UNIVERSIDADE FEDERAL DE MINAS GERAIS
CIÊNCIA DA COMPUTAÇÃO

MARCO TÚLIO DE PINHO TAVARES TRISTÃO
ORIENTADOR HEITOR S. RAMOS

RELATÓRIO DE FINAL DE ATIVIDADES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
UNIVERSIDADE FEDERAL DE MINAS GERAIS
2023

Lista de Figuras

1 Exemplo de série temporal [5]. 4
2 Distância JS entre classes iguais e distintas [5]. 5
3 Entropia x Complexidade Estatística [5] 7
4 Porcentagem de acertos. 8
5 Matriz de confusão para o HASC. 9

Lista de Tabelas

3.1 Distância Jensen Shannon (4, 4). 6
3.2 Distância Jensen Shannon (3, 5). 6
4.1 F1-Score para múltiplas bases e parâmetros. 8

Sumário

1 INTRODUÇÃO E REVISÃO TEÓRICA **3**

1.1 Tipos de Classificadores de Séries Temporais 3

1.2 Epítome do Relatório 3

2 OBJETIVOS **4**

3 METODOLOGIA **4**

3.1 Fundamentação Teórica 4

3.2 Experimentos 5

4 RESULTADOS ALCANÇADOS E DISCUSSÃO **6**

1 INTRODUÇÃO E REVISÃO TEÓRICA

Séries temporais são sequências de pontos numéricos que ocorrem sucessivamente ao longo de um intervalo de tempo. Esses dados são captados de sensores a todo o momento em aplicações como sensoriamento de deslocamento humano [5], monitoramento de batimentos cardíacos [1], detecção de consumo energético [2, 9], entre várias outras. Na necessidade de se analisar tais dados a um nível mais profundo do que o possibilitado somente pela intuição humana, há o esforço para se criar algoritmos computacionais de classificação para as séries temporais. Tais algoritmos devem, por meio da identificação de informações e padrões, normalmente não visíveis por uma análise humana, classificar diferentes séries em classes, tal que cada classe possui características que aproximam as diferentes séries que a forma.

1.1 Tipos de Classificadores de Séries Temporais

No contexto dos algoritmos de classificação de séries temporias (TSC), podemos classificá-los nos seguintes grupos [7]:

1. algoritmos baseados em extração de características (feature based): são extraídas features da série para alimentar o classificador em um único pipeline;
2. algoritmos baseados em intervalos (interval based);
3. algoritmos baseados em shapelets (shapelet based);
4. algoritmos baseados em dicionários (dictionary based): a entrada do classificador é um histograma dos padrões que se repetem ao longo da série;
5. algoritmos baseados em convoluções (convolution based): utiliza-se de operações de convolução e pooling para criar o feature space do classificador;
6. algoritmos baseados em deep learning (deep learning based): utiliza-se redes neurais no processo de classificação;
7. algoritmos baseados em distância (distance based) realizam um cálculo de distância para estimar a similaridade entre séries. Para isso, é muito utilizado o método dos nearest-neighbors (NN).

No trabalho produzido, transformamos a série original de pontos em uma série de símbolos e após isso utilizamos NN para classificá-la.

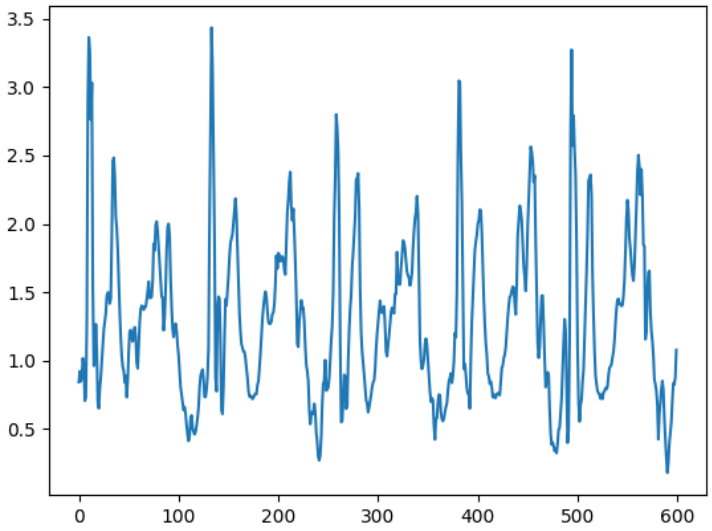
1.2 Epítome do Relatório

A primeira transformação que aplicamos à série original é a Symbolic Aggregate Approximation (SAX) [6], que agrega a cada ponto da série a informação de amplitude. Na sequência utilizamos uma janela deslizante para agrupar pontos subsequentes, agregando assim o valor de temporalidade a cada símbolo da nova série. O trabalho realizado foca-se em avaliar a pertinência dessa transformação em potencializar a efetividade do classificador implementado.

Para isso, a seguir apresentaremos experimentos com múltiplos bancos de dados e métricas que mostram a evolução da precisão do classificador quando aplicamos as

transformações propostas aos dados. Na seção 2 apresentaremos o objetivo do trabalho, na seção 3 a metodologia e os experimentos realizados e na seção 4 apresentaremos e discutiremos os resultados obtidos.

Figura 1: Exemplo de série temporal [5].



2 OBJETIVOS

Classificadores de séries temporais (TSC) são utilizados em várias aplicações do mundo real. O banco de dados [1] contém informações sobre os batimentos cardíacos de indivíduos ao longo do tempo. A partir dessas, é possível prever se os batimentos possuem alguma anomalia, e assim identificar possíveis riscos à saúde do paciente. Aplicações ligadas à mobilidade urbana também são muito beneficiadas por classificadores. A base do HASC 2011 [5] possui dados das coordenadas espaciais de indivíduos em um intervalo de tempo. A partir desses dados, inferimos se o indivíduo estava andando, o se ele estava correndo, ou parado, por exemplo.

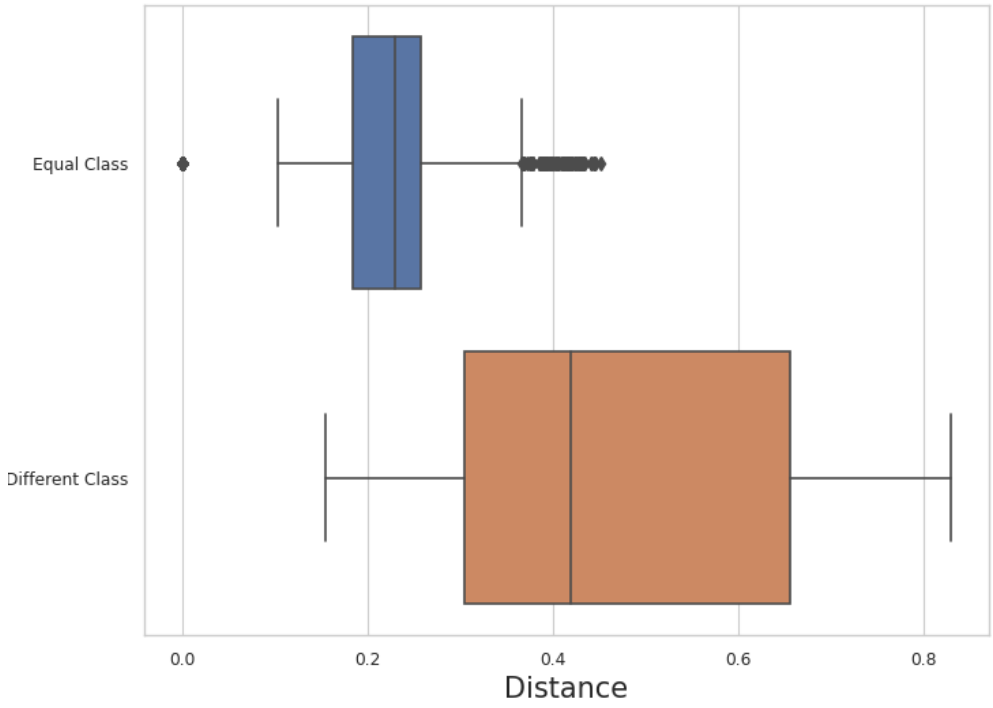
Visto a grande variedade de usos de TSC's, buscamos formular um novo método de TSC que seja capaz de aprimorar a capacidade de classificação das séries em comparação com outros métodos já existentes na literatura. Para isso, estudamos uma transformação de séries temporais e sua aplicação em classificadores, analisando seus impactos nos resultados da classificação.

3 METODOLOGIA

3.1 Fundamentação Teórica

A primeira etapa da manipulação de séries temporais que propomos é a discretização da série por meio do SAX. De acordo com o Teorema do Limite Central (CLT), a distribuição amostral da média da série se aproxima de a distribuição normal conforme

Figura 2: Distância JS entre classes iguais e distintas [5].



a série aumenta. Assim, considere uma distribuição normal entre os limites inferiores e superiores de amplitude de uma série. O SAX divide os pontos em intervalos consecutivos de amplitude de acordo com a área abaixo da curva da distribuição normal, de forma que todos os intervalos possuam a mesma área. Assim, todos os pontos em cada intervalo recebem a mesma representação simbólica. Ou seja, caso se deseje três intervalos de amplitude, o resultado final após o SAX será uma série formada com os símbolos a, b, c.

O segundo passo é, por meio de uma janela deslizante, agrupar símbolos consecutivos. A janela se move um símbolo por vez, mantendo assim comprimento original quase inalterado. O objetivo dessa transformação é captar o aspecto da temporalidade. Ou seja, considere as sequências $a - b - c$ e $c - b - a$. Perceba que uma sequência se altera em ordem crescente de amplitude, enquanto a outra se altera em ordem decrescente. Apesar de serem formadas pelos mesmos símbolos (a, b e c), elas serão transformadas em novos símbolos distintos, pois representam um fluxo distinto em relação ao tempo. Este fluxo é o que desejamos capturar, e que não estava presente anteriormente.

Após a transformação, usamos o algoritmo de K-Nearest-Neighbors [8] para performar a classificação das séries temporais.

3.2 Experimentos

No trabalho executado, realizamos uma pesquisa exploratória sobre o potencial das manipulações de dados propostas na classificação de séries temporais. Para essa análise, calculamos métricas que nos possibilitaram identificar avanços quando usamos as transformações propostas.

A primeira métrica calculada foi a distância Jensen Shannon (JS) entre instâncias da mesma classe e entre instâncias de classes diferentes. Caso séries de classes iguais possuam uma distância JS média menor, isso indicaria uma maior separabilidade entre as classes. O melhor resultado obtido foi para a base do HASC 2011 [5]. Para ela, é possível perceber claramente que a distância JS é considerávelmente menor entre exemplos da mesma classe.

Tabela 3.1: Distância Jensen Shannon (4, 4).

Bancos de dados	Mesma classe		Classes distintas	
	Média	Mediana	Média	Mediana
[1]	0.179	0.144	0.183	0.148
[2, 9]	0.540	0.830	0.6195	0.831
[3]	0.098	0.097	0.112	0.111
[5]	0.199	0.228	0.473	0.418
[4]	0.085	0.081	0.097	0.096

Tabela 3.2: Distância Jensen Shannon (3, 5).

Bancos de dados	Mesma classe		Classes distintas	
	Média	Mediana	Média	Mediana
[1]	0.233	0.197	0.238	0.203
[2, 9]	0.341	0.200	0.510	0.653
[3]	0.104	0.108	0.124	0.120
[5]	0.283	0.218	0.453	0.400
[4]	0.078	0.077	0.089	0.087

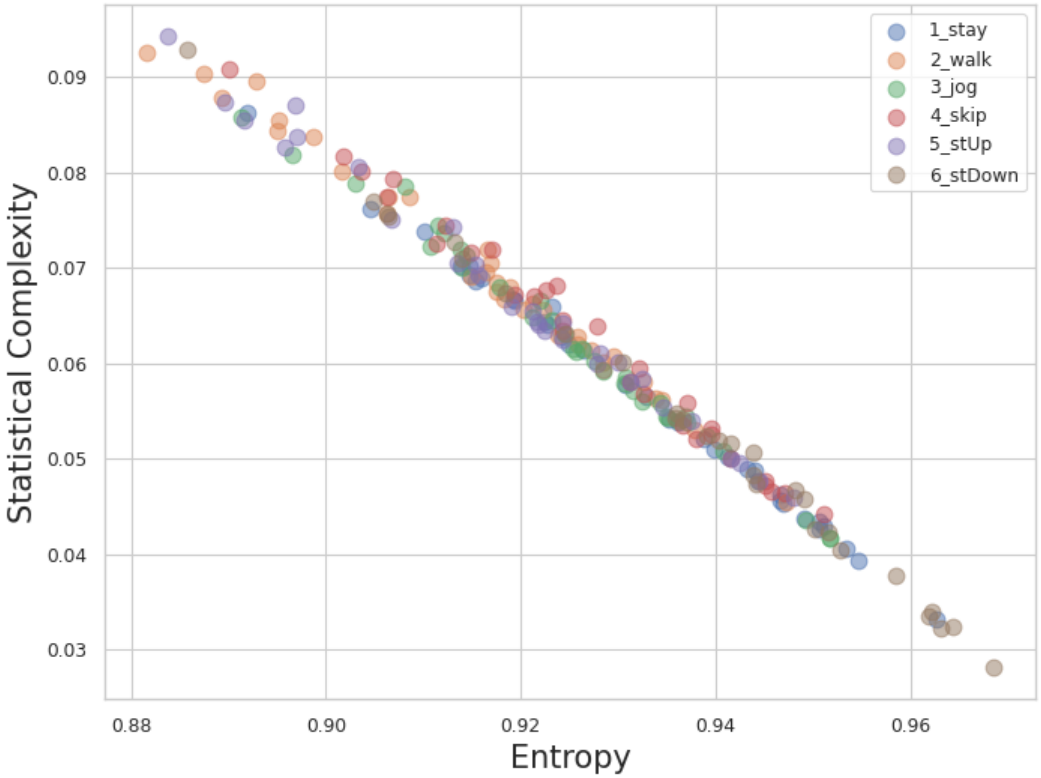
As tabelas 3.2 (tamanho da janela e número de divisões no SAX igual a 4) e 3.2 (tamanho da janela igual a 3 e número de divisões no SAX igual a 5) mostra os resultados da distribuição Jensen Shannon para diferentes bancos de dados usados nos experimentos. Nota-se que pares da mesma classe para diferentes bases de dados têm sim uma distância JS menor na média.

Outro experimento feito foi a observação da distribuição de pontos por classe para o gráfico de entropia por complexidade estatística. O resultado desse experimento pode ser visto na figura 3.

4 RESULTADOS ALCANÇADOS E DISCUSSÃO

Nessa seção apresentamos os resultados obtidos ao executar o classificador implementado com diferentes bancos de dados. Comparamos os resultados para o cenário de rodarmos o classificador realizando e não realizando as transformações propostas, para que possamos analisar a diferença causada por elas.

Figura 3: Entropia x Complexidade Estatística [5]



Nos nossos testes, encontramos os melhores valores com o banco de dados HASC [5] e quando o número de divisões no SAX é igual a 4. A figura 4a mostra o percentual de acertos do classificador para o HASC usando diferentes combinações de parâmetros.

As combinações de parâmetros são as seguintes, sendo o primeiro valor o o número de partições no SAX e o segundo o tamanho da janela, respectivamente: (3, 3), (3, 4), (3, 5), (4, 3), (4, 4), (4, 5), (5, 3), (5, 4), (5, 5).

Vemos que, apesar da porcentagem de acertos não ser grande, as transformações que realizamos representam um avanço em relação à série original no que tange ao potencial de classificação. No entanto, é perceptível uma diferença grande de performance entre os bancos de dados. A imagem 4b mostra um resultado muito mais próximo do que o exibido na 4a.

Também computamos a matriz de confusão do resultado. Novamente, o resultado mais pertinente pertence ao HASC. Temos na figura 5 a comparação da matriz de confusão dos resultados do classificador para os dados originais e para os dados transformados. Em ambos os casos usamos os parâmetros (4, 4).

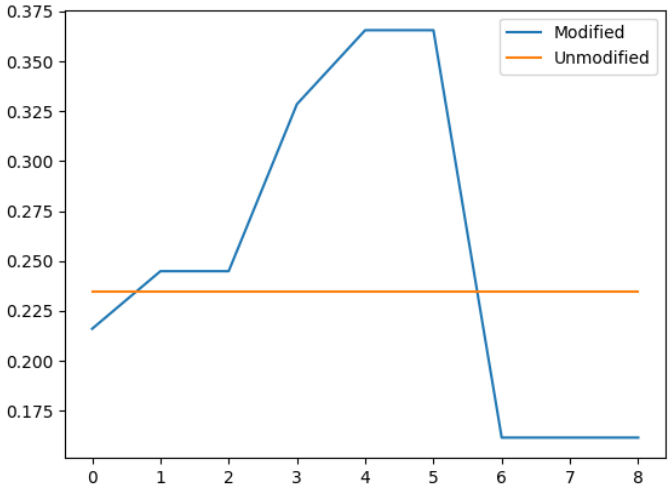
Por fim, a tabela 4 mostra o comparativo do F1-Score para múltiplos bancos de dados e opções de parâmetros. Percebe-se que não há uma escolha de parâmetro que seja universalmente melhor. Também, temos que a nossa transformação não é capaz de aperfeiçoar o resultado do classificador para todas as bases de dados. Por exemplo, os resultados da base [4] não têm melhora quando aplicamos as modificações propostas aos seus dados. No entanto, apesar de não ser uma solução geral, nossa proposta apresenta melhoras para a maior parcela dos casos estudados, e por isso pode merecer mais análises e estudos sobre como aplicá-la de uma maneira a aumentar os seus benefícios.

Tabela 4.1: F1-Score para múltiplas bases e parâmetros.

Bancos de dados	Com transformação					Original
(Janela x Partições SAX)	(3, 3)	(3, 4)	(4, 3)	(4, 4)	(5, 3)	
[1]	0.618	0.618	0.618	0.618	0.618	0.612
[2, 9]	0.114	0.112	0.342	0.322	0.231	0.326
[3]	0.750	0.750	0.750	0.750	0.604	0.500
[5]	0.216	0.244	0.328	0.365	0.161	0.234
[4]	0.345	0.345	0.500	0.345	0.630	0.630

Figura 4: Porcentagem de acertos.

(a) HASC 2011 [5].



(b) ACSF1 [2, 9].

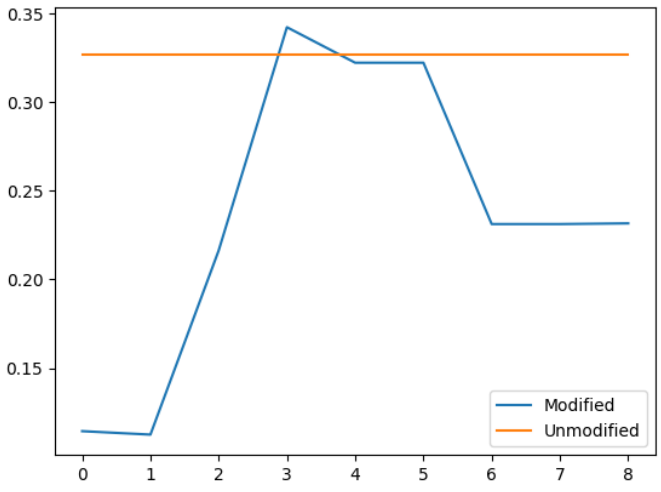
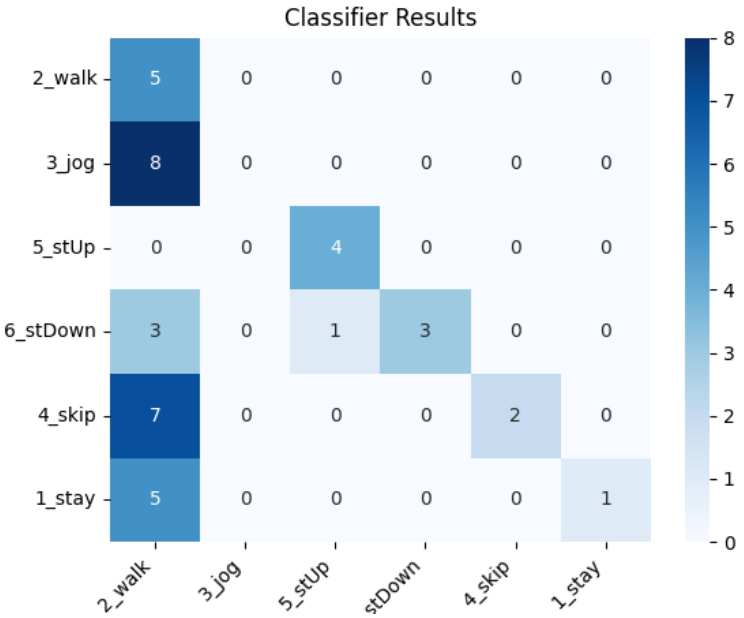
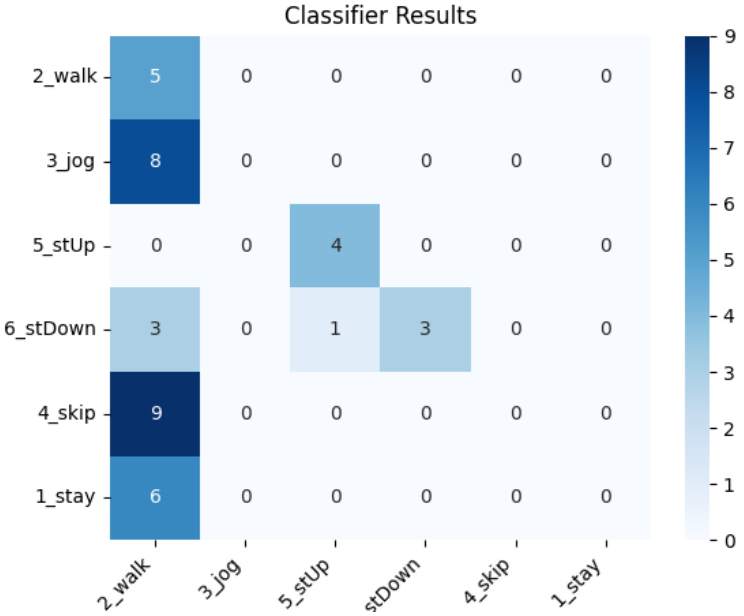


Figura 5: Matriz de confusão para o HASC.

(a) Com transformações.



(b) Sem transformações.



Referências

[1] P. Bentley, G. Nordehn, M. Coimbra, and S. Mannor. The PASCAL Classifying Heart Sounds Challenge 2011 (CHSC2011) Results. http://www.peterjbentley.com/heartchallenge/index.html.

[2] Christophe Gisler, Antonio Ridi, D. Zujferey, Omar Abou Khaled, and Jean Hennebert. Appliance consumption signature database and recognition test protocols. pages 336–341, 05 2013.

[3] Jon Hills, Jason Lines, Edgaras Baranauskas, James Mapp, and Anthony Bagnall. Classification of time series by shapelet transformation. *Data Mining and Knowledge Discovery*, 28, 05 2013.

[4] Jon Hills, Jason Lines, Edgaras Baranauskas, James Mapp, and Anthony Bagnall. Classification of time series by shapelet transformation. *Data mining and knowledge discovery*, 28:851–881, 2014.

[5] Nobuo Kawaguchi, Nobuhiro Ogawa, Yohei Iwasaki, Katsuhiko Kaji, Tsutomu Terada, Kazuya Murao, Sozo Inoue, Yoshihiro Kawahara, Yasuyuki Sumi, and Nobuhiko Nishio. Hasc challenge: Gathering large scale human activity corpus for the real-world activity understandings. page 27, 03 2011.

[6] Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15:107–144, 2007.

[7] Matthew Middlehurst, Patrick Schäfer, and Anthony Bagnall. Bake off redux: a review and experimental evaluation of recent time series classification algorithms, 2023.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] Patrick Schäfer and Ulf Leser. Fast and accurate time series classification with WEASEL. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, nov 2017.