Time Series Classification using Shape Features based on Angle Statistics

Bionda Rozin[1] **and Daniel Carlos Guimarães Pedronette**[1]

[1]Departamento de Estatística, Matemática Aplicada e Computação (DEMAC)
Univerisidade Estadual Paulista (UNESP)
Rio Claro – SP, Brazil

Abstract. *Time series have great applicability in the most diverse scenarios, including the scientific, agricultural, economic domains, among others. Therefore, creating effective representations of a time series is a challenging task, as it allows more accurate analysis and, consequently, more assertive results and conclusions in various machine learning tasks. One of the main tasks is classification, which can be performed from different computational representations of time series. The main objective of this work is to improve the efficiency of classification tasks using a representation of time series obtained by the Beam Angle Statistics algorithm, a contour feature extractor based on angular statistics.*

Resumo. *Séries temporais possuem grande aplicabilidade nos mais diversos cenários, incluindo os domínios científicos, agrícola, econômico, entre outros. Portanto, criar representações efetivas de uma série temporal é uma tarefa desafiadora, pois possibilita análises mais precisas e, consequentemente, obtenção de resultados e conclusões mais assertivas em diversas tarefas de aprendizado de máquina. Uma das principais tarefas associadas é a classificação, que pode ser realizada a partir de diferentes representações computacionais das séries temporais. Este trabalho tem como principal objetivo melhorar a eficácia de tarefas de classificação, utilizando uma representação das séries temporais obtida pelo algoritmo Beam Angle Statistics, um extrator de características de contorno baseado em estatísticas angulares.*

1. Introdução

As origens da agricultura remontam do período neolítico, e a adoção de práticas de cultivo e colheita, inicialmente de grãos, como o trigo, marcam a sedentarização da espécie humana e foram cruciais para a formação das primeiras civilizações do mundo [Mazoyer and Roudart 2006]. O cultivo e colheita, no geral, são extremamente dependentes de condições climáticas e de solo ideais para cada planta cultivada, as quais variam ao logo das estações do ano. A essa variação, está atrelado o conceito de sazonalidade, ou seja, um conjunto de mudanças que ocorrem em intervalos regulares. Uma das formas de analisar e estudar a sazonalidade é por meio de séries temporais, as quais, resumidamente, são um conjunto de observações feitas a respeito de um certo dado ou medida em intervalos regulares ao longo do tempo [Pino 2014]. Séries temporais são extremamente importantes no ramo agrícola, mas sua apliação não está restrita a esse escopo, sendo também utilizadas no mercado financeiro [Pincus and Kalman 2004], análises estatísticas [Pecora et al. 1995], consumo de produtos [Campbell and Mankiw 1989], incidência de doenças [C. Baskaran 2018], dentre outras aplicações.

A classificação de séries temporais é uma tarefa que permite a identificação de padrões que as caracterizam e possibilita a atribuição a uma dada classe ou categoria pré-definida [Bagnall et al. 2017]. Tal tarefa permite um diversificado conjunto de aplições e análise das séries temporais, possibilitando a detecção de ciclos normais e arritmias em um ecocardiograma [Volna et al. 2015], por exemplo. Comumente, a classificação de séries temporais é feita a partir de uma extração de características (*features*), seguido da aplicação de um algoritmo classificador [Bagnall et al. 2017]. Diversas abordagens têm sido propostas para extração de características de séries temporais, dentre as quais pode-se destacar: dados de distância [Geler et al. 2020], intervalos [Deng et al. 2013], forma [Ye and Keogh 2011], dentre outras.

Séries temporais podem ser representadas não só por um vetor de *features*, mas também por imagens [Wang and Oates 2015a]. É possível fazer uma visualização das séries por meio de gráficos, de imagens computacionais como *Gramian Angular Fields* [Wang and Oates 2015a] ou *Markov Transition Fields* [Wang and Oates 2015a], coordenadas polares, dentre outras. Tais imagens permitem que algoritmos de aprendizado de máquina disponham de diferentes representações ou perspectivas para a análise de séries temporais [Wang and Oates 2015b].

Diferentes formas de classificação de séries temporais a partir de imagens têm sido exploradas na literatura. Torres et al. [Torres et al. 2013] utiliza representações em gráficos de séries temporais para realizar um estudo da fenologia em vegetações de Cerrado, no interior do estado de São Paulo. As séries temporais são obtidas a partir das imagens da vegetação, as quais são representadas graficamente. Na figura obtida, um descritor de imagens baseado em formas é aplicado e, a partir dessa etapa, as séries temporais são caracterizadas, permitindo, assim, a definição do melhor horário do dia para caracterizar os padrões de mudança de uma folha de certa espécie de planta.

Wang et al. [Wang and Oates 2015a, Wang and Oates 2015b] representa séries temporais de 20 conjuntos de dados, de diferentes domínios, utilizando imagens denominadas *Gramian Angular Summation Fields* (GASF) e *Gramian Angular Difference Fields* (GADF), que representam a correlação temporal em cada ponto da série temporal. De forma análoga, ocorre o uso de *Markov Transition Fields* (MTF), que representa o campo de probabilidades de transição para uma série temporal discretizada. A partir dessa representação, são extraídas características multinível , as quais são utilizadas para tarefas de classificação a partir de um classificador SVM.

Zhang et al. [Zhang et al. 2020a] propõe uma nova forma de representar séries temporais por meio de imagens, o *Motif Difference Field* (MDF), o qual representa a unidade dos padrões da estrutura temporal de uma série temporal em intervalos longos e curtos. Além disso, a partir das imagens MDF, as séries temporais são classificadas a partir das redes neurais do tipo *Fully Convolution Network* (FCN).

Tabar et al. [Tabar and Halici 2017] propõe uma aplicação médica da classificação de séries temporais por sua representação visual. Partindo de eletroencefalografias realizadas enquanto os pacientes realizavam tarefas com suas mãos, são obtidas imagens das encefalografias a partir da aplicação da transformada de Fourier de tempo curto, utilizada para determinação da frequência senoidal, em que o conteúdo das fases da seção local pertencem a um sinal que varia conforme o tempo. Essas imagens são classificadas por redes neurais convolucionais aliadas ao *Stack Autoencoder*, para melhoria dos resultados. Tais métodos também são executados separadamente com propósito de avaliação.

Zhang et al. [Zhang et al. 2020b] propõe a representação gráfica de séries temporais por um método chamado *Multi-Scaled Signed Recurrence Plots* (MS-RP) e posterior classificação por redes neurais ResNet. O MS-RP se trata do uso de diferentes conjuntos de parâmetros associados aos *Recurrence Plots* [Eckmann et al. 1987], gerando, assim, diferentes imagens para uma mesma série temporal.

Considerando as abordagens discutidas, pode-se observar que, embora a classificação de séries temporais por meio de estratégias baseadas em imagens não seja uma novidade na literatura, há muitos desafios de pesquisa em aberto na área. As informações codificadas em propriedades de forma e contorno das imagens obtidas a partir de séries temporais incluem diversas características relevantes para tarefas de classificação. Neste cenário, este trabalho propõe a representação de séries temporais para tarefas de classificação por meio de estatísticas angulares calculadas a cada ponto, considerando a vizinhança em uma janela de análise.

A ideia central baseia-se na premissa de que as estatísticas angulares são capazes de representar as tendências de crescimento ou queda da série temporal. Para a representação proposta, foi utilizado um algoritmo de descrição de contornos, baseado em estatísticas angulares, o *Beam Angle Statistics* (BAS) [Arica and Yarman-Vural 2003]. Inspirado pela representação proposta em [Torres et al. 2013] para tarefas de recuperação, este trabalho difere em aspectos relevantes, aplicando a representação proposta em tarefas de classificação e utilizando diferentes classificadores. Resultados positivos de acurácia foram alcançados em experimentos considerando diversos conjuntos de dados.

O restante do artigo é organizado como segue: a Seção 2 descreve os métodos utilizados para a classificação de séries temporais; a Seção 3 discute os experimentos realizados; a Seção 4: descreve os resultados obtidos a partir da aplicação da metodologia proposta; e por fim a Seção 5 discute as conclusões do artigo.

2. Metodologia

Séries temporais possuem informações codificadas em suas formas e contornos, as quais podem ser exploradas para a representação eficaz das séries. As tendências de ascenção ou queda dos valores da série temporal podem ser representadas analisando-se o ângulo formado entre o ponto atual (em análise) e seus vizinhos em uma janela fixa, em intervalos anteriores e posteriores. Dessa forma, a estatística angular obtida para cada amostra da série temporal pode ser utilizada para gerar uma nova representação da mesma.

A Figura 1 apresenta uma visão geral do método proposto. A primeira etapa consiste na aplição do método *Beam Angle Statistics* (BAS) [Arica and Yarman-Vural 2003] para obtenção da nova representação, ilustrada à direita. Em seguida, as representações geradas são utilizadas por diferentes classificadores.

Os detalhes das etapas são discutidos nas próximas seções. Na Seção 2.1, é apresentado o extrator de características baseado em estatísticas angulares. Na Seção 2.2, são descritos os classificadores empregados no modelo proposto.

2.1. Beam Angle Statistics (BAS)

Este descritor de formas baseado em contorno é capaz de detectar concavidades e convexidades presentes em um contorno qualquer [Arica and Yarman-Vural 2003] e extrair características baseado em tais informações.

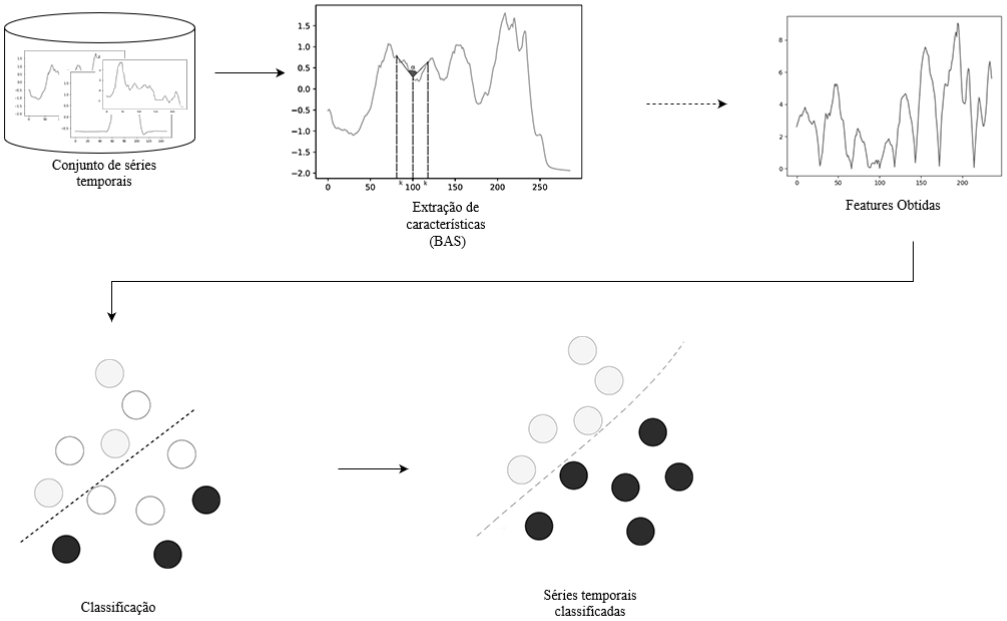

Figura 1. Visão geral do método proposto.

Seja $P = (p_1, p_2, ..., p_n)$ uma sequência de pontos de um contorno, em que n é a quantidade de pontos em P. Para cada ponto p_i, em que $i = k, k+1, k+2, ..., n-k$, sendo k um parâmetro previamente definido que determina o tamanho da vizinhança, existem os pontos p_{i-k} e p_{i+k}, que são os vizinhos determinados de acordo com o tamanho da vizinhaça. Baseado nos pontos p_{i-k}, p_i e p_{i+k}, são definidas duas retas: a e b. A reta a é determinada pelos pontos $(i-k, p_{i-k})$ e (i, p_i) e a reta b determinada pelos pontos (i, p_i) e $(i+k, p_{i+k})$. A Figura 2 ilustra o procedimento.

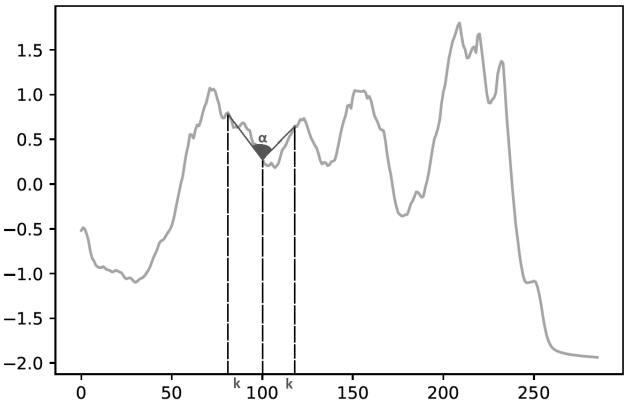

Figura 2. Retas a, b e ângulo α para ponto p_i ($i = 100$ e k arbitrário menor que 50.)

Os coeficientes angulares das retas a e b podem ser definidos como:

$$m_a = \frac{\Delta y}{\Delta x} = \frac{p_i - p_{i-k}}{i - (i-k)}, \tag{1}$$

$$m_b = \frac{\Delta y}{\Delta x} = \frac{p_{i+k} - p_i}{(i+k) - i}. \tag{2}$$

Seja α_i o ângulo formado pela intersecção entre as retas a e b no ponto p_i, o ângulo pode ser definido conforme a Equação 3:

$$\alpha_i = \arctan \frac{|m_b - m_a|}{1 + m_b * m_a} \tag{3}$$

Uma curva (ou contorno) P é representada pelo algoritmo *Beam Angle Statistics* conforme uma sequência de ângulos α_i, definida por $A = (\alpha_1, \alpha_2, ..., \alpha_{n-2k})$. Uma série temporal S definida por meio de uma curva P é representada por A como um conjunto de características para tarefas de classificação.

2.2. Classificadores

A classificação é uma tarefa de aprendizado supervisionado [Bramer 2007] cujo objetivo é definir um modelo, a partir de um conjunto de treinamento que esteja previamente rotulado, que seja capaz de rotular as amostras que não possuem informações. Nesta seção, são descritos os classificadores utilizados no método proposto. Foram utilizadas implementações presentes na biblioteca scikit-learn [Pedregosa et al. 2011], na linguagem de programação Python.

2.2.1. Support Vector Machine (SVM)

O classificador SVM é baseado na busca de superfícies que dividam os dados corretamente, de acordo com as classes definidas a partir de um conjunto rotulado utilizado como treinamento [Cortes and Vapnik 1995]. Cada classe é determinada a partir da separação de dados que pertencem à classe daqueles que não pertencem. Assim, para conjuntos de dados com mais classes, temos mais superfícies de decisão. A forma como são estabelecidas as fronteiras entre as classes é dependente de diversos parâmetros, principalmente do kernel selecionado. A Figura 3 ilustra o funcionamento do classificador em uma fronteira de decisão.

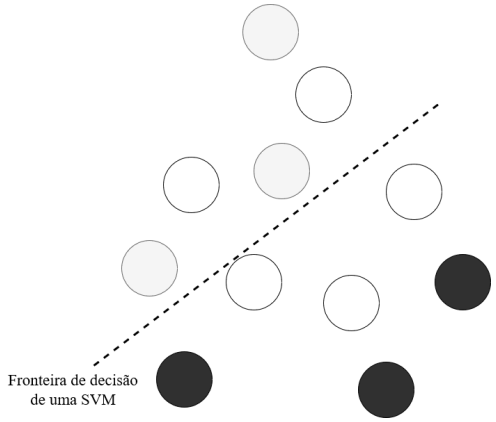

Figura 3. Demonstração do funcionamento de um classificador do tipo SVM.

2.2.2. K-Nearest Neighbors (kNN)

Os classificadores do tipo kNN atribuem classes aos elementos não rotulados a partir de seus k-vizinhos mais próximos pertencentes ao conjunto de treinamento [Bramer 2007]. A classe atribuida a um elemento não rotulado será correspondente à classe que ocorreu com mais frequência na vizinhança definida por k. A Figura 4 traz o exemplo do funcionamento de um classificador kNN.

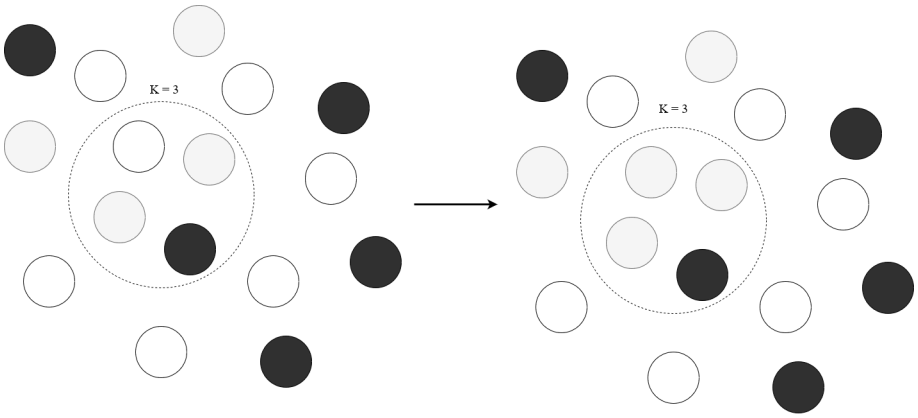

Figura 4. Demonstração do funcionamento de um classificador do tipo kNN.

2.2.3. Random Forest Classifier (RFC)

Este classificador é composto por um conjunto de árvores de decisão, formado a partir de um conjunto de treinamento, em que cada árvore de decisão conclui sobre a classe de um elemento do conjunto de teste. A classificação de cada elemento é feita conforme a classe escolhida pela maior parte das árvores de decisão [Breiman 2001]. A Figura 5 exemplifica a classificação utilizando o método Random Forest.

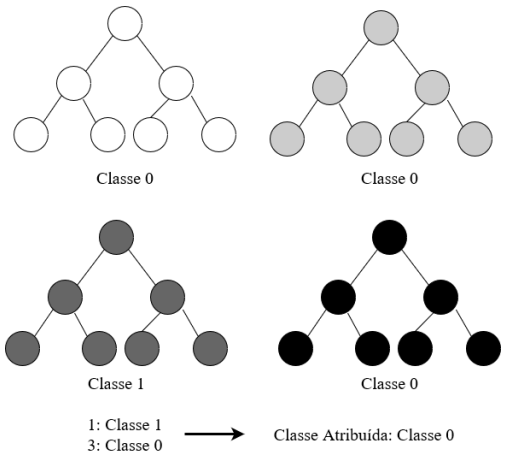

Figura 5. Funcionamento geral de um classificador do tipo Random Forest.

3. Avaliação Experimental

Esta seção descreve os experimentos realizados para a avaliar a acuráca de tarefas de classificação de séries temporais seguindo a metodologia proposta na Seção 2. Na

Subseção 3.1 estão descritos os conjuntos de dados utilizados, na Subseção 3.2 é descrita a métrica de avaliação considerada e, na Subseção 3.3, é descrito o protocolo experimental utilizado.

3.1. Conjuntos de Dados

Para a execução dos experimentos, foram considerados sete conjunto de dados de séries temporais de diferentes domínios, descritos a seguir:

- **Coffee** [1]: conjunto que contém duas classes, as quais correspondem a espectrografias de grãos de café do tipo Arabica e Robusta e é baseado na distinção entre tais classes. O *dataset* é composto por 56 séries temporais de tamanho 286, dividido em 28 séries temporais de treinamento e 28 de teste.
- **GunPoint**[1]: este conjunto de dados é baseado no movimento de sacar uma arma de um coldre, apontar a um alvo e devolver a arma ao coldre. O *dataset* contém duas classes, *Gun-Draw*, em que atores utilizam uma réplica de uma arma para a realização do movimento e *Point*, na qual utilizam o dedo indicador para apontar ao alvo. Composto por 200 séries de comprimento 150, dentre as quais, 50 delas compoem o conjunto de treinamento e 150 compoem o conjunto de teste.
- **Cylinder-Bell-Funnel (CBF)**[1]: este é um *dataset* simulado, em que os dados das classes são compostos por um ruído normal padrão aliado a um termo de compensação, o qual é diferente em cada classe. Este *dataset* é composto por 30 elementos que formam o conjunto de treinamento e 900 que integram o conjunto de teste, totalizando 930 elementos, de tamanho 128, divididos em 3 classes.
- **ECG5000**[1]: *Dataset* composto por 5000 batimentos cardíacos selecionados aleatóriamente, em que os valores foram obtidos por anotação automática. Os batimentos cardíacos estão divididos em 5 classes e possuem tamanho 140. O *dataset* é separado em 500 batimentos cardíacos que compoem o conjunto de treinamento e 4500 batimentos formam o conjunto de teste.
- **Beef**[1]: este *dataset* é composto por cinco classes de espectogramas de carne bovina, sendo compostas por carnes puras e carnes adulteradas por diferentes tipos de resíduos. É um *dataset* com 60 elementos, dividido em 30 séries de treinamento e 30 séries de teste, de comprimento 470.
- **Wine**[1]: conjunto de dados composto por espectrogramas de vinhos. Este conjunto de dados é composto por 111 séries temporais, de tamanho 234, divididas em 2 classes. 57 elementos compoem o conjunto de treinamento e 54, o conjunto de teste.
- **Strawberry**[1]: *dataset* composto por 983 espectógrafos de tamanho 235, divididos em duas classes. O *dataset* é baseado em diferenciar morangos de morangos adulterados ou outros tipos de frutas. Dentre os elementos, 613 destes são utilizados como treinamento e 370 são utilizados para testagem dos classificadores.

3.2. Acurácia

A medida de acurácia foi utilizada para a avaliação dos resultados obtidos. Esta métrica é responsável por calcular a porcentagem de elementos que foram rotulados corretamente dentro de um conjunto de dados que foram rotulados [Saito and Rehmsmeier 2015]. A acurácia é definida pela Equação 4.

[1] `http://www.timeseriesclassification.com/dataset.php`

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

Em que TP são os verdadeiros positivos (*True Positives*: amostras que foram classificadas corretamente), FP são os falsos positivos (*False Positives*: amostras classificadas incorretamente), TN são os verdadeiros negativos (*True Negatives*: amostras que, corretamente, deixaram de ser classificadas) e FN são os falsos negativos (*False Negatives*: amostras que, mesmo estando corretas, não foram consideradas).

3.3. Protocolo Experimental

Esta seção apresenta o protocolo utililzado para a realização dos experimentos. Inicialmente, os *datasets* descritos na Seção 3.1 foram rotulados pelos classificadores descritos em 2.2, utilizando os próprios valores das séries temporais como *features*. Estes resultados são relevantes para estabelecer uma comparação qualitativa entre as *features* originais e aquelas obtidas a partir da aplicação do algoritmo *Beam Angle Statistics*. Dessa forma, para os mesmos *datasets*, são extraídas as *features* de contorno, conforme a metodologia proposta.

Em relação aos conjuntos de treinamento e teste usados para cada *dataset*, todas as tarefas de classificação foram realizadas conforme a separação de conjuntos sugerida na literatura. Esta divisão se encontra disponível na biblioteca pyts [Faouzi and Janati 2020], da linguagem de programação Python. Não houve procedimento de validação cruzada.

Os valores do parâmetro k são determinantes para a obtenção de representações eficazes, uma vez que valores de k muito altos podem ignorar informações relevantes para as *features* e valores de k muito baixos podem considerar ruído em sua análise. Dessa forma, a análise dos resultados foi conduzida utilizando duas abordagens diferentes: valores de k específicos para cada *dataset* e valores de k calculados de acordo com um abordagem automática.

O valor específico por *dataset* foi definido experimentalmente, variando k em intervalos de 5 e tomando-se o valor que atingiu a maior acurácia pelo classificador SVM, no conjunto de testes, uma vez que não há conjuntos de validação nos *datasets* utilizados. Contudo, tal forma de definir k pode representar uma limitação do método. Dessa forma, foi proposta uma abordagem para a definição do parâmetro.

O cálculo automático do parâmetro k foi realizado de acordo com o desvio padrão do conjunto de dados. Seja $D = \{X_1, X_2, ..., X_n\}$ um *dataset* com n séries temporais X_i, é calculado o desvio padrão s_i entre os valores de cada série temporal X_i. Seja z a média de desvio padrão do conjunto dados, de forma que $z = 1/n \sum_{X_i \in D} s_i$. O tamanho da vizinhança k é definido de acordo com o valor de z, o tamamho da série temporal n e uma constante γ, de forma que $k = z \times n \times \gamma$. Foi utilizado o valor $\gamma = 0.09$, definido experimentalmente e utilizado de forma comum para todos os conjuntos de dados.

Os parâmetros dos classificadores foram, em sua maioria, configurados de acordo com o padrão da biblioteca *scikit-learn*. Para o classificador *kNN*, foi considerada uma vizinhança $k = 3$. Para o classificador *SVM*, foi considerado um kernel linear e parâmetro de regularização $C = 10$. Para o classificador *RFC*, todos os hiperparâmetros utilizados foram os padrões da biblioteca *scikit-learn*. A acurácia considerada para o *RFC* foi relativa à média de 10 execuções, junto do desvio padrão, pois se trata de um classificador não determinístico. Não foi realizada uma busca exaustiva sobre os melhores parâmetros

a serem utilizados nos classificadores, o que abre a possibilidade de haver resultados com maior acurácia.

4. Resultados

Nesta seção estão reportados os resultados obtidos a partir da aplicação da metodologia proposta para os diferentes *datasets* selecionados. A Tabela 1 reporta os diferentes valores de acurácia obtidos pela classificação dos *datasets*, utilizando as *features* originais e *features* obtidas com a aplicação do algoritmo *Beam Angle Statistics*, para valores de k automáticos. A Tabela 2 apresenta os resultados da aplicação do método obtido com valores específicos de k. O valor de k que atingiu a maior acurácia é reportardo na coluna final. Os melhores resultados para cada classificador, entre as Tabelas 1 e 2, estão marcados em negrito e os melhores resultados dentre os classificadores estão sublinhados.

Tabela 1. Resultados de classificação (acurácia) considerando k automático.

Dataset	Original			Método Proposto		
	SVM	**kNN**	**RFC**	**SVM**	**kNN**	**RFC**
Coffee	**100%**	**100%**	98.57±1.75%	**100%**	96.43%	98.21 ±1.79%
GunPoint	90.67%	**87.33%**	92.73±1.28%	92.67%	85.33	**95.87**±1.88**%**
CBF	88%	83.78%	89 ±0.91%	81.44%	67.89%	71.72 ±2.79%
ECG5000	**91.82%**	93.49%	**93.68** ±0.11**%**	91%	93.56%	92.89 ±0.08%
Beef	**90%**	60%	71.33±4%	76.67%	**63.33%**	72 ±3.4%
Wine	61.11%	55.56%	72.96±3.43%	90.74%	**57.41%**	71.85±3.19%
Strawberry	95.14%	92.43%	96.46±0.33%	96.76%	**94.32%**	96.19±0.25%

Tabela 2. Resultados de classificação considerando k específico por *dataset*.

Dataset	Método Proposto			
	SVM	**kNN**	**RFC**	***k***
Coffee	**100%**	**100%**	**100**±0**%**	15
GunPoint	**93.33%**	84%	**95.87**±0.88**%**	15
CBF	**92.11%**	**88.56%**	**90.48** ±1.11**%**	30
ECG5000	90.87%	**93.6%**	93.53 ±0.11%	15
Beef	76.67%	60%	**76.67**±4.94**%**	35
Wine	**92.59%**	42.59%	**77.04**±7.28**%**	5
Strawberry	**98.11%**	93.78%	**97.03**±0.42**%**	15

Em análise dos resultados das Tabelas 1 e 2, pode-se verificar que, dentre 7 *datasets* analisados, em 5 deles a metodologia proposta apresentou os melhores resultados nas tarefas de classificação, sendo estes resultados melhores para ks específicos conforme o *dataset*. Destaca-se o *dataset Wine*, aliado ao classificador SVM, em que houve 31.48% de ganho na acurácia, demonstrando que o método proposto é capaz de cumprir seu objetivo de aumentar a eficácia de tarefas de classificação de séries temporais.

4.1. Comparação com Outros Métodos

Nesta seção, os melhores resultados obtidos pela aplicação do método proposto são comparados com resultados de acurácia provenientes de dois classificadores de séries temporais da literatura. Na Seção 4.1.1, são descritos os dois classificadores utilizados e na Seção 4.1.2 são apresentadas as comparações entre os resultados.

4.1.1. Classificadores

Para os classificadores considerados nas comparações, foram utilizadas as implementações disponíveis na biblioteca pyts [Faouzi and Janati 2020], em Python. Não houve uma investigação dos melhores parâmetros a serem utilizados nos classificadores, utilizando-se, em sua maioria, os valores padrão sugeridos pela implementação. Para os parâmetros "word_size", "window_size" e "n_bins" utilizou-se, respectivamente, os valores 12, 0.4 e 5, em ambos os *datasets*. Os classificadores utilizados são descritos a seguir:

• **Symbolic Aggregate approXimation and Vector Space Model (SAX-VSM)** Este classificador utiliza representação simbólica para classificar séries temporais. Seu funcionamento é baseado nas seguintes técnicas: *Symbolic Aggregate approXimation* (SAX) e *Vector Space Model* (VSM) baseado no ponderamento *tf*idf* [Senin and Malinchik 2013] para realizar uma tarefa de classificação. O SAX é o que permite a representação simbólica de séries temporais, o qual gerará palavras de tamanho ω a partir de um alfabeto de tamanho α. Após a aplicação deste algoritmo, é construído um modelo do tipo *bag-of-words*, que será utilizado para ter as classes transformadas em um VSM utilizando *term frequencies (tf)* e *inverse document frequencies (idf)*. A predição da classe de uma amostra se dá pela similaridade por cosseno entre o seu vetor *tf* e os vetores *tf*idf* de cada classe.

• **Bag-of-SFA Symbols in Vector Space (BOSS VS)** Neste classificador, as séries temporais são transformadas em histogramas a partir do algoritmo Bag-of-SFA Symbols (BOSS). Um vetor *tf*idf* [Senin and Malinchik 2013] é calculado a partir das somas dos histogramas, para cada classe. A predição da classe de uma amostra, analogamente ao classificador SAX-VSM, se dá pela similaridade por cosseno entre o seu vetor *tf* e os vetores *tf*idf* de cada classe [Schäfer 2016].

4.1.2. Resultados e Comparação

Os resultados da comparação realizada estão apresentados na Tabela 3. Os melhores resultados obtidos para cada *dataset* estão marcados em negrito. Pode-se verificar que o método proposto atingiu ótimos resultados em comparação com as abordagens consideradas. É possível observar que, em cinco, dentre sete casos, o método proposto apresenta resultados superiores a classificadores já presentes na literatura. O maior ganho é notado no *dataset CBF*, sendo este de 42.55%, entre o método proposto e o método BOSS VS.

5. Conclusão

Neste trabalho, foi proposta uma nova metodologia para classificar séries temporais a partir da extração de características de contorno baseadas em estatísticas angulares. As características são extraídas por um algoritmo, *Beam Angle Statistics*, responsável por calcular os ângulos das curvas da série temporal a partir de um parâmetro k. As características obtidas são, então, utilizadas em classificadores tradicionais da literatura, como SVM, kNN e RFC. O método proposto apresentou resultados positivos, equiparando-se

Tabela 3. Comparação entre o métodos proposto e outras abordagens.

Dataset	Método Proposto	SAX-VSM	BOSS VS
Coffee	**100%**	96.43%	92.86%
GunPoint	95.87 $\pm$0.88%	**97.33%**	**97.33%**
CBF	92.11%	**95.22%**	49.56%
ECG5000	**93.6%**	80.62%	91.58%
Beef	**76.67%**	66.67%	60%
Wine	**92.59%**	61.11%	62.96%
Strawberry	**98.11 %**	62.97%	83.78%

a resultados obtidos em outros classificadores de séries temporais da literatura. Em trabalhos futuros, pretendemos investigar o uso da abordagem proposta em conjunto com métodos baseados em aprendizado profundo e também analisar problemas de desbalanceamento em séries temporais.

6. Agradecimentos

Os autores agradecem o apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo - FAPESP (processos 2020/08854-2, 2018/15597-6 e 2017/25908-6) e o Conselho Nacional de Desenvolvimento Científico e Tecnológico (processo #309439/2020-5).

Referências

Arica, N. and Yarman-Vural, F. (2003). Bas: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters*, 24:1627–1639.

Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31:606–660.

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

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

C. Baskaran, N. S. (2018). Time series analysis of swine flu literature during 1991-2013. *International Journal of Library Science and Information Management*, 2:38–48.

Campbell, J. Y. and Mankiw, N. G. (1989). Consumption, Income and Interest Rates: Reinterpreting the Time Series Evidence. In *NBER Macroeconomics Annual 1989, V. 4*, pages 185–246. National Bureau of Economic Research, Inc.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Deng, H., Runger, G., Tuv, E., and Vladimir, M. (2013). A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153.

Eckmann, J.-P., Kamphorst, S., and Ruelle, D. (1987). Recurrence plots of dynamical systems. *Europhysics Letters (epl)*, 4:973–977.

Faouzi, J. and Janati, H. (2020). pyts: A python package for time series classification. *Journal of Machine Learning Research*, 21(46):1–6.

Geler, Z., Kurbalija, V., Ivanovic, M., and Radovanovic, M. (2020). Weighted knn and constrained elastic distances for time-series classification. *Expert Systems with Applications*, 162:113829.

Mazoyer, M. and Roudart, L. (2006). *A History of World Agriculture: From the Neolithic Age to the Current Crisis*. Earthscan, London.

Pecora, Carroll, and Heagy (1995). Statistics for mathematical properties of maps between time series embeddings. *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics*, 52 4:3420–3439.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Pincus, S. and Kalman, R. (2004). Irregularity, volatility, risk, and financial market time series. *Proc. of the National Academy of Sciences of the USA*, 101:13709–13714.

Pino, F. A. (2014). Sazonalidade na agricultura. *Revista De Economia Agrícola (Impresso)*, v. 61:p. 63–93.

Saito, T. and Rehmsmeier, M. (2015). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10:e0118432.

Schäfer, P. (2016). Scalable time series classification. *Data Mining and Knowledge Discovery*, 30:1273–1298.

Senin, P. and Malinchik, S. (2013). Sax-vsm: Interpretable time series classification using sax and vector space model. In *2013 IEEE 13th International Conference on Data Mining*, pages 1175–1180.

Tabar, Y. R. and Halici, U. (2017). A novel deep learning approach for classification of eeg motor imagery signals. *Journal of Neural Engineering*, 14 1:016003.

Torres, R. d. S., Hasegawa, M., Tabbone, S., Almeida, J., dos Santos, J. A., Alberton, B., and Morellato, L. P. C. (2013). Shape-based time series analysis for remote phenology studies. In *IEEE Int. Geoscience and Remote Sensing Symposium*, pages 3598–3601.

Volna, E., Kotyrba, M., and Habiballa, H. (2015). Ecg prediction based on classification via neural networks and linguistic fuzzy logic forecaster. *The Scientific World Journal*, 2015:205749.

Wang, Z. and Oates, T. (2015a). Encoding time series as images for visual inspection and classification using tiled convolutional neural networks. In *Workshops at the twenty-ninth AAAI conference on artificial intelligence*.

Wang, Z. and Oates, T. (2015b). Imaging time-series to improve classification and imputation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, page 3939–3945. AAAI Press.

Ye, L. and Keogh, E. (2011). Time series shapelets: A novel technique that allows accurate, interpretable and fast classification. *Data Mining Know. Discovery*, 22:149–182.

Zhang, Y., Gan, F., and Chen, X. (2020a). Motif difference field: An effective image-based time series classification and applications in machine malfunction detection. In *Conf. on Energy Internet and Energy System Integration (EI2)*, pages 3079–3083.

Zhang, Y., Hou, Y., Zhou, S., and Ouyang, K. (2020b). Encoding time series as multi-scale signed recurrence plots for classification using fully convolutional networks. *Sensors*, 20:3818.