias.

4.3 Avaliação da Classificação

Esta seção apresenta os resultados obtidos pelo método proposto, nos onze *datatsets* escolhidos, e os compara a outros quatro métodos de classificação de séries temporais encontrados na literatura.

4.3.1 Métodos Avaliados

Além do método proposto, outros quatro métodos de classificação de séries temporais têm seus resultados comparados em todos os conjuntos de dados selecionados. Esta seção resume os métodos selecionados e suas principais características.

***Integrated Squared Error* no RPS**: É o método proposto neste trabalho, detalhado no capítulo 3. Consiste na classificação de séries temporais por meio do método de vizinho mais próximo, utilizando como métrica de distância o *Integrated Squared Error*(ISE) entre as densidades de probabilidade, estimadas a partir do *Kernel Density Estimation*(KDE), das séries temporais no espaço de fases reconstruído (RPS).

Misturas de Gaussianas no RPS: Constrói um modelo de misturas de gaussianas(GMM) utilizando o algoritmo de *Expectation Maximization*(EM), para representar a densidade das séries temporais no RPS e a classificação entre os sinais é feita através de um classificador Bayesiano[10].

Vizinho mais próximo com distância euclidiana: Neste método, uma série temporal de tamanho t é representada por um vetor no espaço t-dimensional. Para classificar uma série temporal, a distância euclidiana de seu vetor a todos os vetores de treinamento é calculada e lhe é atribuída a classe daquele mais próximo. Apesar de muito simples e sensível a desalinhamentos entre as séries[39], consegue resultados competitivos na grande maioria dos conjuntos testados, servindo como um bom *baseline* devido a sua simplicidade de implementação, interpretação e popularidade[34].

***Dynamic Time Warping* (DTW)**: O DTW é uma técnica que utiliza programação dinâmica a fim de alinhar duas séries temporais de forma a minimizar a distância euclidiana entre estas. Dessa forma, propõe-se lidar com uma possível falta de alinhamento entre as séries. A classificação, neste método, também se dá pelo vizinho mais próximo da série que deseja-se classificar.

DTW com *warping window*: Este método é uma extensão do DTW, onde aplica-se uma restrição adicional ao problema de programação dinâmica. Essa restrição evita buscar alinhamentos que estejam muito longe das séries originais e são utilizados tanto para reduzir o tempo de processamento quanto para evitar o *overfitting*[40].

4.3.2 Resultados e Discussão

Nesta seção são apresentados os resultados obtidos pelos métodos apresentados na seção 4.3.1, nos conjuntos de dados apresentados na seção 4.1. A acurácia de classificação foi calculada em cima do conjunto de teste, que, no caso dos *datasets* do UCR, já são pré-determinados, utilizando a equação 4.1.

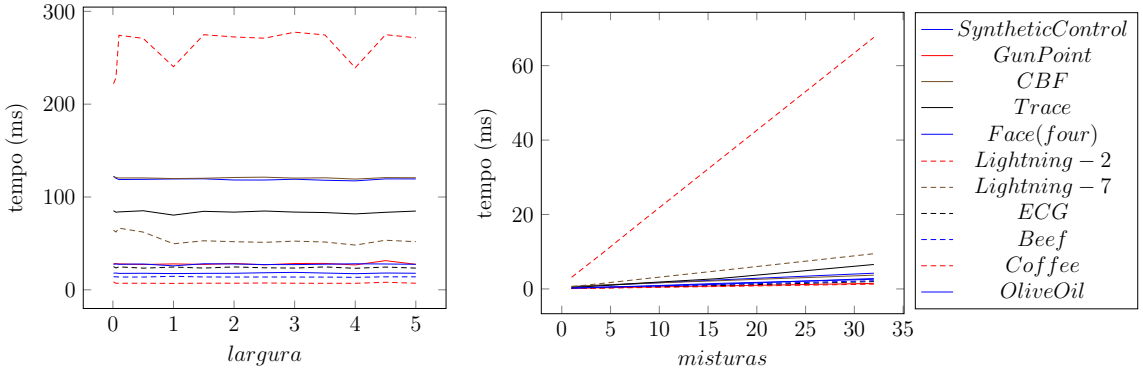
Comparação entre os métodos baseados em RPS

Nesta seção, compara-se o resultado dos dois métodos baseados em RPS: ISE RPS(proposto neste trabalho) e o GMM RPS.

Para cada *dataset* a dimensão foi determinada a partir do método dos falsos vizinhos e o atraso τ a partir do primeiro mínimo da função de informação mútua, métodos apresentados na seção 2.1.1. Com isso, para um determinado *dataset* ambos os parâmetros são iguais nos dois métodos de classificação. A largura de banda do ISE RPS foi variada de 0.1 à 3, enquanto que a quantidade de misturas de gaussiana do GMM RPS foi variada de 1 à 32. A acurácia de classificação nos conjuntos de treinamento pode ser visualizada na tabela 4.6, onde o melhor resultado de cada

Tabela 4.6: Acurácia dos métodos RPS utilizando d e τ escolhido por heurísticas

Nome	ISE RPS	GMM RPS
Synthetic Control	**0.9667**	0.9533
Gun-Point	0.9867	**1**
CBF	0.8722	**0.96**
Trace	0.9600	**1**
Face (four)	0.9432	**0.9545**
Lightning-2	0.7869	**0.8196**
Lightning-7	**0.7534**	0.7260
ECG	**0.8400**	0.83
Beef	0.5667	**0.633**
Coffee	0.9643	**1**
Olive Oil	**0.8667**	0.80



(a) ISE (b) GMM

Figura 4.9: Tempo de execução dos métodos baseados em RPS

conjunto de dados foi reportado.

Na maioria dos casos, ambos os métodos obtém resultados bastante próximos, com exceção dos *datasets* CBF e, onde o GMM RPS obtém uma acurácia 10% superior e do *Olive Oil* onde o ISE RPS obtém um resultado quase 7% superior. Como, neste experimento, a seleção dos parâmetros d e τ foi feita através dos referidos métodos, a diferença nos resultados entre ambos os métodos é devido, apenas, ás outras características dos modelos.

Na figura 4.9 pode-se visualizar o tempo de execução de ambos os métodos em cada um dos conjuntos testados e para cada um dos parâmetros de largura de banda h, do método proposto, e quantidade de misturas de gaussianas, do método baseado em GMM. A largura, como esperado, não influencia no tempo de execução do ISE e portanto, o mesmo mantem-se estável para todas as execuções de um mesmo conjunto, variando apenas entre os conjuntos, devido à quantidade de pontos de cada um deles. Já o método baseado em GMM, como esperado, tem seu tempo de execução ligado a quantidade de gaussianas escolhidas.

Tabela 4.7: Comparação com os métodos baseados em distâncias

Nome	ISE RPS	ISE RPS*	Euclidiana	DTW WW	DTW
Synthetic Control	0.9667	**0.9933**	0.88	0.983	**0.993**
Gun-Point	0.9867	**0.9933**	0.913	0.913	0.907
CBF	0.8722	**1**	0.852	0.996	0.997
Trace	0.9600	**1**	0.76	0.99	**1**
Face (four)	0.9432	**0.9545**	0.784	0.886	0.83
Lightning-2	0.7869	0.8360	0.754	**0.869**	**0.869**
Lightning-7	0.7534	**0.7671**	0.575	0.712	0.726
ECG	0.8400	**0.89**	0.88	0.88	0.77
Beef	0.5667	0.6333	**0.667**	**0.667**	0.633
Coffee	0.9643	**1**	**1**	**1**	**1**
Olive Oil	**0.8667**	**0.8666**	**0.867**	**0.867**	0.833

Como trata-se de um método baseado em vizinho mais próximo, o método proposto possui a desvantagem de precisar calcular a distância de cada uma das instâncias que deseja-se classificar a todas as instâncias de treinamento, para então determinar a que classe esta pertence. Essa característica tem direta influência no tempo de processamento, que é superior ao método GMM, que utiliza um classificador Bayesiano.

Comparação com métodos baseados em distâncias

Nesta seção, compara-se o método proposto com os outros métodos apresentados na seção 4.3.1, que são classificadores baseados em distância ao vizinho mais próximo. Na tabela 4.7 estão listados, os resultados obtidos pelo método proposto e os dos outros métodos. A coluna **ISE RPS** corresponde ao método proposto com os parâmetros d e τ sendo escolhidos a partir dos métodos dos falsos vizinhos e do mínimo da função de informação mútua, respectivamente. Portanto, são os mesmos resultados que os obtidos na tabela 4.6. Já a coluna **ISE RPS***, corresponde ao melhor resultado de classificação obtido pela combinação dos parâmetros onde: $d = \{2, 4, \ldots, 20\}$, $\tau = \{1, 3, 5\}$ e $h = \{0.1, 0.5, 1, 2\}$

Analisando a tabela 4.7, percebe-se que o método proposto, quando utiliza a melhor combinação de parâmetros, supera os métodos clássicos em quase todos os conjuntos testados, com exceção do *Lightning-2* e *Beef*. Enquanto que, quando as heurísticas de seleção de parâmetros são utilizadas, os método supera os métodos clássicos em 4 dos 11 conjuntos, porém, com resultados comparáveis.

Além disso, nota-se que, em todos os casos, é possível obter uma configuração do RPS com um resultado melhor do que aquele obtido pelos parâmetros selecionados através das heurísticas de falsos vizinhos e do mínimo da função de informação mútua. Isso ocorre pois as heurísticas não levam em consideração nenhuma in-

formação relacionada às classes das instâncias e portanto, não têm como objetivo aumentar a separabilidade das classes diferentes e, consequentemente, aumentar a acurácia da classificação.

4.4 Conclusões

Neste capítulo, a proposta da dissertação foi avaliada experimentalmente em diversos conjuntos de dados de classificação de séries temporais. O primeiro conjunto de experimentos expôs o comportamento dos parâmetros que compõem o método e destacou a importância da seleção destes para atingir separabilidade entre as classes. Dentre os parâmetros do RPS, a dimensão d exerce uma influência maior no cálculo da métrica e, consequentemente, na acurácia de classificação. Com isso, especial atenção deve ser exercida na seleção deste parâmetro.

No segundo experimento, o método proposto foi aplicado a uma gama grande de conjuntos de dados de classificação de séries temporais e sua acurácia de classificação foi comparada a outros métodos da literatura. Na comparação com os métodos clássicos, baseados em distâncias, duas implementações foram avaliadas. Na primeira, utilizando as heurísticas para seleção dos parâmetros d e τ, foram obtidos resultados comparáveis e, em alguns casos, superiores. Na segunda implementação a combinação de parâmetros com melhor acurácia foi utilizada e, neste caso, os resultados obtidos foram superiores em quase todos os conjuntos de dados analisados.

Na comparação com um método baseado na construção de modelos paramétricos a partir da representação no espaço de fases, o método proposto, que oferece uma abordagem não paramétrica, obteve resultados comparáveis em grande parte dos conjuntos avaliados. Contudo, o tempo computacional necessário para o cálculo das distâncias entre as séries e a estimativa das funções de densidade de probabilidade pelo KDE torna o método computacionalmente caro quando comparado com esta alternativa.

De modo geral, o método obteve resultados promissores e, portanto, oferece um alternativa aos métodos clássicos e aos baseados em modelos paramétricos, pois é construído a partir de uma fundamentação teórica distinta. Apesar disso, investigações precisam ser feitas a fim de melhorar a performance computacional do método e a seleção de parâmetros.

Capítulo 5

Conclusões

Nesta dissertação propusemos um novo método de classificação de séries temporais. A técnica proposta foi apresentada, discutida e seus resultados em diversos conjuntos de dados foram comparados a métodos clássicos da literatura e também a outro método que utiliza o mesmo tipo de representação derivada das séries temporais.

O método utiliza uma abordagem baseada na suposição de que o espaço de estados do sistema original em que a série temporal foi observada é uma amostra aleatória de uma distribuição desconhecida. Estas distribuições são estimadas de forma não paramétrica e uma medida de divergente entre estas é utilizada como critério de distância para um classificador baseado em vizinhos mais próximos.

Quando comparado com os métodos clássicos, também baseados em distâncias entre séries temporais e utilizando o mesmo classificador, o método obteve resultados promissores e competitivos. Ao selecionar os parâmetros utilizando uma abordagem empírica, ao invés de utilizar as heurísticas clássicas, o método foi capaz de obter resultados de classificação superiores em quase todos os conjuntos de dados testados.

Quando comparado a outro método cujo modelo do espaço de fase reconstruído é empregado, o método proposto apresenta resultados comparáveis na maioria dos conjuntos de dados. Contudo, o método se mostra computacionalmente caro, em decorrência da alta complexidade da estimava das densidades pelo KDE, maior do que a baseada em misturas de gaussianas conforme a quantidade de instâncias aumenta.

5.1 Trabalhos Futuros

Um fator determinante para um bom resultado do método proposto é a escolha dos parâmetros de reconstrução do espaço de fases. As heurísticas são capazes de estimar tais parâmetros, porém não são direcionadas para a melhora na classificação. Dessa forma, seria interessante desenvolver uma metodologia de escolha de parâmetros

supervisionada que utilizasse a informação das classes do conjunto de treinamento e escolha os parâmetros que aumentem a separabilidade das mesmas.

Com o objetivo de melhorar a performance computacional do método, técnicas para a remoção de instâncias redundantes do conjunto de treinamento poderiam ser investigadas e plugadas ao método proposto e, com isso, reduzir o número de distâncias a serem calculadas na etapa de classificação. Além disso, estruturas auxiliares, como as *K-d tree*, poderiam ser utilizadas a fim de aproximar o cálculo dos vizinhos mais próximos também provendo um ganho de velocidade do método.

Referências Bibliográficas

[1] KANTZ, H., SCHREIBER, T. *Nonlinear time series analysis*, v. 7. Cambridge university press, 2004.

[2] ESLING, P., AGON, C. "Time-series data mining", *ACM Computing Surveys (CSUR)*, v. 45, n. 1, pp. 12, 2012.

[3] DA S. LUZ, E. J., SCHWARTZ, W. R., CHÁVEZ, G. C., et al. "ECG-based heartbeat classification for arrhythmia detection: A survey", *Computer Methods and Programs in Biomedicine*, v. 127, pp. 144–164, 2016.

[4] AL-FAHOUM, A. S., QASAIMEH, A. M. "A practical reconstructed phase space approach for ECG arrhythmias classification", *Journal of medical engineering & technology*, v. 37, n. 7, pp. 401–408, 2013.

[5] LUZ, E. J. D. S., SCHWARTZ, W. R., CÁMARA-CHÁVEZ, G., et al. "ECG-based heartbeat classification for arrhythmia detection: A survey", *Computer methods and programs in biomedicine*, v. 127, pp. 144–164, 2016.

[6] NEJADGHOLI, I., MORADI, M. H., ABDOLALI, F. "Using phase space reconstruction for patient independent heartbeat classification in comparison with some benchmark methods", *Computers in Biology and Medicine*, v. 41, n. 6, pp. 411–419, 2011.

[7] TSELAS, N., PAPAPETROU, P. "Benchmarking dynamic time warping on nearest neighbor classification of electrocardiograms". In: *Proceedings of the 7th International Conference on PErvasive Technologies Related to Assistive Environments*, p. 4. ACM, 2014.

[8] SAKOE, H., CHIBA, S. "Dynamic programming algorithm optimization for spoken word recognition", *IEEE transactions on acoustics, speech, and signal processing*, v. 26, n. 1, pp. 43–49, 1978.

[9] ZHONG, S., GHOSH, J. "HMMs and coupled HMMs for multi-channel EEG classification". In: *Proceedings of the IEEE International Joint Conference on Neural Networks*, v. 2, pp. 1254–1159, 2002.

[10] POVINELLI, R. J., JOHNSON, M. T., LINDGREN, A. C., et al. "Time series classification using Gaussian mixture models of reconstructed phase spaces", *IEEE Transactions on Knowledge and Data Engineering*, v. 16, n. 6, pp. 779–783, 2004.

[11] SAYED, K. S., KHALAF, A. F., KADAH, Y. M. "Arrhythmia classification based on novel distance series transform of phase space trajectories". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 5195–5198. IEEE, 2015.

[12] ALBANO, A.-M., MEES, A., DE GUZMAN, G., et al. "Data requirements for reliable estimation of correlation dimensions". In: *Chaos in biological systems*, Springer, pp. 207–220, 1987.

[13] ALLIGOOD, K. T., SAUER, T. D., YORKE, J. A. "Chaos". In: *Chaos: An Introduction to Dynamical Systems*, Springer, pp. 105–147, 1997.

[14] TAKENS, F. "Detecting strange attractors in turbulence". In: *Dynamical systems and turbulence, Warwick 1980*, Springer, pp. 366–381, 1981.

[15] SAUER, T., YORKE, J. A., CASDAGLI, M. "Embedology", *Journal of statistical Physics*, v. 65, n. 3-4, pp. 579–616, 1991.

[16] POVINELLI, R. J., JOHNSON, M. T., LINDGREN, A. C., et al. "Statistical models of reconstructed phase spaces for signal classification", *IEEE Transactions on Signal processing*, v. 54, n. 6, pp. 2178–2186, 2006.

[17] CAO, L. "Practical method for determining the minimum embedding dimension of a scalar time series", *Physica D: Nonlinear Phenomena*, v. 110, n. 1, pp. 43–50, 1997.

[18] FRASER, A. M., SWINNEY, H. L. "Independent coordinates for strange attractors from mutual information", *Physical review A*, v. 33, n. 2, pp. 1134, 1986.

[19] JENSSEN, R., PRINCIPE, J. C., ERDOGMUS, D., et al. "The Cauchy–Schwarz divergence and Parzen windowing: Connections to graph theory and Mercer kernels", *Journal of the Franklin Institute*, v. 343, n. 6, pp. 614–629, 2006.

[20] PARZEN, E. "On estimation of a probability density function and mode", *The annals of mathematical statistics*, v. 33, n. 3, pp. 1065–1076, 1962.

[21] DUDA, R. O., HART, P. E., STORK, D. G. *Pattern classification*. John Wiley & Sons, 2012.

[22] HEIDENREICH, N.-B., SCHINDLER, A., SPERLICH, S. "Bandwidth selection for kernel density estimation: a review of fully automatic selectors", *AStA Advances in Statistical Analysis*, v. 97, n. 4, pp. 403–433, 2013.

[23] BOWMAN, A. W. "An alternative method of cross-validation for the smoothing of density estimates", *Biometrika*, v. 71, n. 2, pp. 353–360, 1984.

[24] SILVERMAN, B. W. *Density estimation for statistics and data analysis*, v. 26. CRC press, 1986.

[25] PRINCIPE, J. C. *Information theoretic learning: Renyi's entropy and kernel perspectives*. Springer Science & Business Media, 2010.

[26] CHERNOFF, H. "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations", *The Annals of Mathematical Statistics*, pp. 493–507, 1952.

[27] BHATTACHARYYA, A. "On a measure of divergence between two multinomial populations", *Sankhyā: the indian journal of statistics*, pp. 401–406, 1946.

[28] MATUSITA, K. "Decision rules, based on the distance, for problems of fit, two samples, and estimation", *The Annals of Mathematical Statistics*, pp. 631–640, 1955.

[29] COVER, T. M., THOMAS, J. A. "Elements of information theory". 1991.

[30] PATRICK, E., FISCHER, F. "Nonparametric feature selection", *IEEE Transactions on Information Theory*, v. 15, n. 5, pp. 577–584, 1969.

[31] LISSACK, T., FU, K.-S. "Error estimation in pattern recognition via L-distance between posterior density functions", *IEEE Transactions on Information Theory*, v. 22, n. 1, pp. 34–45, 1976.

[32] ADHIKARI, B. P., JOSHI, D. D. *Distance, discrimination et résumé exhaustif.* 1956.

[33] ZHOU, S. K., CHELLAPPA, R. "From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel hilbert space", *IEEE transactions on pattern analysis and machine intelligence*, v. 28, n. 6, pp. 917, 2006.

[34] KEOGH, E., KASETTY, S. "On the need for time series data mining benchmarks: a survey and empirical demonstration", *Data Mining and knowledge discovery*, v. 7, n. 4, pp. 349–371, 2003.

[35] XING, Z., PEI, J., KEOGH, E. "A brief survey on sequence classification", *ACM SIGKDD Explorations Newsletter*, v. 12, n. 1, pp. 40–48, 2010.

[36] MÖRCHEN, F. "Time series feature extraction for data mining using DWT and DFT". 2003.

[37] YE, L., KEOGH, E. "Time series shapelets: a new primitive for data mining". In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 947–956. ACM, 2009.

[38] LIAO, T. W. "Clustering of time series data—a survey", *Pattern recognition*, v. 38, n. 11, pp. 1857–1874, 2005.

[39] ANTUNES, C. M., OLIVEIRA, A. L. "Temporal data mining: An overview". In: *KDD workshop on temporal data mining*, v. 1, p. 13, 2001.

[40] XI, X., KEOGH, E., SHELTON, C., et al. "Fast time series classification using numerosity reduction". In: *Proceedings of the 23rd international conference on Machine learning*, pp. 1033–1040. ACM, 2006.

[41] CHEN, Y., KEOGH, E., HU, B., et al. "The UCR Time Series Classification Archive". July 2015. `www.cs.ucr.edu/~eamonn/time_series_data/`.

[42] SALVADOR, S., CHAN, P. "Toward accurate dynamic time warping in linear time and space", *Intelligent Data Analysis*, v. 11, n. 5, pp. 561–580, 2007.

[43] AL-NAYMAT, G., CHAWLA, S., TAHERI, J. "SparseDTW: a novel approach to speed up dynamic time warping". In: *Proceedings of the Eighth Australasian Data Mining Conference-Volume 101*, pp. 117–127. Australian Computer Society, Inc., 2009.

[44] KEOGH, E. J., PAZZANI, M. J. "Scaling up dynamic time warping for data-mining applications". In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 285–289. ACM, 2000.

[45] JEONG, Y.-S., JEONG, M. K., OMITAOMU, O. A. "Weighted dynamic time warping for time series classification", *Pattern Recognition*, v. 44, n. 9, pp. 2231–2240, 2011.

[46] DENG, K., MOORE, A. W., NECHYBA, M. C. "Learning to recognize time series: Combining arma models with memory-based learning". In: *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*, pp. 246–251. IEEE, 1997.

[47] MOON, T. K. "The expectation-maximization algorithm", *IEEE Signal processing magazine*, v. 13, n. 6, pp. 47–60, 1996.

[48] WAIBEL, A., HANAZAWA, T., HINTON, G., et al. "Phoneme recognition using time-delay neural networks", *IEEE transactions on acoustics, speech, and signal processing*, v. 37, n. 3, pp. 328–339, 1989.

[49] LINDGREN, A. C., JOHNSON, M. T., POVINELLI, R. J. "Speech recognition using reconstructed phase space features". In: *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, v. 1, pp. I–60. IEEE, 2003.

[50] SCOTT, D. W. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[51] PHAM, D., CHAN, A. "Control chart pattern recognition using a new type of self-organizing neural network", *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, v. 212, n. 2, pp. 115–127, 1998.

[52] RATANAMAHATANA, C. A., KEOGH, E. "Everything you know about dynamic time warping is wrong". In: *Third Workshop on Mining Temporal and Sequential Data*. Citeseer, 2004.

[53] ROVERSO, D. "Multivariate temporal classification by windowed wavelet decomposition and recurrent neural networks". In: *3rd ANS international topical meeting on nuclear plant instrumentation, control and human-machine interface*, v. 20. Citeseer, 2000.

[54] SAITO, N. "Local feature extraction and its applications using a library of bases", *Topics in Analysis and Its Applications: Selected Theses*, pp. 269–451, 2000.

[55] EADS, D. R., HILL, D., DAVIS, S., et al. "Genetic algorithms and support vector machines for time series classification". In: *International Symposium on Optical Science and Technology*, pp. 74–85. International Society for Optics and Photonics, 2002.

[56] OLSZEWSKI, R. T. *Generalized feature extraction for structural pattern recognition in time-series data*. Relatório técnico, DTIC Document, 2001.

[57] BAGNALL, A., DAVIS, L. M., HILLS, J., et al. "Transformation Based Ensembles for Time Series Classification." In: *SDM*, v. 12, pp. 307–318. SIAM, 2012.