Resumo: Esta dissertação explora a classificação de séries temporais utilizando análise de recorrência, discutindo a discretização de séries temporais e a extração de características para a construção de classificadores. O estudo é aplicado ao domínio musical, mas as técnicas podem ser adaptadas a outros contextos.

Classificação de Séries Temporais baseada em Análise de Recorrência e Extração de Características

Angelo Maggioni e Silva

SERVIÇO DE PÓS-GRADUAÇÃO DA FACOM-UFMS

Data de Depósito:

Assinatura:_____

Classificação de Séries Temporais baseada em Análise de Recorrência e Extração de Características[1]

Angelo Maggioni e Silva

Orientador: *Prof. Dr. Renato Porfirio Ishii*

UFMS - Campo Grande
Agosto/2016

[1]Trabalho Realizado com Auxílio do CNPq Proc. No: 131802/2014-3

À minha esposa Yara.
Ao meu orientador Renato Ishii.

Agradecimentos

A Deus, pelo refúgio, fortaleza e socorro em horas de angústia.
A minha família, Yara e Nícolas, pelo incentivo.
A meus pais João e Marlene, que com amor, sempre me aconselham e instruem.
A meu irmão Artur, meu amigo.
Aos meus colegas Jean, Lucas Rocha, Lucas de Souza, Hudson, Clóvis, Fábio, Cleison e Eduardo.
À equipe do PET-Sistemas.
Ao CNPq, CAPES e FUNDECT.

Resumo

A identificação de padrões em fluxos de dados contínuos tem despertado o interesse científico, seja na detecção de falhas em sistemas, identificação de operações fraudulentas em transações bancárias, propagação de doenças ou ainda na preservação do meio ambiente. A categorização destes dados, concomitante com a ampliação do sensoriamento e monitoramento de diversos outros domínios, motiva a busca por soluções práticas e eficientes que auxiliem na busca por padrões recorrentes. A extração de conhecimento dos dados, quando dependentes do tempo, exige um tratamento especial e a mineração dos dados apresenta-se como uma atividade valiosa. Neste trabalho, é proposta uma abordagem chamada DSP-Class para classificação de séries temporais utilizando Descritores de Textura aplicados em Gráficos de Recorrência (RP). São utilizados 14 conjuntos de dados reais relacionados a vocalizações de aves, identificação de insetos, categorização de reações químicas, dentre outros. O objetivo desta pesquisa é verificar a utilização das características texturais de RPs em algoritmos de aprendizagem, tais como *Support Vector Machine* (SVM) e C5.0, aplicando a Decomposição de Modo Empírico (EMD) na classificação de séries temporais. Também é analisada a influência estocástica-determinística presentes nos fluxos. Verifica-se desempenho ruim do algoritmo 1NN, considerado estado-da-arte, em séries predominantemente estocásticas ou determinísticas e desempenho 67.66% superior da abordagem DSP-Class, uma vez que as características texturais distinguem classes de séries temporais mais satisfatoriamente que a busca por similaridade utilizada no algoritmo 1NN nos dados analisados. Verifica-se inclusive, resultados 18,67% superiores àqueles obtidos por pesquisas semelhantes que utilizam outras características presentes em séries temporais.

Palavras-chave: Gráficos de Recorrência; Extração de Características; Séries Temporais; Classificação; Decomposição de Modo Empírico.

Abstract

Identify patterns in continuous data streams has attracted scientific interest for detecting system failures, identify fraudulent transactions in banks, the spread of diseases and also in the preservation of the environment. The increased volume of data produced in the last decade, concomitant with the number of sensors motivates the search for practical and efficient solutions that help data categorization. The extraction of knowledge about them, when time-dependent, requires special treatment and mining data becomes a valuable activity. In this paper, we propose an approach called DSP-Class for time series classification using texture descriptors applied Recurrence Charts (RP). We adopt 14 real datasets related to sound recognition, signals processing, chemical reactions and another produced by Big Data which demand high processing capacity. The purpose of this research is to verify the use of textural features in machine learning algorithms, such as SVM and C 5.0, applying Decomposition Empirical Mode (EMD) in time series classification. It is also analysed the stochastic-deterministic influence present in data streams. It presented poor results of 1-Nearest Neighbour algorithm when the data stream is mostly deterministic or stochastic. Our approach outperforms a traditional preprocessing approach applied on an audio stream using coefficients as features in around 18,67% of average accuracy and in around 67,66% the state-of-art algorithm that uses distance as measure.

Keywords: Recurrence plot, feature extraction, texture analysis, time series, classification, empiric mode decomposition.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

AM Aprendizado de Máquina

DCT Transformada Discreta de Cosseno

DTW *Dynamic Time Warping*

FN Falso Negativo

FP Falso Positivo

GLCM *Grey Level Co-ocorrence Matrix*

IA Inteligência Artificial

LBP *L*ocal Binary Pattern

RP *Recurrence Plot*

SAX *Symbolic Aggregate approXimation*

SFTA *Segmentation-based Fractal Texture Analysis*

ST Série Temporal

SVM *Support Vector Machine*

TFP Taxa de Falso Positivo

TVP Taxa de Verdadeiro Positivo

VN Verdadeiro Negativo

VP Verdadeiro Positivo

Lista de Algoritmos

CAPÍTULO

1

Introdução

A procura por técnicas eficientes para manipular e sumarizar os dados tem sido alvo de pesquisas recentes para, por exemplo identificar o momento ideal de realizar uma operação na bolsa de valores ou na agricultura para classificar excelentes épocas de plantio. Ao categorizá-los, o conhecimento acerca deles pode ser utilizado de maneira que ofereça vantagem competitiva.

A classificação e o agrupamento de dados também pode ser vista no reconhecimento de padrões. Pulseiras móveis sem fio permitem o registro de batimentos cardíacos, caminhos percorridos e até horas dormidas (Gyselinckx et al., 2005). A caracterização e extração de informações acerca destes dados pode beneficiar a saúde do ser humano por exemplo ao identificar um desvio de padrão.

Os dados que apresentam comportamentos recorrentes utilizados nesta pesquisa apresentam uma dependência temporal, pois se busca classificar informações a partir de comportamentos periódicos. Considere eletrocardiogramas de duas pessoas, uma saudável e outra enferma do coração. Cada eletrocardiograma é representado por um fluxo de dados contendo a intensidade dos pulsos elétricos em intervalos de tempo pré-determinados. A única diferença entre os fluxos está apenas na mudança em suas amplitudes, no qual o diagnóstico de um especialista pode ser auxiliado por um programa de computador que os analise e os classifique, por exemplo, em um eletrocardiograma de uma pessoa saudável ou de outra enferma do coração.

Ao classificar um fluxo de dados, grupos podem ser formados e rótulos atribuídos a eles. Autores como Silva (2014) dedicam-se à construção de programas de computador para classificar fluxos de dados. Em seu trabalho os fluxos são interpretados como séries temporais oriundas do bater de asas de

insetos. Tal pesquisa auxilia na identificação e classificação de espécies de insetos vetores de doenças em cidades ou aqueles que parasitam culturas e reduzem a produção agrícola.

A classificação de séries temporais permeia muitos domínios e soluções interessantes têm sido empregadas para caracterizá-las e classificá-las. Pereira (2013) utiliza técnicas de modelagem[1] de séries temporais como apoio à escolha da carteira de investimentos mais rentável na bolsa de valores. Silva et al. (2013b), inspirados pela compressão de vídeos e imagens[2], propuseram um algoritmo para classificar a similaridade entre séries temporais baseado nos Gráficos de Recorrência (do inglês: *Recurrence Plot* - RP) formados por elas.

A utilização de RPs formados por séries temporais é outro ponto chave desta pesquisa. Como cada RP reflete o comportamento da série temporal (Marwan et al., 2007), analisá-los para verificar suas similaridades extrapola a área compreendida por "processamento de sinais" e invade a área de "processamento de imagens".

A união destas duas áreas oferece um excelente ganho de informação para realizar a tarefa de classificação de séries temporais. Ao analisar um fluxo de dados da perspectiva de sinal, regressões podem ser realizadas e comportamentos preditos. No entanto, sob a ótica de uma imagem, singularidades podem ser suavizadas e agrupamentos feitos. Combinar técnicas destes dois domínios corrobora na eficiência da classificação e identificação de padrões nos fluxos de dados, uma vez que cada área extrai características distintas. Tal união é verificada no trabalho de Souza et al. (2014). O autor realiza uma investigação quanto a utilização de RPs para classificar séries temporais coletadas em vários domínios. Utilizando 38 datasets o pesquisador alcançou 78,71% de acurácia média na classificação extraindo características texturais dos RPs formados pelas séries. Apesar de propor uma metodologia "genérica", o autor consegue superar técnicas tradicionais de classificação de séries temporais.

Também abordando fluxos de dados como imagens, Rios (2013) provê uma metodologia para apoiar classificadores através da modelagem individual de componentes da série. O autor propõe a utilização da Decomposição de Modo Empírico e da Decomposição em Componentes Estocásticos/Determinísticos aplicados em séries temporais para auxiliar na classificação de padrões meteorológicos (chuvas e radiação ultravioleta) em escala global.

Para séries temporais que apresentam muitas observações, demasiadamente extensas, realizar a classificação delas é uma tarefa custosa, pois exige

[1] Exemplo: Modelos Auto-Regressivos (AR) e Auto-Regressivos com Médias Móveis (ARIMA).

[2] Técnica normalmente aplicada para vídeos MPEG e imagens JPEG (Pennebaker e Mitchell, 1993; Silva et al., 2013b), na qual compara-se a similaridade entre quadros dos vídeos e os pixels das imagens para reduzir o espaço ocupado em disco.

comparações. Quando elas possuem comprimentos significativos, o tempo gasto de processamento é proporcional ao seu comprimento. Logo, eliminar as observações desnecessárias reduz o custo e o tempo de computação despendido. Neste sentido, foram elencadas as seguintes perguntas:

- Descrever uma série temporal sob qual perspectiva (sinal/imagem) fornece maior acurácia na classificação?
- A utilização de filtros nas séries aumenta a taxa de acertos?
- Qual volume de dados é suficiente para classificar uma série temporal?

1.1 Hipótese

Conforme pesquisas mencionadas anteriormente, a hipótese principal deste trabalho é:

A extração das características de gráficos de recorrência através de descritores de textura, apoiada pela decomposição de modo empírico, contribui para o aumento da acurácia de classificadores de séries temporais.

1.2 Objetivo Geral

Define-se o seguinte objetivo geral desta pesquisa: *Realizar a extração de características de gráficos de recorrência combinada com a técnica de decomposição em modo empírico sob séries temporais a fim de construir um modelo de inferência por meio de algoritmos de aprendizado de máquina com o objetivo de aumentar a acurácia de classificadores.*

1.3 Objetivos Específicos

No sentido de alcançar o objetivo geral busca-se coletar atributos (*features*) de séries temporais a partir de gráficos de recorrência por meio dos descritores de textura: GLCM (*Grey Level Co-ocorrence Matrix*), SFTA (*Segmentation-based Fractal Texture Analysis*), LBP (*Local Binary Pattern*), Filtro de Gabor e DCT2-D (*Transformada Discreta de Cosseno Bi-dimensional*) (Apatean et al., 2008; Costa et al., 2012).

A influência do acréscimo de descritores de textura também é alvo de investigação. Inclusive, é averiguado quais *features* oferecidas pelos descritores proporcionam maior poder discriminativo de classes, para construção de um classificador de séries temporais. Também é realizada uma avaliação do

desempenho classificatório dos algoritmos de AM em séries temporais, utilizando a Decomposição de Modo Empírico (EMD) (Rios, 2013) e a Decomposição em Componentes Estocásticos/Determinísticos (Rios, 2013) como apoio, trabalhando como um filtro para os descritores de textura, antes da etapa de Extração de Características. Em seguida, os resultados são comparados ao estado da arte e com trabalhos relacionados.

1.4 Principais Contribuições

Este trabalho busca aplicar técnicas da área de processamento de sinais e de imagens para minerar dados e construir classificadores de séries temporais utilizando práticas da área de AM. Unir estes campos demonstra resultados promissores para identificação de propriedades do sinal que influenciam na classificação de uma série temporal.

Utilizando dados de diversos domínios, como aqueles provenientes do canto de pássaros ou da concentração de cloro presente na água, todos interpretados como séries temporais, a abordagem proposta não se detém a uma esfera exígua pois permeia-se em setores de classificação de dados, análise de sinais e imagens.

É feita a reconstrução de uma série temporal no espaço fase, como descrito no Capítulo 2 , nesta etapa, experimentos iniciais verificaram a influência estocásticas/determinística presente nas séries e seus resultados demonstram uma influência destes componentes na elaboração de um classificador de séries, quando utiliza-se *features* obtidas do RP. Os resultados obtidos foram publicados no 31^{o} *Simpósio Internacional de Computação Aplicada - 2016*.

Neste trabalho publicado, a abordagem proposta também confronta a pesquisa de Pereira (2013) para classificação de séries temporais oriundas de músicas, na qual se reduz a quantidade de observações necessárias para classificar uma série temporal em 4.000 observações, sendo necessárias apenas 1.000 observações, ao invés de 5.000 eventos presentes na série.

Ainda como contribuição desta pesquisa, outro artigo está em desenvolvimento a partir dos resultados alcançados. Verificou-se que ao analisar cada componente presente na série temporal e variar o nível de determinismo mínimo para sua separação a acurácia do classificador é variada em até $7,31\%$ quando utilizada *features* determinísticas e em $11,04\%$ quando utilizada as estocásticas.

Utilizando técnicas de seleção das *features* mais representativas, ranqueou-se aquelas que melhor descrevem os RPs. Verificou-se desnecessário utilizar todas *features* para construção do classificador de vocalizações, desta forma, reduziu-se a quantidade de *features* em $22,34\%$ com a técnica de

Ganho de Informação (IG) e em $98,93\%$ com a técnica RFE (*Recursive Feature Elimination*), tendo um aumento de $7,74\%$ na acurácia do classificador quando utiliza-se a técnica IG.

Este trabalho está organizado da seguinte maneira: O Capítulo 2 apresenta as definições formais necessárias à compreensão desta pesquisa e trabalhos relacionados. No Capítulo 3, a abordagem de pré-processamento de séries temporais proposta é apresentada. No Capítulo 4 resultados dos experimentos são verificados e por fim, no Capítulo 5, são apresentadas as conclusões.

CAPÍTULO

2

Definições e Trabalhos Relacionados

Séries temporais são aplicadas nas mais diversas áreas, como na agrometeorologia por auxiliar especialistas na detecção da melhor época de plantio, na engenharia para revelar falhas estruturais, na biologia propiciando a identificação de vetores patológicos e na computação, para extração de conhecimento em problemas *Big Data* (Chino, 2014; Karvelis et al., 2013; Silva, 2014).

Neste capítulo, serão apresentadas definições quanto a Séries Temporais na Seção 2.1, conceitos relacionadas a Gráficos de Recorrência na Seção 2.2, métodos de extração de features na Seção 2.3, a técnica de Decomposição de Modo Empírico na Seção 2.4 e métricas de aferição dos resultados e conceitos de Aprendizagem de Máquina na Seção 2.5. Trabalhos relacionados ao tema central desta dissertação de mestrado são apresentados na Seção 2.6.

2.1 Séries Temporais

Uma série temporal $T = \{t_1, t_2, t_3, ..., t_n\}$ pode ser representada como um vetor de n observações coletadas ao longo do tempo. Podem ser exemplos de séries temporais: valores mensais de temperatura de uma cidade, índices da bolsa de valores ou a precipitação atmosférica diária de uma determinada região coletada em um ano.

Não caracterizam séries temporais: *"O número de acidentes de trânsito em São Paulo durante o mês de Janeiro"* ou *"As medições de temperatura realizadas em bairros diferentes da cidade de Campo Grande no mês de Dezembro"*. Ambos exemplos demonstram uma, e apenas uma, contagem de eventos, negligencia-se o fator temporal ao considerar somente um único instante de tempo, ao invés de um intervalo, por exemplo: *"número de acidentes diá-*

rios durante o mês de Janeiro" ou *"medições diárias de temperatura no mês de Dezembro"*.

Quando as observações são realizadas em intervalos de tempo equidistantes, ela é dita **discreta**, como demonstrado na Figura 2.1 (B), na qual é utilizada uma série temporal hipotética de amplitude $[-1:1]$ e comprimento $[0:20]$. Quando as observações são realizadas continuamente no tempo, a série é dita **contínua**, Figura 2.1 (A). Um exemplo de série temporal contínua pode ser *"o registro de marés no porto de Santos"*.

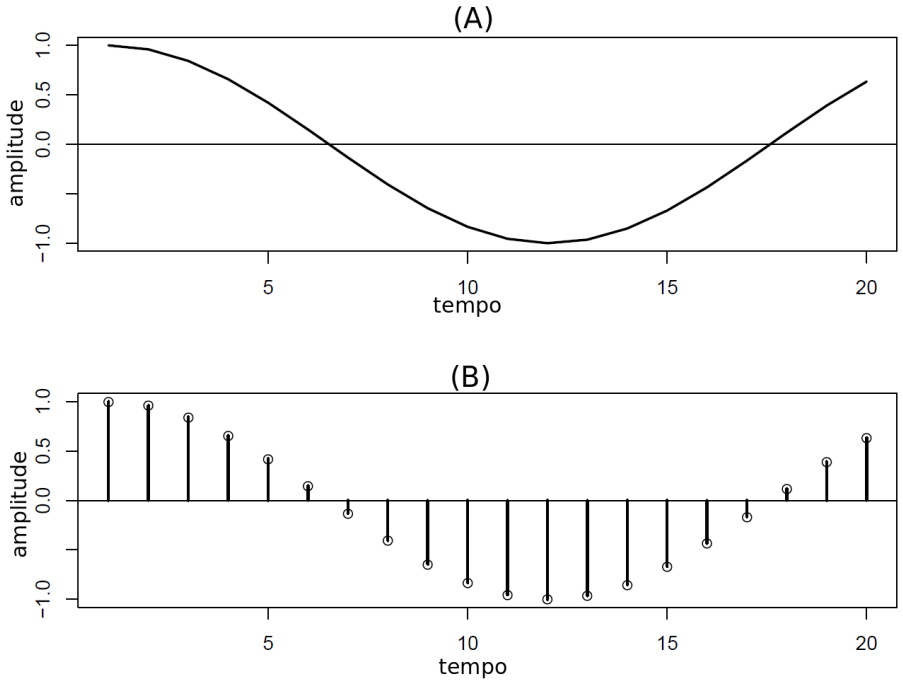

Figura 2.1: Série temporal contínua (A) e discreta (B) (Rios, 2013).

Morettin e Toloi (2006) descrevem uma série temporal T_t composta por $t = 1, ..., n$ observações como a soma de três componentes (Equação (2.1)), com $\mathcal{T}_t$ revelando a tendência, expondo o comportamento ou trajetória da série, $\mathcal{S}_t$ demonstrando a sazonalidade ou o comportamento repetitivo da série, enquanto a_t reproduz um componente aleatório, normalmente um ruído.

$$T_t = \mathcal{T}_t + \mathcal{S}_t + a_t. \tag{2.1}$$

Uma série temporal ainda pode ser classificada como **estacionária** ou **não-estacionária**. Séries estacionárias apresentam um certo equilíbrio estatístico no decorrer do tempo, com suas observações evoluindo com média $\mu(t)$ e variância $\sigma(t)$ constantes, refletindo alguma forma de equilíbrio estável (Equação (2.2)). As não-estacionárias, por sua vez, apresentam a média e a variância variáveis de acordo com o subconjunto de observações (Box et al., 2008;

Morettin e Toloi, 2006). Séries reais, em sua maioria, apresentam características não-estacionárias, como as econômicas e financeiras, além daquelas com comportamento explosivo, semelhante ao crescimento de uma colônia de bactérias.

$$\mu(t) = \mu \ , \ \sigma^2(t) = \sigma^2. \tag{2.2}$$

Descrever o comportamento de um fenômeno por meio de uma série temporal, quando modelado por leis da física, permite realizar predições exatas em qualquer instante do tempo. Seja o exemplo de calcular a trajetória de um míssil lançado à uma direção e velocidade conhecidas: o cálculo de seu curso seria inteiramente **determinístico**[1] caso o ambiente fosse perfeitamente controlado (Box et al., 2008).

Poucos eventos podem ser puramente determinísticos. Fatores como a velocidade do vento, no exemplo do míssil, influenciam no comportamento, alterando o resultado previsto no modelo inicial. Nestes casos, um modelo que pondere a influência aleatória e seja capaz de calcular a probabilidade de um evento futuro, dentro de um limite especificado, é chamado **estocástico** (Box et al., 2008).

Definição 1 (Variável Aleatória) *Uma variável aleatória X qualquer, definida no contexto de um espaço de probabilidades, é uma variável que pode assumir valores reais, sendo possível para qualquer $x \in \mathbb{R}$ se obter uma probabilidade Pr, tal que $Pr(X \leq x)$.*

Informalmente, podemos dizer que uma variável aleatória pode assumir diferentes valores numéricos definidos para cada evento elementar de um espaço amostral Ω, cada qual com uma probabilidade associada (Azevedo Filho, 2009).

Definição 2 (Espaço de Probabilidade) *Espaço de probabilidade é uma tripla $(\Omega, \mathcal{A}, \mathcal{P})$ formada por um conjunto Ω não vazio de f eventos associados a uma probabilidade P, com $0 \leq P \leq 1$ e $\mathcal{A}$ uma álgebra[2].*

Definição 3 (Processo Estocástico) *Seja I um conjunto arbitrário, um processo estocástico é uma família $Z = \{Z(t),\ t \in I\}$, tal que, para cada $t \in I$, $Z(t)$ é uma variável aleatória.*

Um processo estocástico consiste em um grupo de variáveis aleatórias definidas em um mesmo espaço de probabilidades $(\Omega, \mathcal{A}, \mathcal{P})$ (Morettin e Toloi,

[1] O termo determinístico significa que as observações atuais dependem somente das anteriores.

[2] Uma álgebra, dentro do nosso contexto, pode ser interpretada como um relacionamento do evento f no espaço de probabilidades P.

2006). Uma série temporal é dita estocástica quando seus eventos são descritos por um processo estocástico, em que valores futuros podem depender de observações passadas ou surgirem aleatoriamente (Box et al., 2008).

2.1.1 Técnicas de Modelagem de Séries Temporais

Esta seção descreve técnicas de modelagem eficazes para descrever o comportamento de séries temporais analisando observações passadas. Utilizam-se propriedades de sistemas dinâmicos para caracterizar a evolução dos eventos reconstruindo-os no espaço fase. Este espaço considera somente relações entre estados do sistema, removendo o componente temporal para realizar a sua modelagem.

Eventos presentes em séries estocásticas normalmente possuem alguma dependência com valores passados, além da presença de um ruído. Uma propriedade fundamental de sistemas dinâmicos, denominada *recorrência*, quantifica esta influência. Ela identifica a periodicidade de um evento[3] baseada na modelagem do sistema em um conjunto de estados possíveis, no espaço Euclidiano com m dimensões. O uso desta técnica permite investigar tendências, realizar predições e fazer agrupamentos (Ishii et al., 2011; Takens, 1981).

No contexto de sistemas dinâmicos, Takens (1981) afirma em seu teorema que uma série temporal $T = \{t_1, t_2, t_3, ..., t_n\}$ pode ser reconstruída em um estado no espaço fase $T_n(m, \tau) = (t_n, t_{n+\tau}, ..., t_{n+(m-1)\tau})$ onde m é a dimensão embutida (*embedding dimension*) e τ representa a dimensão de separação ou tempo de atraso (*time delay*).

Para a identificação do número de dimensões de separação e embutidas na série temporal, nesta dissertação, foram consideradas as técnicas *Auto-Mutual Information* (AMI) e *False Nearest Neighbors* (FNN) (Fraser e Swinney, 1986; Kennel et al., 1992). Seja o atrator de Lorenz com parâmetros $\sigma = 10$, $\rho = 28$, $\beta = 8/3$ e 100 observações para exemplificar a extração das dimensões (Figura 2.2).

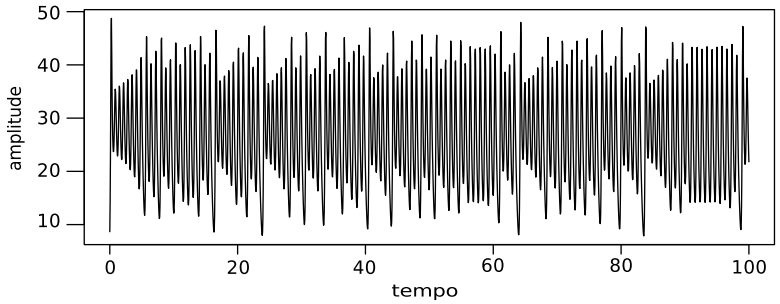

Figura 2.2: Atrator de Lorenz.

A técnica AMI realiza diversos deslocamentos temporais e calcula a pro-

[3]Ou a frequência de sua repetição.

babilidade dos eventos ocorrerem para cada deslocamento. O resultado produzido após aplicá-la na série temporal de Lorenz pode ser visto na Figura 2.3. Fraser e Swinney (1986) propõem a escolha do primeiro mínimo como dimensão de separação. No atrator de Lorenz, verificam-se apenas 3 dimensões necessárias.

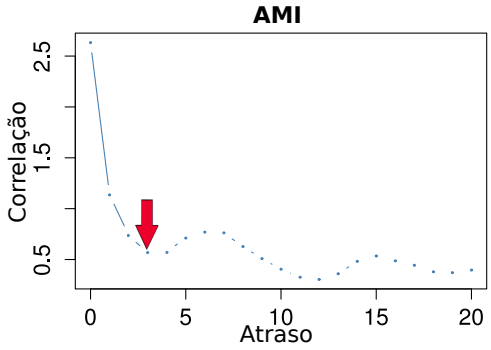

Figura 2.3: Identificação de 3 dimensões de separação.

A técnica FNN, responsável por calcular o número de dimensões embutidas, é executada após o cálculo da dimensão de separação. Kennel et al. (1992) propõem iniciar o cálculo do número de dimensões embutidas com 1 unidade, em seguida, uma nova unidade é adicionada e a distância entre os vizinhos é novamente calculada. Caso a ela aumente, os pontos são considerados falsos vizinhos, caracterizando a necessidade de mais dimensões para reconstruir o comportamento da série. Os autores sugerem que o número de dimensões embutidas adotada deve ser o próximo valor abaixo de $0{,}3$ da taxa de correlação, neste caso, 3 dimensões embutidas são necessárias para caracterizar a série de Lorenz (Figura 2.4).

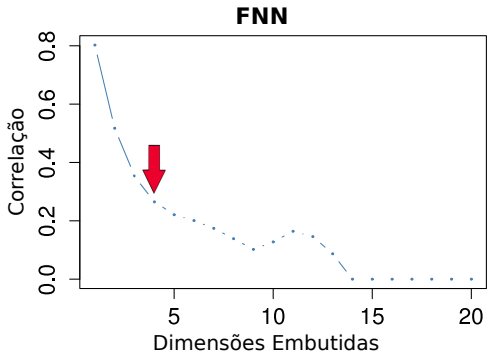

Figura 2.4: Identificação de 3 dimensões embutidas.

Após identificar tais dimensões, pode-se reconstruir a série temporal no espaço multidimensional com a representação gráfica dada pela Figura 2.5, o que permite investigar, por exemplo, os estados do sistema.

A identificação de tais dimensões se faz necessária para a construção do Gráfico de Recorrência[4] (RP) da série temporal. Uma vez completa sua recons-

[4]Representação bidimensional de uma série temporal na qual destaca-se a recorrência dos

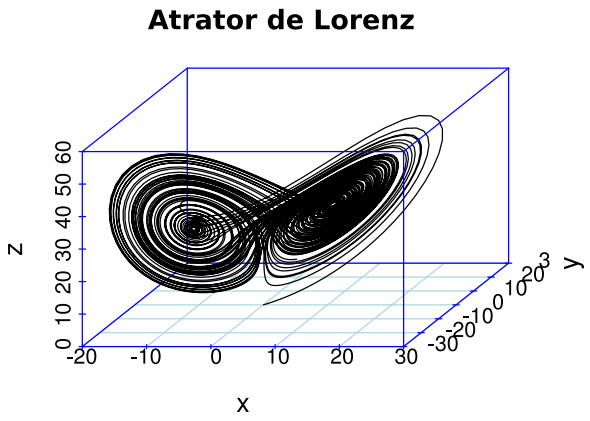

Figura 2.5: Esboço tridimensional do atrator de Lorenz reconstruído no espaço fase.

trução no espaço fase, a recorrência, o determinismo, a entropia[5] e outras métricas que descrevem o fluxo de dados[6] são obtidas pela análise do RP, através da Análise Quantitativa de Recorrência (*Recurrence Quantification Analysis* - RQA).

2.2 Gráficos de Recorrência

Uma série temporal, reconstruída em um espaço multidimensional, conhecido como espaço fase, permite que as relações entre as observações sejam organizadas em uma estrutura bidimensional chamada de Gráfico de Recorrência (Reccurrence Plot - RP) (Marwan et al., 2007).

O *RP* avalia estados de uma série que podem ser representados por uma trajetória no espaço fase e quantifica a recorrência dos eventos presentes na série temporal. São utilizadas técnicas descritas na Seção 2.1 para identificar as dimensões de separação e embutidas necessárias para reconstrução do espaço fase, a fim de reconstruir os estados da série. Tais estados (pontos) são organizados em uma matriz $\mathbf{R}$, formalmente definida pela Equação (2.3).

$$\mathbf{R}_{i,j} = \begin{cases} 1: & \overrightarrow{x_i} \approx \overrightarrow{x_j} \\ 0: & \overrightarrow{x_i} \not\approx \overrightarrow{x_j} \end{cases} \tag{2.3}$$

Em que $\{\overrightarrow{x}\}_{i=1}^{N}$ caracteriza uma trajetória no espaço fase com N estados possíveis e $\overrightarrow{x_i} \approx \overrightarrow{x_j}$ representa quando um estado x_i tendente a outro x_j, de acordo com uma vizinhança ε. A matriz $\mathbf{R}$ compara os estados do sistema nos instantes i e j. Caso sejam semelhantes, $\mathbf{R}_{i,j} = 1$, caso contrário, $\mathbf{R}_{i,j} = 0$.

Um RP é definido pela Equação (2.4), sendo N o número de observações, $\overrightarrow{x_i}$, $\overrightarrow{x_j} \in \mathbb{R}^d$, Θ uma função degrau, ou seja, $\Theta(x) = 0$ se $x < 0$ ou $\Theta(x) = 1$ caso

eventos.

[5]Taxa de incerteza, desordem, quantidade de informação presente no sistema.

[6]Fluxo de dados neste contexto pode ser interpreto como série temporal.

contrário, e $\|\cdot\|$ uma norma[7].

$$\mathbf{R}_{i,j}(\varepsilon) = \Theta(\varepsilon - \| \overrightarrow{x_i} - \overrightarrow{x_j} \|), \text{ i, j = 1,....,N} \tag{2.4}$$

A vizinhança ε é muito importante e esta medida deve ser justa o suficiente para abranger a quantidade de estados no espaço fase tal que o número de vizinhos represente estados recorrentes. Zbilut e Webber (1992) sugerem utilizar 10% da média dos valores observados na série temporal, Rios (2013) propõe utilizar apenas $0,5$ do desvio padrão dos valores observados e Marwan et al. (2007) aconselha estudar o sistema em questão para estimar empiricamente a escolha de ε.

Neste trabalho o parâmetro ε adotado corresponde a $0,5$ do desvio padrão dos valores observados em cada série temporal. Tal escolha pode estar relacionada a um desempenho ruim do classificador de séries temporais, pois ao negligenciá-lo, eventos presentes nas séries que apresentam valores elevados (outliers) podem não estar representados no RP. Consequentemente, por exemplo, as características extraídas podem ser de má qualidade, ou seja, baixa capacidade de representar uma determinada classe.

Eventos na série temporal são refletidos no RP por meio de pontos, abordados nesta dissertação de mestrado como pixels. Considerá-los como pixels permite o uso de técnicas de extração de características apresentadas na Seção 2.3, como aquelas que medem o contraste, a energia e os níveis de cinza contidos em um RP.

A Figura 2.6 exemplifica a construção de um RP a partir de uma série temporal utilizada nos experimentos com 167 observações, 7 dimensões de separação, 3 embutidas e $\varepsilon = 1$. A presença de pontos isolados representam eventos raramente repetidos, ou seja, alta **estocasticidade** (Marwan et al., 2007; Rios, 2013).

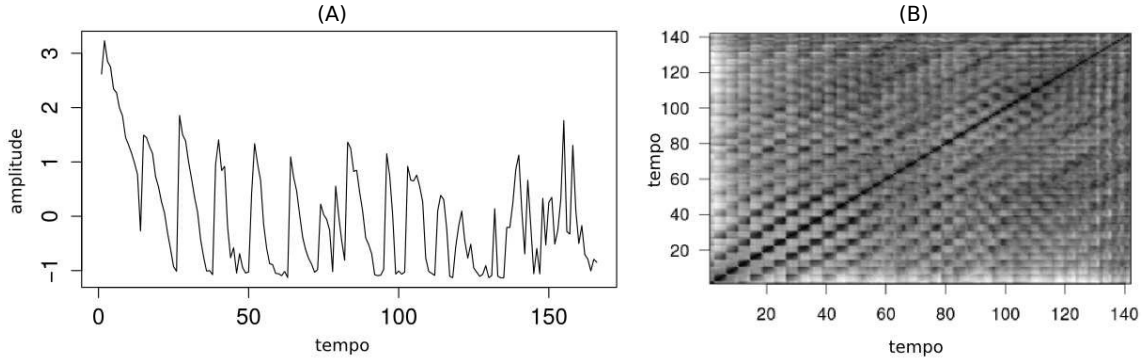

Figura 2.6: Série temporal predominantemente estocástica do dataset *Cloro*.

Na Figura 2.7, uma série temporal também utilizada nos experimentos com

[7]Normalmente emprega-se uma medida de distância definindo-se um limitante inferior (Marwan et al., 2007; Rios, 2013).

275 observações, 4 dimensões de separação, 1 dimensão embutida e $\varepsilon = 1$, percebemos e formação de estruturas bem definidas, denominadas texturas, no contexto de imagens. Linhas diagonais, verticais, formas geométricas ou áreas escuras caracterizam estados repetitivos, ou seja, alta influência **determinística**.

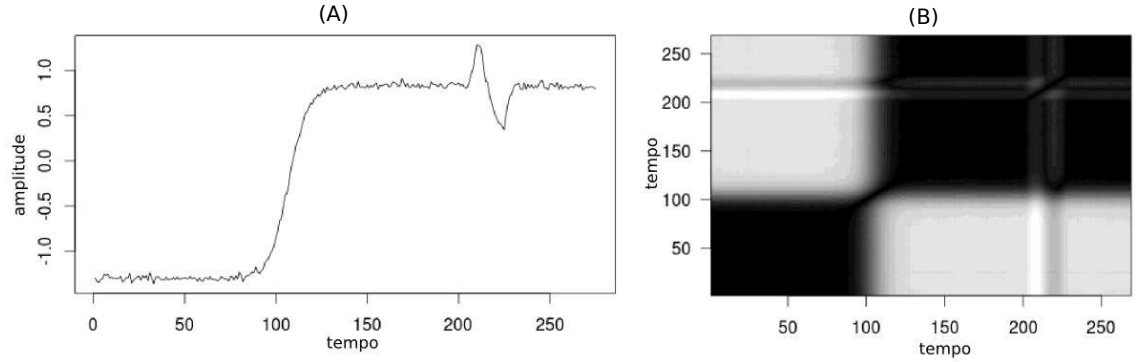

Figura 2.7: Série temporal predominantemente determinística do dataset *Trace*.

Através de uma simples análise visual, em alguns casos, pode-se concluir a presença de padrões frequentes, permitindo caracterizar um RP. Existem técnicas formais para quantificar esses padrões, uma delas chama-se Análise Quantitativa de Recorrência (*Recurrence Quantification Analysis* - RQA) (Marwan et al., 2007; Zbilut e Webber, 1992).

2.2.1 Análise Quantitativa de Recorrência - RQA

A técnica RQA permite descrever um RP a partir de sua estrutura, ou seja, a partir da distribuição dos pontos de recorrência baseando-se em sua densidade, na ocorrência de linhas verticais e horizontais, utilizando uma medida de proximidade ε fixa, como descrito na Equação (2.4) (Marwan et al., 2007).

A taxa de determinismo encontrada para a Figura 2.6 (B) corresponde a 28.98%, levando a caracterizá-la como predominantemente estocástica, enquanto para a Figura 2.7 (B), uma série com taxa de determinismo 99.98% é considerada determinística.

Uma medida obtida pela técnica RQA denomina-se Taxa de Recorrência (do inglês: *Recurrence Rate - RR*). Ela é dada pela Equação (2.5) para uma medida de vizinhança ε e uma matriz de recorrência $\mathbf{R}$ com N observações.

$$RR(\varepsilon) = \frac{1}{N^2} \sum_{i,j=1}^{N} \mathbf{R}_{i,j}(\varepsilon). \tag{2.5}$$

Nesta Equação, computa-se a taxa de repetição dos pontos de recorrência. Outra medida, denominada Taxa de Determinismo, ou previsibilidade do

sistema, é obtida pela Equação (2.6), sendo l o comprimento das linhas diagonais e $P(l)$ o seu respectivo histograma. Estima-se a porcentagem de pontos de recorrência que formam linhas diagonais no RP de comprimento mínimo l_{min}.

$$DET = \frac{\sum_{l=l_{min}}^{N} lP(l)}{\sum_{i,j=1}^{N} \mathbf{R}_{i,j}} \tag{2.6}$$

Ao todo, a técnica RQA fornece onze medidas capazes de descrever um RP formado por uma série temporal analisando-o sob diferentes perspectivas, algumas delas são: *Taxa de Recorrência - RR*, *Taxa de Determinismo - DET*, *Taxa de Recorrência de linhas verticais - LAM*, *Comprimento médio das linhas diagonais - L*, *Comprimento da maior linha diagonal - L_{max}*, *Entropia de Shannon - rENTR*.

Para as séries utilizadas como exemplo, uma do dataset de *Cloro* e outra do dataset *Trace* (Figs. 2.6 e 2.7), os dados dispostos na Tabela 2.1 resumem as métricas RQA[8].

	Série de Cloro	**Série Trace**
RR	12.71	52.09
DET	52.69	99.97
LAM	67.63	99.97
L	2.39	75.16
L_{max}	6	274
rENTR	0.5166	0.9835472

Tabela 2.1: Dados RQA de séries do dataset *Cloro* e *Trace*.

Encontram-se diversas aplicações de cada medida, por exemplo, através da quantificação do nível de determinismo da série para aplicar uma discretização que melhor traduz seu comportamento. Ou ainda, como na pesquisa realizada por Rios (2013), em que o autor utiliza a medida DET para analisar a estocasticidade e o determinismo presente em uma série temporal.

2.3 Extração de Atributos de Gráficos de Recorrência

Um RP criado a partir de uma série temporal quando analisado sob a perspectiva de uma imagem, permite investigá-lo quanto à disposição dos pontos que o compõe. Esta seção descreve técnicas para analisar e quantificar a textura formada pelo agrupamento dos pontos de cada RP, com a finalidade de extrair features para introduzi-las em um algoritmo de AM.

A definição de textura na biologia permite caracterizar uma patologia. Na arquitetura ela pode ser utilizada para descrever um ambiente. Seja qual for

[8]Considerando ε sendo $0,5$ do desvio padrão nos valores observados na série.

o domínio, todas as superfícies possuem uma textura que auxilia a descrevê-las. Para uma imagem em tons de cinza, Haralick et al. (1973) definem textura como um atributo gerado a partir da análise estatística da disposição (organização) dos pixels da imagem.

Sob a perspectiva de uma imagem, um RP formado por uma série temporal possui, entre outros, os seguintes atributos: *auto-correlação, contraste, dissimilaridade, energia, entropia, homogeneidade e variância* (Haralick et al., 1973).

2.3.1 Local Binary Pattern - LBP

Esta técnica de reconhecimento de padrões em imagens permite analisar a textura por meio da comparação entre a intensidade de um pixel central g_c com P pixels g_p delimitados por um raio R (Ojala et al., 2002). A textura extraída de uma imagem é caracterizada pela Equação (2.7), na qual obtem-se um vetor de atributos $\vec{f}_{LBP}$.

$$\vec{f}_{LBP} = \sum_{p=0}^{P-1} s(g_p - g_c)^{2^p} \tag{2.7}$$

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0. \end{cases} \tag{2.8}$$

A operação $s(g_p - g_c)$ tem a capacidade de revelar diferenças de textura. Ela registra as ocorrências de vários padrões na vizinha de um pixel central (g_c) para construção de um histograma. Em regiões onde não são encontradas diferenças texturais, o operador resulta em 0. Quando encontrado uma leve diferença na textura, o operador regista a maior diferença na direção do gradiente.

Em uma matriz 3×3 (Figura 2.8 (a)), um valor limitante é definido baseado no pixel central, intensidades superiores a ele recebem 1 e inferiores 0 (Figura 2.8 (b)). O valor de cada pixel vizinho ao central é multiplicado pelos seus respectivos pesos definidos na Figura 2.8 (c) para obter os oito pixels na Figura 2.8 (d).

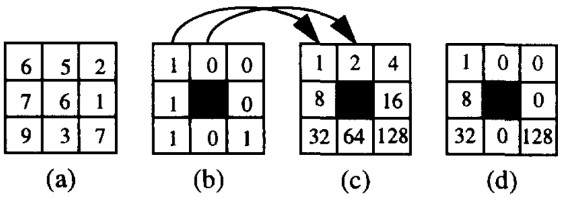

Figura 2.8: Exemplo dos passos da técnica LBP (Ojala et al., 1996).

Nesta técnica, obtêm-se até $2^8 = 256$ unidades de texturas para compor

um histograma. O vetor de features $\vec{f}_{LBP}$ é constituído pela concatenação dos histogramas produzidos pela análise dos pixels de uma imagem.

2.3.2 Grey Level Co-ocorrence Matrix - GLCM

O descritor de textura GLCM realiza uma abordagem estatística sob os pixels criando matrizes, denominadas *matrizes de co-ocorrência*, construídas a partir dos níveis de cinza sob diferentes orientações (Haralick et al., 1973). A técnica GLCM pode ser denominada como um histograma que fornece a frequência de ocorrência de um pixel $P_{(i,j,d,\theta)}$ a uma distância d em uma direção θ, comumente $0°$, $45°$, $90°$ e $135°$, conforme Figura 2.9.

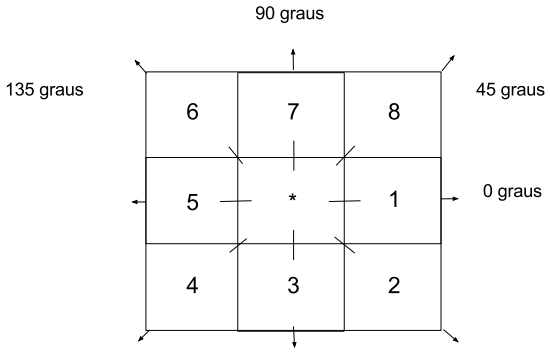

Figura 2.9: Exemplo do método de extração de características GLCM (Haralick et al., 1973).

Tais matrizes permitem descrever a frequência entre os tons de cinza e aplicar técnicas estatísticas como *Correlação, Contraste, Uniformidade e Entropia* encontradas na imagem para compor um vetor de features $\vec{f}_{GLCM}$.

2.3.3 Filtro de Gabor

Este filtro tem por finalidade realçar determinados aspectos ou atenuar ruídos. Sua utilização pode ser verificada em algoritmos de reconhecimento de padrões em íris de seres humanos, no diagnóstico de patologias e outras aplicações (Hollingsworth et al., 2009). Sua constituição é de funções senoides complexas e bi-dimensionais modeladas por uma função mãe também bi-dimensional (Manjunath e Ma, 1996).

Dada uma imagem $I(m,n)$, Manjunath e Ma (1996) definem a Transformada Wavelet de Gabor (TWG) $g_{m,n,}$ pela Equação (2.9), atentando-se para a operação $*$ representando o conjugado complexo[9], sendo $I(m,n)$ uma imagem com largura m e altura n e uma região de interesse possuindo dimensões $x \times y$.

$$g_{m,n}(x,y) = \int I(x_1,y_1) g_{m,n} * (x-x_1, y-y_1) dx_1 dy_1 \tag{2.9}$$

[9]O conjugado complexo de um número $z = a + bi$ é representado por $\bar{z} = a - bi$.

Os atributos que descrevem $I(m,n)$ são obtidos a partir da média μ_{mn} e do desvio padrão σ_{mn} em função da TWG, segundo a região de interesse ($x' \times y'$), conforme Equações (2.10) e (2.11).

$$\mu_{mn} = \int\int |W_{m'n'}(xy)| dxdy \tag{2.10}$$

e

$$\sigma_{mn} = \sqrt{\int\int (|W_{m'n'}(x,y)| - \mu_{mn})^2 dxdy} \tag{2.11}$$

O vetor de atributos $\vec{f}_{Gabor}$, definido pela Equação (2.12), é construído usando μ_{mn} e σ_{mn} como componentes.

$$\vec{f}_{Gabor} = [\mu_{00}\sigma_{00}, \mu_{01}\sigma_{01}, \ldots, \mu_{nm}\sigma_{nm}] \tag{2.12}$$

2.3.4 Segmentation-based Fractal Texture Analysis - SFTA

O algoritmo de extração de atributos SFTA, desenvolvido por Costa et al. (2012), constrói o vetor de características a partir de uma imagem em tons de cinza. Um esboço da extração de características é apresentado na Figura 2.10, na qual a imagem é decomposta para extração de três componentes: a intensidade média de cinza $\overline{\mathcal{V}}$, o número de bordas identificadas $\mathcal{A}$ e as dimensões fractais $\mathcal{D}$ (Costa et al., 2012; Mamani, 2012).

Utiliza-se o conceito de dimensão fractal para medir a irregularidade de uma imagem, neste caso, de um RP. Este conceito baseia-se na ideia de medir o tamanho de objetos quando a geometria euclidiana apresenta resultados insatisfatórios (de Assis et al., 2008). Seja o exemplo de mapear o litoral brasileiro e medir sua similaridade com a costa africana. A utilização de técnicas bi-dimensionais para realizar tal tarefa pode não ser a mais indicada, considerando a densidade de informações presentes nas imagens.

A técnica SFTA pode ser dividida em duas partes principais: decompor uma imagem acinzentada em várias imagens binárias através do algoritmo *Two-Thereshold Binary Decomposition - (TTBD)* e a segunda parte sendo o cômputo das dimensões fractais e o cálculo da intensidade de cinza (Costa et al., 2012).

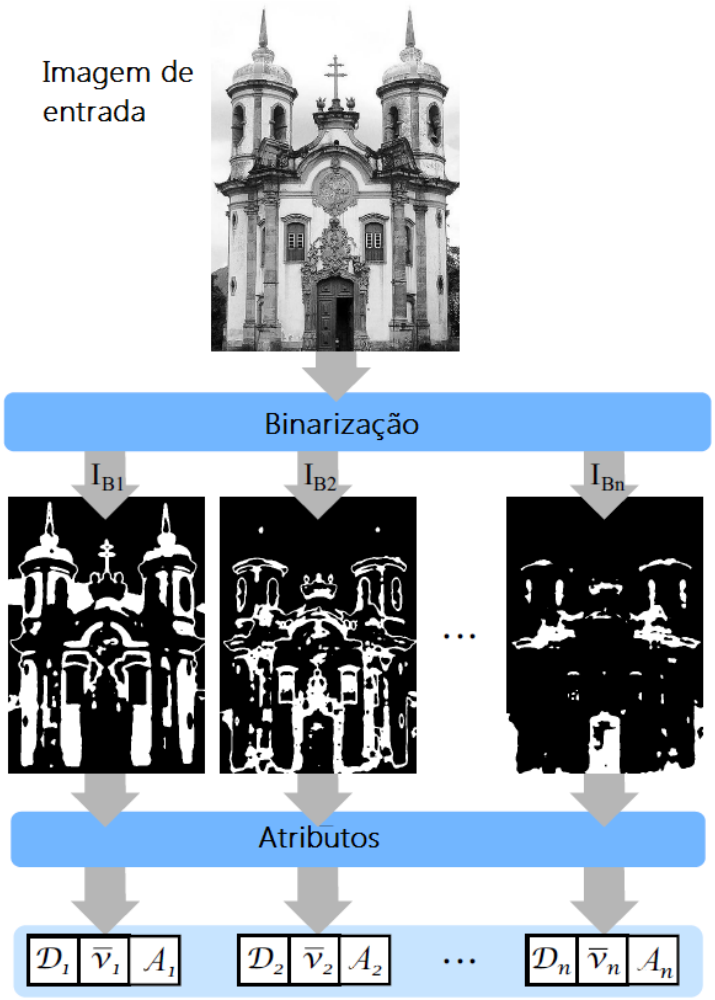

Figura 2.10: Reconstrução da imagem para coletas dos atributos (Costa et al., 2012).

Algoritmo 1: *Segmentation-based Fractal Texture Analysis*

Entrada: Uma imagem I e o número de theresholds n_t

Saída: Vetor de características $\vec{f}_{SFTA}$

início

$T \leftarrow \text{MultiLevelOtsu}(I, n_t)$;

$T_A \leftarrow \{\{t_i, t_{i+1}\} : t_i, t_{i+1} \in T, i \in [1.. \mid T \mid -1]\}$

$T_B \leftarrow \{\{t_i, n_l\} : t_i \in T, i \in [1.. \mid T \mid]\}$

$i \leftarrow 0$

repita

$I_b \leftarrow \text{TwoThresholdSegmentation}(I, t_l, t_u)$

$\Delta(x,y) \leftarrow \text{FindBorders}(I_b)$

$\vec{f}_{SFTA}[i] \leftarrow \text{BoxCounting}(\Delta)$

$\vec{f}_{SFTA}[i+1] \leftarrow \text{MeanGrayLevel}(I, I_b)$

$\vec{f}_{SFTA}[i+2] \leftarrow \text{PixelCount}(I_b)$

$i \leftarrow i+3$

até *enquanto* $T_A \neq 0$ *e* $T_B \neq 0$;

fim

O algoritmo TTBD é aplicado na binarização de uma imagem e utiliza-se do método de Otsu (Huang e Wang, 2009) com diferentes limiares (*threshold number - nt*), sendo escolhido empiricamente. Para cada limiar (*nt*), são identificados $t_1, t_2, t_3, ..., t_n$ níveis de cinza a fim de computar a média, o tamanho e a dimensão fractal de cada limiar utilizando o algoritmo *Box Counting* (Schroeder, 1992).

O vetor de características $\vec{f}_{SFTA}$ (Figura 2.10) é obtido a partir do cálculo da quantidade de dimensões fractais $\mathcal{D}_{1...n}$, da intensidade média de pixels cinza $\overline{\mathcal{V}}_{1...n}$ e da quantidade de bordas identificadas $\mathcal{A}_{1...n}$ para cada limiar.

2.3.5 Transformada Discreta de Cosseno

A Transformada Discreta de Cosseno (DCT), em especial a bi-dimensional (DTC-2D), utilizada normalmente para compressão de imagens, está relacionada à transformada de Fourier. Seja uma imagem I com m representando sua altura e n sua largura, a definição da DCT-2D usada no padrão de imagens *JPEG*[10], é dada pela Equação (2.13) (Pennebaker e Mitchell, 1993).

$$DTC\text{-}2D(m,n) = \frac{1}{4}C(m)C(n)\sum_{m=0}^{7}\sum_{n=0}^{7} f(x,y)*, \\ \cos\frac{(2x+1)m\pi}{16}\cos\frac{(2x+1)n\pi}{16}. \tag{2.13}$$

em que

$$C(m), C(n) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{se } m,n = 0, \\ 1, & \text{caso contrário.} \end{cases}$$

A transformada bi-dimensional DTC-2D produz um vetor de atributos $\vec{f}_{DTC2D}$ com tamanho proporcional ao da matriz utilizada para representar a imagem, no caso da Equação (2.13) são extraídos 64 (8×8) atributos, similar ao utilizado na compressão de imagens *JPEG* (Pennebaker e Mitchell, 1993).

2.4 Decomposição de Modo Empírico

A revisão sistemática da literatura realizada por Rios e Mello (2012) apresenta as técnicas de decomposição de ST mais utilizadas, sendo: Transformada de Fourier (*Fourier Transform - FT*), Transformada de Ondaletas (*Wavelet Transfor - WT*), Análise de Componentes Principais (*Principal Component Analysis - PCA*) e a técnica de Decomposição de Modo Empírico (*Empirical Mode Decomposition - EMD*).

[10]JPEG: do inglês *Joint Photographic Experts Group*, é o nome dado a um método utilizado para comprimir imagens (Pennebaker e Mitchell, 1993).

A técnica de EMD reduz a diferença entre "picos" e "vales", ou seja, diminui a sinuosidade da série (Huang et al., 1998). Ela consiste na extração de um número finito de Funções de Modo Intrínseco (do inglês: *Intrinsic Mode Functions - IMF*) aplicado recursivamente em um sinal, até obter um resíduo ou atingir um limitante pré-estabelecido (Huang et al., 1998; Rios e Mello, 2012). Um sinal, ou série temporal, é composto por um modo oscilatório. Ao aplicar EMD, ocorre uma redução da escala do padrão oscilatório sem perda da informação (Rios, 2013). Se todas as IMFs forem somadas, incluindo o resíduo, o sinal original é recuperado (Huang et al., 1998).

Ao utilizá-la como técnica de análise de séries temporais, possui como característica interessante a independência da estocasticidade, linearidade e estacionariedade (Huang et al., 1998). Aplicável a diferentes contextos, pode-se aplicá-la na fase de pré-processamento da série, atuando como um filtro, para extração de características ou para entender melhor o comportamento da série (Karvelis et al., 2013; Rios, 2013; Wu e Huang, 2009).

Ao aplicar EMD, obtém-se um subconjunto de IMFs através de um processo chamado *sifting*. Este processo identifica pontos de máximo locais $max(T_{1,\dots,t})$ e mínimos locais $min(S_{1,\dots,t})$ em uma série temporal qualquer $T_{i=1,\dots,t}$. Em seguida, interpolam-se os pontos de máximo produzindo uma *spin-line-cúbica-superior* e outra *spin-line-cúbica-inferior* utilizando $min(S_{1,\dots,t})$.

Um exemplo de aplicação da técnica EMD pode ser visto na Figura 2.11, onde os pontos de máximo e de mínimo são destacados com círculos e a série temporal original com uma linha pontilhada (Figura 2.11 (a)). As *spin-lines* são demonstradas com linhas tracejadas (Figura 2.11 (b) e (c)).

Em seguida, computa-se a média $\mu(t)$ entre cada máximo e mínimo local demonstrado na Figura 2.11 (d) com uma linha tracejada. Após o cálculo de $\mu(t)$, subtrai-se $\mu(t)$ de $T(t)$, acrescentando o resultado em um subconjunto chamado H_i, o qual possuirá a série temporal T com parão oscilatório reduzido para a *i-ésima* decomposição (*i-th IMF*).

Estes passos, representados no Algoritmo 2, podem ser utilizados para extrair diversas IMFs até um critério de parada ser atingido, como o desvio padrão estar entre $0,2$ e $0,3$, ou a série remanescente ser uma função constante (Huang et al., 1998).

O conjunto $H_{i=1,\dots,n}$ contem todas as séries (IMFs) obtidas após a aplicação do processo *sifting*. Na Figura 2.12, tem-se um exemplo deste processo com a extração de 7 IMFs ($n = 7$) a partir da série temporal formada a partir de um sistema Lorenz com parâmetros $\sigma = 10$, $\rho = 28$ e $\beta = 8/3$ com um comportamento aleatório criado para 100 observações (Alligood et al., 1997). Esta série é a mesma utilizada como exemplo para identificação das dimensões de separação e embutidas na Subseção 2.1.1.

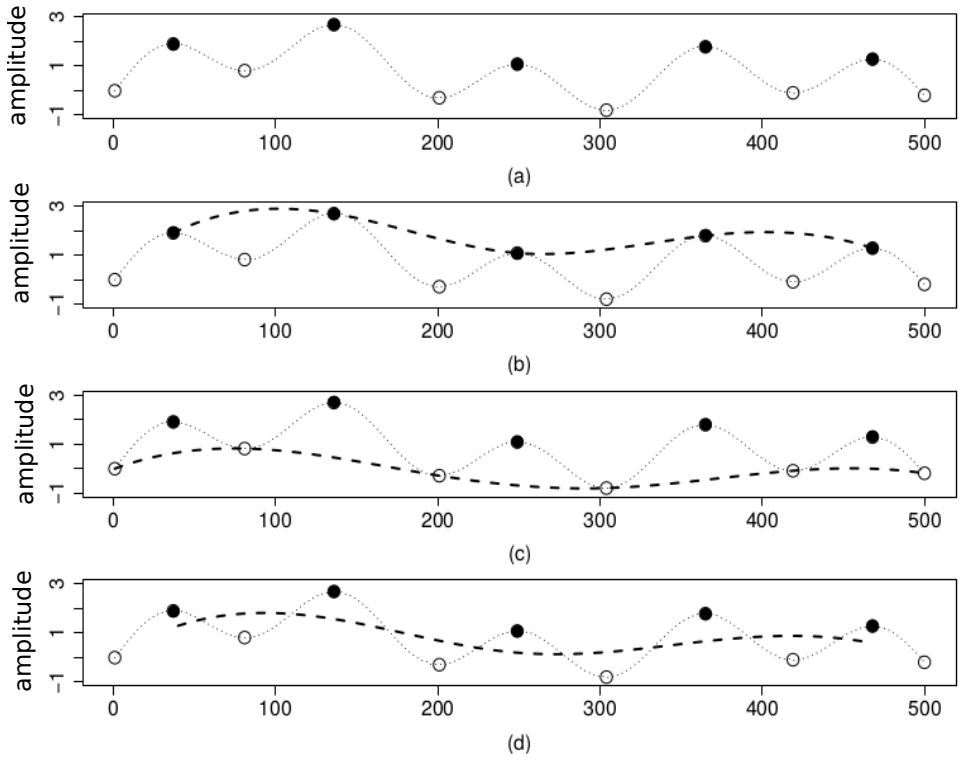

Figura 2.11: Exemplo de aplicação da técnica EMD para uma série temporal T qualquer. São demonstrados 4 passos ((a), (b), (c) e (d)) para identificação da média $\mu(t)$ oriunda das *spin-lines*.

Para cada função $H_i(t)$, com $i = 1...7$, tem-se uma IMF, após a sétima decomposição, neste caso, obteve-se o resíduo $r(t)$. Por definição, um IMF deve atravessar o eixo das abcissas (cruzamentos de zero) em quantidade igual ou diferente em até uma unidade e, ainda, a média observada entre o valor máximo e mínimo local[11] deve ser zero (Huang et al., 1998).

Algoritmo 2: Decomposição de Modo Empírico

Entrada: Uma série $T(t)$

Saída: H_i, com $0 \leq i \leq n$

início

repita

$max \leftarrow$ Encontre o máximo local de $T(t)$;

$min \leftarrow$ Encontre o mínimo local de $T(t)$;

Realiza Interpolação (max);

Realiza Interpolação (min);

Calcule a média $\mu(t) \leftarrow \frac{min(t)+max(t)}{2}$;

Subtraia $\mu(t) - T(t)$ e chame de H_i;

até *a série ser apenas ruído, um valor constante ou outro critério de parada* ;

fim

[11] O máximo e o mínimo local refere-se a amplitude observada da série/sinal em relação ao eixo das ordenadas.

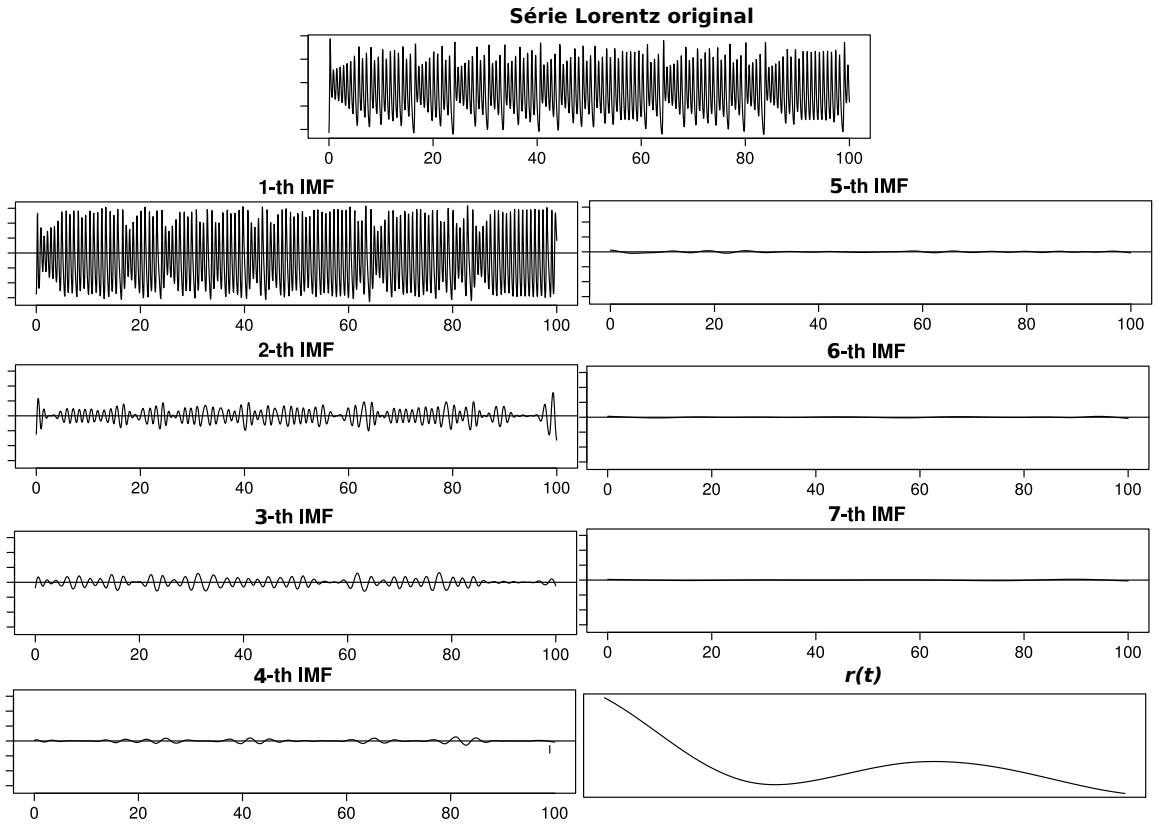

Figura 2.12: Exemplo da aplicação da técnica EMD com extração de 7 IMFs na série temporal formada pelo sistema *Lorenz*.

2.5 Aprendizado de Máquina

Esta seção descreve algoritmos de Aprendizado de Máquina (AM) utilizados para classificação de dados. Anderson et al. (1986) definem AM como o campo de pesquisa da Inteligência Artificial cujo objetivo é assimilar características de amostras de dados.

A partir das amostras, constrói-se um modelo classificador para rotular instâncias do problema. Entende-se como classificação o processo de atribuir um rótulo a uma determinada informação, isto é, atribuir uma classe[12] à uma instância. Na Figura 2.13, tem-se um conjunto de dados composto por atributos e classes, ao qual aplica-se uma técnica de AM com o objetivo de construir um classificador.

Haykin (1998) define três paradigmas de aprendizado utilizados na geração de um classificador: supervisionado, não-supervisionado e por reforço. No primeiro paradigma, o aprendizado ocorre com exemplos rotulados. O algoritmo é treinado a partir de um conjunto de dados na qual as classes são conhecidas. No paradigma não-supervisionado, as instâncias não são rotuladas, cabe ao algoritmo de aprendizado agrupá-las de acordo com uma medida. No paradigma por reforço o aprendizado ocorre por meio de recompensas (ou

[12]A classe descreve o fenômeno de interesse.

não) de acordo com o desempenho do classificador.

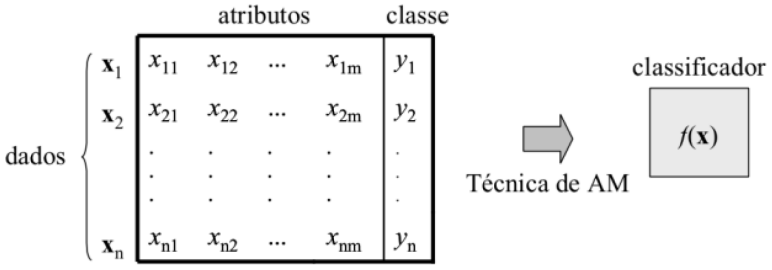

Figura 2.13: Um conjunto de dados $(x_1, x_2, ...x_n)$ de n amostras rotuladas $(y_1, y_2, ...y_n)$ é utilizado para construção de um classificador $f(x)$ utilizando técnicas de Aprendizagem de Máquina (AM).

Com o objetivo de construir um modelo classificador de séries temporais eficiente, metodologias existentes na literatura norteiam o processo de aprendizagem e a validação dos resultados. Billingsley (2013) descreve em seu *Teorema do Limite Central* a necessidade da realização de no mínimo 30 repetições de um experimento para poder afirmar, por exemplo: *"a taxa de acerto do classificador tente a permanecer constante"*.

Tais repetições ainda devem obedecer alguns critérios de amostragem para que o resultado do experimento possua validade. No contexto de classificação, para cada uma, das 30 rodadas de experimentação, deve-se construir aleatoriamente e de maneira estratificada[13], dois conjuntos de dados. O primeiro destinado a treinar o modelo classificador de dados e o outro a testá-lo, pois a abordagem utilizada nesta dissertação será a supervisionada, justificado por outros autores realizarem esta separação e para que os resultados experimentais sejam comparáveis a eles.

2.5.1 Seleção de Features

A seleção de features tem como objetivo identificar características do conjunto de dados que sejam úteis sob o aspecto da aprendizagem para a construção do modelo classificador (Hall, 1999). Esta tarefa proporciona a redução da dimensionalidade dos dados, a remoção de features irrelevantes ou redundantes e ainda permite que os algoritmos sejam executados mais rapidamente (Dash e Liu, 1997; Guyon e Elisseeff, 2003).

Existem na literatura diversas técnicas para solucionar o problema de seleção de features, dentre elas, encontram-se as baseadas em *Filtros* e as chamadas *Wrappers*.

[13]Preservando a proporção de exemplos de cada classe.

Técnica baseada em Filtros

Uma técnica de filtragem de features denomina-se *Information Gain* (IG). Diferente de outras técnicas, esta (Figura 2.14) não utiliza algoritmos de aprendizagem para maximizar a escolha das features.

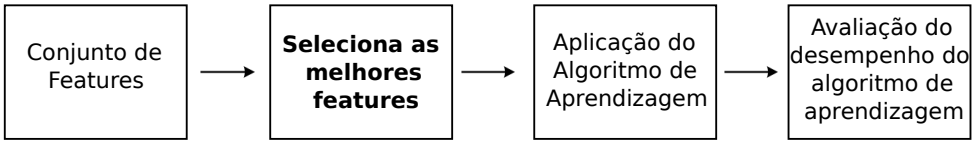

Figura 2.14: Em destaque, ação de um método baseado em filtro para seleção de features.

A técnica de filtragem IG quantifica o volume de informação (em bits) sobre a predição de uma classe dada uma entropia como medida de impureza em um conjunto de dados S, neste caso, de features coletadas a partir da análise de textura de RPs.

O IG computa o ganho de informação, Gain$(S,\ f)$, através da redução da entropia proporcionada pelo particionamento (p_c) em k classes de S, para cada feature f de S (Mitchell, 1997).

$$\text{Entropia}(S) = \sum_{c=1}^{k} -p_c * \log_2 p_c \tag{2.14}$$

$$\text{Ganho}(S,f) = \text{Entropia}(S) - \text{Entropia}(S|f) \tag{2.15}$$

O primeiro termo da Equação (2.15) refere-se a entropia calculada para o conjunto original de features enquanto o segundo termo, refere-se ao subconjunto S particionado utilizando a feature f como critério de particionamento. Valores elevados de entropia refletem alta taxa de informação, enquanto valores menores traduzem maior pureza no conjunto de dados.

$$\text{Taxa de Ganho}(S,f) = \frac{\text{Ganho}(S,f)}{\text{Entropia}(S|f)} \tag{2.16}$$

Um *ranking de features*[14] que melhor descreve o conjunto de dados pode ser criado utilizando a quantidade de informação oferecida por cada uma delas. Este passo é definido pela Equação (2.16) na qual Quinlan (1986) sugere utilizar somente as features tal que a Taxa de Ganho (*Gain Ratio*) seja acima da média.

Técnica Wrapper

Um método wrapper de seleção de features denomina-se RFE (*Recursive Feature Elimination*). Ele realiza um ranking das melhores features dada a

[14]Features mais significativas ordenadas de maneira decrescente.

sua importância na construção do modelo classificador. Para tal, usa-se um algoritmo de aprendizagem no passo de seleção de features (Figura 2.15).

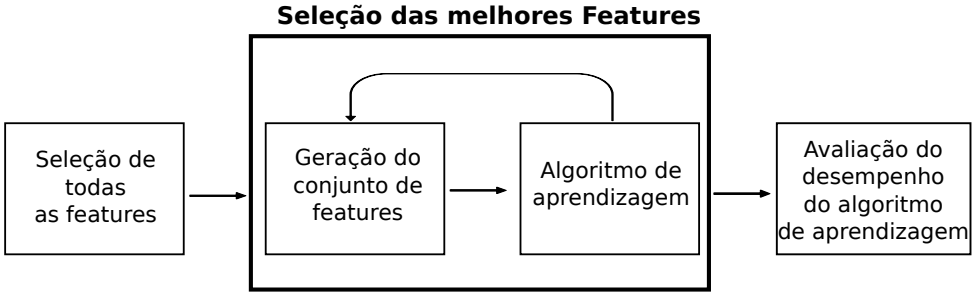

Figura 2.15: Em destaque, ação de um método wrapper para seleção de features.

A técnica RFE empregada nesta pesquisa tem como métricas para construção do ranking de features a Acurácia (Equação (2.17)) e o Kappa (Equação (2.19)) e para a construção do modelo classificador intermediário o algoritmo *Random Forest* (RF) (Guyon et al., 2002). A partir do algoritmo de aprendizagem RF, constrói-se uma estrutura em árvore para ordenar as features por importância, sendo a mais relevante armazenada na raiz da árvore. Os passos da técnica wrapper RFE são demonstrados no Algoritmo 3.

Algoritmo 3: Pseudocódigo do método wrapper RFE.

Entrada: Conjunto rotulado $S = \{f_1, ..., f_n\}$ de n features
Saída: Uma lista R das melhores features.
início
  repita
    $model \leftarrow$ RF-train(S);
    $R \leftarrow$ rankFeatures(S, $model$);
    $f^* \leftarrow$ lastFeatures(R);
    $S \leftarrow S - f^*$
  até $S = \{\}$;
fim

A técnica RFE envolve a construção de um subconjunto de características chamado out-of-bag (OOB) para aferição dos resultados. Este conjunto é criado para testar o desempenho do classificador RF nas features selecionadas. Ele não possui relação com os dados originais reservados para teste e aferição do classificador final.

A importância de cada feature é dada assim que ela entra no conjunto de aprendizagem do algoritmo RF e sai para o conjunto OOB. Este passo de permutação do conjunto de aprendizagem para o OOB permite medir o impacto na Acurácia e no Kappa dos rótulos preditos.

Intuitivamente, features irrelevantes não mudam o erro de predição (Liaw e Wiener, 2002). A cada iteração do algoritmo, uma lista f^* de features que não impactam na taxa de acertos do modelo é removida.

Ao final, um ranking R das features que mais impactaram no desempenho do algoritmo RF é devolvida para ser utilizada no algoritmo de aprendizagem desejado. Este ranking deve oferecer a maximização dos resultados para o conjunto de treino, podendo ocasionar *overfitting* (Granitto et al., 2006), ou seja, o classificador possui pouca capacidade de generalização, apresentando bons resultados apenas quando utilizado os dados de treino, enquanto no de teste, resultados aquém do esperado[15].

2.5.2 Algoritmos de Aprendizagem

Esta seção descreve algoritmos de aprendizagem utilizados nesta pesquisa, bem como suas definições básicas e métricas de aferição do processo de aprendizagem. O objetivo do algoritmo de aprendizagem é a construção de um modelo classificador, neste caso, de séries temporais rotuladas. São descritos brevemente dois algoritmos de AM, as Máquinas de Vetores Suporte (Support Vectors Machines - SVMs) e o C5.0.

Baseado em Teorias Estatísticas, o algoritmo SVM demonstra um bom desempenho na tarefa de classificação e regressão de dados. Aplicável em diversas áreas, apresenta resultados excepcionais no domínio da biologia, informática e medicina (Aggarwal e Zhai, 2012; Byun e Lee, 2002; Silva, 2014). Smola et al. (2000) descrevem como vantagem sua boa capacidade de generalização (ao evitar *overfitting*) e bom desempenho com problemas de dimensões elevadas.

Considere um problema binário[16], o objetivo das SVMs é separar as instâncias utilizando uma função construída a partir da análise de alguns exemplos do conjunto de dados (fase de treinamento). Esta função deve ser capaz de separar exemplos desconhecidos (capacidade de generalização do modelo). A ideia básica das SVMs é diferenciar elementos do conjunto de dados através da construção de um hiperplano (f), realizando uma separação binária (Figura 2.16) com margem máxima (B), ao contrário de uma margem qualquer (A).

Quando a separação por um hiperplano não é suficiente, como em problemas não linearmente separáveis (Figura 2.17 (A)), são utilizadas funções reais que mapeiam as entradas para um novo espaço, denominado *espaço de características*, tornando-as linearmente separáveis (Figura 2.17 (B)).

A escolha destas funções (Φ) que mapeiam as entradas para este novo "espaço" introduz o conceito de *Kernel*. Um Kernel K é uma função que recebe dois pontos x_i e x_j do espaço de entradas e computa o produto escalar $\Phi(x_i) \cdot \Phi(x_j)$ no espaço de características (Figura 2.17 (B)) (Haykin, 1998). Den-

[15] Pouca eficiência.

[16] Adaptações são realizadas para problemas do tipo *multiclasses*, sendo realizada decomposição "um-contra-todos" e "todos-contra-todos".

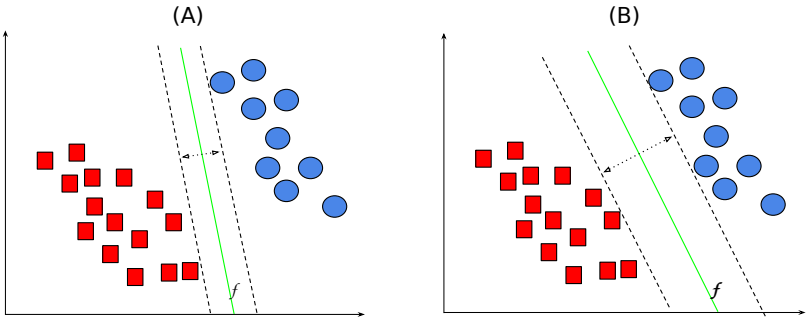

Figura 2.16: Exemplo de separação linear.

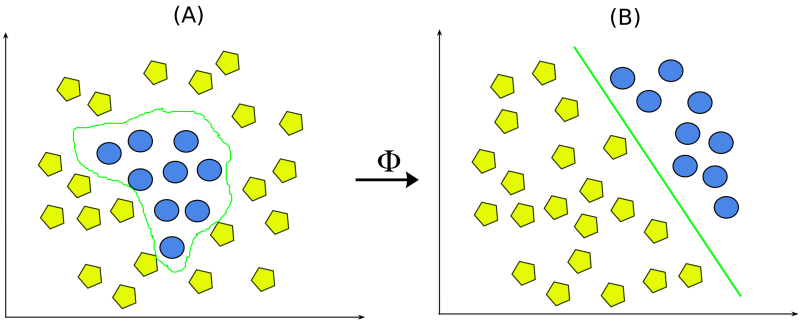

Figura 2.17: Exemplo de separação não linear.

tre os Kernels mais usados, destaca-se o *Polinomial*, o *Gaussiano* e o *Sigmoidal* (Hsu et al., 2010). A Figura 2.17 mostra o exemplo do uso de uma função Kernel Polinomial para construção de um espaço de características linearmente separável.

Outro algoritmo de classificação baseado em árvore de decisão é o C5.0 (Quinlan et al., 2008), uma evolução do C4.5 (Quinlan, 1993). Este oferece melhorias em relação ao seu antecessor quanto à velocidade de execução, eficiência do uso de memória, incorporação de aprendizado por reforço, dentre outras. Assim como seu antecessor, este algoritmo baseia-se no Ganho de Informação para descobrir quais features proporcionam maior redução da entropia para defini-las como nós de uma árvore hipotética.

O conhecimento adquirido pelo algoritmo é representado em nós, armazenando as features mais representativas próximas à "raiz". Inicialmente, todas as *features* são normalizadas e colocadas em um mesmo conjunto C. Em seguida, escolhe-se uma *feature* f e verifica-se a razão do ganho de informação pelo particionamento de C em relação a f. Faz-se o mesmo até que todas as features sejam processadas registrando-se o maior ganho de informação obtido. Em seguida, a *feature* f que ofereceu o maior ganho de informação é removida de C e posicionada na raiz. Repetem-se estes passos até que todas as features sejam processadas (removidas de C).

2.5.3 Métricas de Aferição dos Resultados

Esta seção descreve métricas de aferição das predições geradas pelos classificadores. Dado um conjunto de séries temporais utilizado como treinamento, a fim de construir um classificador, este deve ser aferido com um outro conjunto, agora de teste, quanto a sua capacidade de classificação de séries desconhecidas.

Dado um problema de classificação binária, na qual existem apenas duas classes (P e N), algumas notações básicas do domínio de AM são utilizadas para medir a qualidade de um classificador, dentre elas: PV (Positivos Verdadeiros), FP (Falsos Positivos), FN (Falsos Negativos) e NV (Negativo Verdadeiros). Utilizando esta notação, cria-se uma representação visual chamada *Matriz de Confusão* para tabular as predições do classificador e os rótulos reais dos exemplos de teste.

	Referência P	**Referência N**
Classificador P	PV	FN
Classificador N	FP	NV

Tabela 2.2: Estrutura de uma Matriz de Confusão.

A partir da análise da Tabela 2.2, pode-se extrair Acurácia (*Accuracy*). Ela mede a proporção dos resultados corretos (VP + VN) dentre todos os casos examinados (VP + VN + FP + FN). Quando o domínio de classificação apresenta desbalanceamento dos exemplos nas classes, outras medidas de aferição do classificador podem ser usadas, como a Precisão (*Precision*) e a Cobertura (*Recall*). A Precisão mede a proporção de exemplos positivos que foram classificados corretamente e a Cobertura mede a proporção de exemplos que foi classificada corretamente como exemplos positivos (Equação (2.17)).

$$\text{Acurácia} = \frac{\text{VP}+\text{VN}}{\text{VP}+\text{VN}+\text{FP}+\text{FN}}, \qquad \text{Precisão} = \frac{\text{VP}}{\text{VP}+\text{FP}}, \qquad \text{Cobertura} = \frac{\text{VP}}{\text{VP}+\text{FN}}. \tag{2.17}$$

A relação[17] entre a Precisão e a Cobertura provê uma terceira métrica interessante chamada Medida-F (*F-Measure*), constituindo-se da média harmônica entre a Precisão e a Cobertura (Equação (2.18)).

$$\text{Medida-F} = \frac{\text{Precisão} * \text{Cobertura}}{\text{Precisão} + \text{Cobertura}}. \tag{2.18}$$

Outra medida estatística, o Índice Kappa, mede a relação entre os rótulos atribuídos pelo classificador e os rótulos reais do conjunto de teste. Utilizando a Matriz de Confusão (Tabela 2.2), define-se a Equação (2.19) para obter do Ín-

[17] Baixos valores de Precisão podem indicar um alto número de Falsos positivos enquanto baixos valore de Cobertura podem indicar alto número de falsos negativos.

dice Kappa, na qual p é a proporção em que existe acordo e em p_e a proporção de expectativa de acordo (Cohen et al., 1960; Landis e Koch, 1977).

$$\text{Kappa} = \frac{p - p_e}{1 - p_e}. \tag{2.19}$$

Landis e Koch (1977) sugerem utilizar como referência a Tabela 2.3 para interpretar o Índice Kappa, sendo valores iguais ou inferiores a 0 quando há total discordância entre os rótulos preditos pelo classificador e sua referência, enquanto valores maiores que $0,8$ representam "excelente concordância" entre os rótulos atribuídos pelo classificador e os reais dos dados utilizados para teste.

Índice Kappa	**Nível de Concordância**
< 0	Sem concordância
0 - 0,19	Baixa concordância
0,19 - 0,39	Fraca concordância
0,40 - 0,59	Moderada concordância
0,60 - 0,79	Elevada concordância
0,80 - 1,00	Excelente concordância

Tabela 2.3: Interpretação do índice Kappa.

2.6 Trabalhos Relacionados

A revisão bibliográfica realizada por esta pesquisa sobre análise e classificação de séries temporais traz trabalhos julgados apropriados e relevantes à área de Inteligência Artificial, Análise de Sinais e Processamento de Imagens, cada qual no seu domínio de aplicação sendo investigados: os dados utilizados, as técnicas de extração e representação do conhecimento, os algoritmos de aprendizado e o desempenho alcançado.

Uma complexa busca literária foi realizada consultando os principais repositórios digitais cujas publicações apresentassem relevância e validação científica. É apresentada nesta seção o resultado desse levantamento e os resultados obtidos por cada trabalho, bem como uma breve análise a respeito da abordagem utilizada pelos autores para solucionar problemas de classificação envolvendo séries temporais.

Dentre eles, Rios e De Mello (2013) descrevem a importância de analisar separadamente os componentes estocásticos e determinísticos de séries temporais. Seu trabalho contribui com uma metodologia de separação destes componentes utilizando Decomposições de Modo Empírico (EMD). Basicamente, a separação deles ocorre por sucessivas aplicações de EMD. Para cada IMF extraído, ocorre a sua reconstrução no espaço fase e formação do seu respectivo RP para aplicação da técnica RQA. A partir dela, afere-se o nível de de-

terminismo (DET) presente e caso seja maior que um pré-definido *threshold*, este IMF é reservado em um conjunto de IMFs determinísticos (Rios, 2013). Após a extração de todos IMFs, aqueles que foram adicionados ao conjunto de IMFs determinísticos, são somados para compor o componente determinístico, enquanto aqueles que não atingiram o *threshold* mínimo, são somados para compor o componente estocástico.

Dispondo de 14 anos de dados meteorológicos do volume de chuva e da incidência de raios ultravioleta em diferentes países, Rios e De Mello (2013) encontraram padrões que relacionam a luminosidade e a temperatura. Uma investigação decorrente desse trabalho seria a utilização da sua abordagem para auxiliar na classificação de dados temporais ruidosos, trabalhando como um filtro, proporcionando melhor acurácia na classificação, ao analisar o componente determinístico da série separado do estocástico.

Biem et al. (2013) descrevem uma ferramenta produzida pelo departamento de pesquisa e desenvolvimento da *IBM-Nova York* para gerenciar, visualizar e detectar anomalias em séries temporais multivariadas em tempo real. Os dados coletados referem-se ao uso de CPU, a vazão das informações pela rede, ao uso do disco rígido, dentre outros dados provindos de sensores coletados em intervalos de 15 segundos.

Basicamente, de forma não-supervisionada, a classificação de uma anomalia está relacionada a sua diferença entre a observação atual em relação às observações da última hora. Devido à necessidade da aplicação ser em tempo real, para a representação da série utiliza-se uma variação do polinômio de Laguerre implementado de maneira recursiva. Dada essa representação, calcula-se a variância e o desvio padrão de cada observação dos últimos 60 minutos e compara-se com a observação atual, obtêm-se um erro relativo ao previsto pela representação polinomial e o observado.

O aprendizado do classificador de anomalias ocorre automaticamente ajustando os coeficientes da representação polinomial da série. A identificação da anomalia baseia-se na diferença entre o dado esperado e o obtido, ou seja, não possui um valor fixo, pois é modificado conforme os dados surgem, sendo computado de maneira ponderada em que as observações mais recentes tem maior importância.

Após a detecção de uma anomalia, ela pode ser classificada como: em diminuição, em diminuição acentuada, em forte queda, em crescimento, em crescimento acentuado, oscilante, com falha, anormal (*outlier*), com granularidade incomum e não-anômalo. A classificação de uma anomalia está relacionada com uma análise estatística realizada dos eventos da última hora. Experimentos realizados por Biem et al. (2013) alcançaram 94% de acurácia para detecção de anomalias em um conjunto com 514 séries temporais.

Uma investigação quanto ao armazenamento das séries que permita a descoberta de padrões frequentes, poderia auxiliar na identificação de anomalias desconhecidas. Técnicas de visualização das anomalias detectadas podem ser exploradas para auxiliar não somente na identificação, mas também na gestão do conhecimento.

A pesquisa realizada por Silva (2014) e Silva et al. (2013a) tem como objetivo comparar métodos de classificação por similaridade e por extração de atributos que possam ser utilizados no contexto da classificação de insetos. Este trabalho realizou uma aproximação ao tema de processamento de sinais de áudio pois os dados foram coletados por um sensor óptico. Segundo o autor, iniciativa semelhante tem sido realizada para reconhecimento de fala, instrumentos musicais e espécie de animais.

Os experimentos foram realizados a fim de obter melhores resultados na classificação de insetos, foram conduzidos utilizando vários algoritmos tradicionais incluindo variações de kernels e de parâmetros, diferentes técnicas de representação do sinal (Representação Temporal, Espectral e Cepstral) e de decomposição (em senos e cossenos) para obter $81,87\%$ de acurácia na classificação por similaridade e $87,33\%$ de acurácia na classificação por extração de atributos. O melhor resultado foi obtido pelo algoritmo SVM com *kernel* RBF treinado com atributos mel-cepstrais. Para análise dos classificadores por busca de similaridade, o melhor resultado foi obtido utilizando o algoritmo kNN com a Distância Topsoe. O trabalho inclui como anexo, uma métrica chamada de *Recurrence Patterns Compression Distance (RPCD)* para medir a similaridade entre séries temporais baseada em RP e na *Distância Kolmogorov* na qual define um índice de similaridade entre séries temporais. Experimentos realizados pelo autor demonstram que a métrica supera a Distância Euclidiana em 73,68% e a DTW em 52,63%.

Silva (2014), realizou a aplicação de um filtro para suavizar a série temporal e obter melhores resultados, no entanto não há descrição de como o filtro age na série nem há comparação com outros filtros, deixando uma possibilidade de pesquisa futura.

O trabalho realizado por Baydogan et al. (2013) consiste em implementar um framework para mineração de dados em séries temporais baseado na extração de atributos. Seu funcionamento baseia-se em dividir recursivamente a série temporal em intervalos e para cada intervalo (e sub-intervalo), calcula-se o declive da "regressão linear", média, variância e informações quanto a localização desse intervalo na série temporal para compor um vetor de atributos.

Cada vetor de atributos gerado pelos intervalos é armazenado em um conjunto chamado de Bag-of-Features (BoF) para treinar um classificador. Os autores realizaram experimentos com datasets do repositório UCR (Keogh et al.,

2006) tanto para algoritmos supervisionados (Random Forest e SVM) quanto não-supervisionados (K-means) com a Distância Euclidiana. Posterior análise deste artigo, cabe responder quanto aos intervalos formados serem disjuntos ou não.

O trabalho realizado por Souza et al. (2014) utiliza quarto descritores de texturas em gráficos de recorrência (LBP, GLCM, SFTA e Gabor). Com 38 datasets do repositório *UCR*, os autores realizaram a coleta de 823 features dos gráficos de recorrência formados por cada série temporal a fim de treinar um algoritmo de AM para a tarefa de classificação. Valendo-se de parametrização empírica dos algoritmos descritores de textura, os autores obtiveram sucesso em 26,31% dos casos utilizando somente o algoritmo SVM, acurácia superior à algoritmos tradicionais, como o 1NN-DTW e 1NN-Euclidiano. Carece de investigação a utilização de outros algoritmos para classificação, como Misturas Gaussianas ou Árvore de Decisão, além da possibilidade de agregar atributos extraídos por outros descritores. Lacunas encontradas neste trabalho motivaram nossa abordagem para investigar a influência determinística e estocástica nas séries temporais.

O trabalho conduzido por Pereira (2013) e por Pereira e de Mello (2014) identifica diversas medidas descritivas de séries temporais, tais como coeficientes de *Wavelet*, da transformada de *Fourier*, expoentes de *Hurst*, dentre outros para serem utilizados como features. O autor expõe a criação de um algoritmo de agrupamento de fluxo de dados contínuos chamado *Time Series Stream (TS-Stream)*. Este algoritmo não-supervisionado aprende com novas entradas no fluxo, em tempo real, ele adapta-se por calcular a variância ponderada e a entropia do conjunto assim que novos dados chegam. O autor realizou experimentos com dados de áudio e com séries provindas de ações financeiras, identificando por exemplo, ações mais rentáveis na bolsa de valores. Decorrente da análise deste trabalho, carece uma investigação a eficiência das medidas descritivas utilizadas comparadas com outras, inclusive com às obtidas pela análise textural de RPs.

Ishii e De Mello (2012) realizaram uma otimização de acesso a dados em sistemas distribuídos sem requerer nenhuma informação sobre solicitações antigas. Características como linearidade, estacionaridade e estocasticidade são exploradas por uma taxonomia hierárquica. Através dela, o modelo[18] que melhore descreve o fluxo de dados é aplicado para realizar predições quanto a leituras e escritas em arquivos distribuídos.

A Tabela 2.4 sintetiza as técnicas utilizadas, palavras chaves e pontos interessantes dos trabalhos relacionados neste trabalho. Verifica-se uma lacuna de trabalhos que realizam *análise* e *classificação* de séries temporais (Colunas

[18]ARIMA, AR, Polinomial e RBF.

3 e 4).

Autor (ano)	Realiza previsão?	Realiza classificação?	Analisa a série temporal?	Utiliza RP?	Aplica EMD?
Rios (2013)	SIM		SIM	SIM	SIM
Biem (2013)	SIM	SIM			
Baydogan (2013)		SIM			
Silva (2014)		SIM			
Souza (2014)		SIM		SIM	
Ishi (2012)	SIM		SIM	SIM	
Pereira (2014)	SIM	SIM	SIM		SIM

Tabela 2.4: Trabalhos relacionados

Têm-se como *análise de séries temporais* (Tabela 2.4 - Coluna 4), o uso de técnicas e ferramentas da área de processamento de sinais como apoio a tarefas de classificação ou regressão. E ainda, se o trabalho aborda séries temporais como segmentos fluxo de dados, para reconstrução no espaço fase e análise da influência dos componentes estocásticos e determinísticos.

Já *classificação de séries temporais* (Tabela 2.4 - Coluna 3) , verifica-se se o trabalho soluciona o problema de caracterizar um segmento temporal e atribuir um rótulo a ele.

Relacionar estas áreas, motiva este trabalho a verificar/relacionar a influência determinística e estocástica presente nas séries temporais com o desempenho de classificadores. Também desperta o interesse em investigar a performance de classificadores construídos a partir de features texturais, oriundas de Gráficos de Recorrência.

CAPÍTULO

3

DSP-Class

Este capítulo apresenta a proposta denominada DSP-Class (*Data Stream Preprocess - Classification*), para classificação de séries temporais utilizando descritores de textura em gráficos de recorrência (RPs) e Decomposição em Modo Empírico (EMD). Dados os conceitos necessários apresentados no Capítulo 2, aqui são reveladas as etapas envolvidas da abordagem, sendo: aplicação da decomposição de modo empírico (Subseção 3.1.1), análise da série temporal (Subseção 3.1.2), reconstrução no espaço fase (Subseção 3.1.3), emprego dos descritores de texturas (Subseção 3.1.4) e, por fim, a etapa de aprendizado (Subseção 3.1.5).

3.1 DSP-Class

Inspirado na área de sistemas dinâmicos, processamento de sinais e análise textural de imagens, a abordagem proposta visa quantificar a disposição dos pixels e a intensidade do nível de cinza dos RPs, a fim de utilizá-las como features para algoritmos de classificação da área de aprendizagem de máquina (AM).

Seus passos independentes podem ser aplicados em séries temporais de origens variadas, como aquelas oriundas de sensores de luz, de som e inclusive aquelas derivadas de logs de sistemas computacionais. São utilizadas séries temporais discretas e de mesmo comprimento, como aquelas da Figura 3.2 do conjunto de dados *OSULeaf*.

O diagrama esquemático da Figura 3.1 mostra a aplicação do DSP-Class em fluxos de dados de áudio. O primeiro passo realiza a decomposição em modo empírico dos fluxos. O segundo passo analisa a série temporal quanto às

dimensões de separação e embutida. No terceiro passo ocorre a reconstrução no espaço fase e formação do RP. No quarto passo são utilizados descritores de textura no RP para extração das features. No quinto, e último passo, aplicam-se algoritmos de AM parar construção de um classificador de séries temporais.

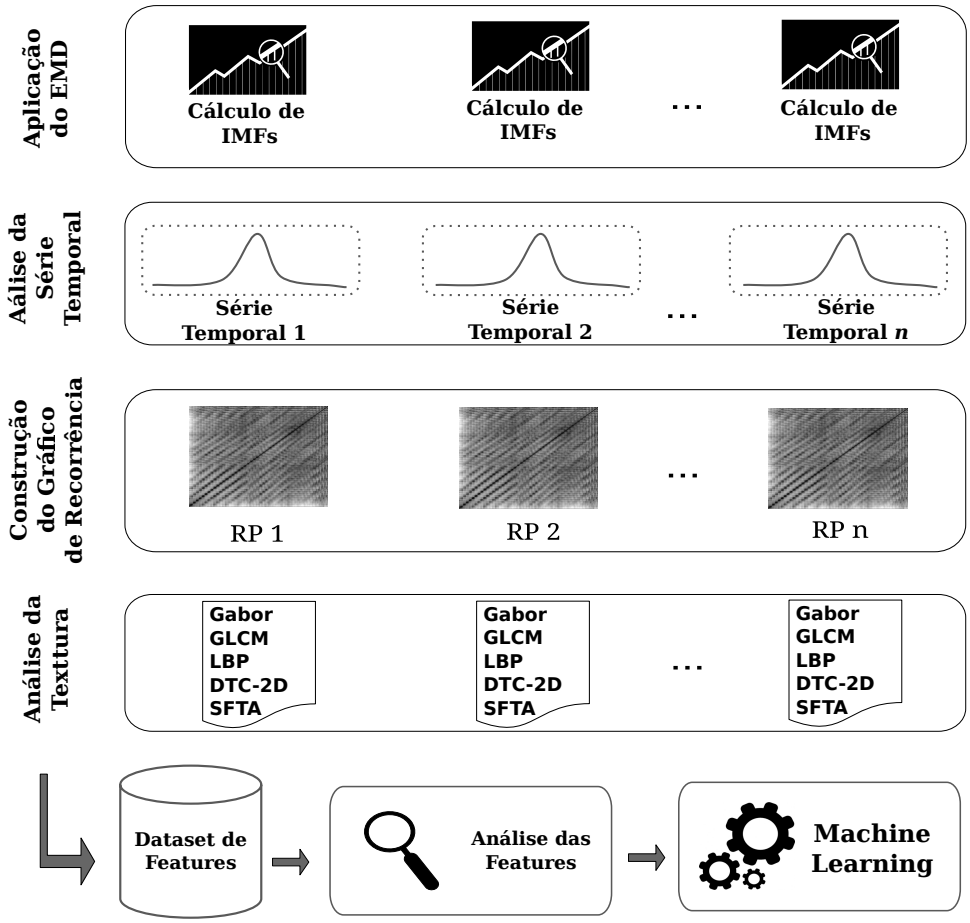

Figura 3.1: Diagrama esquemático da abordagem de pré-processamento de séries temporais *DSP-Class*. Contemplam 5 passos: Aplicação do EMD, Análise da Série Temporal, Construção do RP e Análise de sua textura e por fim, a etapa de seleção de features e aprendizado.

3.1.1 Aplicação da Decomposição de Modo Empírico

A aplicação da decomposição de modo empírico ocorre para separação dos componentes estocásticos e determinísticos presentes nos fluxos de dados (Rios e De Mello, 2013). A abordagem proposta vale-se da metodologia descrita por Rios e De Mello (2013), na qual mede-se o nível de determinismo presente em cada IMF, caso ele seja maior que um pré-definido *threshold*, o IMF em questão é reservado para compor o componente determinístico. Aqueles que não atinjam este limiar, farão parte do componente estocástico.

Nos experimentos realizados, uma análise do número de IMFs extraídos das séries originárias do bater de asas de insetos é feita, variando-se o *threshold*

com início em 0,4 e final em 0,95. Também é relacionada a influência estocástica/determinística com a acurácia obtida por classificadores utilizando as features extraídas do RP, ora apenas do componente determinístico, ora apenas do estocástico. Dos domínios analisados, alguns resultados demonstram que o uso desta abordagem é satisfatório, em outros, a DSP-Class exige uma parametrização refinada.

3.1.2 Análise da Série Temporal

Esta etapa de processamento, dentro do contexto de sistemas dinâmicos, consiste na análise da série temporal. Envolve a quantificação das dimensões de separação (τ) e embutida (m). Após a obtenção destas dimensões, uma série temporal pode ser reconstruída em seu espaço fase, na qual o componente temporal é removido e regressões podem ser realizadas a fim de compreender tendências (de Mello, 2009).

A identificação de τ ocorre pela técnica AMI (Fraser e Swinney, 1986) e a definição de m acontece pela técnica FNN (Kennel et al., 1992). Além destas dimensões, é necessário estipular um raio ε de vizinhança, normalmente definido empiricamente como metade do desvio padrão da série temporal (Marwan et al., 2007). Com estas informações, Marwan et al. (2007) apresentam uma técnica para reconstruir uma série temporal no espaço fase, chamada Gráficos de Recorrência (*Recurrence Plot - RP*).

A identificação destes parâmetros é necessária para compreender estatisticamente estados recorrentes do fluxo de dados, neste caso representados por séries temporais, como discutido no Capítulo 2.

3.1.3 Construção do Gráfico de Recorrência

Um RP descreve os estados recorrentes no espaço fase por meio de pontos, interpretados neste trabalho como pixels, devido a necessidade dos descritores de textura serem aplicados em imagens. Os padrões de recorrência são regularidades frequentes associadas a comportamentos interessantes, úteis para classificação (Silva, 2014). Seja a Figura 3.2 com exemplos de RPs construídos a partir da amostra de 6 séries temporais de 3 possíveis classes do dataset *OSULeaf* (Classe A, B e C).

É possível observar nos RPs forte semelhança de sua aparência quando são da mesma classe (Coluna 1, 2 e 3), ainda sendo possível distingui-los com uma simples inspeção visual. Características como a homogeneidade dos pontos, os tons de cinza e a formação de linhas verticais e diagonais, permitem agrupá-los em suas respectivas classes. Para cada série temporal, constrói-se um RP a fim de extrair estas features.

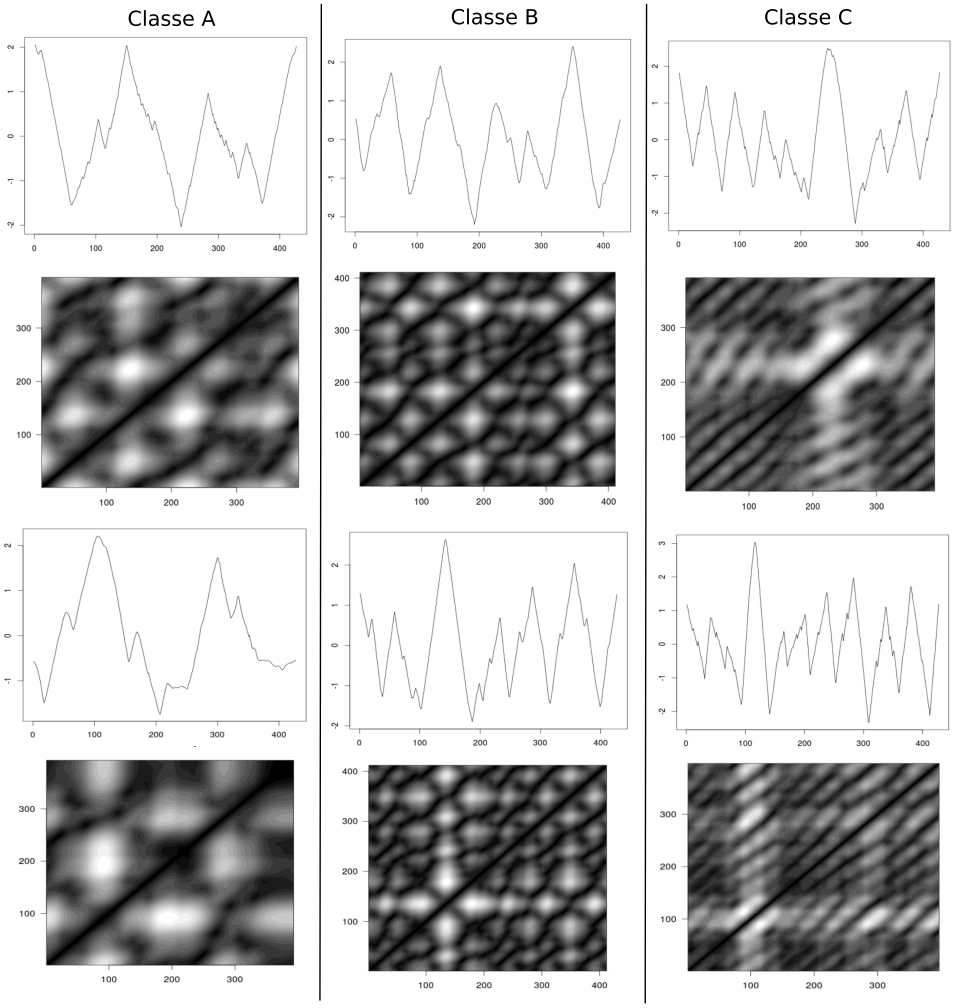

Figura 3.2: 6 RPs de 3 classes (A, B e C) e suas respectivas séries temporais do dataset OSULeaf.

O tamanho de um RP ($n \times n$), no caso das séries da Figura 3.2 com $n = 427$, está diretamente ligado à quantidade de eventos presentes no fluxo de dados observado. A abordagem DSP-Class permite ser adaptada para séries com diferentes comprimentos, ou seja, fluxos de tamanhos variados ($n = 100, 150...$), inclusive para janelamento em fluxos de dados contínuos. No entanto, esta faculdade será contemplada em trabalhos futuros.

Independente do tamanho da séries, após a etapa de construção de um RP, ocorre a quantificação de sua aparência por algoritmos capazes de descrever a textura de imagens.

3.1.4 Análise da Textura

Esta seção descreve algoritmos utilizados para quantificar a textura de um RP. Sua análise ocorre com o uso de metodologias que interpretam uma imagem segundo uma regra, como discutido anteriormente, seja por meio de propriedades estatísticas (GLCM, Gabor) ou pontos de interesse (DTC-2D). Cada descritor de textura de imagem exige uma parametrização, normalmente um

ângulo, uma distância (GLCM) ou uma escala (Gabor). Tais parâmetros exigem uma configuração empírica para maximizar a descrição das imagens pela análise de suas texturas.

Estes parâmetros determinam a quantidade de informação presente no conjunto de dados, por exemplo, ao determinar um raio maior para construir um histograma de pixels (uso da DCT-2D), a área utilizada para compor o histograma também é maior, consequentemente, a descrição do local revela-se com maior precisão. A escolha dos parâmetros relaciona-se com a capacidade de processamento e o tempo de resposta exigido para cumprir a tarefa.

Análogo a um RP, para uma imagem qualquer, os argumentos utilizados nos algoritmos descritores de textura quantificam o nível de detalhe de um ambiente, traduzem cenários, e ainda, permitem compactar imagens mais satisfatoriamente (Apatean et al., 2008; Costa et al., 2012; Pennebaker e Mitchell, 1993). Sua escolha ocorre de maneira empírica.

Para o descritor de textura GLCM, Haralick et al. (1973) o definem como um método para contabilizar a frequência da ocorrência de 2 pixels separados por uma distância considerando um ângulo. Estes dados são armazenados em uma matriz, chamada de matriz de co-ocorrência. Os parâmetros utilizados pelo descritor GLCM são descritos no Capítulo 4, sendo investigadas distâncias $d = \{1,2,3,4,5\}$ e ângulos $\Theta = \{0°,45°,90°,135°\}$.

Outro descritor de textura utilizado na abordagem de pré-processamento DSP-Class chama-se Filtro de Gabor. No processo de classificação de imagens, Pollen e Ronner (1983) iniciaram pesquisas biológicas de como o ser humano percebe e identifica características texturais do mundo real. Os autores concluíram que a classificação e diferenciação de objetos reais, ocorre através da escolha de uma orientação, de uma frequência e de uma dimensão espacial.

O descritor de textura DTC-2D utilizado na abordagem DSP-Class, vale-se da Transformada Discreta de Cosseno, em especial a bidimensional para extrair as features de um RP. Utilizando um RP em escala de cinza, ele é dividido em blocos de tamanho 16×16 pixels. Para cada bloco, quantifica-se a ocorrência de variações de brilho, sendo que áreas que ocorrem muitas variações de brilho são armazenadas com menor precisão, enquanto áreas que possuem poucas variações são armazenadas com maior precisão. Este processo chama-se *Quantização*, ligado diretamente à qualidade de compactação da imagem (Pennebaker e Mitchell, 1993).

Cada descritor de textura cria um vetor de atributos que caracteriza o RP através dos descritores GLCM, DCT-2D e filtros de Gabor. A abordagem DSP-Class concatena-os criando um único vetor. Em seguida, cada vetor obtido é inserido no conjunto para treinar e testar o classificador de séries temporais. Técnicas de AM são utilizadas para seleção das features mais representativas

(úteis) a fim de obter um ótimo desempenho de classificação.

3.1.5 Seleção de Features e Algoritmos de AM

Após analisar a série temporal para reconstruí-la no espaço fase e extrair features texturais do RP, o último passo consiste em utiliza-las para inferir um modelo classificador de séries temporais.

A tarefa de classificação de séries temporais presente neste trabalho pode ser representada como um mapeamento $f : X \rightarrow Y$, sendo X o domínio de séries temporais e Y as possíveis classes mapeadas por uma função f (Last et al., 2001). Sua construção dá-se por técnicas de AM com os algoritmos SVM e C5.0, frisando a necessidade do mapeamento f ocorrer evitando tanto overfitting quanto underfitting. O mapeamento f deve ser flexível o suficiente para descobrir a classe correta de uma série temporal.

Neste contexto de classificação de séries, a literatura apresenta o uso de diferentes técnicas para realizar a análise e atribuição de um rótulo (classe) à uma série (classificação), seja comparando eventos presentes nos fluxos, como por exemplo o uso do DTW-Euclidiano realizando distorções no eixo temporal para encontrar a similaridade máxima entre segmentos temporais.

Estas técnicas caracterizam-se por utilizar uma medida de distância para encontrar a melhor semelhança entre trechos do fluxo, apresentando resultados estatisticamente satisfatórios em alguns domínios de aplicação (Keogh et al., 2006). No entanto, outras abordagens mostram desempenho classificatório superior no domínio temporal, quando este é caracterizado por muitas observações e fluxos de dados ruidosos, como por exemplo no domínio musical.

Buscar a similaridade em fluxos de dados de áudio pode ser uma tarefa custosa devido o número de observações[1]. Diante disso, abordagens de extração de features têm sido aplicadas para caracterizá-los. As features extraídas devem ter alto poder discriminatório, sendo capaz de representar bem a série temporal e oferecer boa distinção entre as possíveis categorias.

Silva (2014), ao utilizar o algoritmo SVM com kernel RBF para classificação e utilizando features extraídas diretamente do sinal (atributos mel-cepstrais), obteve resultados superiores quando comparado ao algoritmo 1NN-DTW ou 1NN-Euclidiano para a classificação de sons de insetos. Assim como em outros trabalhos relacionados, soluções ótimas são alcançadas extraindo características das séries temporais e armazenando-as em vetores de features, para em seguida utilizá-las nos algoritmos de AM, como SVM e C5.0.

Este último passo da abordagem DSP-Class, compreende o uso das features extraídas no passo anterior para identificar as mais representativas no

[1] Por exemplo, um segundo de música em CD possui 44.000 observações.

conjunto de treino, além de induzir um classificador de fluxo de dados. A seleção das features mais discriminantes é alcançada por técnicas baseadas em Filtros e também as chamadas Wrappers, detalhadas no Capítulo 2. Elas buscam eliminar as features redundantes, diferente de outras técnicas que buscam criar novos atributos a partir de suas combinações[2].

Após a seleção das features, a indução do modelo classificador é investigada utilizando os algoritmos de aprendizagem SVM e o C5.0 nas features previamente selecionadas. A escolha destes algoritmos de AM é justificada pelos resultados satisfatórios no contexto de classificação, sendo utilizados por outras pesquisas da área permitindo a comparação dos resultados com features oriundas dos RP. O primeiro algoritmo de AM possui características de otimização matemática enquanto o segundo, redução da entropia e construção de uma árvore de decisão, ambos estatísticos (Cortes e Vapnik, 1995; Quinlan et al., 2008).

Com interesse de provar a eficiência do pré-processamento DSP-Class e ainda confrontar seu desempenho classificatório com outras abordagens, a aferição do modelo classificador é feita utilizando diversas métricas, dentre elas a Acurácia, o F-score e o teste de concordância Kappa.

Interpretado como imagem, um RPs pode produzir imagens com 64 MB, dependendo do formato, da compressão e densidade dos pixels durante a fase de criação da imagem. A complexidade da abordagem é $O(n^2)$, relacionado diretamente ao tamanho do RP e a necessidade de analisar a matriz completamente para computo das features. O Capítulo 4 descreve diversos experimentos com séries de 166 a 8.000 eventos[3].

Como a abordagem DSP-Clas realiza a extração de características com diferentes descritores de textura, totalmente independentes, o problema apresenta características paralelizáveis, permitindo sua adaptação para execução em múltiplos cores ou diferentes estações de trabalho. Inclusive, a extração de características e a construção do modelo classificador de séries temporais pode ser modificada para fluxos contínuos infinitos e janelamento adaptativo, sendo para séries temporais com tamanhos diferentes ou aquelas em que o número de eventos são infindáveis.

Os experimentos apresentados no Capítulo 4 buscam analisar a abordagem DSP-Class sob diferentes aspectos, seja investigando a influência do aumento da quantidade de descritores de textura ou seu uso em diferentes domínios de aplicação (ex.: músicas e processamento de sinal). Também é apurado o emprego de técnicas de seleção de features e algoritmos de aprendizagem.

[2]Exemplo de técnica: PCA.

[3]Experimento realizado com segmentos temporais de 1 segundo representando trechos de músicas.

CAPÍTULO

4

Resultados Experimentais

Neste capítulo são apresentados experimentos realizados com séries temporais de diversos domínios. Para cada série, é executado o pré-processamento DSP-Class de coleta dos atributos texturais originados pela análise do RP, descritos no Capítulo 2. Utilizando aprendizagem supervisionada, os resultados apontam que é possível construir um classificador de séries temporais adotando tais atributos com relevante poder classificatório, superando algoritmos tradicionais da área que utilizam uma medida de distância (1NN) e abordagens que exploram outras features de séries temporais para construir um classificador.

Também é realizada uma análise da influência determinística presente em cada série temporal com o objetivo de relacioná-la com a taxa de acertos. Esta investigação dá-se pelo cálculo do determinismo médio (*DET*) presente nas séries, utilizando o pacote disponível na linguagem *R* chamado *crqa* (Coco e Dale, 2014; R Core Team, 2015).

Experimentos inciais com séries provenientes do repositório UCR (Keogh et al., 2006) demonstram que a abordagem DSP-Class é excelente naquelas cuja influência estocástica ou determinística é predominante, enquanto o desempenho do pré-processamento DSP-Class é menor para aquelas em que não existe a dominância estocástica nem a determinística, ou seja, apresenta níveis de similares determinismo e estocasticidade. Um exemplo pode ser visto nos datasets *Crickets* (Figura 4.1), com uma taxa[1] de determinismo médio de $42,66\%$ e estocasticidade em $57,33\%$.

Para o cálculo do número de dimensões embutidas e de separação, utilizou-se o pacote *RTisean*, implementado em *R* (R Core Team, 2015), enquanto para

[1] Taxas obtidas utilizando o pacote *crqa*.

a extração de features, utilizou-se a ferramenta *Matlab* (MathWorks, 2015).

Seja um conjunto de m séries temporais com n eventos cada, f features são extraídas de cada série através de descritores de textura, como: o filtro de Gabor, GLCM ou DTC-2D (discutidas no Capítulo 2), para compor um dataset de features (Figura 3.1).

Após a coleta de features, inicia-se a fase de Aprendizado de Máquina (AM) a partir das features extraídas utilizando os pacotes: *caret* e *e1071* (Kuhn, 2015; Meyer et al., 2014), além de outros como *perfmeas* e *FSelector* para identificação das melhores features e cálculo do desempenho classificatório. A parametrização desta fase é realizada automaticamente, a fim de maximizar a acurácia apenas para o conjunto de treino, reservando o de teste. Ambos são construídos de maneira aleatória a partir da população original, porém, de maneira estratificada[2].

As métricas utilizadas para avaliar o classificador são a acurácia média, o f-score médio, e o índice Kappa, todos coletados a partir de 15 ou 30 repetições dos experimentos utilizando os algoritmos de aprendizado SVM ou C5.0, para posterior cômputo de suas médias. Para cada experimento, criam-se novos conjuntos de treino e teste, a fim de evitar que os classificadores de séries temporais sejam construídos sempre com os mesmos exemplos e em respeito ao Teorema do Limite Central (Billingsley, 2013).

Os experimentos descritos neste capítulo, detalhados em cada seção, são realizados de maneira encadeada e evolutiva, demonstrando o desempenho do DSP-Class em diferentes domínios temporais e mesclando configurações.

Examina-se a influência determinística e estocástica na Seção 4.1, o desempenho em séries com baixa taxa de determinismo e o acréscimo de descritores de textura na Seção 4.2. Na Seção 4.3, são investigadas técnicas de seleção de atributos. Por fim, na Seção 4.4, o uso da técnica de decomposição em componentes estocásticos e determinísticos, apoiados pela EMD, é investigado como apoio, ocorrendo antes do aprendizado.

4.1 Experimento com dados do repositório UCR

Este experimento inicial tem como propósito validar a possibilidade da construção de um classificador de fluxo de dados discretos, representados como séries temporais, utilizando o pré-processamento DSP-Class. Em seguida, relaciona-se a acurácia obtida pelo classificador no conjunto de teste com a taxa de determinismo médio das séries de cada dataset utilizado (Tabela 4.1).

Os dados desta pesquisa são provenientes do repositório *UCR Time Series*

[2]Preserva-se da distribuição de classes.

dataset	# treino	# teste	comprimento	# classes
Beef	30	30	470	5
Chlorine	467	3.840	166	3
Coffee	28	28	286	2
CricketX	390	390	300	2
CricketY	390	390	300	2
CricketZ	390	390	300	2
Diatom.	16	306	345	4
Fish	175	175	463	7
OliveOil	30	30	570	4
OSULeaf	200	242	427	6
Trace	100	100	275	4

Tabela 4.1: Detalhes dos datasets UCR (Keogh et al., 2006).

Archive[3]. Somam ao todo 11 datasets gentilmente cedidos por *Eamonn Keogh*, professor da Universidade de Riverside, CA. Este repositório é referência como benchmark para testar algoritmos de classificação de séries temporais, possui séries rotuladas, discretas e de mesmo tamanho, facilitando a reprodução dos experimentos e a comparação com outras técnicas de pré-processamento (Baydogan et al., 2013; Silva, 2014; Souza et al., 2014).

Apesar de oferecer 86 conjunto de dados, alguns datasets foram descartados por apresentarem um número muito pequeno[4] de observações, impossibilitando a reconstrução no espaço fase, além de outros serem sintéticos, criados para experimentos específicos.

Os conjuntos de dados utilizados neste experimento são listados na Tabela 4.1. Nela, é apresentado o número de séries utilizadas para treino e teste, a quantidade de classes e o comprimento (número de observações) de cada série temporal. Este repositório oferece os dados já rotulados e separados em conjuntos de treino e teste. Para cada uma, das 30 rodadas de experimentação, estes conjuntos foram reconstruídos preservando sua proporcionalidade original.

Os resultados para este experimento de classificação de séries temporais são apresentados na Tabela 4.2. É utilizado o descritor de textura DCT-2D com blocos de tamanho 16, retornando 256 features. Para o descritor GLCM computam-se d direções, com $d = \{1,2,3,4,5,6\}$, devolvendo 21 features em cada direção, tais como: taxa de autocorrelação dos pixels, contraste, energia, entropia e homogeneidade. A parametrização do filtro de Gabor vale-se de 5 escalas e 6 orientações. Ao todo, são extraídas 646 features após realizar a etapa de seleção de features em cada série temporal utilizando estes descritores de textura.

Neste experimento, é verificado inclusive o desempenho do algoritmo tradi-

[3]Disponível em: http://www.cs.ucr.edu/~eamonn/time_series_data.
[4]Por exemplo, as séries do dataset *Italy Power Demand* possui apenas 24 observações.

cional 1NN, com a distância Euclidiana aplicado diretamente nas séries temporais. Além disso, o resultado obtido por pesquisas semelhantes, com o mesmo conjunto de dados, é alvo de comparação para verificar a utilidade da abordagem proposta (Baydogan et al., 2013; Silva, 2014; Souza et al., 2014).

dataset	1NN-Euc	DSP-Class-SVM	DSP-Class-C5.0	BoW-SVM	TFRP	RPCD	*DET*
Beef	48,89%	50,66%	42,66%	**73,30%**	63,30%	63,33%	60,47%
Chlorine	68,34%	69,27%	63,76%	59,00%	**70,00%**	51,09%	11,53%
Coffee	74,16%	92,62%	90,71%	96,40%	96,43%	**100,00%**	30,77%
Cricket_X	61,20%	53,84%	51,95%	69,70%	64,10%	**70,77%**	43,73%
Cricket_Y	58,41%	50,20%	46,63%	68,70%	63,85%	**73,85%**	41,14%
Cricket_Z	58,66%	52,79%	52,60%	70,50%	63,33%	**70,77%**	44,04%
Diatom.	95,08%	93,05%	80,87%	88,90%	92,48%	**96,41%**	48,92%
Fish	80,26%	72,97%	70,19%	**97,10%**	88,00%	87,43%	47,00%
OliveOil	79,19%	67,74%	58,96%	76,70%	**86,67%**	83,33%	52,00%
OSULeaf	60,61%	74,77%	76,65%	84,70%	**92,98%**	64,46%	32,00%
Trace	86,51%	**100,00%**	97,13%	**100,00%**	—	—	80,00%

Tabela 4.2: Acurácia obtida utilizando a abordagem DSP-Class com os algoritmos SVM e C-5.0. Também são mostrados resultados logrados por pesquisas semelhantes (Seção 2.6). Aqueles em destaque exibem o melhor desempenho.

A princípio, investiga-se qual seria a acurácia obtida utilizando os algoritmos SVM e C5.0 com a abordagem DSP-Class, chamada DSP-Class-SVM quando empregado o algoritmo SVM e DSP-Class-C5.0, caso utilize o C5.0. A parametrização destes algoritmos de AM foi realizada utilizando a biblioteca *caret* (Kuhn, 2015), efetuando uma busca em grade (*grid-search*) para maximizar a acurácia do classificador[5] no conjunto de treino, reservando o de teste.

A escolha dos algoritmos de aprendizagem SVM e C5.0 decorre do levantamento bibliográfico (Capítulo 2) apontá-los como meio eficiente para construção de classificadores . Ademais, seu uso permite avaliar as features texturais relacionando a acucurácia obtida pela abordagem DSP-Class com pesquisas semelhantes.

A partir da Tabela 4.2 verifica-se um bom desempenho classificatório (5/11) de ambas abordagens, quando comparado ao 1NN aplicado diretamente na série temporal, ou seja, sem realizar o pré-processamento, com acurácia de pelo menos 90% para os datasets *Coffee*, *Diatom* e *Trace*, com 95% de nível de confiança.

Em especial, para o conjunto de séries temporais do dataset *Trace*, a acurácia, o índice Kaapa e o F-score obtido pela abordagem DSP-Class-SVM é de 100% para todas estas métricas, enquanto para o dataset *Coffe* é de 92,62%, 85,05% e 46,02% e para o dataset *Diatom* é de 95,08%, 93,08% e 90,43% respectivamente.

Entretanto, dentro do contexto de classificação de séries temporais, investigando pesquisas semelhantes, como BoW-SVM (Baydogan et al., 2013), TFRP (Souza et al., 2014) e RPCD (Silva, 2014), que utilizam o mesmo conjunto de

[5]Construído utilizando a técnica de 5-*repeated* 3-*folds-cross-validation*.

dados e aplicam técnicas de pré-processamento diferentes, notam-se resultados melhores aos obtidos neste experimento para a maioria dos conjuntos de séries temporais. Inclusive, o algoritmo 1NN supera a abordagem proposta em alguns casos.

Deste modo, investigando a origem de tal desempenho aquém do esperado, verificou-se a relação entre a acurácia obtida pela abordagem DSP-Class e a taxa de determinismo, com objetivo de relacionar a influência estocástica e a determinística com taxa de acertos do classificador. Utilizou-se a métrica *DET* concebida pelo pacote *crqa* para realizar a investigação (Coco e Dale, 2014; Marwan et al., 2007). Esta métrica é calculada a partir da *Análise Quantitativa do Gráfico de Recorrência - RQA*, discutida no Capítulo 2. Resultados apresentados na Tabela 4.2, coluna *DET*, refletem a taxa de determinismo médio presente nas séries de cada conjunto.

Relacionando o determinismo médio com a acurácia obtida pelo algoritmo 1NN e as abordagem DSP-Class-SVM e DSP-Class-C-5.0, sumarizados na Figura 4.1, observa-se uma relação de alta acurácia da abordagem DSP-Class com conjuntos de dados que possuem séries temporais que apresentam alta taxa de determinismo ou estocasticidade.

Verifica-se que para os conjuntos que apresentam taxa de determinismo entre $0,41$ e $0,52$, ou seja, para os conjuntos *CricketX*, *CricketY*, *CricketZ*, *Fish*, *Diatom* e *OliveOil*, o desempenho do algoritmo 1NN é superior ao DSP-Class. Quando a taxa de determinismo é superior a $0,52$, ou seja, para os conjuntos *Beef* e *Trace*, ou ainda quando inferior à $0,41$, conjuntos *Chlorine*, *Coffee* e *OSULeaf*, a abordagem DSP-Class apresenta melhores resultados.

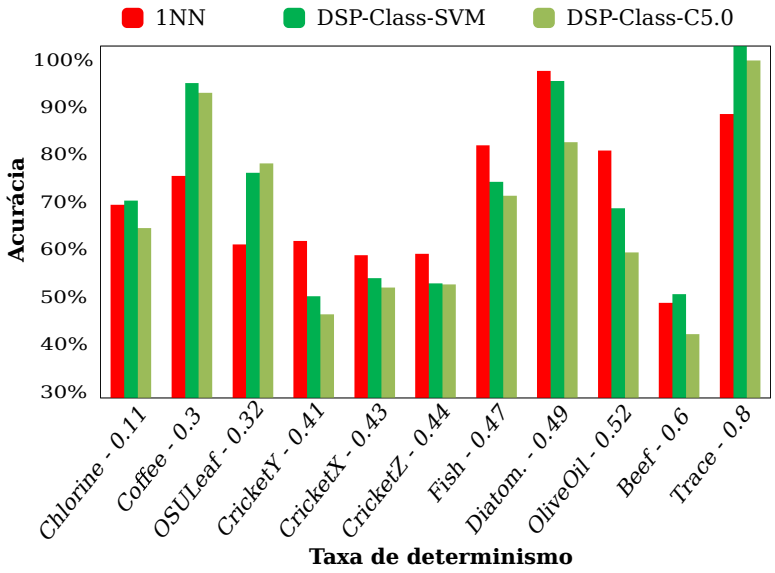

Figura 4.1: Relação entre a Acurácia (eixo y) e a taxa de Determinismo (eixo x) para datasets UCR.

Como este repositório contem séries temporais com ruído praticamente inexistente e aplicáveis a contextos bem específicos, ele foi utilizado apenas

para validar a hipótese desta pesquisa da aplicabilidade do DSP-Class. Além disso, verificou-se a influência determinística-estocástica presente nas séries relacionando-a com o desempenho classificatório, evidenciado uma performance excepcional da abordagem DSP-Class para conjuntos altamente estocásticos ou altamente determinísticos (Figura 4.1).

Tais resultados motivaram a procura por domínios de aplicação do DSP-Class para verificar sua potencialidade em classificação de séries temporais, logo, decidiu-se compará-lo com a abordagem realizada por Pereira (2013) no contexto de classificação de séries temporais oriundas de músicas.

No domínio musical, a discretização de séries temporais oferece muitas observações, além da presença de ruídos, motivando a ratificação do desempenho do DSP-Class. Ademais, Pereira (2013) extrai outras features e compartilha a metodologia de construção do conjunto de dados, permitindo a comparação dos resultados.

4.2 Experimento com gêneros musicais

Neste experimento realizado com séries temporais complexas oriundas de músicas, motivado pelos resultados do experimento da Seção 4.1, investiga-se o poder classificatório da abordagem DSP-Class utilizando os algoritmos de aprendizado SVM (DSP-Class-SVM) e C5.0 (DSP-Class-C5.0) com features obtidas pelos descritores GLCM, DCT, Gabor e agora incluindo aquelas obtidas pelos descritores SFTA e LBP.

Os resultados alcançados superam em 68% o algoritmo estado-da-arte 1NN aplicado diretamente nas séries e também supera aqueles obtidos por Pereira (2013) no mínimo em 18,67%, na qual o autor utiliza outras features para construir o classificador de séries temporais. Features como os coeficientes como os da Transformada de *Fourier*, *Wavelet* e *Cossenos* e expoentes como os de *Lyapunov* e de *Hurst*.

O objetivo principal deste experimento é verificar se os coeficientes dos modelos podem oferecer melhores features, comparadas àquelas obtidas pela análise textural dos RPs dentro do contexto de classificação de séries temporais.

Também é realizado neste experimento um estudo quanto à influência do acréscimo de descritores de textura, para extrair mais features. Como hipótese, ao oferecer mais features para os algoritmos de AM, ocorre um aumento da acurácia do classificador.

A construção deste conjunto de dados é realizada da seguinte forma: a partir de 40 músicas[6] podendo ser apenas de quatro gêneros (Figura 4.2 (A)),

[6] *Free Music Archive* (http://freemusicarchive.org) acessado em 22 de agosto de 2015

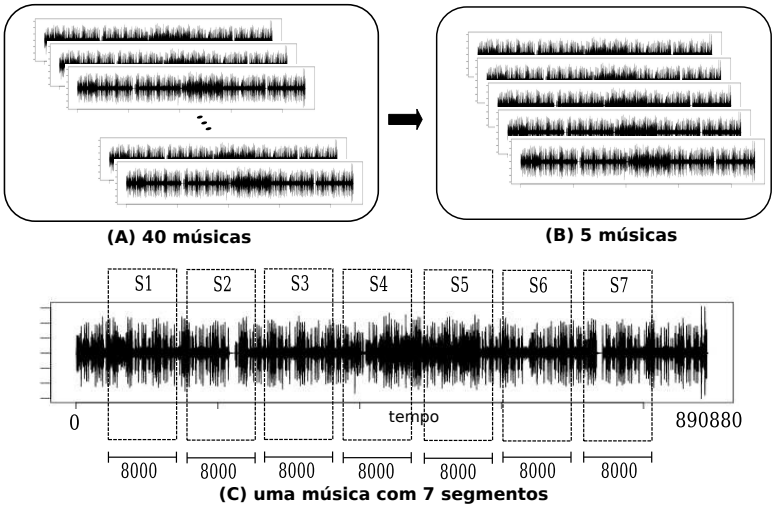

Figura 4.2: Detalhes de um segmento de música.

selecionam-se aleatoriamente 5 músicas (Figura 4.2 (B)). De cada uma destas cinco músicas, também são extraídos aleatoriamente 7 segmentos, representados na Figura 4.2 (C) por *S1*, *S2*, *S3*, *S4*, *S5*, *S6* e *S7*, cada um com 8.000 observações para compor um dataset de 35 séries temporais, descrito na Tabela 4.3.

O rótulo (classe) de cada segmento, é o nome da música de origem, sendo alvo do classificador. O objetivo é identificar de qual música o segmento pertence. Todos os segmentos extraídos podem possuir o mesmo gênero, ou ainda serem de gêneros totalmente diferentes, não sendo o propósito deste experimento identificar qual o gênero musical da série temporal. A Tabela 4.3 na coluna *DET* apresenta a taxa de determinismo médio encontrada nas séries utilizadas no experimento.

Desde que cada música selecionada possa possuir 4.919.976 observações, é necessário realizar o *downsampling* para 8KHz com o intuito dos experimentos serem compatíveis com Pereira (2013). Este passo reduz o número de observações de uma música para até 890.880 eventos (Figura 4.2 (C)).

O classificador deve ser capaz de identificar a música de origem do segmento de áudio, representado por apenas 1 segundo de observação do trecho de música. Este único segundo é amostrado utilizando o pacote *monitoR* (Hafner e Katz, 2015) a uma taxa de amostragem de 8KHz, ou seja, produz 8.000 observações por segundo.

Dataset	**# classes**	**# séries**	**comprimento**	**# treino**	**# teste**	*DET*
Músicas	5	35	8.000	25 séries	10 séries	0,028

Tabela 4.3: Detalhes do dataset de músicas.

4.2.1 Experimento com três descritores de textura

Inicialmente, são utilizados apenas 3 descritores de textura, sendo que para o descritor DCT-2D são coletadas 256 features utilizando a taxa de compressão igual a 16, para o descritor GLCM sendo computadas 21 features a partir de cada direção $d = \{1,2,3,4,5,6\}$, sendo as principais: *autocorrelação*, *energia*, *entropia*, *contraste*, e *homogeneidade*, enquanto para o descritor Gabor é utilizada 1 escala e 2 orientações. Ao todo, são coletadas 646 features após a realização do pré-processamento DSP-Class para cada série temporal utilizando estes descritores.

Realizadas 30 rodadas de experimentação descritas sumariamente na Tabela 4.4, verifica-se para a abordagem DSP-Class-SVM uma acurácia média de 84,33%, um índice Kappa de 80,42%, apontando *Excelente concordância*, precisão de 84,33%, cobertura de 85,28% e F-Measure de 83,3%.

Conjunto de Dados	**DSP-Class-SVM**	**DSP-Class-C**5.0	1**NN**
1	100%	90%	20%
2	70%	50%	20%
3	100%	60%	20%
4	90%	80%	20%
5	80%	60%	20%
6	90%	100%	20%
7	60%	80%	20%
8	90%	90%	20%
9	80%	90%	20%
10	80%	60%	20%
11	100%	90%	20%
12	100%	100%	20%
13	70%	60%	20%
14	90%	70%	20%
15	70%	80%	20%
16	70%	100%	20%
17	70%	80%	10%
18	90%	70%	20%
19	90%	90%	20%
20	90%	60%	20%
21	70%	50%	20%
22	70%	60%	20%
23	90%	70%	20%
24	90%	50%	20%
25	100%	70%	20%
26	90%	70%	20%
27	80%	90%	20%
28	80%	70%	20%
29	90%	70%	20%
30	90%	70%	20%
Acurácia Média	84,33%	74,33%	19%

Tabela 4.4: Resultados de 30 rodadas de experimentos utilizando apenas 3 descritores de textura.

Resultados demonstram um desempenho superior da abordagem DSP-Class-SVM comparado à DSP-Class-C5.0 e muito maior quando utilizado o

algoritmo considerado *estado-da-arte* 1NN. Verifica-se também resultados expressivos quando comparado àqueles obtidos por Pereira (2013), na qual o autor obteve uma Acurácia de 68,33% e Kappa 60,46% utilizando o algoritmo J48.

Diferente de Pereira (2013) que analisa 5 segundos de música (40.000 observações), a abordagem DSP-Class-SVM opera com apenas 1 segundo, 4 segundos a menos. Utilizando somente 8.000 observações, 32.000 não foram necessárias ser analisadas e ainda o classificador alcançou uma acurácia 16% maior.

Motivado pelos resultados obtidos com apenas 3 descritores de textura, investigou-se o desempenho de acrescentar mais features utilizando mais descritores de textura, ainda no contexto de classificação de séries temporais.

4.2.2 Experimento com cinco descritores

Inicialmente, empregam-se três descritores de textura: GLCM, DCT-2*D* e Gabor como descrito na Seção 4.2.1. Verificou-se que para este conjunto de dados formado por séries temporais oriundas de áudio (Tabela 4.3), a abordagem DSP-Class logra bons resultados. Têm-se como hipótese o enriquecimento do conjunto de features, ao utilizar também os descritores SFTA e LBP, a taxa de acertos do classificador é aumentada.

A partir de 21 rodadas de experimentação, a hipótese é verificada com um aumento de 2,67% na acurácia, 2,67% na precisão, 1,62% no recall e 2,16% no F-Measure, alcançando 87%, 87%, 86,9% e 83,13% respectivamente. Infelizmente o índice Kappa sofreu uma redução de 0,67%, alcançando 83,75%, no entanto ainda descreve uma excelente concordância entre os rótulos preditos pelo classificador e os rótulos reais.

Considera-se que o acréscimo destes dois descritores não oferece resultados expressivos para o classificador. Neste contexto, utilizado 24 features produzidas pelo descritor SFTA, seu uso acarreta um aumento de 28% no tempo processamento. O mesmo ocorre com o acréscimo do descritor LBP, com tempo incrementado em 25% para obter 105 features.

Na Seção 4.3 são exploradas técnicas de seleção das features mais representativas para construção do classificador de séries temporais, uma técnica *Filter* e outra *Wrapper*.

4.3 Experimento com vocalização de aves

A identificação de espécies de aves a partir do som que elas emitem tem chamado a atenção de pesquisadores para compreender processos migratórios e calcular o número de animais de uma espécie (Pinho e Marini, 2014). No

meio acadêmico, técnicas de identificação de aves através de suas vocalizações tem sido alvo de competições[7] como incentivo à pesquisa e a preservação do meio ambiente (Goëau et al., 2014).

Motivando este experimento, as séries temporais produzidas no meio ambiente por aves, quando coletadas, normalmente apresentam peculiaridades que dificultam a identificação das aves, como ruídos de insetos, a presença de sons produzidos por outras aves, além da dificuldade imposta pelo meio ambiente de aproximar-se do animal para capturar o som, ocasionando um "sinal pouco audível".

A presença destas características torna o processo de identificação de aves um trabalho custoso, pois algumas ainda tem hábitos que impedem a sua visualização para classificação. Logo, resta coletar sua melodia e identificar sua espécie a partir do seu som.

É utilizado o conjunto de dados disponível no acervo *Xeno-canto*[8] com o propósito de criar um modelo computacional capaz de classificar automaticamente espécies de aves a partir da análise da série temporal produzida por seus cantos. Escolheu-se este repositório por oferecer um dos maiores acervos de vocalizações de aves além de participar de diversos concursos na tarefa de identificação de espécies de aves (Goëau et al., 2014; Lopes et al., 2011a; Vellinga e Planqué, 2011).

O repositório *Xeno-canto* oferece uma base de dados com 9.376 espécies incluindo informações geográficas da gravação do áudio, a autoria e a hora de gravação. Escolheu-se arbitrariamente 3 espécies[9] de aves presentes no pantanal sul-mato-grossense para aplicar a abordagem DSP-Class e construir um modelo computacional capaz de classificar a espécie da ave a partir de sua vocalização.

A Tabela 4.5 apresenta a organização do conjunto de dados utilizando nos experimentos e nas Figuras 4.3, 4.4 e 4.5 exemplos de vocalizações produzidas por cada espécie.

Dataset	**# classes**	**# séries**	**comprimento**	**# treino**	**# teste**	*DET*
Vocalizações	3	90	2.000	63 séries	27 séries	76,17

Tabela 4.5: Detalhes do dataset de vocalizações

Como as gravações apresentam diferentes durações, foi realizada uma segmentação manual de trechos interessantes da gravação para construção do conjuto de dados. Dado um arquivo de áudio, trabalhos encontrados descrevem outras maneiras de identificar o canto de uma ave, por exemplo, de forma

[7] LifeCLEF 2015 Bird task - http://www.imageclef.org/lifeclef/2015/bird.

[8] Disponível em http://www.xeno-canto.org/ acessado em 10 de maio de 2016.

[9] 30 *aves - Cercomacra melanaria*, 30 *aves - Cyanocorax cyanomelas* e 30 *aves - Sporophila hypochroma*.

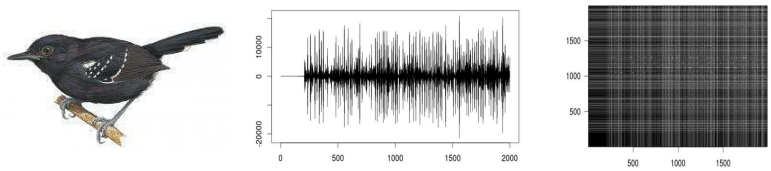

Figura 4.3: Exemplo de série temporal e RP produzido pelo canto da espécie *Cercomacra melanaria*.

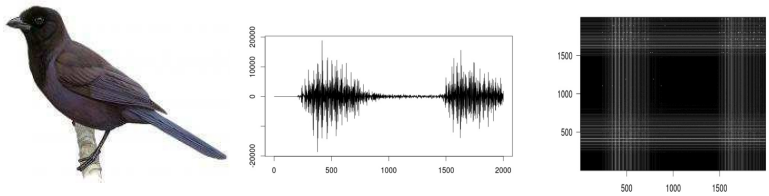

Figura 4.4: Exemplo de série temporal e RP produzido pelo canto da espécie *Cyanocorax cyanomelas*.

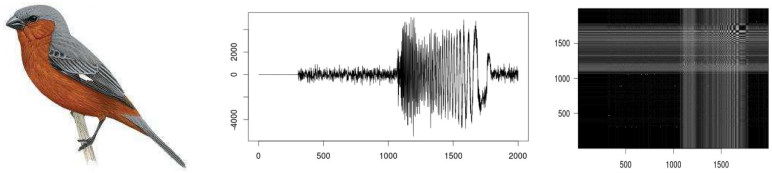

Figura 4.5: Exemplo de série temporal e RP produzido pelo canto da espécie *Sporophila hypochroma*.

não-supervisionada definem uma amplitude mínima (*threshold*) de 10% na série temporal para identificar quando a ave em questão começa e termina seu canto, para em seguida, realiza-se o corte deste trecho (Stowell e Plumbley, 2014). Algumas gravações duram 3 segundos enquanto outras mais de 5 minutos. Empiricamente verificou-se que analisar apenas 0,5 segundo de gravação já produz bons resultados classificatórios para as aves em questão[10].

O **objetivo primário** deste experimento é verificar a aplicabilidade da abordagem DSP-Class no contexto de classificação de séries temporais oriundas de vocalizações de aves. O intuito é verificar o poder classificatório oferecido pela abordagem DSP-Class de maneira a caracterizar as aves utilizando as features texturais coletadas nos RPs. Como **objetivo secundário**, tem-se a utilização de técnicas de seleção de features para melhorar o desempenho dos algoritmos de AM. Também objetiva-se criar um ranking das features e seus descritores que caraterizam mais adequadamente um RP.

Os resultados da aplicação da abordagem DSP-Class-SVM nos dados coletados (Tabela 4.5) são apresentados na Tabela 4.6 a partir de 30 execuções, sendo realizadas com o mesmo dataset, apenas intercalando elementos do conjunto de treino e de teste, para os resultados terem validade estatística de acordo com o Teorema do Limite Central (Billingsley, 2013).

[10]Para cada série temporal, foi realizado o *downsampling* de 2KHz.

Para cada série temporal, é aplicado o descritor de textura Gabor com 2 escalas e 4 orientações, o descritor GLCM para 6 direções, o descritor LBP com raio de tamanho 8 utilizando histogramas normalizados e o descritor DTC-2D com matrizes de 16 unidades. Ao todo, são identificadas 940 features para cada série temporal.

Os resultados classificatórios utilizando o algoritmo SVM (DSP-Class-SVM) após 30 rodadas de experimentação são apresentados na Tabela 4.6. Para a seleção dos melhores parâmetros do algoritmo de AM, foi realizada uma busca em grade e para a construção do modelo, escolheu-se o núcleo *RBF* com controle de treinamento definido para 5 validações cruzadas repetidas 3 vezes.

DSP-Class-SVM					
Repetição	**Acurácia**	**Kappa**	**Precisão**	**Cobertura**	**F-measure**
1	92,59%	88,80%	92,13%	93,27%	92,49%
2	92,59%	88,77%	92,13%	94,44%	92,79%
3	96,30%	94%	96,30%	97,62%	96,80%
4	92,59%	88,89%	92,59%	92,96%	92,58%
5	88,89%	83,40%	89,63%	89,63%	88,89%
6	88,89%	83,33%	88,89%	88,89%	88,89%
7	92,59%	88,84%	92,59%	93,27%	92,29%
8	92,59%	88,46%	91,67%	93,94%	92,19%
9	92,59%	88,68%	93,52%	91,67%	91,83%
10	92,59%	88,66%	91,53%	92,46%	91,83%
11	85,19%	77,12%	84,80%	88,14%	86%
12	92,59%	88,82%	92,13%	92,80%	92,29%
13	92,59%	88,68%	93,52%	91,67%	91,83%
14	92,59%	88,80%	93,27%	92,13%	92,49%
15	92,59%	88,75%	92,21%	91,90%	91,90%
16	88,89%	82,80%	90,74%	90,30%	90,39%
17	92,59%	88,84%	92,59%	93,27%	92,29%
18	96,30%	94,33%	96,30%	97,22%	96,59%
19	92,59%	88,54%	93,52%	92,46%	92,70%
20	92,59%	88,89%	92,96%	92,59%	92,58%
21	88,89%	82,47%	87,04%	89,68%	87,71%
22	88,89%	83,20%	88,80%	90,24%	89,31%
23	100%	100%	100%	100%	100%
24	92,59%	88,89%	92,59%	92,59%	92,59%
25	92,59%	88,41%	94,87%	91,11%	92,44%
26	92,59%	88,75%	92,21%	91,90%	91,90%
27	88,89%	82,98%	89,56%	90,28%	89,77%
28	81,48%	72,39%	81,85%	83,33%	80,47%
29	85,19%	77,31%	82,68%	85%	82,54%
30	92,59%	87,14%	94,07%	89,44%	91,06%
Média	91,48%	86,96%	91,56%	91,81%	91,25%

Tabela 4.6: Experimento com vocalizações de aves utilizando todas as 940 features.

Lopes et al. (2011b), adotando um dataset semelhante com 3 espécies de aves do Estado de Santa Catarina, também formado por 101 vocalizações[11], alcançaram acurácia média de 99,70% em seus experimentos, utilizando o framework *MARSYAS*[12] para extração de features *mel-cepstrais* dos sons emitidos pelas aves e o algoritmo *Naive Bayes* para classificação. Vale ressaltar que as features utilizadas por Lopes et al. (2011b) são equivalentes àquelas explora-

[11] 34 gravações da espécie *Cercomacra tyrannina*, 35 gravações da espécie *Thamnophilus doliatus* e 32 gravações da espécie *Taraba major*

[12] Music Analysis, Retrieval and Synthesis for Audio Signals

das por Silva (2014) com insetos.

Os experimentos aqui realizados demonstram que a abordagem DSP-Class-SVM oferece resultados similares a de outras pesquisas, aplicadas na classificação de aves por meio de suas vocalizações. Sua utilização alcança acurácia média de $91,48\%$, enquanto outra abordagem alcança $99,70\%$.

Investigou-se inclusive a utilização de técnicas de seleção de features para otimizar a fase de construção (treinamento) dos classificadores. Para tal, utilizou-se a técnica de **Ganho de Informação**[13] (IG) para cálculo da entropia do conjunto de features.

A entropia mede a quantidade de informação existente em um conjunto, neste caso, de features texturais. A quantificação deste elemento permite "rankear" as features que oferecem mais informação dentro do conjunto analisado.

Este passo de seleção de features eliminou em média $22,34\%$ das features $(210/940)$. A média de ganho de informação para as features selecionadas manteve-se próximo a $0,6022$, sendo o maior ganho obtido pelas features do descritor GLCM $(0,86)$ e o menor ganho pelo descritor LBP $(0,33)$. A Tabela 4.7 resume as 30 rodadas de experimentos utilizando as features selecionadas pela técnica GI.

DSP-Class-SVM					
Repetição	**Acurácia**	**Kappa**	**Precisão**	**Cobertura**	**F-measure**
1	100%	100%	100%	100%	100%
2	100%	100%	100%	100%	100%
3	100%	100%	100%	100%	100%
4	100%	100%	100%	100%	100%
5	96,30%	93,38%	96,30%	97,92%	96,96%
6	100%	100%	100%	100%	100%
7	100%	100%	100%	100%	100%
8	100%	100%	100%	100%	100%
9	96,30%	94,33%	96.30%	97.22%	96.59%
10	100%	100%	100%	100%	100%
11	96.30%	94.41%	96.30%	96.97%	96.45%
12	100%	100%	100%	100%	100%
13	100%	100%	100%	100%	100%
14	100%	100%	100%	100%	100%
15	100%	100%	100%	100%	100%
16	100%	100%	100%	100%	100%
17	100%	100%	100%	100%	100%
18	100%	100%	100%	100%	100%
19	100%	100%	100%	100%	100%
20	100%	100%	100%	100%	100%
21	100%	100%	100%	100%	100%
22	100%	100%	100%	100%	100%
23	100%	100%	100%	100%	100%
24	96.30%	94.41%	96.30%	96.97%	96.45%
25	100%	100%	100%	100%	100%
26	100%	100%	100%	100%	100%
27	100%	100%	100%	100%	100%
28	100%	100%	100%	100%	100%
29	100%	100%	100%	100%	100%
30	100%	100%	100%	100%	100%
Média	99,51%	99,22%	99,51%	99,64%	99,55%

Tabela 4.7: Experimento com vocalizações de aves utilizando somente as melhores features identificadas pela técnica IG.

[13] Utilizou-se o pacote FSelector (Romanski e Kotthoff, 2014)

A realização da seleção de features neste experimento ofereceu um aumento na acurácia classificatória em 7,74%, aproximando-se ainda mais dos resultados obtidos por outras pesquisas, apenas 0,48% abaixo de Lopes et al. (2011b).

Outra técnica para selecionar as melhores features de vocalizações de aves chama-se **Recursive Feature Elimination** (RFE). Considerada como *Wrapper*, ela consiste em utilizar um algoritmo de aprendizagem a partir dos dados de treino.

O algoritmo de aprendizagem utilizado pela técnica pela técnica RFE chama-se Random Forest (Kuhn, 2015). O alvo de maximização escolhido é a acurácia por meio de 5 validações cruzadas. Na Tabela 4.8 é apresentado o resultado resumido da seleção de features para uma, das 30 rodadas de experimentação. Nela, verifica-se com uma estrela a seleção de apenas 10 features (redução de 98,93% na quantidade de features) para construção do classificador na fase de treino. Utilizando-as, a taxa de acerto mantem-se próximo à 100%.

Número de features	Acurácia	Kappa	Selecionada
2	95,4	92,9	
3	95,3	92,7	
4	95,5	93,1	
5	96,9	95,3	
6	96,9	95,3	
7	96,9	95,3	
8	96,9	95,3	
9	98,5	97,7	
10	100	100	*
11	100	100	
12	100	100	
13	100	100	
14	100	100	
15	100	100	
16	100	100	
17	100	100	
18	100	100	
19	100	100	
20	100	100	
21	100	100	
22	100	100	
23	100	100	
24	98,5	97,7	
25	98,5	97,7	
.	.	.	
.	.	.	
.	.	.	

Tabela 4.8: Seleção das melhores features com a técnica RFE.

Uma outra visualização possível é através da Figura 4.6. A partir dela percebe-se que o acréscimo de outras features é desnecessário na construção do modelo classificador, pois a acurácia mantêm-se constante, entre 98% e 100%, até incluir a 940ª feature.

A melhor feature selecionada é obtida pelo descritor LBP, seguidas por aquelas obtidas pelo descritor GLCM, confirmando o poder descritivo destes descritores, sendo as mesmos indicados pela técnica de seleção de features IG.

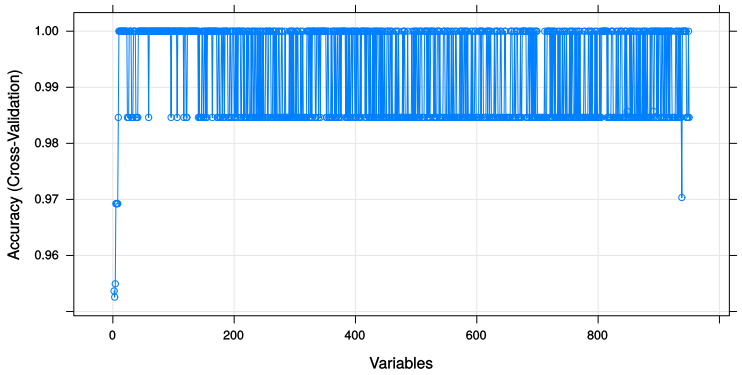

Figura 4.6: Acurácia obtida na seleção de features pelo algoritmo Random Forest utilizando na técnica RFE.

As features obtidas pelos descritores Gabor e DTC-2D não foram consideradas relevantes. Os resultados da aplicação do classificador de séries temporais utilizando a técnica de seleção de features RFE no conjunto de teste, construído com as 10 melhores features, é apresentado na Tabela 4.9.

Após a realização destes experimentos, com resultados sumarizados na Figura 4.7, verifica-se que a aplicação da técnica RFE obteve um desempenho $4,08\%$ superior comparado a utilização de todas as features para construção do classificador. Apesar de apresentar uma acurácia média de $95,56\%$, verifica-se um possível *overfitting* do classificador. No conjunto de treino a acurácia obtida manteve-se em média superior a 98% (Figura 4.6), enquanto no de teste, a acurácia foi menor ($95,56\%$).

De forma semelhante, o uso da técnica de seleção de features IG ofereceu um aumento na acurácia do classificador em $8,03\%$. Demonstrando ser mais eficiente que o uso da técnica RFE em $3,95\%$ oferecendo um acurácia de $99,51\%$.

A aplicação das técnicas de seleção de features revela, neste contexto, a importância do descritor de textura LBP e GLCM para construção do modelo classificador de séries temporais. O ranking de features produzido por ambas técnicas demonstram a baixa influência dos descritores Gabor e DTC-2D na construção do classificador. Estes resultados fomentam o aprofundamento das técnicas de extração e seleção de features, para melhorar ainda mais os resultados.

4.4 Experimento com dados de insetos

Este experimento tem como **objetivo primário** avaliar o desempenho classificatório de algoritmos de AM utilizando a abordagem DSP-Class nos dados provenientes da tese de doutorado de Silva (2014). A séries são representadas

DSP-Class-SVM					
Repetição	**Acurácia**	**Kappa**	**Precisão**	**Cobertura**	**F-measure**
1	100.00%	100.00%	100.00%	100.00%	100.00%
2	100.00%	100.00%	100.00%	100.00%	100.00%
3	96.30%	94.44%	96.30%	96.67%	96.28%
4	88.89%	83.12%	87.96%	90.74%	88.50%
5	96.30%	94.33%	96.30%	97.22%	96.59%
6	92.59%	88.61%	94.44%	93.94%	93.64%
7	96.30%	94.40%	95.83%	96.97%	96.19%
8	100.00%	100.00%	100.00%	100.00%	100.00%
9	100.00%	100.00%	100.00%	100.00%	100.00%
10	100.00%	100.00%	100.00%	100.00%	100.00%
11	96.30%	94.43%	96.67%	96.67%	96.49%
12	88.89%	82.98%	88.50%	90.30%	89.14%
13	100.00%	100.00%	100.00%	100.00%	100.00%
14	96.30%	94.41%	96.30%	96.97%	96.45%
15	100.00%	100.00%	100.00%	100.00%	100.00%
16	96.30%	94.30%	95.24%	97.22%	95.99%
17	96.30%	94.40%	96.97%	95.83%	96.19%
18	96.30%	94.33%	96.30%	97.22%	96.59%
19	96.30%	94.40%	95.83%	96.97%	96.19%
20	70.37%	54.04%	70.66%	75.24%	70.70%
21	92.59%	88.93%	93.33%	93.33%	92.59%
22	96.30%	94.27%	97.22%	96.67%	96.80%
23	100.00%	100.00%	100.00%	100.00%	100.00%
24	100.00%	100.00%	100.00%	100.00%	100.00%
25	100.00%	100.00%	100.00%	100.00%	100.00%
26	100.00%	100.00%	100.00%	100.00%	100.00%
27	100.00%	100.00%	100.00%	100.00%	100.00%
28	85.19%	77.59%	84.17%	89.74%	84.49%
29	100.00%	100.00%	100.00%	100.00%	100.00%
30	85.19%	77.73%	85.83%	87.50%	85.61%
Média	95,56%	93,22%	95,60%	96,32%	95,61%

Tabela 4.9: Experimento com vocalizações de aves utilizando somente as melhores features identificadas pela técnica RFE.

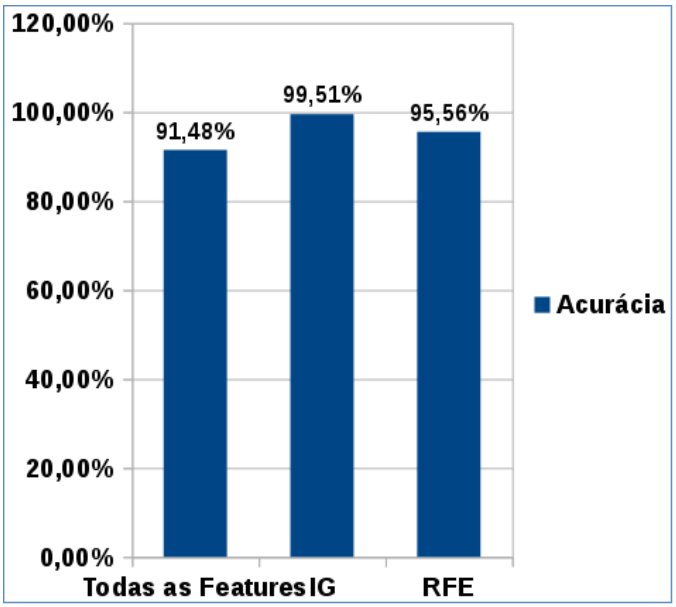

Figura 4.7: Desempenho classificatório médio em 30 rodadas de experimentação com a abordagem DSP-Class utilizando técnicas de seleção de atributos e sem a sua utilização.

por sinais de áudio produzidas pelo bater de asas dos insetos.

Sabe-se que os insetos tem participação na destruição de plantações, gerando perdas na produtividade em lavouras. Sabe-se ainda da existência de

alguns transmissores de doenças, especialmente a Dengue, Ziza e Chikungunya, propagadas pelo mosquito *Aedes*. Simultaneamente são benéficos para o equilíbrio do meio ambiente, servindo inclusive como indicadores de qualidade do ar, da água, do solo e ainda atuando como agentes polinizadores.

A pesquisa desenvolvida por Silva (2014) ocupa-se da coleta de séries temporais produzidas por insetos, ao passarem por um sensor óptico. Quando um inseto passa pelo feixe de luz, um fotosensor (Figura 4.8) captura a variação da luz, resultado da oclusão parcial do feixe causada pela passagem do inseto (Silva, 2014).

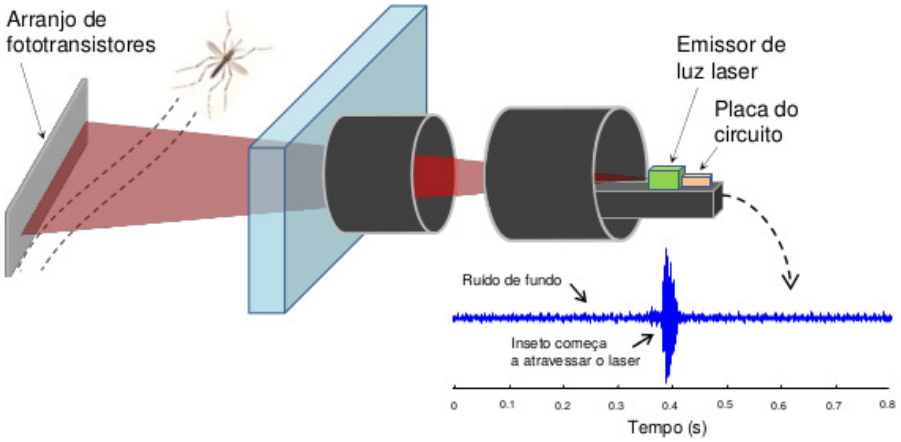

Figura 4.8: Fotosensor (Silva, 2014).

Ao atravessar o feixe de luz, o bater de asas do inseto produz oscilações ópticas, interpretadas como séries temporais. Os dados foram coletados em um ambiente controlado, a uma taxa de amostragem de 16KHz, sendo suficientes para caracterizar os sinais, sendo que os mesmos normalmente estão na faixa de 100Hz a 1KHz. As séries produzidas pelos insetos (Figura 4.9) são rotuladas automaticamente, pois a coleta deu-se de maneira controlada em insetários com a presença de apenas uma espécie.

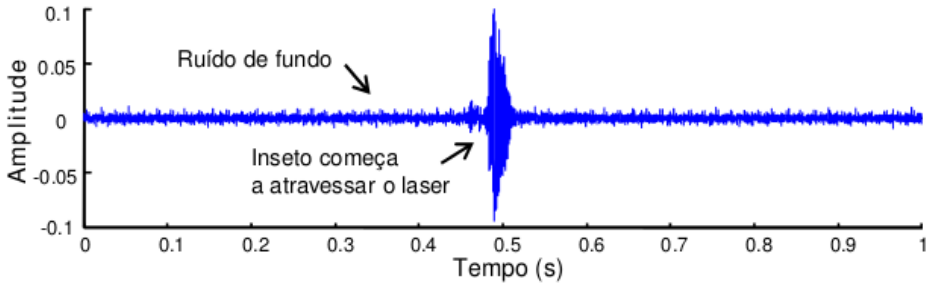

Figura 4.9: Exemplo de série temporal produzida pelo mosquito *Aedes aegypti* ao passar pelo sensor óptico (Silva, 2014).

A Tabela 4.10 apresenta o conjunto de dados utilizado nos experimentos realizados por Silva (2014), sendo os mesmos utilizados nesta pesquisa. Através dela observa-se um desbalanceamento de classes, mantido neste experimento

para permitir a comparação dos resultados. Também preservou-se a distribuição da quantidade de elementos utilizados para treinar e testar o classificador, sendo 33% e 67% respectivamente.

Espécies	**# Treino**	**# Teste**	**# Total**	**Dist. class.** %
Aedes aegypti	1.585	3.171	4.756	26,25
Anopheles gambiae	470	941	1.411	7,79
Apis mellifera	170	341	511	2,82
Cotinis mutabilis	58	114	172	0,95
Culex quinquefasciatus	1.045	2.092	3.137	17,32
Cules tarsalis	1.770	3.539	5.309	29,31
Drosophila melanogaster	259	518	777	4,29
Musca domestica	448	895	1.343	7,14
Psychodidae diptera	233	466	699	3,86
Total	6.038	12.077	18.115	100

Tabela 4.10: Detalhes do conjunto de dados de Insetos.

As séries produzidas pelo sensor possuem diferentes comprimentos, algumas com apenas 500 observações e outras com 2.000, referentes à passagem do inseto pela armadilha. Para aplicação da abordagem DSP-Class neste conjunto de dados, em cada sinal de áudio é identificado o valor cuja amplitude é máxima. A partir dele, um deslocamento é realizado para considerar apenas 300 observações à esquerda e outras 300 observações à direita dele. Logo, todas as séries temporais utilizadas possuem o mesmo tamanho (600 observações). Verificou-se empiricamente que selecionar trechos deste tamanho apresenta bons resultados classificatórios.

Exemplo de um RP produzido a partir de uma série temporal oriunda do inseto *Aedes aegypti* é visto na Figura 4.10. Nela também é mostrada a série temporal original e um esboço do inseto.

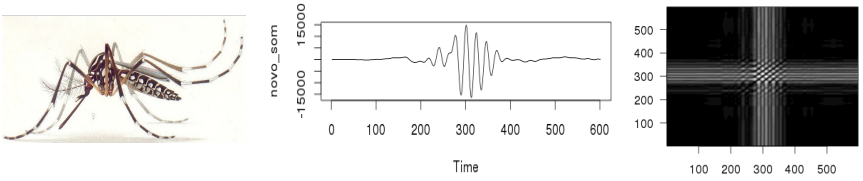

Figura 4.10: Exemplo de RP gerado pelo inseto *Aedes aegypti*.

Na Figura 4.11 é apresentado um exemplo da série temporal produzida pela mosca doméstica ao passar pelo peixe de luz do sensor óptico.

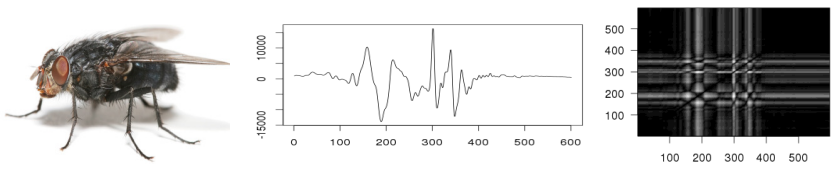

Figura 4.11: Exemplo de RP gerado pelo inseto *Musca domestica*.

Pesquisas realizadas por Rios (2013) e por Ishii e De Mello (2012) demonstram a necessidade de analisar individualmente o componente estocástico e o determinístico presente nas séries temporais. Segundo os autores, esta avaliação individual resulta em previsões com maior taxa de acerto e melhor compreensão do comportamento da série, seja para regressões ou classificações. Como **objetivo secundário**, têm-se a avaliação dos resultados produzidos pelos classificadores quando são construídos a partir das features extraídas dos componentes, comparando-os quando são construídos utilizando as features extraídas da série original.

Como hipótese, tem-se que: *ao separar o componente estocástico do determinístico e aplicar a abordagem DSP-Class individualmente, ocorre aumento na taxa de acerto do classificador quando utilizado apenas um componente em sua construção, comparado ao uso da abordagem DSP-Class na série original com ambos componentes*.

Esta hipótese norteia a investigação para concepção de dois classificadores, um gerado a partir de features do componente determinístico e outro do estocástico. A pesquisa realizada por Rios (2013) compartilha o pacote *TS-Decomposition*, disponível em R, para separar o componente estocástico do determinístico presente em séries temporais.

Este experimento ainda tem por **objetivo terciário**, avaliar a influência do *threshold* na acurácia do classificador. Este parâmetro influencia na quantidade de IMFs que integram o componente determinístico (Rios, 2013). Aumentando-o, a quantidade de IMFs combinados para formar o componente determinístico tende a aumentar, enquanto aqueles destinado a compor o estocástico, diminuem. Logo, investiga-se a taxa de determinismo (*threshold*) e sua relação com a acurácia classificatória utilizando a abordagem DSP-Class.

Na Subseção 4.4.1 são exploradas as séries originais, isto é, sem considerar os componente estocásticos e determinísticos. Nelas, aplica-se a abordagem DSP-Class. Em seguida, na Subseção 4.4.2, são examinados os componentes da séries e sua influência na taxa de acerto do classificador, bem como nível de determinismo empregado para extração deste componentes.

4.4.1 Experimento com as series originais

Neste experimento, o alvo consiste na aplicação do pré-processamento DSP-Class no conjunto de dados formado por séries temporais utilizadas por Silva (2014). Ao todo, são utilizados 5 descritores de textura para construir o dataset de features, sendo aquelas oriundas do filtro de Gabor, GLCM, SFTA, LBP e DCT-2D.

Antes de realizar o passo de aprendizagem, aplicou-se a técnica de seleção de features IG para identificar as mais representativas, em seguida, o algo-

ritmo de aprendizagem SVM foi utilizado (DSP-Class-SVM) para construção do classificador de séries temporais.

Após a realização do pré-processamento oferecido pela abordagem DSP-Class-SVM nas séries temporais e realização de 15 repetições alternando elementos, ora para treino ora para teste, os resultados classificatórios são demonstrados na Tabela 4.11.

DSP-Class-SVM					
Repetição	**Acurácia**	**Kappa**	**Precisão**	**Cobertura**	**F-measure**
1	71,28%	63,77%	65,81%	65,75%	65,21%
2	70,86%	63,37%	64,86%	64,99%	64,57%
3	71,14%	63,74%	65,07%	65,45%	65,03%
4	71,84%	64,56%	65,46%	67,81%	66,40%
5	71,30%	63,88%	65,03%	64,72%	64,56%
6	70,70%	63,08%	64,21%	64,21%	63,84%
7	72,03%	64,76%	66,36%	66,42%	65,99%
8	70,69%	63,12%	64,25%	64,99%	64,21%
9	71,09%	63,52%	63,65%	65,88%	64,25%
10	71,59%	64,16%	65,27%	65,72%	64,99%
11	72,03%	64,64%	63,57%	69,57%	65,78%
12	71,14%	63,52%	62,82%	68,35%	65,01%
13	70,21%	62,48%	64,22%	64,53%	63,66%
14	71,95%	64,63%	63,79%	68,92%	65,88%
15	70,98%	63,40%	64,50%	66,28%	65,11%
Média	71,25%	63,77%	64,59%	66,23%	64,96%

Tabela 4.11: Resultado classificatório da abordagem DSP-Class-SVM no conjunto de dados de séries temporais de insetos.

Os dados apresentados refletem uma acurácia média distante da alcançada por Silva (2014) (87,33%), motivando uma análise mais profunda das séries. Esta investigação dar-se-á separando o componente estocástico do determinístico de cada série temporal, para em seguida, aplicar a abordagem DSP-Class de extração das features e construção de um classificador de séries temporais.

4.4.2 Experimento com o componente determinístico e estocástico

Este experimento utiliza o componente determinístico e o estocástico da série temporal, tratados separadamente, para construção de classificadores de séries temporais. Tem-se como hipótese que, ao utilizar somente um componente, seja determinístico ou estocástico para extração das features oriundas do RP, o classificador de séries obtem maior acurácia classificatória, quando comparado a utilização das séries originais.

Compreende-se os seguintes passos para realização deste experimento: construir dois datasets, um formado por componentes temporais determinísticos e outro por estocásticos, aplicar a abordagem DSP-Class em cada dataset, e por fim, comparar os resultados variando o nível de determinismo.

Um exemplo da ambos componentes é visto na Figura 4.12. Ao variar o *threshold*, o número de IMFs extraídos é alterado, consequentemente ocorrem

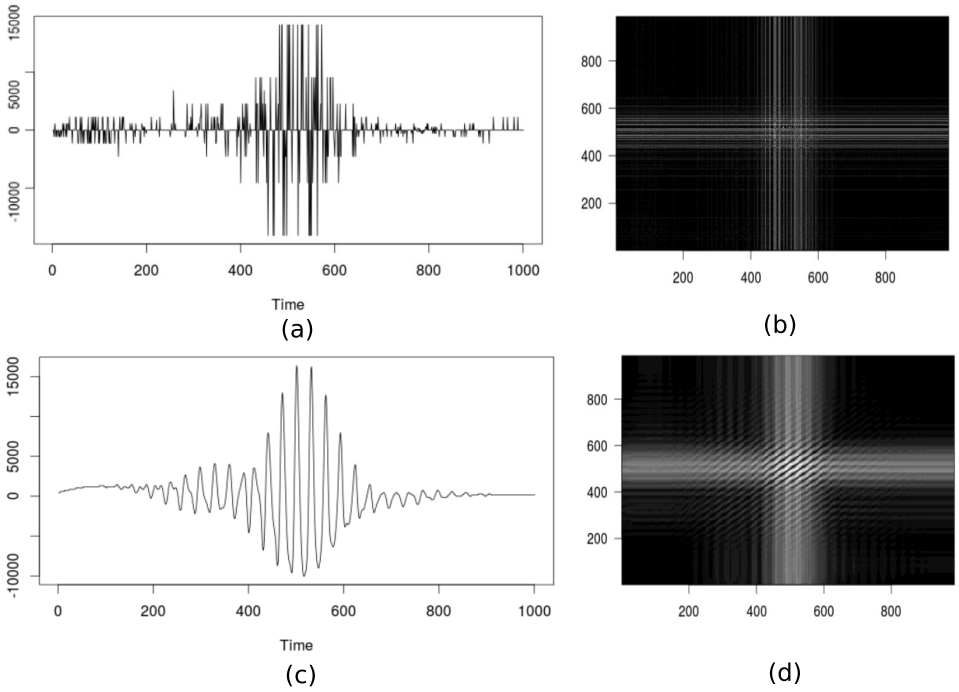

Figura 4.12: Componente estocástico (a) e seu respectivo RP (b). Componente determinístico (c) e seu respectivo RP (d). Taxa de determinismo utilizada de 95% em uma série da classe *Aedes aegypti*.

novos arranjos de estados recorrentes refletidos no RP. Ao elevar este parâmetro, eleva-se também a quantidade de IMFs extraídos das séries temporais para compor o componente determinístico. De maneira análoga, a redução do *threshold* interfere na redução do número de IMFs extraídos das séries para compor o componente estocástico. Para *threshold* entre 40% à 80%, extraem-se até 4 IMFs, enquanto verifica-se até 5 IMFs com *threshold* de 90%, e até 6 IMFs para *threshold* de 95% nos dados utilizados.

Pesquisas demonstram a necessidade de identificar o *threshold* empiricamente, sugerindo $0,95$ como nível de determinismo (DET) ideal para considerar um componente como determinístico (Rios e De Mello, 2013).

Experimentos foram conduzidos variando o *threshold*, iniciando em 40%, com degraus de $0,5$%, até atingir o limiar de 95% na tarefa de construir um classificador de séries temporais oriundas do bater de asas de insetos.

Ao relacionar o determinismo com a acurácia obtida pelo classificador, verifica-se resultados satisfatórios com threshold menores que 70%. Nota-se também que o uso do componente determinístico para construir o RP e extrair as features oferece melhores resultados, quando comparado àquelas extraídas do componente estocástico.

No entanto, os resultados obtidos pela separação dos componentes de séries temporais e o uso da abordagem DSP-Class-SVM, mostrou-se aquém daqueles obtidos por Silva (2014). O autor atingiu acurácia de $87,33$%, superior em $24,96$% quando comparado a esta ao utilizar apenas o componente deter-

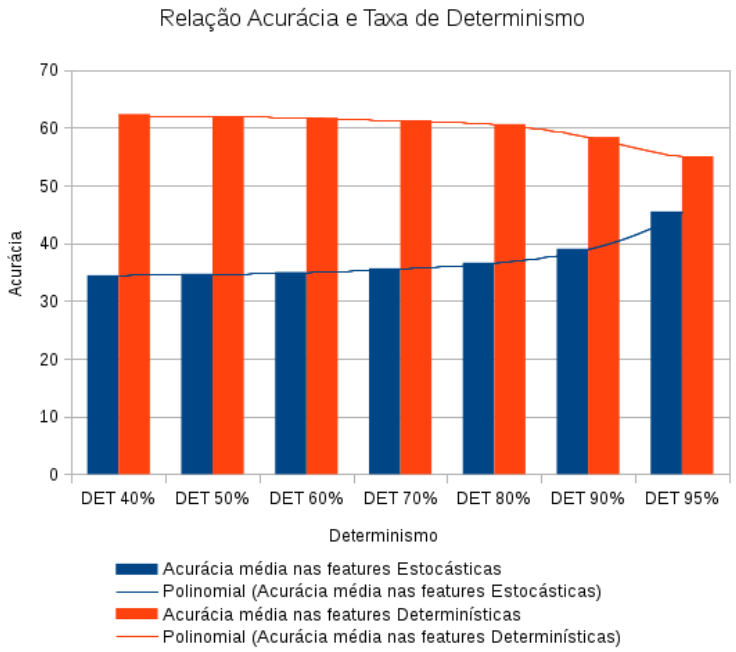

Figura 4.13: Relação da acurácia obtida pela abordagem DSP-Class-SVM com a variação da taxa de determinismo utilizada para separação dos componentes estocásticos e determinísticos nas séries dos insetos.

minístico e threshold de 40% (62,37% de acurácia da abordagem DSP-Class).

Quando comparado aos resultados da Tabela 4.11, na qual são utilizadas as séries originais, os resultados alcançados por Silva (2014) são superiores em 16,08% (71,25% de acurácia obtida pela abordagem DSP-Class nas séries originais).

A quantidade de informação presente no componente estocástico é significativa, pois ao removê-lo, o desempenho da abordagem DSP-Class é reduzido em 6,88% (71,25% − 62,37%), confirmando a quantidade de informação presente no componente estocástico influenciando a acurácia do classificador.

A realização dos experimentos com séries temporais oriundas de insetos demonstraram-se satisfatórios, apesar do desempenho inferior ao esperado. A investigação do uso de componentes da série temporal, ao invés da própria série, permite quantificar a sua influência nos algoritmos de AM. Comprova-se que o uso da série original fornece features mais representativas que as obtidas pelos componentes da séries temporal.

CAPÍTULO

5

Conclusões

A abordagem DSP-Class para extração de características de Gráficos de Recorrência mostrou-se eficaz para séries temporais de diversos domínios. 0s descritores de textura normalmente utilizados em imagens mostraram-se aliados à caracterização de Gráficos de Recorrência.

As features extraídas são capazes de distinguir classes diferentes de séries temporais. Sua aplicação demonstrou-se eficiente para séries que demonstram grande influência estocástica ou grande influência determinística, situação que outras técnicas de classificação expõe resultados inferiores.

Investigou-se o desempenho classificatório no domínio de sons produzidos por seres humanos, aves ou insetos. Para alguns domínios, os resultados atingidos pela abordagem DSP-Class mostraram-se superiores a de outras.

Verificou-se que as features extraídas de RPs são mais adequadas para descrever séries temporais no domínio de gêneros musicais. Quando comparado àquelas obtidas pela análise do sinal, o classificador SVM na abordagem DSP-Class-SVM oferece taxa de acerto $18,67\%$ maior. Além disso, utiliza 4 segundos a menos, necessitando analisar 8.000 eventos ao invés de 40.000[1].

Os resultados alcançados pela abordagem DSP-Class-SVM equiparou-se a outra pesquisa na área de vocalização de aves, logrando $99,51\%$ de acurácia com uso de quanto descritores de textura com o uso de técnicas de seleção de features. Uma delas (RFE) revela que apenas 10 features texturais são necessárias para construir um classificador de séries temporais com acurácia de $95,56\%$ e *F-measure* de $95,61\%$.

Ainda como contribuições deste trabalho, tem-se a publicação de artigos científicos, o primeiro vinculado ao desempenho excepcional no contexto de

[1] A uma taxa de amostragem de 8.000 hertz, ou seja, 1.000 eventos por segundo.

gêneros musicais relacionando níveis de determinismo e estocasticidade, publicação aceita na 31º *ACM Sysposium on Applied Computing*, realizado em Abril de 2016 (Silva e Ishii, 2016). Outros trabalhos científicos a serem publicados em conferências da área decorrem da análise dos diferentes *thresholds* utilizados e sua relação com a acurácia do classificador. Também, os resultados verificados utilizando a abordagem DSP-Class para classificação de vocalizações de aves serão enviados para publicação.

Como trabalho futuro, tem-se o aperfeiçoamento das ferramentas para permitirem o paralelismo, reduzindo o tempo dispendido na extração das features, sendo uma grande dificuldade encontrada. Além disso, propõe-se avaliar o desempenho classificatório das features com outros algoritmos de AM e a adequação desta abordagem para fluxo de dados contínuos em séries temporais.

Ainda como pesquisa decorrente deste trabalho, há a necessidade de investigar a decomposição em componentes estocásticos e determinísticos como apoio a tarefa de classificação. Para o conjunto de dados de insetos analisado, a abordagem DSP-Class mostrou-se aquém do esperado, seja utilizando as features extraídas da serie original ou utilizando as features estocásticas ou determinísticas. Uma investigação decorrente deste trabalho consiste na análise da união destes componentes, eliminando-se o ruído para verificação de um possível aumento na taxa de acertos do classificador.

Referências Bibliográficas

Aggarwal, C. C. e Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data*, páginas 163–222. Springer. Citado na página 27.

Alligood, K. T., Sauer, T. D., e Yorke, J. A. (1997). Chaos: An introduction to dynamical systems. Citado na página 21.

Anderson, J. R., Michalski, R. S., Carbonell, J. G., e Mitchell, T. M. (1986). *Machine learning: An artificial intelligence approach*, volume 2. Morgan Kaufmann. Citado na página 23.

Apatean, A., Rogozan, A., e Bensrhair, A. (2008). Objects recognition in visible and infrared images from the road scene. In *Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on*, volume 3, páginas 327–332. IEEE. Citado nas páginas 3 e 39.

Azevedo Filho, A. (2009). *Introdução à Estatística Matemática Aplicada*. Scotts Valley. Citado na página 9.

Baydogan, M. G., Runger, G., e Tuv, E. (2013). A bag-of-features framework to classify time series. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2796–2802. Citado nas páginas 32, 45, e 46.

Biem, A., Feng, H., Riabov, A., e Turaga, D. S. (2013). Real-time analysis and management of big time-series data. *IBM Journal of Research and Development*, 57(3/4):8–1. Citado na página 31.

Billingsley, P. (2013). *Convergence of probability measures*. John Wiley & Sons. Citado nas páginas 24, 44, e 53.

Box, G. E. P., Jenkins, G. M., e Reinsel, G. C. (2008). *Time series analysis: forecasting and control*. John Wiley & Sons. Citado nas páginas 8, 9, e 10.

Byun, H. e Lee, S.-W. (2002). Applications of support vector machines for pattern recognition: A survey. In *Pattern recognition with support vector machines*, páginas 213–236. Springer. Citado na página 27.

Chino, D. Y. T. (2014). *Mineração de padrões frequentes em séries temporais para apoio à tomada de decisão em agrometeorologia*. PhD thesis, Universidade de São Paulo. Citado na página 7.

Coco, M. I. e Dale, R. (2014). *crqa: Cross-Recurrence Quantification Analysis for Categorical and Continuous Time-Series*. R package version 1.0.5. Citado nas páginas 43 e 47.

Cohen, J. et al. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46. Citado na página 30.

Cortes, C. e Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297. Citado na página 41.

Costa, A. F., Humpire-Mamani, G., e Traina, A. J. M. (2012). An efficient algorithm for fractal analysis of textures. In *Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on*, páginas 39–46. IEEE. Citado nas páginas xv, 3, 18, 19, e 39.

Dash, M. e Liu, H. (1997). Feature selection for classification. *Intelligent data analysis*, 1(1):131–156. Citado na página 24.

de Assis, T. A., Miranda, J. G. V., de Brito Mota, F., Andrade, R. F. S., e de Castilho, C. M. C. (2008). Geometria fractal: propriedades e caracterısticas de fractais ideais. *Revista Brasileira de Ensino de Fısica*, 30(2):2304. Citado na página 18.

de Mello, R. F. (2009). *Sistemas dinâmicos e técnicas inteligentes para a prediçao de Comportamento de Processos: Uma Abordagem Para Otimizaçao de Escalonamento em Grades Computacionais*. PhD thesis, PhD dissertation, Intituto de Cincias Matemáticas e de Computa cao-USP. Citado na página 37.

Fraser, A. M. e Swinney, H. L. (1986). Independent coordinates for strange attractors from mutual information. *Physical review A*, 33(2):1134. Citado nas páginas 10, 11, e 37.

Goëau, H., Glotin, H., Vellinga, W.-P., Planqué, R., Rauber, A., e Joly, A. (2014). Lifeclef bird identification task 2014. In *CLEF2014*. Citado na página 52.

Granitto, P. M., Furlanello, C., Biasioli, F., e Gasperi, F. (2006). Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products. *Chemometrics and Intelligent Laboratory Systems*, 83(2):83–90. Citado na página 27.

Guyon, I. e Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182. Citado na página 24.

Guyon, I., Weston, J., Barnhill, S., e Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422. Citado na página 26.

Gyselinckx, B., Van Hoof, C., Ryckaert, J., Yazicioglu, R. F., Fiorini, P., e Leonov, V. (2005). Human++: autonomous wireless sensors for body area networks. In *Custom Integrated Circuits Conference, 2005. Proceedings of the IEEE 2005*, páginas 13–19. IEEE. Não citado no texto.

Hafner, S. D. e Katz, J. (2015). *monitoR: Acoustic template detection in R*. R package version 1.0.3. Citado na página 49.

Hall, M. A. (1999). *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato. Citado na página 24.

Haralick, R. M., Shanmugam, K., e Dinstein, I. H. (1973). Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, (6):610–621. Citado nas páginas xv, 16, 17, e 39.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, 2nd edition. Citado nas páginas 23 e 27.

Hollingsworth, K. P., Bowyer, K. W., e Flynn, P. J. (2009). The best bits in an iris code. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(6):964–973. Citado na página 17.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2010). A practical guide to support vector classification. Citado na página 28.

Huang, D.-Y. e Wang, C.-H. (2009). Optimal multi-level thresholding using a two-stage otsu optimization approach. *Pattern Recognition Letters*, 30(3):275–284. Citado na página 20.

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., e Liu, H. H. (1998). The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis.

Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 454(1971):903–995. Citado nas páginas 21 e 22.

Ishii, R. P. e De Mello, R. F. (2012). An online data access prediction and optimization approach for distributed systems. *Parallel and Distributed Systems, IEEE Transactions on*, 23(6):1017–1029. Citado nas páginas 33 e 61.

Ishii, R. P., Rios, R. A., e Mello, R. F. (2011). Classification of time series generation processes using experimental tools: a survey and proposal of an automatic and systematic approach. *International Journal of Computational Science and Engineering*, 6(4):217–237. Citado na página 10.

Karvelis, P., Tsoumas, I. P., Georgoulas, G., Stylios, C. D., Antonino-Daviu, J. A., e Climente-Alarcon, V. (2013). An intelligent icons approach for rotor bar fault detection. In *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*, páginas 5526–5531. IEEE. Citado nas páginas 7 e 21.

Kennel, M. B., Brown, R., e Abarbanel, H. D. (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Physical review A*, 45(6):3403. Citado nas páginas 10, 11, e 37.

Keogh, E., Xi, X., Wei, L., e Ratanamahatana, C. A. (2006). The ucr time series classification/clustering homepage. *URL= http://www. cs. ucr. edu/˜ eamonn/time_series_data*. Citado nas páginas xvii, 32, 40, 43, e 45.

Kuhn, M. (2015). *caret: Classification and Regression Training*. R package version 6.0-41. Citado nas páginas 44, 46, e 56.

Landis, J. R. e Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, páginas 159–174. Citado na página 30.

Last, M., Klein, Y., e Kandel, A. (2001). Knowledge discovery in time series databases. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 31(1):160–169. Citado na página 40.

Liaw, A. e Wiener, M. (2002). Classification and regression by randomforest. *R news*, 2(3):18–22. Citado na página 26.

Lopes, M. T., Gioppo, L. L., Higushi, T. T., Kaestner, C. A., Silla Jr, C. N., e Koerich, A. L. (2011a). Automatic bird species identification for large number of species. In *Multimedia (ISM), 2011 IEEE International Symposium on*, páginas 117–122. IEEE. Citado na página 52.

Lopes, M. T., Koerich, A. L., Nascimento Silla, C., e Kaestner, C. A. A. (2011b). Feature set comparison for automatic bird species identification. In *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, páginas 965–970. IEEE. Citado nas páginas 54 e 56.

Mamani, G. E. H. (2012). *Seleção supervisionada de características por ranking para processar consultas por similaridade em imagens médicas*. PhD thesis, Universidade de São Paulo. Citado na página 18.

Manjunath, B. S. e Ma, W.-Y. (1996). Texture features for browsing and retrieval of image data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(8):837–842. Citado na página 17.

Marwan, N., Carmen Romano, M., Thiel, M., e Kurths, J. (2007). Recurrence plots for the analysis of complex systems. *Physics Reports*, 438(5):237–329. Citado nas páginas 2, 12, 13, 14, 37, e 47.

MathWorks (2015). *Image Processing Toolbox*, volume 9. MathWorks, Inc. Citado na página 44.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., e Leisch, F. (2014). *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*. R package version 1.6-4. Citado na página 44.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill. Citado na página 25.

Morettin, P. A. e Toloi, C. (2006). *Análise de séries temporais*. Blucher. Citado nas páginas 8 e 9.

Ojala, T., Pietikäinen, M., e Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59. Citado nas páginas xv e 16.

Ojala, T., Pietikäinen, M., e Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987. Citado na página 16.

Pennebaker, W. B. e Mitchell, J. L. (1993). *JPEG: Still image data compression standard*. Springer Science & Business Media. Citado nas páginas 2, 20, e 39.

Pereira, C. M. e de Mello, R. F. (2014). Ts-stream: clustering time series on data streams. *Journal of Intelligent Information Systems*, 42(3):531–566. Citado na página 33.

Pereira, C. M. M. (2013). *Agrupamento de séries temporais em fluxos contínuos de dados*. PhD thesis, Universidade de São Paulo. Citado nas páginas 2, 4, 33, 48, 49, e 51.

Pinho, J. e Marini, M. (2014). Birds' nesting parameters in four forest types in the pantanal wetland. *Brazilian Journal of Biology*, 74(4):890–898. Citado na página 51.

Pollen, D. A. e Ronner, S. F. (1983). Visual cortical neurons as localized spatial frequency filters. *Systems, Man and Cybernetics, IEEE Transactions on*, (5):907–916. Citado na página 39.

Quinlan, Vipin, Wu, X., Kumar, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Philip, S. Y., et al. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1–37. Citado nas páginas 28 e 41.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1(1):81–106. Citado na página 25.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. Citado na página 28.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. Citado na página 43.

Rios, R. (2013). *Improving time series modeling by decomposing and analyzing stochastic and deterministic influences*. PhD thesis, Instituto de Ciências Matemáticas e de Computaçâo, Universidade de São Paulo, São Carlos. Citado nas páginas 2, 3, 8, 13, 15, 21, 31, e 61.

Rios, R. A. e De Mello, R. F. (2013). Improving time series modeling by decomposing and analyzing stochastic and deterministic influences. *Signal Processing*, 93(11):3001–3013. Citado nas páginas 30, 31, 36, e 63.

Rios, R. A. e Mello, R. (2012). A systematic literature review on decomposition approaches to estimate time series components. *INFOCOMP Journal of Computer Science*, 11(3-4):31–46. Citado nas páginas 20 e 21.

Romanski, P. e Kotthoff, L. (2014). *FSelector: Selecting attributes*. R package version 0.20. Citado na página 55.

Schroeder, M. R. (1992). *Fractals, chaos, power laws : minutes from an infinite paradise*. W. H. Freeman. Citado na página 20.

Silva, A. e Ishii, R. (2016). A new time series classification approach based on recurrence quantification analysis and gabor filter. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, páginas 955–957. ACM. Citado na página 66.

Silva, D. F. (2014). *Classificação de séries temporais por similaridade e extração de atributos com aplicação na identificação automática de insetos*. PhD thesis, Universidade de São Paulo. Citado nas páginas xvi, 1, 7, 27, 32, 37, 40, 45, 46, 55, 57, 59, 61, 62, 63, e 64.

Silva, D. F., De Souza, V., Batista, G. E., Keogh, E., e Ellis, D. P. (2013a). Applying machine learning and audio analysis techniques to insect recognition in intelligent traps. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, volume 1, páginas 99–104. IEEE. Citado na página 32.

Silva, D. F., Souza, V., De, M., e Batista, G. E. (2013b). Time series classification using compression distance of recurrence plots. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, páginas 687–696. IEEE. Citado na página 2.

Smola, A., Bartlett, P., Schölkopf, B., e Schuurmans, D. (2000). Introduction to large margin classifiers. Citado na página 27.

Souza, V., Silva, D. F., e Batista, G. E. (2014). Extracting texture features for time series classification. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, páginas 1425–1430. IEEE. Citado nas páginas 2, 33, 45, e 46.

Stowell, D. e Plumbley, M. D. (2014). Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. *PeerJ*, 2:e488. Citado na página 53.

Takens, F. (1981). *Detecting strange attractors in turbulence*. Springer. Citado na página 10.

Vellinga, W.-P. e Planqué, R. (2011). The xeno-canto collection and its relation to sound recognition and classification. Citado na página 52.

Wu, Z. e Huang, N. E. (2009). Ensemble empirical mode decomposition: a noise-assisted data analysis method. *Advances in adaptive data analysis*, 1(01):1–41. Citado na página 21.

Zbilut, J. P. e Webber, C. L. (1992). Embeddings and delays as derived from quantification of recurrence plots. *Physics letters A*, 171(3):199–203. Citado nas páginas 13 e 14.