

APLICAÇÕES DE INTELIGÊNCIA ARTIFICIAL E CIÊNCIA DE DADOS

Julio C. Uzinski
Caio C. E. de Abreu
Bruno R. de **Oliveira**

organizadores



Pantanal Editora

2020

Julio Cezar Uzinski
Caio Cesar Enside de Abreu
Bruno Rodrigues de Oliveira
(Organizadores)

**Aplicações de
Inteligência Artificial e
Ciência de Dados**



Pantanal Editora

2020

Copyright© Pantanal Editora
Copyright do Texto© 2020 Os Autores
Copyright da Edição© 2020 Pantanal Editora
Editor Chefe: Prof. Dr. Alan Mario Zuffo
Editores Executivos: Prof. Dr. Jorge González Aguilera
Prof. Dr. Bruno Rodrigues de Oliveira

Diagramação: A editora
Edição de Arte: A editora
Revisão: Os autor(es), organizador(es) e a editora

Conselho Editorial

- Prof. Dr. Adailson Wagner Sousa de Vasconcelos – OAB/PB
- Profa. Msc. Adriana Flávia Neu – Mun. Faxinal Soturno e Tupanciretã
- Profa. Dra. Albys Ferrer Dubois – UO (Cuba)
- Prof. Dr. Antonio Gasparetto Júnior – IF SUDESTE MG
- Profa. Msc. Aris Verdecia Peña – Facultad de Medicina (Cuba)
- Profa. Arisleidis Chapman Verdecia – ISCM (Cuba)
- Prof. Dr. Bruno Gomes de Araújo - UEA
- Prof. Dr. Caio Cesar Enside de Abreu – UNEMAT
- Prof. Dr. Carlos Nick – UFV
- Prof. Dr. Claudio Silveira Maia – AJES
- Prof. Dr. Cleberton Correia Santos – UFGD
- Prof. Dr. Cristiano Pereira da Silva – UEMS
- Profa. Ma. Dayse Rodrigues dos Santos – IFPA
- Prof. Msc. David Chacon Alvarez – UNICENTRO
- Prof. Dr. Denis Silva Nogueira – IFMT
- Profa. Dra. Denise Silva Nogueira – UFMG
- Profa. Dra. Dennyura Oliveira Galvão – URCA
- Prof. Dr. Elias Rocha Gonçalves – ISEPAM-FAETEC
- Prof. Me. Ernane Rosa Martins – IFG
- Prof. Dr. Fábio Steiner – UEMS
- Prof. Dr. Gabriel Andres Tafur Gomez (Colômbia)
- Prof. Dr. Hebert Hernán Soto González – UNAM (Peru)
- Prof. Dr. Hudson do Vale de Oliveira – IFRR
- Prof. Msc. Javier Revilla Armesto – UCG (México)
- Prof. Msc. João Camilo Sevilla – Mun. Rio de Janeiro
- Prof. Dr. José Luis Soto Gonzales – UNMSM (Peru)
- Prof. Dr. Julio Cezar Uzinski – UFMT
- Prof. Msc. Lucas R. Oliveira – Mun. de Chap. do Sul
- Prof. Dr. Leandro Argente-Martínez – ITSON (México)
- Profa. Msc. Lidiane Jaqueline de Souza Costa Marchesan – Consultório em Santa Maria
- Prof. Msc. Marcos Pisarski Júnior – UEG
- Prof. Dr. Mario Rodrigo Esparza Mantilla – UNAM (Peru)
- Profa. Msc. Mary Jose Almeida Pereira – SEDUC/PA
- Profa. Msc. Nila Luciana Vilhena Madureira – IFPA
- Profa. Dra. Patrícia Maurer
- Profa. Msc. Queila Pahim da Silva – IFB
- Prof. Dr. Rafael Chapman Auty – UO (Cuba)
- Prof. Dr. Rafael Felipe Ratke – UFMS
- Prof. Dr. Raphael Reis da Silva – UFPI

- Prof. Dr. Ricardo Alves de Araújo – UEMA
- Prof. Dr. Wéverson Lima Fonseca – UFPI
- Prof. Msc. Wesclen Vilar Nogueira – FURG
- Profa. Dra. Yilan Fung Boix – UO (Cuba)
- Prof. Dr. Willian Douglas Guilherme – UFT

Conselho Técnico Científico

- Esp. Joacir Mário Zuffo Júnior
- Esp. Maurício Amormino Júnior
- Esp. Tayronne de Almeida Rodrigues
- Esp. Camila Alves Pereira
- Lda. Rosalina Eufrausino Lustosa Zuffo

Ficha Catalográfica

Dados Internacionais de Catalogação na Publicação (CIP) (eDOC BRASIL, Belo Horizonte/MG)	
A642	<p>Aplicações de inteligência artificial e ciência de dados [recurso eletrônico] / Organizadores Julio Cezar Uzinski, Caio Cesar Enside de Abreu, Bruno Rodrigues de Oliveira. – Nova Xavantina, MT: Pantanal, 2020. 100p.</p> <p>Formato: PDF Requisitos de sistema: Adobe Acrobat Reader Modo de acesso: World Wide Web ISBN 978-65-88319-26-0 DOI https://doi.org/10.46420/9786588319260</p> <p>1. Inteligência artificial. 2. Automação. 3. Processamento de dados. I. Uzinski, Julio Cezar. II. Abreu, Caio Cesar Enside de. III. Oliveira, Bruno Rodrigues de.</p> <p style="text-align: right;">CDD 001.535</p>
Elaborado por Maurício Amormino Júnior – CRB6/2422	

O conteúdo dos livros e capítulos, seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva do(s) autor (es). O download da obra é permitido e o compartilhamento desde que sejam citadas as referências dos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Pantanal Editora

Rua Abaete, 83, Sala B, Centro. CEP: 78690-000.
 Nova Xavantina – Mato Grosso – Brasil.
 Telefone (66) 99682-4165 (Whatsapp).
<https://www.editorapantanal.com.br>
contato@editorapantanal.com.br

APRESENTAÇÃO

Ciência de dados (*Data science*) e Inteligência artificial (*Artificial intelligence*) podem ser consideradas como algumas das áreas da ciência mais importantes de nosso tempo. Ademais, não estão apenas no mundo acadêmico ou no futuro como no filme *The terminator*, estão ao nosso redor nesse exato momento em situações simples, por exemplo, são empregadas para sugerir qual o próximo emoji que você enviará em uma conversa no seu aplicativo de bate papo favorito. Ciência de dados se refere a um campo de estudo muito amplo que se vale de métodos científicos para se obter informações e percepções de conjuntos de dados, enquanto que a inteligência artificial refere-se ao que chamamos de inteligência das máquinas. As aplicações dessas ferramentas estão relacionadas às tecnologias que nos envolvem em nosso cotidiano, e.g., as redes sociais ou comércio eletrônico, bem como aplicações que escapam ao senso comum, como em cibersegurança ou astrofísica.

Tais abordagens, às vezes, são utilizadas juntas, de forma intercalada ou separadas. Uma pesquisa poderá contar com ambas as ferramentas ou apenas com uma delas. Nesse livro, o leitor encontrará aplicações diversas dessas ferramentas em diferentes perspectivas. Em seis capítulos são apresentados resultados de pesquisas de dezesseis pesquisadores sobre temas diversos que dão uma demonstração do poder da Inteligência Artificial e da Ciência de Dados.

O primeiro capítulo traz uma comparação dos resultados da Transformada Wavelet (WT) e do Filtro Savitzky-Golay (SG) na estimação da tendência em séries temporais. Os resultados são obtidos a partir de uma série temporal artificial e uma série de dados reais, i.e, a cotação da moeda americana (Dólar US\$) frente à moeda brasileira (Real R\$).

Considerando que a principal abordagem em Aprendizado de Máquinas é a construção de algoritmos que, por meio das experiências e aprendizado, possam ser melhorados automaticamente, o segundo capítulo desse livro traz uma avaliação de um algoritmo usando Redes Neurais Artificiais LSTM. Essa abordagem é justificada pela necessidade de reconhecimento de padrões de comportamento de presença de elementos em um ambiente para realizar a predição de ações no local por meio dos dados coletados fornecendo uma base para o desenvolvimento de um sistema doméstico inteligente.

Já no terceiro capítulo, uma metodologia para descrição de Casos de Uso (Uc) no desenvolvimento de sistemas de informação é apresentada. A motivação para a criação desta metodologia é que muitos UC's são descritos com pouca riqueza de detalhes, o que prejudica a qualidade dos sistemas de informação.

O quarto capítulo traz uma aplicação bastante diversa de todas as outras: um estudo dos aspectos e causas da obesidade por meio do algoritmo de agrupamento Farthest First para agrupar

pessoas com informações demográficas, socioeconômicas, de biotipo e de conjunto de ações observáveis recorrentes.

No quinto capítulo é desenvolvida uma aplicação utilizando regressão logística como um mapeamento do espaço de atributos não-linearmente separável baseado em um tipo especial de funções denominadas de Golden Wavelets. Nesse texto, os autores buscam tratar a matemática envolvida no método de aprendizado de máquina denominado de Regressão Logística e propor a utilização de funções wavelets para mapeamento do espaço de atributos.

Para encerrar esse e-book com chave de ouro, o sexto capítulo, apresenta uma classificação de cenas acústicas utilizando a Transformada Wavelet e Aprendizagem de Máquina. Os autores dedicam a redação à apresentação dos principais métodos de Classificação de Cenas Acústicas (CCA) e os principais atributos utilizados para esta classificação, exemplos de aplicações, e uma investigação adicional centrada na utilização da Transformada Wavelet Contínua (TWC).

Caríssimo(a) leitor(a), convidamos você à leitura de Aplicações de Inteligência Artificial e Ciência de Dados, e esperamos que a mesma seja um deleite ilimitado e que contribua de alguma forma com suas aspirações.

Julio Cezar Uzinski

SUMÁRIO

Apresentação	4
Capítulo I	7
Obtenção de Tendências em Série Temporal empregando a Transformada Wavelet e o filtro Savitzky-Golay.....	7
Capítulo II	24
Aplicação de Redes Neurais Artificiais LSTM em domótica inteligente	24
Capítulo III	32
Uma Metodologia Simples para Descrição de Casos de Uso no Desenvolvimento de Sistemas de Informação.....	32
Capítulo IV	49
Aspectos e causas da obesidade: uma análise utilizando algoritmo de agrupamento <i>Fartbest First</i>	49
Capítulo V	59
Mapeamento de espaço de atributos não-linearmente separável baseado na Golden wavelet: aplicação na regressão logística	59
Capítulo VI	85
Classificação de Cenas Acústicas Utilizando a Transformada Wavelet Contínua e Aprendizagem de Máquina	85
Índice Remissivo	99

Obtenção de Tendências em Série Temporal empregando a Transformada Wavelet e o filtro Savitzky-Golay

Recebido em: 16/09/2020

Aceito em: 24/09/2020

 10.46420/9786588319260cap1

Julio Cezar Uzinski^{1*} 

Bruno Rodrigues de Oliveira² 

Caio Cesar Enside de Abreu³ 

INTRODUÇÃO

O mundo corporativo e os cientistas em geral estão inclinados cada vez mais a utilizarem aplicações de predição do comportamento de processos que variam ao longo do tempo, por meio da análise dos dados passados destes processos, para subsidiar tomadas de decisões e até mesmo definir políticas econômicas e comportamentais corporativas. Dentre alguns exemplos podemos citar: predição do preço das *comodities*; do comportamento do mercado de ações; previsão do número de infectados por determinado vírus; previsões de terremotos; do impacto ambiental causado por decisões a serem consolidadas, reconhecimento e identificação de padrões, dentre outros (Fu, et al., 2008).

Nesse contexto, a ciência de dados, a inteligência artificial, o aprendizado de máquina, a própria estatística, e diversas outras técnicas dentro de alguns desses campos científicos possuem uma gama de alternativas para realizar tais predições (Picasso, et al, 2019; Puchalsky, 2018; Radityo, 2017).

Uma série temporal é um conjunto de observações, cada uma sendo registrada em um dado momento, representando o comportamento de um processo em função do tempo (Puchalsky, 2018). Dentre os objetivos vinculados ao estudo de séries temporais, destacamos a estimativa de suas componentes, especialmente, a componente de tendência, que possibilita compreender o comportamento do processo (ou da série) no longo prazo. Por outro lado, a componente aleatória ou as oscilações, que representam a maior variabilidade da série, possibilitam a compreensão de seu comportamento no curto prazo (Schlüter e Deuschle, 2010).

Diversas técnicas podem ser utilizadas para a estimação da componente de tendência de uma série temporal, tais como filtros Média Móvel, Savitzky-Golay e Transformada Wavelet. Alguns

¹ Rua das Gaivotas, 325, Sinop – MT. CEP 78556-398.

² Editora Pantanal. E-mail: bruno@editorapantanal.com.br.

³ Universidade do Estado de Mato Grosso – UNEMAT, Rua Santa Rita, 128, Alto Araguaia-MT. CEP 78780-000. E-mail: caioenside@unemat.br

* Autor correspondente: uzinski.jc@gmail.com.

exemplos são encontrados em: Carbone (2009), que utiliza a remoção de tendência pela Média Móvel para estudar o parâmetro de Hurst de fractais com dimensão arbitrária; Zhang (2005) implementa a Transformada Wavelet para obtenção e remoção da linha de base de sinais de eletrocardiograma; Kordestani e Zhang (2020) que implementam um filtro Savitzky-Golay com o objetivo de localizar e quantificar a tendência do dano em estruturas de pontes sujeitas a movimentos.

O objetivo deste trabalho é a comparação dos resultados da Transformada Wavelet e do Filtro Savitzky-Golay na estimação da componente de tendência em séries temporais. Para tanto, primeiramente utilizamos uma série temporal artificial a fim de compreender como as abordagens mencionadas atuam e, em seguida, implementamos uma aplicação para a série da cotação³ da moeda americana (Dólar U\$) frente à moeda brasileira (Real R\$).

FUNDAMENTOS

Série Temporal

Uma série temporal é uma realização do processo estocástico $X_t(\omega)$, $t \in Z$, definido no espaço $Z \times \Omega$, sendo que $X_t(\omega)$ é uma variável aleatória sobre um espaço de probabilidade (Ω, \mathcal{A}, P) para um t fixo, onde Ω é o conjunto de todos os eventos elementares possíveis, \mathcal{A} é uma coleção dos subconjuntos $A \in \Omega$, ω é um evento elementar, P uma função de probabilidade e $Z = \{0, 1, 2, 3, \dots\}$ é um conjunto de índices (Fuller, 1996).

A série temporal mencionada acima pode ser vista como uma coleção $\{X_t: t \in Z\}$ de variáveis aleatórias. Por questões de compatibilidade de notação em relação aos conceitos de processamento digital de sinais, podemos reescrever a série temporal $\{X_t: t \in Z\}$ como um função do tempo t , ou seja, $x(t)$ com $t \in \{0, 1, 2, \dots\}$.

Alguns modelos de séries temporais do tipo não-estacionária⁴ podem ser escritos como a soma entre uma componente de tendência T_t , uma sazonal S_t e uma aleatória ou irregular V_t (Fuller, 1996, p. 475), de acordo com a equação (1).

$$X_t = T_t + S_t + V_t, \quad (1)$$

onde V_t também pode ser composta por algum ruído.

Como mencionado anteriormente, um dos objetivos no estudo de séries temporais é a estimativa de suas componentes. Estimar a componente de tendência é importante, por exemplo, pois possibilita compreender o comportamento da série no longo prazo, enquanto que a estimação da componente aleatória, que representa a maior variabilidade da série, possibilita a compreensão de seu

³ Conforme disponibilizado pelo portal br.investing.com.

⁴ Em uma série não-estacionária a média e a variância da série são funções do tempo e a série é gerada por um mecanismo estocástico não-estacionário (Fuller, 1996).

comportamento no curto prazo (Schlüter e Deuschle, 2010). Várias técnicas podem ser aplicadas na estimação da tendência da série temporal, tais como os filtros Média Móvel, Savitzky-Golay e Transformada Wavelet. Nestes casos, todas implementam filtragens do tipo suavização.

Filtro Savitzky-Golay

No trabalho apresentado em Savitzky e Golay (1964), os autores propuseram um método de suavização de dados, com capacidade de aumentar a precisão destes sem distorcer a tendência do sinal, baseando-se na aproximação local polinomial por mínimos quadrados locais. Além disso, os autores mostraram que ajustar um polinômio a um conjunto de amostras de entrada e então avaliar o polinômio resultante em um único ponto dentro do intervalo de aproximação é equivalente à convolução discreta com uma resposta de impulso fixa. Os filtros passa-baixa obtidos por este método são amplamente conhecidos como filtros Savitzky-Golay (Schafer, 2011).

Várias pesquisas têm mostrado que a propriedade de preservação da forma de pico dos filtros Savitzky-Golay é interessante em aplicações de processamento de sinais, como por exemplo, no processamento de sinais de eletrocardiograma (Referência). Quando este filtro é generalizado para duas dimensões, pode também ser empregado em processamento de imagens (Schafer, 2011). Aplicações desse tipo de filtro no estudo de séries temporais também podem ser encontradas em diversos artigos, dentre eles: Wang e Pan e Xia (2019), Liu e Zhan (2016) e Cao et al. (2018).

Considera-se o grupo de $2M + 1$ amostras centradas em $n = 0$, de onde se obtém os coeficientes de um polinômio

$$p(n) = \sum_{k=0}^N a_k n^k, \quad (2)$$

que minimiza o erro quadrático médio da aproximação para o grupo de amostras de entrada centradas em $n = 0$,

$$\epsilon_N = \sum_{n=-M}^M (p(n) - x[n])^2 = \sum_{n=-M}^M \left(\sum_{k=0}^N a_k n^k - x[n] \right)^2. \quad (3)$$

A análise é a mesma para qualquer outro grupo de amostras de entrada $2M + 1$ (Schafer, 2011).

De acordo com Schafer (2011), a saída $y[0]$ em $n = 0$ é $y[0] = p(0) = a_0$, isto é, o valor da saída é igual ao 0-ésimo coeficiente polinomial. O valor da saída da próxima amostra é obtido deslocando o intervalo de análise para a direita em uma amostra e repetindo o ajuste polinomial e a avaliação no local central. Isso é repetido em cada amostra da entrada, e a cada iteração é produzido um valor da sequência de saída $y[n]$. Em cada posição, o valor de saída suavizado obtido pela amostragem do polinômio ajustado é idêntico a uma combinação linear fixa do conjunto local de amostras de entrada, ou seja, o conjunto de $2M + 1$ amostras de entrada dentro do intervalo de aproximação são

efetivamente combinados por um conjunto fixo de coeficientes de ponderação que podem ser calculados uma vez para uma dada ordem polinomial N e intervalo de aproximação de comprimento $2M + 1$. Assim, o vetor de saída pode ser obtido por uma convolução discreta da forma

$$y[n] = \sum_{m=-M}^M h[m]x[n-m] = \sum_{m=n-M}^{n+M} h[n-m]x[m]. \quad (4)$$

Para determinar os coeficientes do filtro segue-se o seguinte procedimento: define-se o vetor dos coeficientes do polinômio $\mathbf{a} = [a_0, a_1, \dots, a_N]^T$ e o vetor de entrada $\mathbf{x} = [x_{-M}, \dots, x_{-1}, x_0, x_1, \dots, x_M]^T$, onde $x_n = x(n)$, e a matriz

$$A = \begin{bmatrix} (-M)^0 & (-M)^1 & \dots & (-M)^N \\ \vdots & \vdots & \ddots & \vdots \\ (-1)^0 & (-1)^0 & \dots & (-1)^0 \\ 1 & 0 & \dots & 0 \\ M^0 & M^1 & \dots & M^N \end{bmatrix}. \quad (5)$$

Deriva-se ϵ_N em relação a esses coeficientes e definimos as derivadas iguais a zero, levando $N + 1$ equações em $N + 1$ incógnitas como segue

$$\sum_{k=0}^N \left(\sum_{n=-M}^M n^{i+k} \right) a_k = \sum_{n=-M}^M n^i x[n], \quad (6)$$

$i = 0, 1, \dots, N$; e então escreve-se essa última equação na sua forma matricial:

$$(A^T A)^{-1} \mathbf{a} = A^T \mathbf{x}; \quad (7)$$

e a partir daí, o vetor de coeficientes pode ser obtido como

$$\mathbf{a} = (A^T A)^{-1} A^T \mathbf{x} = H \mathbf{x}. \quad (8)$$

É importante destacar que é necessário calcular apenas o coeficiente \mathbf{a}_0 , e portanto, determinar a primeira linha da matriz H é suficiente. A matriz H é independente das amostras de entrada e depende apenas da ordem do filtro (N) e do comprimento da janela (M). Então, pela definição de multiplicação de matrizes, a saída será

$$y[0] = a_0 = \sum_{m=-M}^M h_{0,m} x[m], \quad (9)$$

e $h_{0,m} = h[-m]$, onde $-M \leq m \leq M$ (Schafer, 2011).

Transforma Wavelet Discreta

A Transforma Wavelet Discreta, do inglês, *Discrete Wavelet Transform* (DWT), é uma ferramenta fundamental para o estudo de inúmeras aplicações em diversas áreas científicas, entre elas, as séries

temporais. A DWT pode ser vista como uma ferramenta equivalente a Transformada de Fourier, porém com características distintas que a tornam relevante (Percival e Walden, 2000).

Os coeficientes de decomposição em uma base ortogonal de funções wavelet são calculados com um algoritmo rápido, baseado em convoluções discretas com vetores h e g , seguidos de subamostragem na saída. Trata-se de um banco de filtros de dois canais que divide o sinal de entrada em baixas e altas frequências, e são descritos pelas bases ortonormais $\{\psi_{j,n}\}_{n \in \mathbb{Z}}$ e $\{\phi_{j,n}\}_{n \in \mathbb{Z}}$. As funções dessas bases são chamadas funções wavelet e escala, respectivamente, e são dadas pelas equações (10) e (11).

$$\psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t - 2^j n}{2^j}\right), \quad (10)$$

$$\phi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \phi\left(\frac{t - 2^j n}{2^j}\right). \quad (11)$$

Essas fórmulas geram, de acordo com Mallat (2009), por meio das equações recursivas (12) e (13), os coeficientes dos filtros h e g , respectivamente:

$$h[n] = \left\langle \frac{1}{\sqrt{2}} \phi\left(\frac{t}{2}\right), \phi(t - n) \right\rangle, \quad (12)$$

$$g[n] = \left\langle \frac{1}{\sqrt{2}} \psi\left(\frac{t}{2}\right), \phi(t - n) \right\rangle. \quad (13)$$

Uma série $x(t)$ é decomposta pela Transformada Rápida Wavelet, do inglês, *Fast Wavelet Transform* (FWT), como $a_j[n] = \langle x(t), \phi_{j,n}(t) \rangle$ e $d_j[n] = \langle x(t), \psi_{j,n}(t) \rangle$, onde $a_j[n]$ são os coeficientes de aproximação na escala 2^j , sendo $j (j = 1, \dots, J)$ o nível de decomposição. Nas mesmas condições, $d_j[n]$ são os coeficientes de detalhes. Considerando esses coeficientes, a série $x(t)$ é composta pelos coeficientes de aproximação na última escala J e os coeficientes de detalhes nas escalas j , isto é,

$$x(t) = \sum_n a_j[n] \phi_{j,n}(t) + \sum_{j \leq J} \sum_n d_j[n] \psi_{j,n}(t), \quad (14)$$

(Mallat, 2009; Oliveira et al., 2018).

Na FWT, a série no domínio do tempo é convertida para uma representação nas bases wavelets, porém, podendo ser reconstruído no domínio anterior por meio da Transformada Rápida Wavelet Inversa, do inglês, *Inverse Fast Wavelet Transform* (IFWT). A reconstrução é feita por meio de filtros de quadratura espelhada. Os filtros da reconstrução serão os mesmos da decomposição apenas se o banco de filtros for ortogonal (Strang e Nguyen, 1996). Estas duas operações (decomposição e reconstrução)

são denominadas análise e síntese da série temporal, e funcionam conforme esquematizado na Figura 1.

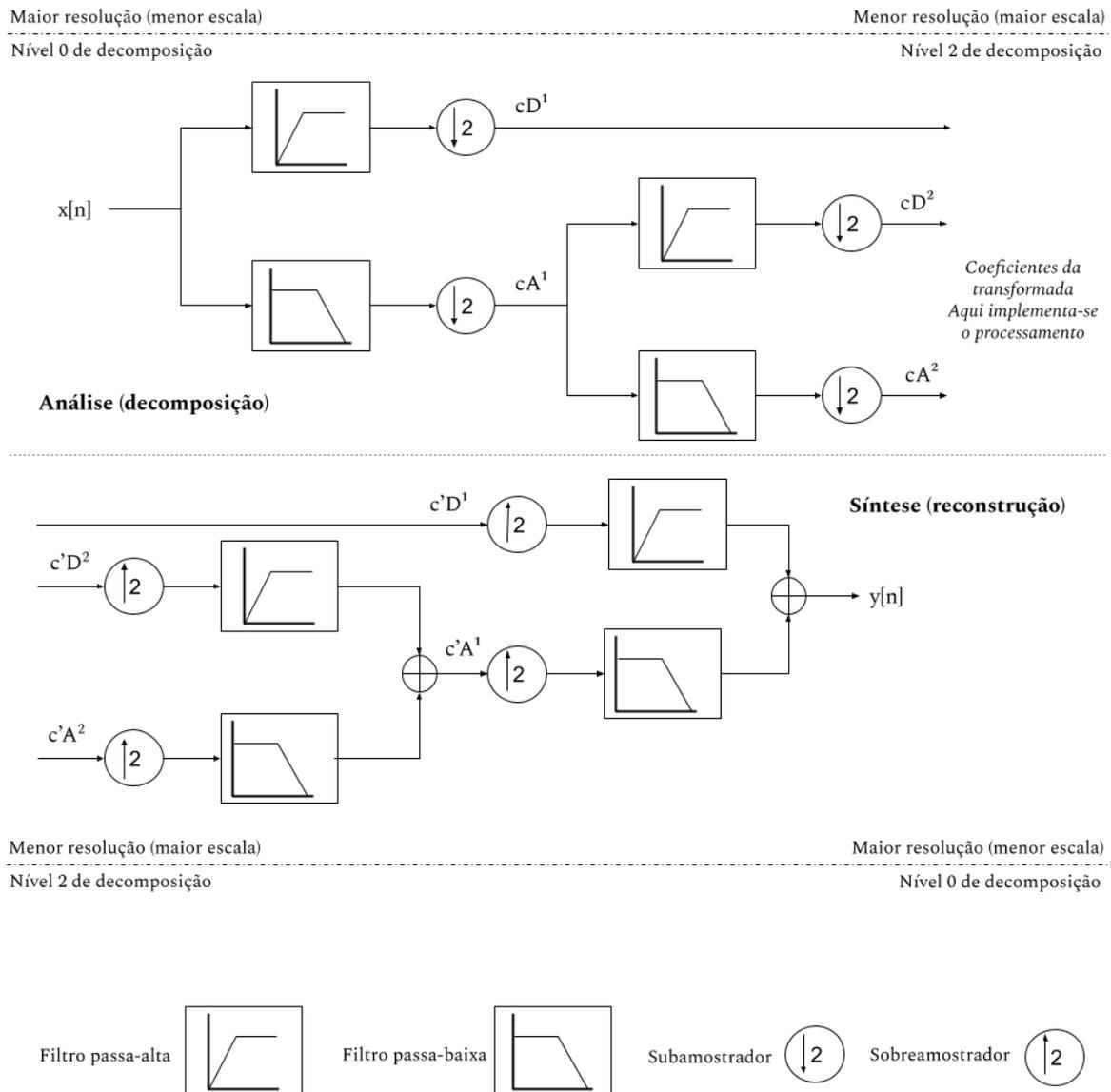


Figura 1. Banco de filtros de dois canais implementado pela Transformada Wavelet Discreta. Fonte: os autores.

De acordo com Stachura (2014), em geral, uma série temporal $x(t)$ é decomposta por meio da análise de multirresolução como

$$x(t) = S(t) + \sum_{j=1}^J D_j(t), \tag{15}$$

onde $S_j(t)$ denota as componentes cíclicas com periodicidade maior que 2^j períodos, e todos os $D_j(t)$ denotam componentes cíclicas com periodicidade entre 2^{j-1} e 2^j períodos. A componente $S(t)$

representa uma tendência (aproximação) de $x(t)$ e os $D_j(t)$ podem representar negócios e ciclos sazonais (detalhes), exceto por $D_1(t)$ que é da maior frequência e representa algum ruído.

A Equação (15) pode ser escrita na forma da Equação (1), ou vice-versa, porém esta última nos leva diretamente a uma associação com a Equação (14). Considere, na Equação (1), $X_t = x(t)$ e:

$$T_t = \sum_n a_j[n] \phi_{j,n}(t), \quad (16)$$

$$S_t + V_t = \sum_{j \leq J} \sum_n d_j[n] \psi_{j,n}(t). \quad (17)$$

Para se obter apenas a tendência T_t na equação (1), uma alternativa bem simples seria considerar todas as componentes de detalhes com $d_j[n] = 0$, o que anularia $S_t + V_t$ (Oliveira et al., 2018). Caso se queira considerar algum ruído, deve-se ter em mente que o mesmo é dado por V_t na Equação (1), e que na Equação (7) é dado por $D_1(t)$.

MATERIAL

Visto que o objetivo do presente trabalho é a estimação da componente de tendência em um série temporal, é necessário que tenhamos acesso a essa componente isoladamente, para comparar com a componente estimada pelo método de suavização aplicado. Uma vez que para séries temporais reais a componente de tendência não é conhecida, propomos a construção de uma série temporal artificial. Esta série é composta pela componente de tendência

$$T_t = \sin(\pi x), \quad (18)$$

com período de 2 segundos, e pela componente irregular (onda dente de serra - *Sawtooth wave*)

$$V_t = \frac{-2}{\pi} \arctan \left(\cot \left(\frac{x\pi}{7} \right) \right), \quad (19)$$

com período de 7 segundos, onde $0 < x \leq 4$ foi amostrado a 100Hz por 4 segundos. A Figura 2 painel (a) ilustra a série temporal artificial X_t , painel (b) a tendência T_t e painel (c) a componente irregular V_t .

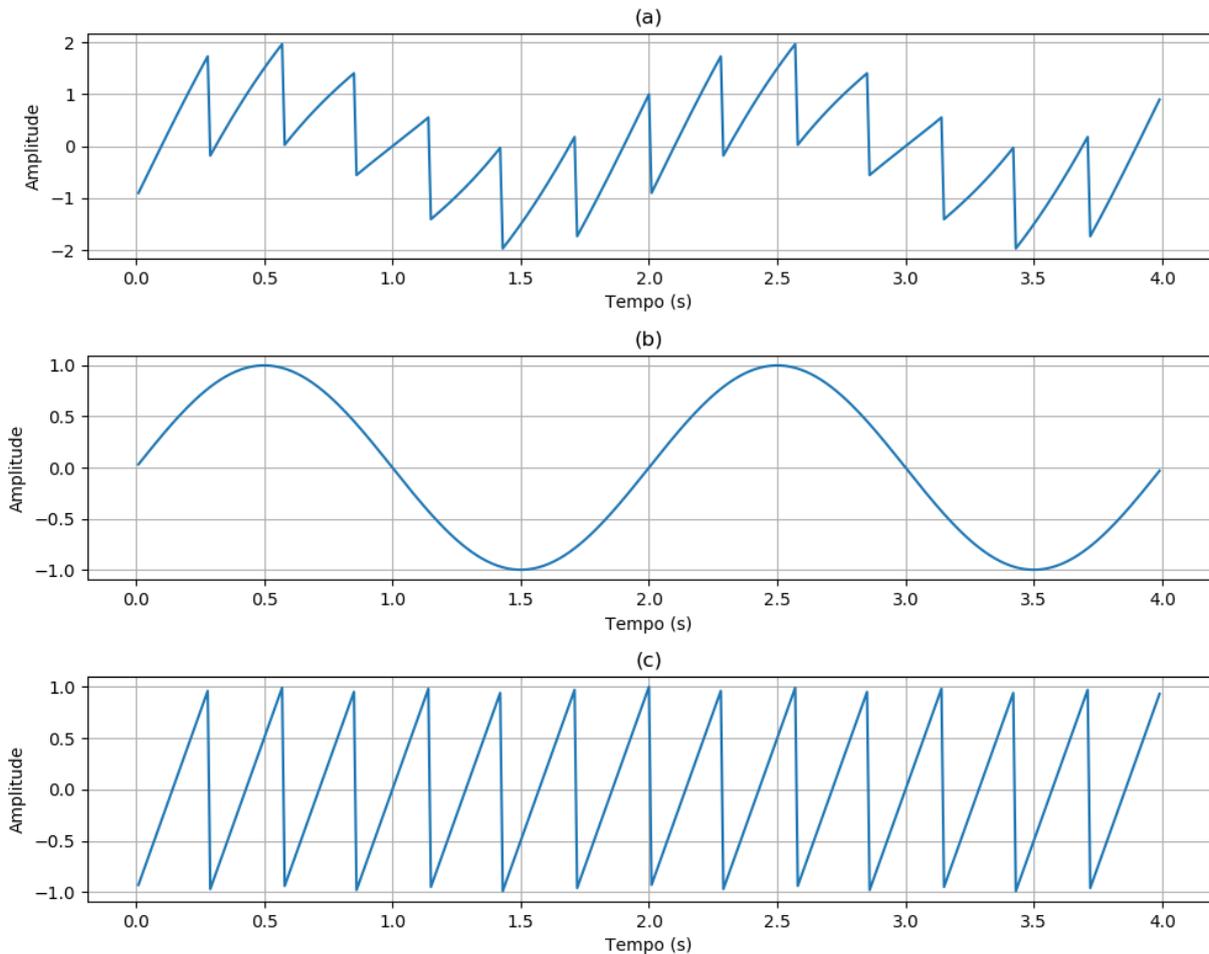


Figura 2. (a) Série temporal, (b) componente de tendência dada pela Equação (10) e (c) componente irregular dada pela Equação (11). Fonte: os autores.

Para a análise de uma série temporal formada por dados reais, que envolve, geralmente, um processo de formação desconhecido e dependente de vários fatores, utilizamos a série da cotação do Dólar Americano frente ao Real Brasileiro, para todo o ano de 2019. Restringimos nossa análise apenas aos valores máximos das cotações diárias do Dólar. A série foi obtida do site br.invesing.com e é composta de 261 amostras. Entretanto, preferimos truncar a série para 256 amostras, somente para melhor visualização dos resultados.

RESULTADOS E DISCUSSÕES

O primeiro experimento consistiu da estimação da tendência da série artificial, descrita na seção anterior, utilizando o método baseado na DWT para várias bases wavelets (com diferentes momentos nulos) de Daubechies e o filtro Savitzky-Golay empregando polinômios de várias ordens. Para esta última abordagem foi considerada uma janela de tamanho 201 amostras, que correspondente a um período da série de tendência mais um, pois o filtro espera uma janela de tamanho ímpar. Para comparar

as tendências estimadas com a tendência real, considerou-se o erro quadrático médio (*Mean Square Error* - MSE), dado pela equação.

$$MSE = \frac{1}{N} \sqrt{x - \hat{x}}, \tag{20}$$

onde N é a quantidade de valores da série original x , sendo \hat{x} a sua estimação. Os resultados obtidos com esta implementação estão exibidos na Tabela 1.

Tabela 1. Resultados MSE do experimento com a série temporal artificial, considerando várias bases wavelets e ordens de polinômio para o filtro Savitzky-Golay.

Base Wavelet	MSE DWT	Ordem Polinômio	MSE Savitzky-Golay
Db2	165,79	1	153,96
Db3	26,77	2	23,31
Db4	8,81	3	23,31
Db5	2,17	4	3,25
Db6	3,35	5	3,25
Db7	22,89	6	2,49
Db8	25,18	7	2,49
Db9	24,30	8	199,99

Pelos resultados da Tabela 1, o menor MSE foi obtido ao empregar o método baseado na DWT com a base Db5. A fim de averiguar que a superioridade do método baseado na DWT não é apenas devida às características da série artificial ilustrada na Figura 1, propõe-se a obtenção de outras 20 séries temporais baseadas nos mesmos modelos das Equações (18) e (19), mas variando seus períodos, isto é: a tendência T_t tem um período que varia aleatoriamente entre 1 e 5 segundos, o que corresponde a senoides de 0,1Hz a 0,5Hz; já a componente irregular ou sazonal, V_t , tem seu período variando aleatoriamente entre 100 e 1000 segundos, correspondendo a frequências de 1Hz a 10Hz. Analogamente ao experimento anterior, variou-se as bases wavelets de Daubechies e a ordem dos polinômios do filtro Savitzky-Golay.

Tabela 2. Resultados para 20 experimentos onde foram escolhidos aleatoriamente os períodos das componentes de tendência e irregular. Aqueles experimentos cujos valores dos períodos são iguais não estão discriminados. Fonte: os autores.

Frequências (Hz)	MSE		Tempo de processamento (segundos)	
	DWT	Savitzky-Golay	DWT	Savitzky-Golay
$T_t: 0,1$ e $V_t: 5$	0,0177	0,0055	0,0311	0,0000
$T_t: 0,1$ e $V_t: 3$	0,0299	0,0115	0,0156	0,0311
$T_t: 0,2$ e $V_t: 2$	0,0638	0,0297	0,0000	0,0155
$T_t: 0,1$ e $V_t: 1$	0,1221	0,0613	0,0155	0,0155
$T_t: 0,4$ e $V_t: 3$	0,0598	0,0321	0,0155	0,0312
$T_t: 0,3$ e $V_t: 1$	0,2165	0,1337	0,0155	0,0000
$T_t: 0,4$ e $V_t: 4$	0,0354	0,0168	0,0156	0,0155
$T_t: 0,3$ e $V_t: 2$	0,1268	0,0243	0,0155	0,0312
$T_t: 0,3$ e $V_t: 4$	0,0316	0,0081	0,0155	0,0000
$T_t: 0,2$ e $V_t: 7$	0,0089	0,0028	0,0155	0,0312
$T_t: 0,4$ e $V_t: 8$	0,0093	0,0059	0,0155	0,0156
$T_t: 0,1$ e $V_t: 9$	0,0069	0,0014	0,0156	0,0155
$T_t: 0,3$ e $V_t: 7$	0,0080	0,0036	0,0156	0,0000
$T_t: 0,3$ e $V_t: 5$	0,0165	0,0074	0,0156	0,0000
$T_t: 0,3$ e $V_t: 6$	0,0113	0,0050	0,0156	0,0000
$T_t: 0,1$ e $V_t: 4$	0,0214	0,0083	0,0155	0,0311
Média	0,0565	0,0223	0,0154	0,0164
Estatística H	6,611		0,005	
p -valor	0,010		0,945	

Analisando os resultados da Tabela 2, o valor da estatística H do teste de Kruskal-Wallis e os p -valores associados, e considerando que a hipótese nula H_0 seja que ambas as abordagens possuem a mesma distribuição, para um nível de significância de 5%, concluímos pela rejeição desta hipótese em relação ao MSE e pela sua aceitação em relação ao tempo de processamento. Portanto, como a média para o filtro Savitzky-Golay é menor, então esta abordagem é melhor que o método baseado na DWT para as variações da série artificial analisada. Por outro lado, em relação ao tempo de processamento, ambas abordagens são estatisticamente iguais.

A Figura 2 ilustra as componentes de tendências com menores MSE (obtidas dos 20 experimentos aleatórios da Tabela 2), estimadas pelas abordagens implementadas em comparação com a tendência original. Podemos destacar dessas comparações que a aplicação do filtro Savitzky-Golay resulta em uma componente de tendência estimada com algumas distorções nos picos e vales dessa componente, enquanto que a estimação resultante do método baseado na DWT é mais suave e similar à tendência original. Porém, podemos também observar que nas bordas da série o filtro Savitzky-Golay provê uma estimação mais próxima à componente de tendência original que aquela estimação obtida pelo método baseado na DWT, sendo este o principal motivo desse método ter apresentado um MSE menor, pois podemos notar que nos demais segmentos, a componente estimada pelo método da DWT coincide muito bem com a componente original.

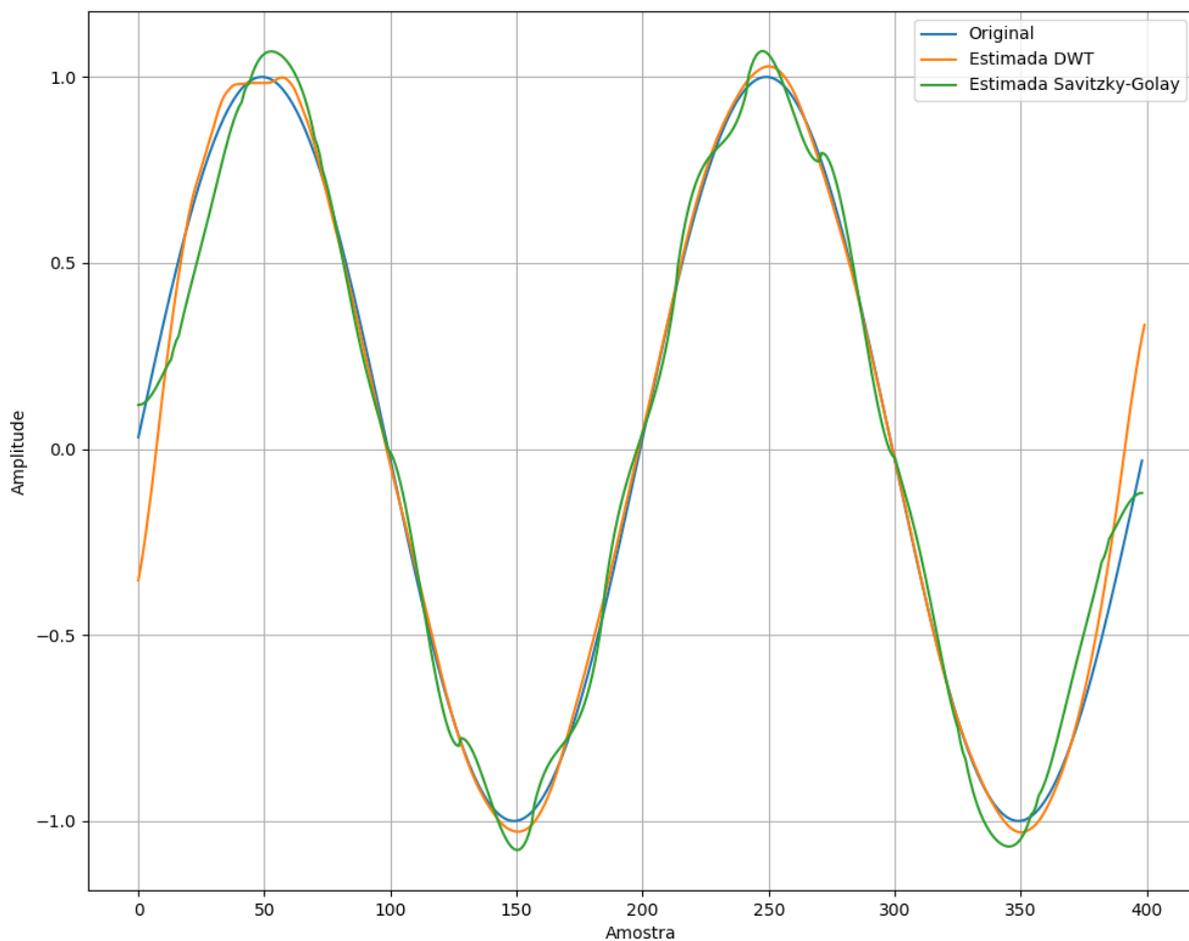


Figura 2. Comparação das séries de tendências estimadas pelo método baseado na DWT (curva alaranjada) e pelo filtro Savitzky-Golay (curva esverdeada), com a tendência original (curva azul). Fonte: os autores.

Para verificar se é o método de extensão das bordas que está ocasionando os problemas mencionados, outros 20 experimentos aleatórios foram executados, mas desta vez alternando os

métodos de extensão de bordas utilizados pela DWT. As médias de 20 MSEs estão exibidos na Tabela 3 para distintos modos de extensão das bordas da série temporal.

Tabela 3. Resultados médios dos MSEs de 20 experimentos considerando diferentes métodos de extensão das bordas da série temporal. Fonte: os autores.

Média dos MSEs		Modo de extensão da DWT*	Estatística H	p-valor	Hipótese H_0
DWT	Savitzky-Golay				
0,0305	0,0286	zero	0,059	0,808	Não rejeita
0,0601	0,0370	constante	5,418	0,020	Rejeita
0,0505	0,0272	simétrico	8,240	0,004	Rejeita
0,0202	0,0143	periódico	0,165	0,685	Não rejeita
0,0415	0,0172	suave	16,921	0,000	Rejeita
0,0211	0,0248	antissimétrico	0,237	0,626	Não Rejeita
0,0525	0,0158	antirreflexo	14,971	0,000	Rejeita

* Para informações sobre os modos de extensão consultar a biblioteca PyWavelets: <https://pywavelets.readthedocs.io/en/latest/ref/signal-extension-modes.html#modes>.

Nota-se pelos resultados da Tabela 3, analisando a última coluna, que apenas para os modos de extensão denominados de zero, periódico e antissimétrico, os resultados apresentados pelo filtro Savitzky-Golay foram estatisticamente superiores, tomando um nível de significância de 5%. Para os demais modos, ambas abordagens não apresentaram resultados estatisticamente diferentes. Portanto, pelo que foi destacado anteriormente, podemos afirmar que para os demais modos os problemas evidenciados nas bordas da série temporal afetam menos a estimação da componente de tendência. Vale destacar que para os experimentos cujos resultados estão discriminados na Tabela 2, foi empregado o modo de extensão denominado de reflexão.

No quarto experimento, foi adicionado à componente de tendência um ruído com distribuição gaussiana, com média nula e variância unitária. Analogamente aos dois experimentos anteriores, também variamos aleatoriamente os períodos das componentes de tendência e irregular nos mesmos intervalos anteriormente mencionados. Os resultados estão exibidos na Tabela 4.

Tabela 4. Resultados para 20 experimentos onde foram escolhidos aleatoriamente os períodos das componentes de tendência e irregular e adicionado um ruído com distribuição gaussiana à componente de tendência. Fonte: os autores.

Frequências (Hz)	MSE		Tempo de processamento (segundos)	
	DWT	Savitzky-Golay	DWT	Savitzky-Golay
$T_t: 0,3$ e $V_t: 5$	0,0388	0,0220	0,0155	1,5422
$T_t: 0,2$ e $V_t: 7$	0,0348	0,0382	0,0160	1,5210
$T_t: 0,2$ e $V_t: 2$	0,0387	0,0316	0,0170	1,7090
$T_t: 0,1$ e $V_t: 9$	0,0223	0,0122	0,0170	1,6660
$T_t: 0,4$ e $V_t: 9$	0,0207	0,0200	0,0160	1,5710
$T_t: 0,2$ e $V_t: 9$	0,0378	0,0196	0,0170	1,7430
$T_t: 0,3$ e $V_t: 4$	0,0256	0,0171	0,0270	1,9791
$T_t: 0,1$ e $V_t: 7$	0,0581	0,0142	0,0160	1,7330
$T_t: 0,1$ e $V_t: 3$	0,0316	0,0081	0,0155	0,0000
$T_t: 0,4$ e $V_t: 6$	0,0382	0,0291	0,0230	1,7851
$T_t: 0,4$ e $V_t: 8$	0,0350	0,0312	0,0190	1,9991
$T_t: 0,4$ e $V_t: 5$	0,0256	0,0181	0,0220	1,6610
$T_t: 0,1$ e $V_t: 6$	0,0119	0,0090	0,0150	1,9321
$T_t: 0,3$ e $V_t: 7$	0,0406	0,0333	0,0270	1,8411
$T_t: 0,2$ e $V_t: 8$	0,0246	0,0166	0,0210	1,5700
$T_t: 0,4$ e $V_t: 7$	0,0226	0,0180	0,0170	1,5150
Média	0,0326	0,0224	0,0192	1,7096
Estatística H	7,317		29,301	
p -valor	0,007		0,0000	

Pelos resultados da Tabela 4, notamos que, similarmente aos resultados da Tabela 2, considerando a estatística H e o p -valor, o filtro Savitzky-Golay gerou resultados superiores ao método baseado na DWT, mas apenas em relação ao MSE, pois para o tempo de processamento, esta última abordagem superou a primeira, pois em média foi aproximadamente 8,9 vezes menor, considerando a mesma hipótese nula e nível de significância estatística.

No quinto e último experimento, considerou-se a série temporal da cotação do Dólar Americano frente ao Real Brasileiro, discriminada na Seção Materiais. As abordagens baseadas na DWT e no filtro Savitzky-Golay foram empregadas com as seguintes configurações: base Db5 e quatro níveis

de resolução; e polinômio de ordem 6 e vários comprimentos de janelas, respectivamente, devido aos resultados constatados anteriormente. A Figura 3 ilustra os resultados obtidos para ambas abordagens, além de adicionar os resultados fornecidos pelo filtro Média Móvel, para fins de comparação.

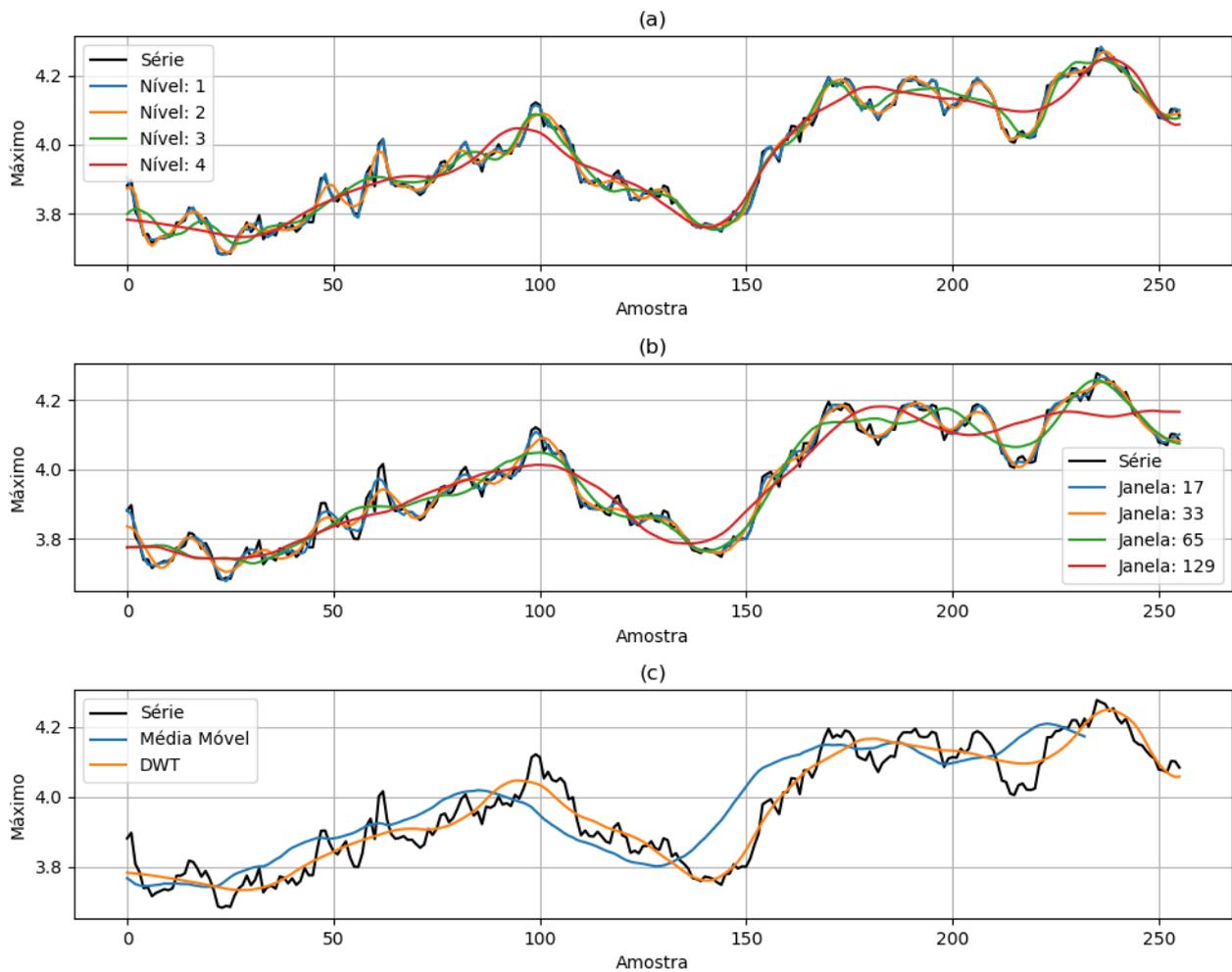


Figura 3. Estimações das componentes de tendências obtidas pelo método baseado na DWT (a) e pelo filtro Savitzky-Golay (b) e comparação das estimativas DWT e Média Móvel (c), para várias configurações. Fonte: os autores.

Nota-se, pela análise dos resultados da Figura 3, que as formas de onda das componentes de tendência estimadas são similares para níveis menores de resolução da DWT e tamanhos menores de janelas do filtro Savitzky-Golay. Ou seja, quanto maior o nível de resolução menos amostras temos nos coeficientes de aproximação devido a operação de subamostragem no banco de filtros, logo, estes coeficientes estão relacionados a uma porção maior da série temporal. Por isso, ao tomarmos uma janela de interpolação maior para o filtro Savitzky-Golay, obtemos resultados similares às estimações obtidas pela DWT, concernentes à tendência da série. Assim, as abordagens apresentam resultados análogos, mas com certas peculiaridades, para parâmetros específicos do nível de resolução e comprimento da janela.

Portanto, quando se pretende analisar as tendências de curto prazo de uma série temporal, deve-se escolher níveis de resolução mais altos para o método baseado na DWT e comprimentos de janela maiores para o filtro Savitzky-Golay, lembrando ainda que a escolha da base wavelet e da ordem do polinômio também influenciam nas estimações. Por outro lado, na análise das tendências de longo prazo, deve-se empregar níveis de resolução mais baixo e janelas de comprimentos menores. Esta constatação é evidente na estimação da tendência exibida na Figura 3 (curva azul), para um nível de resolução, pois a estimação quase que coincide completamente com a série original, enquanto que, na estimação do quarto nível, temos uma tendência que é relativa a média de cerca de 24 dias, já que o período de amostragem da série é de 1 dia, e no quarto nível analisamos as frequências entre $2^4 = 16$ e $2^5 = 32$ amostras (dias), assim:

$$24 = 2^4 + \frac{2^5 - 2^4}{2}. \quad (21)$$

Este resultado pode ser destacado na comparação das estimações da componente de tendência, Figura 3 (c), onde o resultado da DWT no quarto nível de resolução é comparado ao resultado obtido por um filtro Média Móvel empregando 24 elementos para calcular a média. Exceto pelo deslocamento natural da estimação pela Média Móvel, pois este método precisa das amostras futuras para calcular a média, ambas estimações são bem semelhantes em relação ao comportamento da componente de tendência da série.

CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentado e estudado duas abordagens para estimação da componente de tendência de uma série temporal, a DWT e o filtro Savitzky-Golay. Os resultados foram avaliados utilizando o MSE e mostraram que, para a série temporal artificial gerada, o filtro Savitzky-Golay fornece melhores estimativas da componente de tendência dependendo do modo de extensão de bordas utilizado pela DWT. Os resultados mostraram que a abordagem via DWT possui algumas deficiências na estimação da componente de tendência nas bordas da série temporal. Quando utilizado o modo antissimétrico de extensão de bordas, o teste estatístico de Kruskal-Wallis, considerando uma significância estatística de 5%, estabeleceu que não existe diferença estatística entre as abordagens. O mesmo ocorre para o tempo de processamento, mas neste caso o resultado independe do modo de extensão escolhido.

Por outro lado, ao considerar ruído com distribuição gaussiana adicionado à componente de tendência artificial, verificamos que o filtro Savitzky-Golay resultou em um tempo de processamento cerca de 8 vezes maior que o método baseado na DWT, o que evidencia a vantagem da DWT principalmente para aplicações em tempo real.

Nas análises da série temporal do par de moedas dólar/real, constatamos que ambas as abordagens apresentaram resultados similares para certas escolhas dos parâmetros de nível de resolução e comprimento da janela, para as abordagens via DWT e filtro Savitzky-Golay, respectivamente. Nestes casos, uma comparação numérica não foi possível pois a componente de tendência original não é conhecida. Outro fato interessante constatado foi em relação à comparação da DWT com o filtro Média Móvel, pois observou-se que é possível ajustar o nível de resolução da DWT para que este tenha resultados similares ao filtro Média Móvel.

Em trabalhos futuros pretende-se investigar com maior profundidade todas as análises aqui realizadas, sendo que um dos objetivos prováveis será o emprego da *Wavelet Neural Networks*, que possibilita obter bases wavelets adaptadas aos dados analisados.

REFERÊNCIAS BIBLIOGRÁFICAS

- Carbone A (2009). *Detrending Moving Average algorithm: a brief review*. In: 2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH), DOI: 10.1109/TIC-STH.2009.5444412.
- Cao R, Chen Y, Shen M, Chen J, Zhou J, Wang C, Yang W (2018). *A simple method to improve the quality of NDVI time-series data by integrating spatiotemporal information with the Savitzky-Golay filter*. Remote Sensing of Environment, (217): 244-257.
- Fuller W A (1996). *Introduction to Statistical Time Series*. 2 ed. John Wiley & Sons, Inc., New York. 720p.
- Fu TC, Chung FL, Luk R, Ng CM (2008). *Representing financial time series based on data point importance*. Engineering Applications of Artificial Intelligence, 21(2): 277-300.
- Kordestani H, Zhang C (2020). *Direct Use of the Savitzky-Golay Filter to Develop an Output-Only Trend Line-Based Damage Detection Method*. Sensors: 1-15. DOI: 10.3390/s20071983.
- Liu J, Pei Z (2016). *The impacts of smoothing methods for time-series remote sensing data on crop phenology extraction*. IEEE International Geoscience and Remote Sensing Symposium (IGARSS). IEEE.
- Mallat, S (2009). *A wavelet tour of signal processing: The sparse way*. 3 ed. Editora: Academic Press, London.
- Oliveira BR de, Duarte MAQ, Abreu CCE de, Vieira Filho J (2018). *A wavelet-based method for power-line interference removal in ECG signals*. Res. Biomed. Eng, 34(1): 73-86. DOI: 10.1590/2446-4740.01817.
- Percival DB, Walden A. (2000). *Wavelet methods for time series analysis*, Cambridge University Press: Cambridge.
- Picasso A, Merello S, Ma Y, Oneto L, Cambria E (2019). *Technical analysis and sentiment embeddings for market trend prediction*. Expert Systems with Applications, 135: 60-70.

- Puchalsky W, Ribeiro GT, Veiga CP, Freire RZ, dos Santos CL (2018). *Agribusiness time series forecasting using Wavelet neural networks and metaheuristic optimization: An analysis of the soybean sack price and perishable products demand*. International Journal of Production Economics, (203): 174-189.
- Radityo A, Munajat Q, Budi I (2017). *Prediction of Bitcoin exchange rate to American dollar using artificial neural network methods*. In 2017 International Conference on Advanced Computer Science and Information Systems (ICACISIS): 433-438.
- Schlüter S, Deuschle C (2010). Using wavelets for time series forecasting: Does it pay off?. IWQW Discussion Papers, No. 04/2010, Friedrich-Alexander-Universität Erlangen-Nürnberg, Institut für Wirtschaftspolitik und Quantitative Wirtschaftsforschung (IWQW), Nürnberg.
- Stachura M (2014). *Detecting seasonality via wavelet methods*. Studia Ekonomiczne, 207:223-232.
- Strang, G. e Nguyen, T. (1996). *Wavelets and filter banks*. Wellesley, MA: Wellesley-Cambridge Press. (Livro)
- Savitzky A, Golay M (1964). *Smoothing and differentiation of data by simplified least squares procedures*. Analytical chemistry, 36(8): 1627-1639.
- Schafer RW (2011). *What is a savitzky-golay filter?*[lecture notes]. IEEE Signal processing magazine 28(4): 111-117.
- Xiaoxiong W, Pan M, Xia C (2019). *Water level data preprocessing method based on savitzky-golay filter*. 2019 International Conference on Modeling, Simulation and Big Data Analysis (MSBDA 2019). Atlantis Press
- Zhang D (2005). *Wavelet Approach for ECG Baseline Wander Correction and Noise Reduction*. In: 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference. DOI: 10.1109/IEMBS.2005.1616642.

Aplicação de Redes Neurais Artificiais LSTM em domótica inteligente

Recebido em: 03/07/2020

Aceito em: 02/09/2020

 10.46420/9786588319260cap2

Gilberto de Melo Júnior^{1*} 

Matheus Chagas Lee² 

Sanderson Oliveira de Macedo¹ 

INTRODUÇÃO

Muitos avanços vêm acontecendo na área de Inteligência Artificial (IA). Um dos maiores ramos da IA é o Aprendizado de Máquina, onde a abordagem principal é como construir computadores que com experiências e aprendizado possam automaticamente ser melhorados (Jordan e Mitchell, 2015).

Um dos mais utilizados algoritmos de aprendizado de máquina são as Redes Neurais Artificiais (RNAs). Esses algoritmos são modelos abstratos de computadores do que simulam em suas peculiaridades o cérebro humano (Munakata, 2008). Há atualmente várias arquiteturas de RNAs, como por exemplo, Redes Multilayer Perceptrons (MLPs), Redes Neurais Convolucionais (CNNs), Redes de Hopfield, Redes Neurais Recorrentes dentre outras.

As Redes Neurais Recorrentes são redes especialmente voltadas para o processamento de dados sequenciais, linguagem natural e séries temporais. Sua característica fundamental é que ela contém pelo menos uma conexão de *feedback*, para que as ativações possam fluir em um *loop*. Assim, são permitidas a essas redes neurais realizem processamento temporal e aprendam sequências, por exemplo, realizam reconhecimento/reprodução de sequência ou associação/previsão temporal (Zaccone et al., 2017).

As Redes Neurais *Long Short-Term Memory* (LSTM), ou em português, Redes Neurais de Memória de Longo Prazo foram elaboradas em meados dos anos 90 para a solução de um problema comum nas Redes Neurais Recorrentes, o retorno do erro. Ele pode aprender a preencher intervalos de tempo superiores a 1000 etapas mesmo no caso de sequências de entrada incompressíveis e ruidosas, sem perda de recursos de curto espaço de tempo, o que é conseguido por um algoritmo eficiente e baseado em gradiente para uma arquitetura que impõe um fluxo de erro constante (não explodindo nem desaparecendo) através dos estados internos de unidades especiais (desde que o cálculo do gradiente

¹ Docente do Departamento IV no Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

² Discente do curso de Engenharia de Controle e Automação no Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

* Autor correspondente: gilberto.melo@outlook.com

seja truncado em determinados pontos específicos da arquitetura isso não afeta o fluxo de erro de longo prazo) (Hochreiter e Schmidhuber, 1997).

Uma área encontrada para aplicação da Redes Neurais LSTM foi a domótica. A definição de Domótica está relacionado a instalações eletroeletrônicas e sistemas inteligentes no ambiente residencial, permitindo uma gestão, local ou remota, para oferecer aplicações nas áreas de segurança, conforto e comunicação (Linz e Moura, 2010).

Takiuchi et al. (2004) estende o conceito de domótica, a domótica inteligente, onde principal é analisar os dados obtidos pelos sensores de modo a adaptar suas regras de automação ao comportamento dos habitantes. Isso é necessário, pois o comportamento humano sofre mudanças ao longo tempo.

O objetivo desse estudo foi o desenvolvimento e avaliação de um algoritmo usando Redes Neurais Artificiais LSTM para reconhecimento de padrões de comportamento de presença de pessoas em uma residência e assim realizar a predição de ações na residência por meio dos dados coletados pelos sensores, assim, obtendo uma base para o desenvolvimento de um sistema domótico inteligente.

MATERIAL E MÉTODOS

O trabalho inicial foi a realização da coleta de dados para utilização nos algoritmos de predição. A casa de um casal foi o local para aquisição dos dados de presença dos mesmos. Com a anuência do casal, foi instalado dois sensores de presença PIR-HC-SR501.

O PIR-HC-SR501 é um sensor de movimento simples, muito utilizado no mercado, e compatível com muitos microcontroladores. O sensor pode ser alimentado com uma tensão de 5 a 20 volts. O seu range de leitura é entre 3 e 7 metros, assim qualquer movimento dentro do seu range o pino de saída vai para nível lógico alto (Datasheet, 2016).

Para este trabalho foi utilizado dois sensores PIR para aquisição de dados de presença em dois locais diferentes na casa. A disposição dos sensores é ilustrada na Figura 1, onde as marcações em vermelho na planta da casa, indicam onde os mesmos foram instalados.

Para a aquisição dos dados, os sensores foram conectados a placa NodeMCU, plataforma microcontroladora da família ESP8266, criada para o desenvolvimento de projetos de *Internet of Things* (IoT), em português, Internet das coisas (Al Dahoud e Mohamed, 2018). Entre as particularidades dessa placa, destaca-se o baixo custo, suporte integrado a redes WiFi, tamanho reduzido e baixo consumo de energia.

A Figura 2 A) ilustra a pinagem do sensor PIR HC-SR501 e a Figura 2 B) a placa microcontroladora NodeMCU.

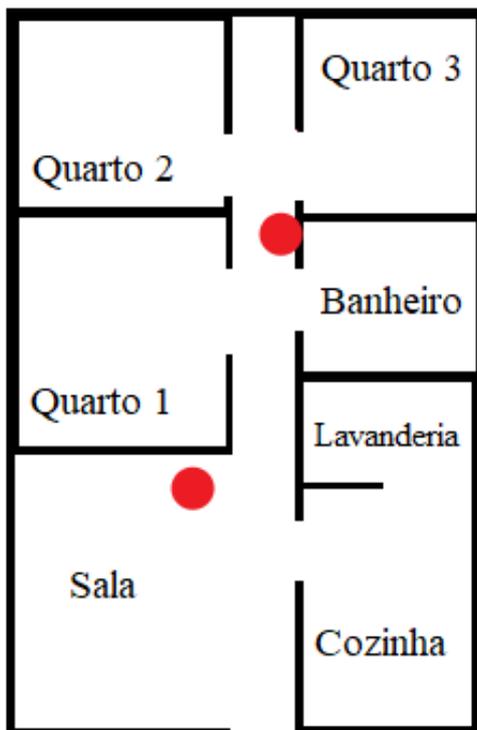


Figura 1. Disposição dos sensores (em vermelho) na casa para coleta dos dados de presença. Fonte: os autores

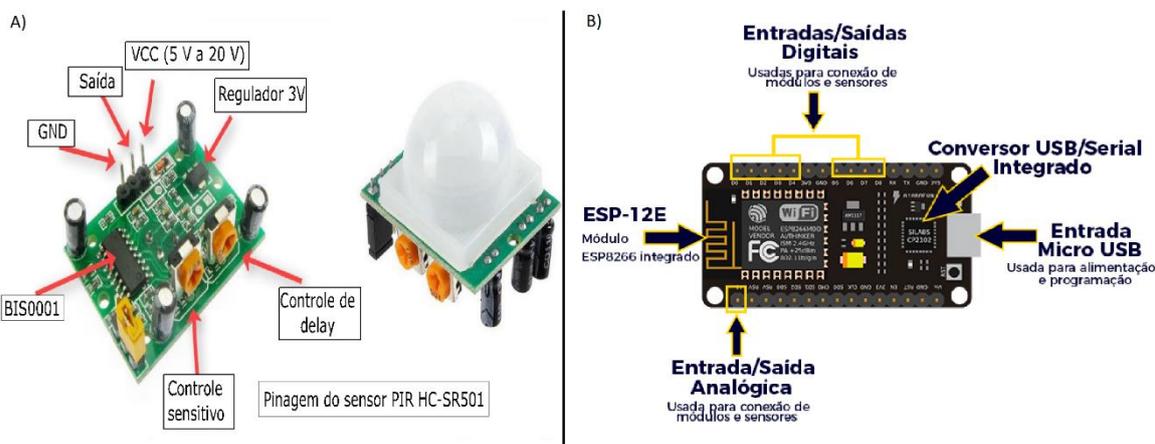


Figura 2. A) Detalhes do sensor PIR HC-SR501. Fonte: Adaptado (Datasheet, 2016); B) Placa NodeMCU – ESP8266. Fonte: Adaptado (Al Dahoud e Mohamed, 2018).

A placa NodeMCU pode ser programada em linguagem LUA ou C/C++, linguagem que o microcontrolador Arduino utiliza. Utilizou a IDE do Arduino para programar o algoritmo de configuração do sensor de presença. O algoritmo foi programado para realizar a aquisição dos dados de presença, a lógica para aquisição é ilustrada na Figura 3.

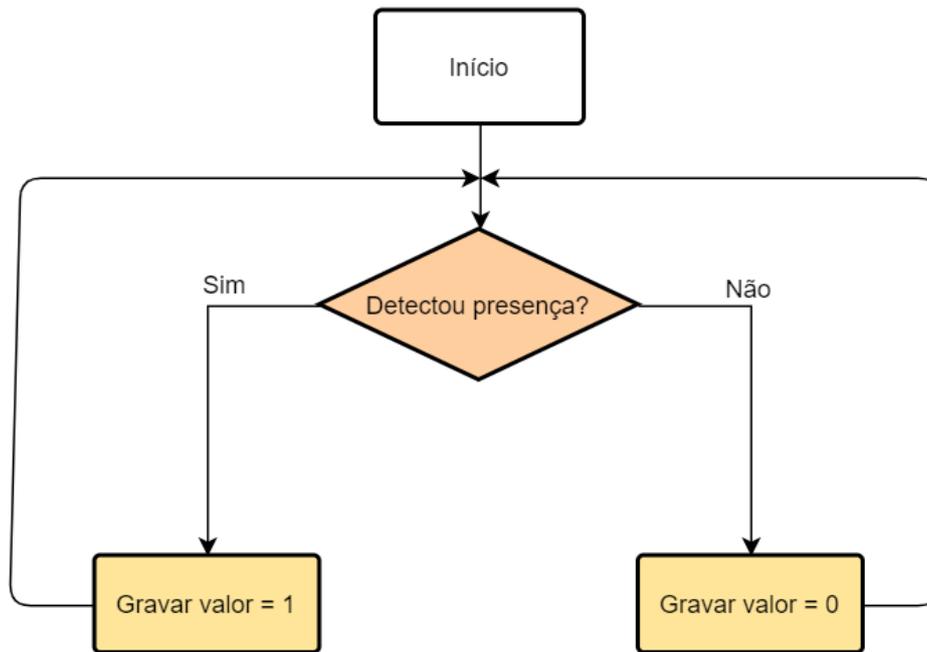


Figura 3. Fluxograma do algoritmo do sensor de presença. Fonte: os autores

Os dados coletados foram enviados a um servidor local responsável pelo armazenamento e atualização dos dados. O servidor foi desenvolvido usando o *micro-framework* Flask em um Raspberry Pi 3 B +. O Flask é destinado a pequenas aplicações com requisitos mais simples, possui um núcleo simples e expansível (Grinberg, 2018).

O módulo Raspberry Pi 3B + é um equivalente em miniatura de um computador pessoal completo em formato miniaturizado, suas dimensões são de apenas 86 x 54 mm. Ele contém todos os componentes necessários para usar o microcomputador como um computador pessoal comum (Saluch, 2018). Possui um processador com um *clock* de 1,4 GHz e com memória RAM de 1 Gb.

Com toda a infraestrutura para aquisição montada e configurada, iniciou-se a preparação da rede neural LSTM. Foi utilizado o *framework* TensorFlow juntamente com a biblioteca Keras. O TensorFlow é um sistema de aprendizado de máquina que opera em larga escala e em ambientes heterogêneos. Ele usa gráficos de fluxo de dados para representar a computação, o estado compartilhado e as operações que modificam esse estado (Abadi et al., 2016).

O Keras é uma biblioteca para rede neural artificial desenvolvida na linguagem Python. Ela funciona como *front-end* em TensorFlow, e foi desenvolvida para facilitar as experiências em redes neurais, possui uma rápida e fácil prototipagem, suporta redes neurais Convolucionais e Recorrentes (Gulli e Pal, 2017).

A rede neural artificial LSTM foi executada em um ambiente virtual, o *Google Colaboratory*, mais conhecido como *Google Colab* ou simplesmente *Colab*. O *Colab* é um projeto de pesquisa para criar protótipos de modelos de aprendizado de máquina em poderosas opções de *hardware*, como *Graphics Processing Units* (GPUs) e *Tensor Processing Units* (TPUs), fornece um ambiente de notebook *Jupyter* sem servidor para desenvolvimento interativo e é gratuito (Bisong, 2019).

Para a aplicação dos dados coletados na rede neural houve um processo de limpeza e transformação dos dados. Os dados “ano” e “mês” foram descartados por não fazerem parte da ciclotimia da série temporal em questão. Além disso, realizou-se a transformação da dimensão tempo em uma sequência temporal com a seguinte característica: [dia_semana, dia, hora, minuto].

A transformação da dimensão tempo em sequência facilitou a observação da dimensão tempo, haja vista que, cada sequência foi observada como um vocabulário para a rede neural LSTM.

A rede neural LSTM foi elaborada com a camada de entrada de dimensão das sequências definidas anteriormente; 100 neurônios artificiais; regularização normal de 0,2; regularização recorrente de 0,2; uma camada densa de saída com 1 neurônio com ativação *sigmoid*. Para compilação da rede foi utilizado: *loss* = *binary_crossentropy*; otimizador = Adam. E para métrica de avaliação da rede neural utilizou-se a acurácia.

RESULTADOS E DISCUSSÕES

Foram coletados em torno de 100 mil linhas de informações no período de 01 de fevereiro de 2019 a 11 de abril de 2019, com frequência de coleta definida a cada um minuto. Cada linha de dado apresentou as informações de: ano, mês, dia, hora, minuto, dia da semana e a presença. A presença é a variável que expressa o valor captado pelo sensor na janela de tempo definida pelas outras variáveis e como mostrado na Figura 3 essa variável possui valores binários (quando há presença captada pelo sensor = 1; quando não há presença captada pelo sensor = 0). Tabela 1 apresenta uma amostra dos dados coletados pelos sensores.

Tabela 1. Amostra dos dados coletados. Fonte: os autores

ano	mês	dia	hora	minuto	dia_semana	presença
2019	2	1	12	41	7	1
2019	2	1	12	42	7	1
2019	2	1	12	43	7	1
2019	2	1	12	44	7	1
2019	2	1	12	45	7	1

A rede neural artificial LSTM construída foi executada no *Colab* utilizando uma placa aceleradora para processamento paralelo, a K-80 da Nvídia. A acurácia foi a métrica de avaliação escolhida para

rede neural. Ela é a taxa de acerto global, ou seja, a proporção de predições corretas em relação ao tamanho do conjunto de dados (Silva et al., 2012).

Os testes iniciais a rede neural atingiu acurácia de 90%, porém foi utilizado reguladores para obtenção de maior generalidade e redução do overfitting. Assim os testes finais apresentaram uma acurácia de 85,27%.

O tamanho do arquivo da rede neural gerado pelo Keras foi de 825 KB possibilitando assim o desenvolvimento de um sistema inteligente em um dispositivo com baixa capacidade computacional.

CONCLUSÃO

Conclui-se que, com a acurácia de 85,27%, a rede neural artificial LSTM obteve um bom desempenho ao detectar padrões de presença de pessoas em uma residência. Assim, utilizando tal rede neural é possível o desenvolvimento de um sistema domótico inteligente baseado no comportamento dos residentes. Em trabalhos futuros, há a possibilidade de avaliar o comportamento da rede neural LSTM com camadas Convolucionais de uma dimensão, analisar outros hiper parâmetros de configuração e o desenvolvimento de um sistema domótico inteligente.

AGRADECIMENTOS

Os autores agradecem o suporte do Instituto Federal de Educação, Ciência e Tecnologia de Goiás (IFG), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERÊNCIAS BIBLIOGRÁFICAS

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Kudlur, M. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)* (pp. 265-283).
- Ahmad, M. A., Eckert, C., & Teredesai, A. (2018, August). Interpretable machine learning in healthcare. In *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics* (pp. 559-560).
- Al Dahoud, A e Mohamed F. (2018). NodeMCU V3 For Fast IoT Application Development.
- Bertoni, A., Dasari, S. K., Hallstedt, S. I., & Andersson, P. (2018). Model-based decision support for value and sustainability assessment: Applying machine learning in aerospace product development. In *DS 92: Proceedings of the DESIGN 2018 15th International Design Conference* (pp. 2585-2596).

- Bisong, E. (2019). Google Colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform* (pp. 59-64). Apress, Berkeley, CA.
- Bonica, A. (2018). Inferring Roll- Call Scores from Campaign Contributions Using Supervised Machine Learning. *American Journal of Political Science*, 62(4), 830-848.
- Culkin, R., & Das, S. R. (2017). Machine learning in finance: The case of deep learning for option pricing. *Journal of Investment Management*, 15(4), 92-100.
- Datasheet, P.I.R. (2016). Sensor (HC-SR501). *Diakses tanggal*, v. 30.
- de Melo, G., Oliveira, S. M., Ferreira, C. C., Vasconcelos Filho, E. P., Calixto, W. P., & Furriel, G. P. (2017, October). Evaluation techniques of machine learning in task of reprobation prediction of technical high school students. In *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)* (pp. 1-7). IEEE.
- Ge, Z., Song, Z., Ding, S. X., & Huang, B. (2017). Data mining and analytics in the process industry: The role of machine learning. *Ieee Access*, 5, 20590-20616.
- Grinberg, M. (2018). *Flask web development: developing web applications with python*. " O'Reilly Media, Inc."
- Gulli, A., & Pal, S. (2017). *Deep learning with Keras*. Packt Publishing Ltd.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Jean, N., Burke, M., Xie, M., Davis, W. M., Lobell, D. B., & Ermon, S. (2016). Combining satellite imagery and machine learning to predict poverty. *Science*, 353(6301), 790-794.
- Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260.
- Knoll, D., Prüglermeier, M., & Reinhart, G. (2016). Predicting future inbound logistics processes using machine learning. *Procedia CIRP*, 52, 145-150.
- Libbrecht, M. W., & Noble, W. S. (2015). Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, 16(6), 321-332.
- Lins, V., & Moura, W. (2010). Domótica: Automação Residencial. *Recife, dez*.
- Munakata, T. (2008). *Fundamentals of the new artificial intelligence: neural, evolutionary, fuzzy and more*. Springer Science & Business Media.
- Saluch, M., Tokarski, D., Grudniewski, T., Chodyka, M., Nitychoruk, J., Woliński, P., ... & Adamczewski, G. (2018). Raspberry PI 3B+ microcomputer as a central control unit in intelligent building automation management systems. In *MATEC Web of Conferences* (Vol. 196, p. 04032). EDP Sciences.
- Silva, R. M., Almeida, T. A., & Yamakami, A. (2012). Análise de desempenho de redes neurais artificiais para classificação automática de web spam. *Revista Brasileira de Computação Aplicada*, 4(2), 42-57.

Takiuchi, M., Melo, É., & Tonidandel, F. (2004). Domótica inteligente: automação baseada em comportamento. *Centro Universitário da FEI–UniFE, São Bernardo do Campo-SP.*

Zaccone, G., Karim, M. R., & Menshawy, A. (2017). *Deep Learning with TensorFlow*. Packt Publishing Ltd.

Uma Metodologia Simples para Descrição de Casos de Uso no Desenvolvimento de Sistemas de Informação

Recebido em: 14/08/2020

Aceito em: 04/09/2020

 10.46420/9786588319260cap3

José Gladistone da Rocha¹ 

Carlo Kleber da Silva Rodrigues^{2*} 

INTRODUÇÃO

Um sistema de informações (SI) que segue o paradigma de inteligência artificial (IA) deve ter a capacidade de interpretar corretamente dados externos, aprender a partir desses dados e, então, utilizar essas aprendizagens para atingir objetivos e realizar tarefas desejadas (Desouza et al., 2020). Por sua vez, a área de Ciência de Dados (CD) considera o estudo e a análise de grandes volumes de dados visando a um efetivo processo de tomada de decisão (Elshawi et al., 2018).

Ao se pensar em um projeto de desenvolvimento de um SI moderno, a união promissora dos dois campos de conhecimento supracitados (i.e, IA e CD) soa naturalmente como a mais indicada. Todavia, essa união somente será de fato promissora se antes for assegurada a correta execução da atividade conhecida como levantamento de requisitos, em que o engenheiro de *software* deve definir os *casos de uso* (do inglês, *Use Case* – UC) pertinentes ao SI em questão.

Os UCs se constituem em unidades básicas de implementação do SI, sendo expressos por meio de descrições narrativas das interações que ocorrem entre os elementos externos (atores) e o SI. Essas descrições devem ser claras, concisas e sem ambiguidades, permitindo o desenvolvimento de um SI de adequada qualidade (Sommerville, 2011; Schmitz, 2018; Russel, 2019).

Neste contexto, este trabalho tem o objetivo precípua de apresentar uma metodologia para descrição detalhada de UCs. A motivação para a criação desta metodologia é que muitos UCs são descritos com pouca riqueza de detalhes, prejudicando a qualidade desse tipo de artefato (Santos e Travassos, 2010). Enfatizam-se nesta metodologia a inclusão de cenários e a técnica de prototipação em maior escala. Para a construção da metodologia, é realizada uma compilação de trabalhos da literatura que trata de Engenharia de Requisitos (ER) e UCs. Para efeito de validação, tem-se a apresentação

¹ Centro de Desenvolvimento de Sistemas (CDS) do Exército Brasileiro, Brasília, DF, Brasil.

² Centro de Matemática, Computação e Cognição (CMCC), Universidade Federal do ABC (UFABC), Santo André, SP, Brasil.

* Autor de correspondência: carlo.kleber@ufabc.edu.br

parcial de um UC real, em que podem ser observadas as principais vantagens advindas da utilização da metodologia.

O restante deste capítulo está dividido da seguinte forma: a Seção 2 se refere a trabalhos da literatura relacionados ao tema; a Seção 3 apresenta a metodologia de descrição de UC aqui proposta; a Seção 4 apresenta um exemplo simplificado de descrição de UC, aplicando a metodologia aqui proposta; e por fim, a Seção 5 trata das conclusões e apontamentos para trabalhos futuros.

TRABALHOS RELACIONADOS

Esta seção discorre sobre algumas pesquisas da literatura recente que contribuem direta ou indiretamente com o objetivo deste trabalho. Em especial, tenciona-se prover o leitor com uma visão sobre as pesquisas mais importantes relacionadas ao problema sob investigação, bem como sobre o estado da arte.

Deboni e Gregolin (2008) desenvolveram um instrumento para apoiar a inspeção de qualidade das descrições dos UCs, que pudesse, ao mesmo tempo, ser utilizado como subsídio para a orientação de equipes de desenvolvimento na elaboração de descrições de UCs com qualidade. Este instrumento, fundamentado principalmente em *checklist*, facilita o encontro de defeitos nas descrições dos UCs, antes que elas passem para a próxima etapa do desenvolvimento do *software*. A integração desse *checklist* na metodologia a ser aqui proposta agrega valor e auxilia sobremaneira na qualidade dos UCs a serem descritos.

Santos e Travassos (2010) descreveram uma avaliação experimental de uma técnica de inspeção de UCs baseada em *checklist* e a sua inserção em um projeto de *software* real. A avaliação foi conduzida por meio de uma estratégia que combina dois tipos de estudos experimentais. O primeiro objetivando avaliar a técnica de *checklist*, e o segundo utilizando a metodologia da pesquisa-ação em um projeto real. Os resultados foram favoráveis à técnica *checklist* em relação à abordagem *ad-hoc*, quando considerado o número de defeitos identificados. Esse estudo vem ao encontro deste trabalho, pois a técnica de *checklist* pode ser empregada para se avaliar a descrição do UC.

Reis e Barbosa (2013) apresentaram uma revisão sistemática para identificar características e atributos de projeto e de sistemas que influenciavam o tempo gasto em atividades de desenvolvimento de requisitos. Baseado nos resultados da revisão, uma técnica inicial de estimativa de esforço para estas atividades foi proposta. Os resultados mostraram que um grande conjunto de características influenciadoras é encontrado e que resultados iniciais de aplicação da técnica são animadores.

Segundo Dennis, Wixom e Roth (2014), os UCs se originaram como uma parte do mundo do desenvolvimento orientado a objetos, mas são aceitos independente da metodologia de desenvolvimento utilizada. Assim, um UC representa como um sistema interage com o ambiente,

ilustrando as atividades realizadas pelos usuários do sistema e as respostas desse sistema. Ainda, segundo os autores, os UCs são empregados para explicar e documentar a interação exigida entre o usuário e o sistema, objetivando realizar a tarefa do usuário. O artefato UC serve dessa forma como um contrato entre o cliente e os desenvolvedores (Kruchten, 2003).

Giroto (2016) afirmou que modelos de processos de negócio podem ser utilizados como fontes de requisitos para a modelagem de SIs. Os requisitos obtidos podem ser representados através de UCs, seja em formato diagramático ou por descrição textual. O autor propôs uma abordagem para apoiar a obtenção de um modelo de UCs a partir de modelos *Business Process Model Notation* (BPMN). Com isso, foram obtidas a representação diagramática e a descrição textual dos UCs. Também, foi proposto integrar essa abordagem à ferramenta *Java Goal Into Object Oriented Standard Extension* (JGOOSE), a qual pode derivar UCs a partir de modelos BPMN.

Alves et al. (2017) apresentaram um relato de experiência de um profissional bibliotecário que assumiu o papel de engenheiro de requisitos, utilizando o método etnográfico (i.e., observação, entrevista e consulta à documentação) para elicitare os requisitos de um sistema real. O estudo de caso foi realizado com o apoio de uma empresa especializada na produção de insetos transgênicos em grande escala (e.g., *Aedes aegypti*) e vigilância de vetor. Em particular, a prototipação em papel foi adotada na fase de validação de requisitos. Os resultados apontaram a interdisciplinaridade como fator importante para o entendimento e modelagem de um sistema centrado no gerenciamento, controle e monitoramento de dados. Ademais, concluiu-se que a elicitação de requisitos é um campo de conhecimento em aberto para que pesquisadores da Ciência da Informação possam prospectar sobre o tema de curadoria de dados. A técnica de prototipação é uma abordagem que se inclui na metodologia aqui proposta.

METODOLOGIA PROPOSTA

Essa seção trata sobre a metodologia aqui proposta, sendo dividida em três subseções. A primeira, “Fundamentos”, refere-se aos conceitos e definições necessárias a serem incorporados na metodologia. A segunda, “Formato”, descreve os elementos constituintes da metodologia em si. E a terceira, “Verificação dos UCs”, apresenta um método para verificação da qualidade dos UCs descritos em acordo com a metodologia proposta.

Fundamentos

A ER é extremamente importante no desenvolvimento de SI, pois aplica o método de obtenção dos requisitos funcionais (RFs), que são a matéria prima dos UC (Cunha, 2017). Segundo Sommerville (2011), a captura de RFs se traduz na execução de quatro etapas: Elicitação (Levantamento); Análise,

Documentação e Validação. A realização desses passos evita a ocorrência de alguns erros na identificação de RFs, a saber, confusão semântica, redundância e inconsistência.

Durante a fase de análise de UCs, poderá ocorrer a identificação de novos RFs mais detalhados (Dennis et al., 2014; Deboni e Gregolin, 2008; Epifanio, 2018). Há ocasiões em que o UC pode se tornar muito grande e complexo. Neste caso, é aconselhável dividi-lo em dois ou mais UCs para que possam se tornar de mais fácil gerenciamento no projeto de desenvolvimento de SI.

Como mencionado, cada UC de um sistema se define pela descrição narrativa das interações que ocorrem entre os elementos externos (atores) e o sistema. A *Unified Modeling Language* (UML) não define uma estrutura textual a ser utilizada na descrição de um UC. Com isso, existem vários estilos de descrição propostos para definir um UC. Há, porém, três dimensões em que o estilo de descrição de um UC pode variar, a saber: o formato; o grau de detalhamento e o grau de abstração, conforme observa-se na Figura 1.

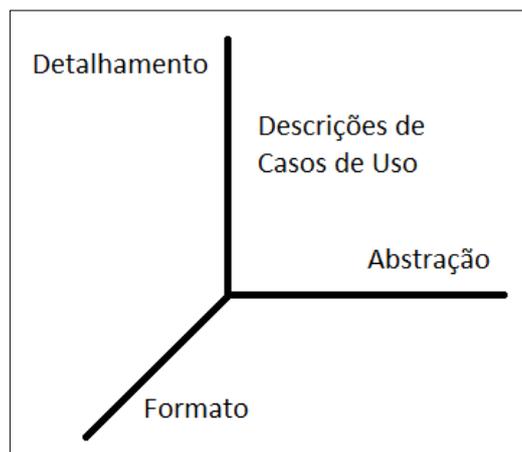


Figura 1. Dimensões de estilos de um UC. Fonte: Adaptado de Bezerra (2007).

O estilo a ser considerada na metodologia proposta de descrição de UC será voltado para um maior detalhamento e um menor grau de abstração, seguindo o formato numerado. No formato numerado, a narrativa é descrita por uma série de passos numerados, como exemplificado na Figura 2. Os outros dois formatos não são objetos do escopo desse trabalho.

- 1) Cliente inser seu cartão no caixa eletrônico
- 2) Sistema apresenta solicitação de senha
- 3) Cliente digita senha
- 4) Sistema valida a senha e exhibe menu de opções
- 5) Cliente indica que deseja realizar um saque
- 6)

Figura 2. Formato numerado na descrição de um UC. Fonte: Adaptado de Bezerra (2007).

O que se pretende é que o UC seja mais detalhado (Schmitz, 2018) e que os passos sejam descritos de maneira clara e sem redundância. O menor grau de abstração está associado à tecnologia que é explicitada na descrição do UC. Sugere-se ainda, para a metodologia proposta, a abordagem de um grau menor de abstração, onde pode-se mencionar alguma tecnologia na descrição do UC.

Quanto ao nível de detalhamento a ser descrito no UC, sugere-se apontar claramente, usando-se a voz ativa, que passos da descrição do UC irão ter um comportamento diferenciado, ou seja, quando, e.g., da inclusão de um UC do tipo *include* e/ou do tipo *extend*. Isso tornará o entendimento do UC mais claro e mais simplificado. O *include* refere-se a um ponto onde o fluxo básico ou alternativo de um UC obrigatoriamente realiza outro UC, desviando o seu fluxo, já no *extend*, esse desvio ocorre atendendo a uma decisão previamente estabelecida.

Sempre que necessário, devem-se incluir figuras ilustrativas dos diversos comportamentos (cenários) do UC ao longo de sua descrição. Isso tornará o UC de mais fácil entendimento e auxiliará sobremaneira a etapa seguinte, ou seja, a da implementação do UC pelos desenvolvedores. Um outro elemento da UML importante para entendimento de UC é a modelagem por meio de diagramas. A Figura 3 apresenta um exemplo de diagrama de UC.

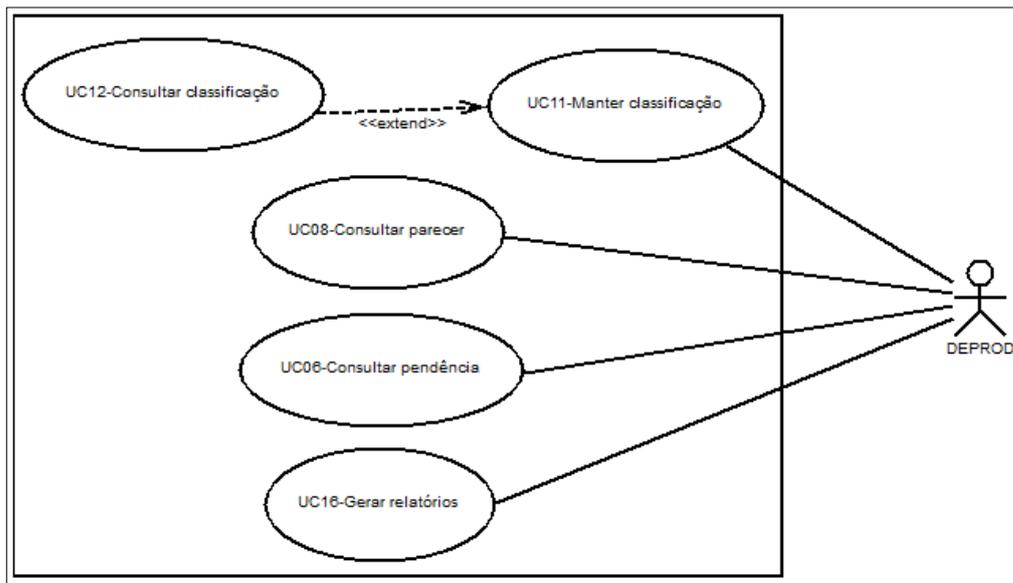


Figura 3. Diagrama de UC. Fonte: Rocha (2014).

Formato

A metodologia proposta constitui-se do seguinte formato: Nome do UC; Numeração; Classificação; Descrição; Atores Principais e Secundários; Pré-condições; Pós-condições; Fluxo Básico ou Principal; Fluxos Alternativos; Fluxos de Exceção; Pontos de Extensão; Requisitos Funcionais; Cenários; Regras de Negócio; Protótipo de tela e Dados Relacionados. Cada elemento da estrutura é detalhado a seguir.

a) Nome: nome do UC que o identifique unicamente em todo o sistema. Preferencialmente iniciar com um verbo no infinitivo. Deve-se evitar uma nomenclatura vaga e que não represente efetivamente o UC.

b) Numeração: é o identificador do UC em todo o sistema. Sugere-se usar o seguinte padrão: UC acrescido do número. Exemplo: UC001. Ele será útil para se ter uma rastreabilidade do UC em todo o sistema.

c) Classificação: identificação da ordem de prioridade do UC. Exemplo: baixa, média e alta.

d) Descrição: é uma descrição resumida da finalidade do UC e o objetivo do ator em utilizar o UC.

e) Ator principal: trata-se de todos os atores que incitam o UC para sua execução.

f) Ator secundário: é o ator que indiretamente interfere no UC ao longo de sua execução. Este, no entanto, não provoca sua execução.

g) Pré-condições: são todas as hipóteses necessárias para que se inicie a execução de um UC. Exemplo: o ator tem que estar autenticado no sistema.

h) Pós-condições: trata-se da situação que ocorrerá após a execução de um UC. Exemplo: usuário cadastrado no sistema.

i) Fluxo básico ou principal: refere-se ao fluxo de passos a ser seguido no caminho feliz ou de maior ocorrência ou mais comum no UC. Deve ser descrito em forma de diálogo entre o ator e o sistema. Preferencialmente, cada passo deve realizar uma ação atômica, isto é, uma única ação. Um exemplo desse fluxo é mostrado na Seção 4.

j) Fluxo alternativo: são todos os fluxos que não atendem as condições do fluxo básico e dele se derivam. Podem ser utilizados para descrever situações de escolha exclusivas entre si. Exemplos são mostrados na Seção 4.

k) Fluxo de exceção: são ações que podem vir a ocasionar erros ao longo do fluxo básico ou alternativo. Esses fluxos tratam de verificar possíveis erros. Descrevem o que acontece quando algo inesperado ocorre na interação entre ator e UC (por exemplo, quando o ator realiza uma operação inválida).

l) Pontos de extensão: indica em que passos do fluxo básico ou alternativo o UC realiza outro UC, seja do tipo *include* ou *extend*.

m) Requisitos funcionais: trata-se de uma lista de todos os RFs que estão incorporados no UC. Isso é útil para se poder fazer uma rastreabilidade no sistema.

n) Cenários: são outras formas de se executar um UC. Em específico, Kruchten (2003) afirma que os cenários são usados no processo *Rational Unified Process* (RUP) para extrair e enfatizar uma sucessão sem igual de ações ou detalhar um UC. A Seção 4 também apresenta exemplo de cenários ocorridos em um UC.

o) Regras de negócio: entende-se por regras de negócio as políticas, condições ou restrições que devem ser consideradas no desenvolvimento de SI (Bezerra, 2007). Deve-se criar uma lista contendo todas as regras de negócio que estão incluídas no UC. Também é útil para se ter uma rastreabilidade no sistema.

p) Protótipos de tela: chamada de prototipação em papel, é uma técnica em que se apresentam as versões em papel das telas do sistema com as quais os usuários interagem, e se projeta um conjunto de cenários que descrevem como o sistema pode ser usado (Nielsen, 1995). Protótipos são representações gráficas de como seria a interface do sistema com o ator de forma simples e objetiva. Tais figuras devem referenciar-se a algum passo do fluxo básico, alternativo ou de exceção. Essa técnica tem a vantagem de poder se criar de forma ágil as versões das interfaces do projeto. Exemplos são apresentados na Seção 4.

q) Dados relacionados: refere-se ao conjunto de dados das tabelas do banco de dados que são afetados pela realização do UC. Por exemplo, pode ser uma tabela contendo as seguintes colunas: Código; Campos; Valor associado (inclusão/alteração) e Tabelas do banco de dados.

Destaca-se da metodologia ora apresentada que dois quesitos são evidenciados: cenários e protótipos de tela. Em descrições de UC normalmente esses dois itens são pouco explorados. O que se quer aqui é exatamente alertar engenheiros de *software* e de requisitos que utilizem com maior intensidade esses dois quesitos apontados. Deve-se destacar também cada cenário que está incluído no fluxo básico, fluxos alternativos e/ou nos fluxos de exceção. Para efeito dessa metodologia, sugere-se ainda enumerar, dentro de cada UC, os cenários para facilitar sua identificação. Da mesma forma, deve-se explorar, sempre que possível, a inserção de protótipos de tela (Cunha, 2017) a cada cenário dos UCs.

Verificação dos UCs

Para imprimir qualidade na descrição de UCs, além de cumprir com o formato proposto na subseção anterior, propõe-se a inclusão do *checklist* (Tabela 1) de descrição de UC, apresentado no trabalho de Deboni e Gregolin (2008), como ferramenta de verificação da qualidade dos UC descritos. A classificação dos impactos para os defeitos encontrados segue a seguinte notação: A para alto; M para médio e B para baixo.

Tabela 1. *Checklist* de descrição de UC. Fonte: Adaptado de (Deboni e Gregolin, 2008).

Nº e Questão	Impacto
1. A descrição de UC é a de um UC representado no diagrama?	A
2. A descrição de UC contém nome do UC, nome do ator, fluxo básico e alternativo? (Se não houver um fluxo alternativo definido, considerar se ele está especificado dentro do fluxo básico.)	A
3. A descrição de UC com mais de uma página contém índice e paginação?	B
4. Se houver tabelas ou figuras, elas têm explicação adicional de forma que fiquem compreensíveis para o leitor?	B
5. Se houver referências, essas são numeradas ou nomeadas da mesma forma na origem (descrição de UC) e no destino (a própria referência)?	B
6. As frases representam um diálogo entre ator e sistema, evidenciando a ação do ator e a resposta do sistema?	M
7. As frases se utilizam de subtítulos para comunicar as ideias chaves dos fluxos de forma mais clara?	B
8. As frases são construídas em voz ativa? (Ex.: “Sistema valida a quantia informada” em vez de “A quantia informada deve ser validada pelo sistema”).	B
9. As frases utilizam o tempo presente?	B
10. São evitados termos sem quantificação precisa, como “muito”, “pouco”, “adequado”, “claro”, “fácil”, “longo”, “curto”, “rápido” “etc”?	M
11. São evitados termos que indicam opção, como “possivelmente”, “alternativamente”, “no caso”, “se”, etc., sem especificar um fluxo alternativo?	M
12. Os termos passíveis de mais de uma interpretação constam em glossário, com clara definição?	A
13. Uma vez utilizado um termo, ele é mantido para referenciar-se ao mesmo elemento?	M
14. São evitados termos que indicam a prematura especificação de interface, tais como “clicar” “botão” etc.?	B
15. As funcionalidades se restringem ao que o sistema deve fazer e não em como, evitando a definição explícita de código na especificação?	M

16. A descrição evita requisitos de negócio sem ação direta ao sistema?	M
17. Há presença de breve descrição ou resumo no início da descrição de UC, que especifique de forma clara o seu propósito?	B
18. O fluxo básico está aparentemente completo?	A
19. O fluxo alternativo está aparentemente completo?	A
20. As frases são numeradas para que possibilitem a rastreabilidade?	M
21. As frases procuram ser objetivas, evitando redundâncias ou presença de informações evidentemente desnecessárias?	M
22. O UC é acompanhado de protótipo de interface a fim de aumentar a sua compreensibilidade?	A
23. O UC é acompanhado de especificação de requisitos não funcionais separadas do fluxo de eventos do UC ou em documento de especificação suplementar?	M
24. O UC é acompanhado de modelo de domínio, mostrando os relacionamentos entre os principais conceitos do sistema, a fim de aumentar a sua compreensibilidade?	M
25. Se houver regras de negócios associadas, estas estão separadas dos fluxos de evento do UC ou em documento de especificação de regras de negócios?	B

Os Quesitos 22, 23 e 25 vão ao encontro do que se propõe a metodologia de descrição de UC proposta, onde o Quesito 22 refere-se à prototipação, o Quesito 23 trata dos fluxos de eventos a serem acionados pelos UCs, ou seja, os diversos cenários que ocorrem em UCs, e o Quesito 25 refere-se a regras de negócio associados aos fluxos de eventos do UC. De posse do UC já descrito deve-se então submetê-lo ao *checklist* apresentado na Tabela 1 para verificar quais erros ou deficiências existem no UC submetido. Assim, fica mais fácil melhorar a qualidade da descrição do UC com as correções. Essa qualidade na descrição do UC irá se refletir na fase seguinte de codificação e, por conseguinte, no SI final.

EXEMPLIFICAÇÃO DE UM CASO DE USO

Com base na metodologia proposta na seção anterior, é apresentado a seguir um exemplo de descrição de UC com base em um estudo de caso real, a saber, Sistema de Cadastramento de Produtos de Defesa do Ministério da Defesa, apresentado no trabalho de Rocha (2014).

Será tomado por base o *caso de uso* UC01 Manter Empresa, conforme pode-se verificar no Diagrama de UC constante na Figura 4.

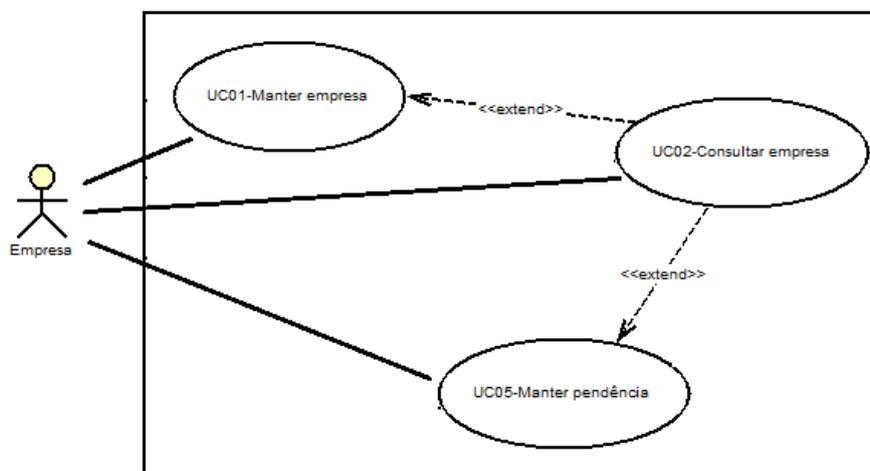


Figura 4. Diagrama de UC do Pacote Empresa. Fonte: Adaptado de Rocha (2014).

A figura original do diagrama é aqui apresentada de forma simplificada, contendo apenas as informações necessárias para o entendimento da metodologia. Observa-se no diagrama que o UC01 Manter Empresa é acionado pelo ator Empresa e possui um UC extensor que é o UC02 Consultar Empresa. Essas informações iniciais serão úteis na hora de se descrever o UC01 Manter Empresa.

A seguir, apresenta-se a descrição, resumida, do UC01 Manter Empresa, sendo que, por maior simplicidade e facilidade de entendimento, não serão apresentados todos os passos do Fluxo Básico e dos Fluxos Alternativos.

Caso de Uso UC01 – Manter Empresa

a) Descrição

Este UC possibilita a realização das operações de alteração, validação de dados cadastrais, validação de pré-cadastro e liberação de usuário da empresa para se autenticar no sistema.

b) Atores

Principal: Empresa

Secundário: Representante da Força e DEPROD.

c) Pré-condições

Ator ter acessado o sistema e estar autenticado.

d) Pós-condições

- 1) Dados da empresa alterados e validados com sucesso.
- 2) Liberação, de usuário da empresa, para autenticação no sistema realizada com sucesso.

e) Fluxo básico

1) sistema realiza os seguintes cenários, dependendo da categoria de usuário do ator autenticado:

- a) Cenário 1 – O ator é a DEPROD: sistema apresenta tela (Figura 5) com as opções “cancelar”, “consultar cadastro” (FA1), “validar cadastro” (FA2) e “validar pré-cadastro” (FA3), “controlar credenciamento” (FA4), “pendências” (FA5) e “pareceres” (FA6);



Figura 5. Tela inicial de manter empresa para cenário 1. Fonte: Rocha (2014)

- b) Cenário 2 – O ator é o Representante da Força: sistema apresenta tela (Figura 6) com as opções “cancelar”, “consultar cadastro” (FA1), “pendências” (FA5) e “pareceres” (FA6);



Figura 6. Tela inicial de manter empresa para cenário 2. Fonte: Rocha (2014).

- c) Cenário 3 – O ator é a Empresa: sistema apresenta tela (Figura 7) com as opções “cancelar”, “consultar cadastro” (FA1) e “alterar cadastro” (FA4).



Figura 7. Tela inicial de manter empresa para cenário 3. Fonte: Rocha(2014).

- 2) ator seleciona a opção “cancelar”.
- 3) sistema retorna para a tela de menu inicial do sistema.
- 4) Fim do fluxo básico e do UC.

f) Fluxos alternativos

1) FA1 – Consultar cadastro

1.1) sistema realiza o caso de uso UC02-Consultar empresa e, dependendo do resultado, realiza os seguintes cenários:

- a) Cenário 4 – se nenhuma empresa tiver sido selecionada, fluxo é direcionado ao passo 1 do fluxo básico.
- b) Caso uma empresa tenha sido selecionada, sistema apresenta tela com dados (D1) da empresa selecionada, a opção “sair”, e ainda as opções, dos seguintes cenários, dependendo da categoria de usuário autenticado:

1) O ator é a DEPROD:

- a) Cenário 5 – empresa selecionada possui status “NÃO DEFINIDO” (Figura 8): “validar pré-cadastro” (FA3);



Figura 8. Opções de consultar cadastro de manter empresa para cenário 5. Fonte: Rocha (2014).

Os Cenários de 6 a 12 foram omitidos.

2) FA2 – Validar cadastro

Omitido.

3) FA3 – Validar pré-cadastro de empresa

Omitido.

4) FA4 – Alterar cadastro da empresa

4.1) Sistema realiza um dos seguintes cenários, dependendo do status da empresa:

a) Cenário 16 – Caso o status da empresa seja “EM CADASTRAMENTO”, o sistema apresenta tela (Figura 13) contendo:

- 1) campos editáveis referentes aos dados (D2) da empresa;
- 2) as opções “salvar” e “sair”.

Tela do UC01 - Manter empresa
FA4 - Alterar cadastro da empresa
Cenário 16
Categoria de usuário: EMPRESA
Cadastro da empresa

CodEmp:
Razão social:
Nome fantasia:
CNPJ:
CEP:
Endereço:
Telefone:
Site:
Mapa:
FAX:
Logomarca:
Status:
UF:
Município:
Credenciamento:
Quem somos:

Figura 9. Alterar cadastro da empresa para cenário 16. Fonte: Rocha(2014).

b) Cenário 17 – caso o status da empresa seja “EM PROCESSAMENTO” ou “PROCESSADO”, o sistema apresenta mensagem “Os dados da empresa não podem ser alterados quando seu status for EM PROCESSAMENTO ou PROCESSADO. Qualquer necessidade de mudança de algum dado deverá ser solicitada diretamente ao DEPROD / MD, via e-mail ou telefone”.

4.2) ator altera dados e seleciona opção “salvar”. Caso opte por “sair” o fluxo para o passo do fluxo básico ou alternativo chamador do FA4.

4.3) sistema valida dados cadastrais (FE1) e, caso estejam em conformidade, grava dados no sistema e envia mensagem “Alteração de cadastro de empresa realizada com sucesso!”.

4.4) sistema encerra o fluxo alternativo e retorna ao passo 1 do fluxo básico.

5) FA5 – Pendências

5.1) Sistema realiza o caso de uso estendido UC05-Manter pendência e retorna ao passo do fluxo alternativo que chamou o FA5.

6) FA6 – Credenciamentos

6.1) Sistema realiza o caso de uso estendido UC09-Manter credenciamento e retorna ao passo do fluxo alternativo que chamou o FA6.

g) Fluxos de Exceção

1) FE1 – Validar dados de cadastro

1.1) sistema verificar se todos os dados obrigatórios estão preenchidos. Caso algum campo não esteja preenchido, envia mensagem informando quais são estes campos e permanece no passo do fluxo alternativo FA que acionou o FE1.

1.2) fim do fluxo de exceção.

h) Ponto de Extensão

1) FA5 – Pendências: sistema realiza o caso de uso UC05-Manter pendência.

2) FA6 – Credenciamentos: sistema realiza o caso de uso UC09-Manter credenciamento.

i) Dados relacionados

O quadro relacionando os dados é muito grande e não cabe na página. Entretanto, tal quadro poderá ser visualizado em Rocha (2014, p. 81).

j) Requisitos Funcionais e Regras de Negócio

1) Requisitos funcionais – conforme a Figura 10.

RF_E01	Registrar pré-cadastro
RF_E02	Validar dados iniciais da empresa
RF_E03	Validar cadastro de empresa
RF_E05	Alterar cadastro da empresa pela empresa
RF_E07	Incluir e retirar pendência de empresa
RF_E08	Gerar relatório de empresas candidatas a EED
RF_A02	Registrar pré-cadastro
RF_A03	Liberar acesso ao sistema

Figura 10. Requisitos funcionais envolvidos no UC01–Manter empresa. Fonte: Adaptado de Rocha (2014).

2) Regras de negócio – conforme Figura 11.

RN08	Restrição para quantidade de usuários
RN10	Exigência de um produto para credenciamento
RN11	Credenciamento de vários CNPJ
RN12	Responsabilidade para liberar pendências
RN16	Restrição de catalogação para credenciamento de empresa e classificação de produto
RN18	Acompanhamento de processo
RN39	Lista de produtos e empresas para CMID

Figura 11. Regras de negócio envolvidas no UC01 – Manter empresa. Fonte: Adaptado de Rocha (2014).

k) Dados Relacionados

Omitido.

CONCLUSÃO E TRABALHOS FUTUROS

Esse capítulo apresentou uma metodologia simples para descrição de UCs visando ao desenvolvimento de SIs, onde foi enfatizada a utilização de cenários e prototipação. Como principal contribuição, espera-se que os UCs descritos sejam de maior facilidade de entendimento e mais completos, resultando em maior qualidade dos SIs desenvolvidos.

Como trabalhos futuros, pretende-se aplicar a metodologia aqui apresentada em UCs de SIs corporativos de distintas áreas de negócio. O objetivo é obter um maior escopo de validação, bem como de possível aperfeiçoamento e/ou customização da própria metodologia.

REFERÊNCIAS BIBLIOGRÁFICAS

Alves RM, Ramos RA, Ramos RP, Burnham TF (2017). Elicitação de Requisitos para o Desenvolvimento de um Sistema de Informação Utilizando a Etnografia: um relato de experiência. *Revista Brasileira de Biblioteconomia e Documentação*, 13:1148-1164.

Bezerra E (2007). *Princípios de Análise e Projeto de Sistemas com UML*. Rio de Janeiro: Elsevier.

Cunha JAC (2017). Engenharia de Requisitos: Análise Sobre o Processo de Engenharia de Requisitos no Desenvolvimento de Sistemas em Teresina – Pi. TCC do Curso de graduação em Análise e Desenvolvimento de Sistemas.

Deboni JEZ, Gregolin R (2008). Inspeção de Qualidade em Descrições de Casos de Uso: Uma Proposta de Modelo e Artefatos. *VII Simpósio Brasileiro de Qualidade de Software*.

Dennis A, Wixom BH, Roth RM (2014). *Análise e Projeto de Sistema*. 5 ed. Rio de Janeiro: LTC.

Desouza KC, Dawson GS, Chenok D (2020). Designing, developing, and deploying artificial intelligence systems: Lessons from and for the public sector. *Business Horizons*, 13(2): 205-213.

Eelshawi R, Sakr S, Talia D, Trunfio P (2018). Big Data Systems Meet Machine Learning Challenges: Towards Big Data Science as a Service. *Big Data Research*, 14:1-11.

Epifanio JC. (2018). Investigação do Ensino de Engenharia de Requisitos na Perspectiva da Academia e da Indústria: Um enfoque em Documentação de Requisitos. Dissertação de mestrado em Sistemas e Computação.

Giroto AN. (2016). Uma proposta para derivar Casos de Uso a partir de modelos BPMN com suporte computacional. Trabalho de Conclusão de Curso para bacharel em Ciência da Computação.

Kruchten P (2003). *Introdução ao RUP – Rational Unified Process*. Rio de Janeiro: Editora Ciência Moderna Ltda.

Nielsen J (1995). Using paper prototypes in home-page design. *IEEE software*, 12(4): 88-89.

Reis JCS, Barbosa MW (2013). Proposta de uma Técnica de Estimativa para Requisitos. *Revista de Sistemas e Computação, Salvador*, 3(1): 3-22.

Rocha JG (2014). Empresas e Produtos Estratégicos de Defesa do Brasil: mapeamento do processo e sua automação por meio de Sistema de Informação (SISCAPE). Trabalho de Conclusão de Curso de Graduação em Sistema de Informação, Centro Universitário Sul de Minas. Varginha-MG.

Russell M. (2019). Supporting Decision Makers with Use Cases; Case Study Results. 17th *Annual Conference on Systems Engineering Research (CSER)*.

Santos PSM, Travassos GH. (2010). Inspeção de Qualidade em Descrições de Casos de Uso: uma Avaliação Experimental em um Projeto Real.

Schmitz RA. (2018). Sistema de Gestão de Pessoas: especificação de requisitos funcionais para os processos seletivos simplificados da UFSC. Dissertação de mestrado em Administração.

Sommerville I (2011). *Engenharia de Software*. 9 ed. Rio de Janeiro: Pearson.

Aspectos e causas da obesidade: uma análise utilizando algoritmo de agrupamento *Farthest First*

Recebido em: 05/08/2020

Aceito em: 17/08/2020

 10.46420/9786588319260cap4

Gilberto de Melo Junior^{1*} 

Sanderson Oliveira de Macedo¹ 

Renato Milhomem de Oliveira Filho² 

Jane Sousa Naves³ 

Géssica Helen de Melo⁴ 

INTRODUÇÃO

Muitos fatores comportamentais acarretam a obesidade em um indivíduo, isto é, há inúmeros fatores variáveis envolvidos. Portanto, a tarefa de classificar quais grupos de comportamentos são mais prevalentes no fator de desenvolvimento da obesidade é muito complexa (Ferreira e Magalhães, 2006). A alimentação descomedida não implica certa e diretamente em obesidade, pois existe um conjunto de fatores demográficos, socioeconômicos e biológicos que são determinantes para tal condição.

A obesidade representa um estado de alto e extenso aumento de gordura corporal total, sendo considerada como um problema mundial de saúde. Ela é caracterizada como uma condição complexa de ordem multifatorial, que leva ao surgimento ou acréscimo de doenças, estando a obesidade abdominal intimamente relacionada com o aumento significativo de doenças cardiovasculares e de ordem metabólica, como a hipertensão, dislipidemia e hiperglicemia (De Oliveira, 2017).

A obesidade é considerada uma doença do grupo de doenças crônicas não transmissíveis, com associações positivas para outros problemas de saúde como os cardiovasculares e cerebrovasculares, distúrbios metabólicos, diferentes tipos de câncer, doenças do aparelho digestivo, entre outras (Rocha-Brischiliari, 2014; Ducan, 2005) que são associadas principalmente com um conjunto de fatores demográficos, socioeconômicos e biológicos (Coelho, 2009).

Monteiro et al. (2003) explicam que as condições socioeconômicas, influenciam os hábitos de vida da população e, conseqüentemente o maior risco de obesidade. Fatores como: educação, renda e

¹ Docente do Departamento IV no Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

² Discente do curso de Engenharia de Controle e Automação no Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

³ Discente do curso de Biomedicina na Universidade Federal de Goiás.

⁴ Discente do curso de Medicina na Faculdade Morgana Potrich, Campus Mineiros.

* Autor correspondente: gilberto.melo@outlook.com

ocupação influenciam nos hábitos comportamentais que, por sua vez, afetam diretamente a ingestão maior de calorias e o gasto energético.

O aumento da prevalência de obesidade entre os jovens foi um dos principais temas de investigação da Saúde Pública na última década. Fundamentado nas observações de maior probabilidade do aparecimento de manifestações clínicas entre os jovens com excesso de peso corporal, a obesidade é considerada um grave fator de risco para morbidade na vida adulta e caracterizada, atualmente, como um dos grandes problemas de Saúde Pública (Cavazzotto et. al., 2014).

Assim, para determinar quais grupos estão mais propensos a serem obesos optou-se por utilizar a teoria de Aprendizagem de Máquina (*Machine Learning*) empregando a abordagem de aprendizagem não-supervisionada e a técnica de Agrupamentos. O objetivo desta técnica é encontrar uma estrutura de *clusters* (grupos) nos dados, onde os objetos pertencentes a cada cluster compartilham alguma característica ou propriedade relevante para o domínio do problema em estudo, ou seja, são de alguma maneira similares (Faceli et. al, 2011).

Dentre os trabalhos relacionados ao uso de *Machine Learning* aplicados a Nutrição, podemos citar: Freddo et al. (2019), onde os autores utilizaram o algoritmo de classificação de Árvore de Decisão para analisar 1339 prontuários de pacientes da Clínica Escola de Nutrição da UFFS. A acurácia apresentada no experimento foi superior de 89% na detecção de doenças relacionadas a alimentação. Já em Ramos et al. (2012), os autores utilizam algoritmos de Árvore de Decisão de *Machine Learning* para avaliação nutricional em crianças. Jacomini et al. (2018) utilizam algoritmos de Árvore de Decisão para obter padrões referentes a obesidade e seus fatores de risco cardiovascular e de comorbidade na cidade de Cascavel, Paraná, Brasil.

Nesse trabalho utilizou-se a técnica de aprendizagem de máquina não-supervisionada de clusterização (agrupamentos) *Farthest First*, para agrupar pessoas com informações demográficas, socioeconômicas, de biotipo e de conjunto de ações observáveis recorrentes, com o objetivo de criar uma base de referência para a análise e identificação de novos usuários com comportamentos tendenciais a obesidade. Essa abordagem resultou em uma nova ferramenta que poderá auxiliar profissionais da saúde e cientistas a identificarem pessoas que tem tendências à obesidade.

MATERIAL E MÉTODOS

A coleta de dados sobre os fatores que influenciam na determinação da obesidade, foi realizada utilizando um questionário online disponibilizado em um curto período de tempo (aproximadamente 1 mês) aos alunos do Instituto Federal de Educação, Ciência e Tecnologia de Goiás (IFG) do campus Goiânia e seus familiares, a fim de obter informações sobre suas rotinas diárias. Os participantes

concordaram com os termos da pesquisa e ficaram cientes que as respostas não seriam divulgadas, mas apenas utilizadas para fins científicos.

O questionário apresentou 29 questões, das quais todas foram elaboradas e discutidas com profissionais da área de saúde. Por meio dos dados coletados no questionário foi possível também calcular o Índice de Massa Corporal (IMC) de cada indivíduo, sendo o mesmo utilizado também como dado na clusterização.

O IMC é uma medida internacional usada para identificar se uma pessoa está no peso ideal ou não. Ele é determinado pela razão entre a massa do indivíduo e o quadrado da sua altura, onde a massa está em quilogramas e a altura é dada em metros (Anjos, 1992).

O questionário utilizado na coleta de dados apresentou as seguintes questões:

- Qual a sua idade?
- Qual a sua altura?
- Qual o seu peso em kg?
- Qual seu estado civil? (a-Solteiro(a) b-Casado(a) c-Divorciado(a) d-Viúvo(a) e- Namorando);
- Qual o seu meio de transporte? (a-Particular b-Público c-Táxi d- A Pé e-Bicicleta);
- Possui filhos? Quantos?
- Qual o seu nível de escolaridade? (a-Ensino Fundamental incompleto b- Ensino Fundamental Completo c-Ensino Médio Incompleto d-Ensino Médio Completo e- Ensino Superior Incompleto f-Ensino Superior Completo g-Especialização);
- Trabalha? Com o que? (a-Escritório/Professor b-Trabalho Pesado c-Domestica/Auxiliar de Serviços Gerais d-Vendedor/Andarilho e-Não Trabalha);
- Qual o seu nível salarial (ou outra remuneração)?
- Quantas refeições você faz por dia?
- Qual a quantidade (em média e em gramas) por refeição?
- Onde você faz suas principais refeições? (a- Casa b-Restaurante c- Leva a Própria Comida d-Casa e Restaurante);
- Consome bebida alcoólica? (a-Quase Nunca b-Socialmente c-Socialmente Exagerado d-Exagerado e-Não Consome);
- Têm o hábito de fumar? (a-Quase Nunca b-Socialmente c-Socialmente Exagerado d-Exagerado e-Não Fuma);

- Pratica alguma atividade física? Quantas vezes por semana? (a-1 b-2 c-3 d-4 e-5 f-6 g-7 h- Não pratico);
- Qual horário do dia? (a- Pela manhã b- Pela tarde c- Pela noite d- Não Pratico);
- Qual exercício físico? (a- Aeróbicos b- Anaeróbicos c- Ambos d- Não Pratico);
- Faz exames de rotina? (a- Sim b-Não);
- Em sua família, existem casos de: (a- Obesidade b- Hipertensão c- Diabetes d-Colesterol Elevado e-Triglicérides Elevado f- Mais de uma opção g- Nenhuma das opções);
- Em média, quantas horas dorme por dia?
- Você se considera: (a-Relaxado(a) b-Paciente c-Extrovertido(a) d-Calmo(a) e- Estressado(a) f- Bipolar g- Ansioso(a) h- Depressiva);
- Toma remédio controlado? (a- Sim b-Não);
- Já fez, ou faz, alguma dieta? (a- Sim b- Às Vezes c- Não);
- Já tomou ou toma *shakes* pra emagrecer ou engordar? (a- Sim b-Não);
- Já tomou ou toma remédio para emagrecer? (a- Manipulado b- Natural (chás, alimentação controlada) c- Farmácia d- Nunca Tomou);
- Quantos copos de água você consome por dia?
- Já fez alguma das seguintes cirurgias? (a-Bariátrica b-Tireoide c-Lipoaspiração d-Bariátrica e Lipoaspiração e-Bariátrica e Tireoide f-Tireoide e Lipoaspiração g-Nenhuma);
- Você possui alguma dessas doenças? (a- Hipertensão b- Colesterol Elevado c- Diabetes d- Triglicérides Elevado e- Mais de uma opção f-Nenhuma das opções);
- Possui vida sexual ativa? (a- Sim b-Não);
- Com qual frequência utiliza dispositivos eletrônicos? (celulares, computadores, etc.);
- Como você se vê de acordo com os corpos apresentado na Figura 1?

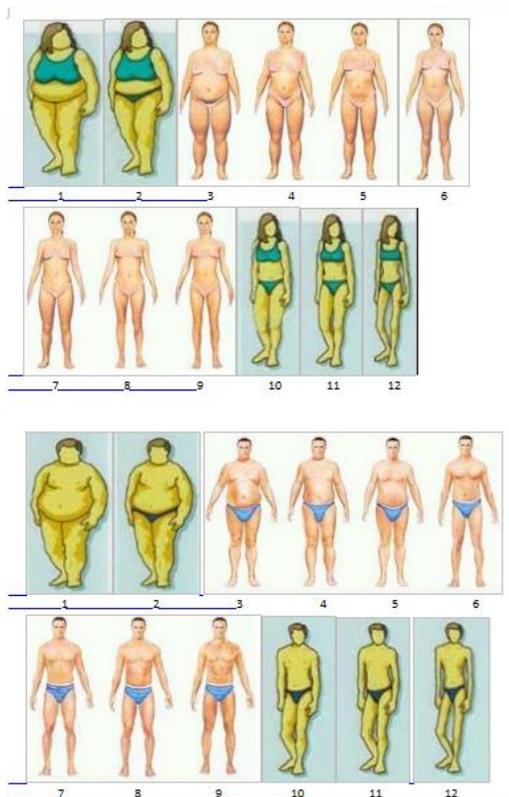


Figura 1. Estereótipos de corpos para o entrevistado escolher com qual se assemelha mais. Adaptado de Cataneo (2005).

ALGORITMO DE AGRUPAMENTO *FARTHEST FIRST*

O algoritmo de agrupamentos utilizado neste trabalho foi o *Farthest First*, que se configura como um algoritmo rápido e guloso. Ele foi proposto por Hochbaum e Shmoysat em 1985 (Vadegar e Yogish, 2014).

Neste algoritmo, k pontos são selecionados como os centros dos clusters. O primeiro centro é selecionado aleatoriamente. O segundo centro é avidamente selecionado como o ponto mais distante do primeiro. Cada centro restante é determinado selecionando o ponto mais distante do conjunto de objetos já escolhidos. Os pontos restantes são adicionados ao cluster cujo centro é o mais próximo (Kumar, 2013).

Ele possui complexidade de tempo $O(nk)$, onde n é o número de objetos no conjunto de dados e k é o número de clusters desejados (Dharmarajan e Velmurugan, 2015).

RESULTADOS E DISCUSSÃO

Foram respondidos 300 questionários, porém, uma averiguação destes mostrou que haviam respostas dados incompletos ou com dados errôneos (sem relação entre as perguntas). Assim, após o descarte desses dados restaram 250 respostas que foram utilizadas para comporem os dados de análises.

Com esses questionários calculou-se o índice de IMC de cada indivíduo, como informação que foi apresentada ao algoritmo para clusterização. Com um trabalho de adequação aos dados, os mesmos foram transformados em arquivo “.arff”. Este tipo de arquivo é utilizado para ser executado no software de código livre de mineração de dados Weka (Hall, 2009), utilizado neste trabalho.

Foi definido a criação de 5 clusters, com a perspectiva de alcançar os 5 grupos definidos pelo IMC (Magro, Normal, Médio, Gordo e Obeso). Embora o cálculo do IMC apresente algumas desvantagens e ele não tenha sido usado para clusterização, o mesmo possibilitou um melhor entendimento visual dos resultados aplicados. Utilizou-se o algoritmo no modo “Classe para avaliação do Cluster”. Neste modo o software inicialmente ignora o atributo de classe e gera o agrupamento. Em seguida, durante a fase de teste, ele atribui classes aos clusters, com base no valor majoritário do atributo de classe em cada cluster. A Tabela 1 apresenta os clusters que o algoritmo *Farthest First* analisou e criou utilizando os dados obtidos com a aplicação do questionário.

Tabela 1. Clusters obtidos após a clusterização utilizando o algoritmo *Farthest First*, juntamente com a porcentagem de cada cluster em relação ao total dos dados e também as respostas de cada pergunta que auxiliou na formação do cluster. Fonte: os autores.

Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
22	47	22	18	17
1.53	1.87	1.63	1.89	1.57
49	125	76	65	64
Casado(a)	Casado(a)	Solteiro(a)	Solteiro(a)	Solteiro(a)
Particular	A pé	Particular	Público	A pé
0	2	0	0	0
Superior	Especialização	Superior	Ensino Médio	Ensino Técnico
Não trabalha	Professor /	Não trabalha	Não trabalha	Professor /
400	20000	0	0	450
3	3	6	5	3
350	500	150	500	800
Leva a própria	Casa	Casa	Casa e	Casa
Não consome	Socialmente	Socialmente	Socialmente	Não consome
Não fuma	Não fuma	Quase nunca	Não fuma	Não fuma
3	Não pratica	5	2	Não pratica
A noite	Não pratica	Pela manhã	Pela tarde	Não pratica
Aeróbicos	Não pratica	Ambos	Ambos	Não pratica
Sim	Não	Sim	Não	Não
Mais de uma	Obesidade	Mais de uma	Diabetes	Hipertensão
7	8	7	9	6
Ansioso(a)	Bipolar	Extrovertido(a)	Calmo(a)	Calmo(a)
Não	Sim	Não	Não	Não
Não	Sim	Sim	Não	Não
Não	Não	Sim	Não	Sim

Nunca tomou	Nunca tomou	Todos	Nunca tomou	Natural
3	4	8	10	5
Nenhuma	Nenhuma	Bariátrica	Nenhuma	Nenhuma
Colesterol	Triglicérides	Nenhuma das	Nenhuma das	Colesterol elevado
Sim	Sim	Não	Não	Não
3	8	16	15	4
6	4	5	11	4
20.13	35.74	28.6	18.19	25.96
Normal	Obeso	Gordo	Magro	Médio

O Cluster 0 definiu o grupo denominado normal, para 44% dos participantes. As características desse grupo foram mulheres, com idade de 20 anos, solteiras, com nível superior incompleto, que mantém hábitos diários considerados saudáveis como: alimentação fracionada e equilibrada, acompanhamento médico regular, além da prática diária de atividades físicas. Apesar dessas participantes terem respondido como afirmativa a presença de doenças crônicas em suas famílias, como hipertensão, diabetes e obesidade, elas não confirmaram a presença de tais diagnósticos para si mesmas. Porém, é importante observar que o IMC dessas participantes foi de 24,6, o que indica que a obesidade está relacionada, de uma forma geral, a estilos de vida e hábitos alimentares.

O Cluster 1 identificou 12% dos participantes propensos ao desenvolvimento da obesidade, denominados pela Organização Mundial de Saúde como obesidade grau 1 ($IMC \geq 30$). Neste grupo, prevaleceu o sexo masculino, com idade de 21 anos. Os participantes negaram a prática regular de atividades físicas, ingestão de altas quantidades de alimentos com intervalos superiores a 3 horas, uso de bebidas alcoólicas e ausência de acompanhamento médico de rotina.

De acordo com Cataneo (2005), a condição socioeconômica, influencia os hábitos de vida da população e, conseqüentemente o risco de obesidade. Fatores, como educação, renda e ocupação influenciam nos hábitos comportamentais que afetam diretamente a ingestão calórica e gasto energético. Assim, quando alimentos saudáveis estão menos disponíveis à população de baixa renda, como peixes, grãos, carnes magras, frutas e verduras, observa-se um aumento da relação entre obesidade e nível socioeconômico. É o que também pode ser analisado pelos resultados do cluster 1. Neste grupo, a renda de 1 salário mínimo, o nível de escolarização e a rotina diária, colaboram para um padrão comportamental propenso ao desenvolvimento da obesidade.

O Cluster 2 mostrou que 7% dos participantes estão abaixo do grupo de risco para sobrepeso, com $IMC < 29$. Os participantes desse grupo são do sexo masculino, com idade de 25 anos e nível de escolaridade em nível de especialização. Neste grupo, observou-se a presença de doenças como o diabetes e triglicérides elevados, uso de bebidas alcoólicas e prática esporádica de atividades físicas.

O Cluster 3 definiu que 34% dos participantes são mulheres, com idade de 43 anos, solteiras, com renda salarial aproximada de 7 salários mínimos e nível superior completo. Em suas rotinas, destaca-se a prática de atividades físicas 2 vezes por semana e a ausência do uso de álcool e tabaco.

Fatores psicológicos como passividade, submissão, dependência, preocupação excessiva com comida, ingestão compulsiva de alimentos, baixa aceitação do esquema corporal, dificuldades de adaptação social, medo de rejeição, sentimento de desamparo, intolerância e culpa podem estar associados a comportamentos de adultos obesos que fazem hiperfagia. O ato de comer, para pessoas ansiosas e instáveis emocionalmente pode ser considerado como um tranquilizador, quando há dificuldades de lidar com a frustração e limites. Assim, esses fatores, integram um quadro multifatorial que pode predispor o sujeito à obesidade, juntamente com outros fatores como os genéticos e ambientais (Cataneo, 2005). É o que verificamos no Cluster 4, em que houve uma prevalência maior de participantes com histórico familiar de obesidade, triglicérides elevadas, ausência de atividades físicas e problemas psicológicos, como o transtorno do humor bipolar, que tem como pano de fundo, um funcionamento de personalidade do tipo ansioso. Adicionalmente, 3% dos participantes do sexo masculino, com idade de 47 anos, casados, com nível superior em nível de pós-graduação e renda salarial aproximada de 3 salários mínimos, afirmaram não praticarem atividades físicas, fazerem uso esporádico de álcool e apresentarem na família problemas de saúde como obesidade.

CONCLUSÃO

Conclui-se que neste estudo onde o objetivo foi a aplicação e análise de aspectos ligados a obesidade utilizando uma técnica de aprendizagem de máquina para agrupar informações de pessoas com características demográficas, socioeconômicas, de biotipo e de comportamento recorrente que foram obtidas por meio de um questionário online. Utilizou-se o algoritmo de aprendizagem não-supervisionado *Farthest First* que analisou os dados e criou cinco agrupamentos de indivíduos, com o objetivo de ser uma base de referência para a análise e identificação de novos usuários com comportamentos que tendem a obesidade. Verificou-se que a utilização desse instrumento pode avaliar satisfatoriamente o padrão comportamental e identificar fatores de risco associados ao desenvolvimento da obesidade. A organização dos dados em clusters permitiu uma visualização das características gerais da população pesquisada, oferecendo informações úteis para a prevenção dos riscos da obesidade e suas comorbidades.

Em trabalhos futuros pretende-se analisar tais grupos utilizando outros algoritmos de aprendizagem não-supervisionada, ou ainda organizando os dados para aplicação em algoritmos supervisionados onde poderemos obter modelos preditivos de classificação.

AGRADECIMENTOS

Os autores agradecem o suporte do Instituto Federal de Educação, Ciência e Tecnologia de Goiás (IFG), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERÊNCIAS BIBLIOGRÁFICAS

- Anjos, LA (1992). Índice de massa corporal (massa corporal. estatura-2) como indicador do estado nutricional de adultos: revisão da literatura. *Revista de Saúde pública*, 26: 431-436.
- Cataneo C, Carvalho AMP, Galindo EMC (2005). Obesidade e aspectos psicológicos: maturidade emocional, auto-conceito, locus de controle e ansiedade. *Psicologia: Reflexão e crítica*, 18(1): 39-46.
- Cavazzotto, TG, Brasil, MR, Oliveira, VM, Silva, SRD, Ronque, ERV, Queiroga, MR, & Serassuelo Junior, H (2014). Nutritional status of children and adolescents based on body mass index: agreement between World Health Organization and International Obesity Task Force. *Revista Paulista de Pediatria*, 32(1): 44-49.
- Coelho, MSPH, MOURA, ECD, & ASSIS, MAAD (2009). Aumento do índice de massa corporal após os 20 anos de idade e associação com indicadores de risco ou de proteção para doenças crônicas não transmissíveis. Monteiro, CA, Conde, WL, Castro, IRRD. A tendência cambiante da relação entre escolaridade e risco de obesidade no Brasil (1975-1997). 2013, *Cadernos de Saúde Pública*, 19: S67-S75.
- Dharmarajan, A., & Velmurugan, T. (2015). Lung cancer data analysis by k-means and farthest first clustering algorithms. *Indian Journal of science and Technology*, 8(15): 1-8.
- de Oliveira Couto, J, Santos, WS, Santos, JR, Simões, AEC, dos Santos, ÉGM, & dos Santos Silva, RJ (2017). Risco cardiovascular, índices antropométricos e percepção de qualidade de vida em idosos. *Scientia Plena*, 13(3).
- Duncan, BB, Duncan, MS, & Schmidt, MI (2005). Inflamação subclínica, obesidade, diabetes e doenças relacionadas. *Revista HCPA*, 25(3): 5-16.
- Faceli K, Lorena AC, Gama J, Carvalho ACPLF de (2011). *Uma Abordagem de Aprendizagem de Máquina*. Rio de Janeiro: LTC.
- Ferreira VA, Magalhães R (2006). Obesidade no Brasil: tendências atuais. *Revista portuguesa de saúde pública*, 24(2): 71-81.
- Freddo AR, Nishiyama MF, Zanuzo K, Koehnlein E (2019). Árvores de Decisão como Método de Mineração de Dados: Análise de Prontuários de uma Clínica Escola de Nutrição. *Revista da Associação Brasileira de Nutrição*, 10(2): 31-37.

- Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1): 10-18.
- Jacomini D, de Bona EM, Johann J (2018). Mineração de dados aplicada aos riscos da obesidade em Cascavel/PR. *Extensio: Revista Eletrônica de Extensão*, 15(30): 26-38.
- Kumar M (2013, November). An optimized farthest first clustering algorithm. In 2013 Nirma University International Conference on Engineering (NUICONE) (pp. 1-5). IEEE.
- Monteiro CA, Conde WL, Castro IRRD (2003). A tendência cambiante da relação entre escolaridade e risco de obesidade no Brasil (1975-1997). *Cadernos de Saúde Pública*, 19: S67-S75.
- Ramos RM, Palmero RMR, Ávalos RG, Lorenzo MMG (2012). Aplicación de métodos de selección de atributos para determinar factores relevantes en la evaluación nutricional de los niños. *Gaceta Médica Espirituana*, 9(1): 7.
- Rocha-Brischiliari SC, Agnolo CMD, Gravena AAF, Lopes TCR, Carvalho MDB, Pelloso SM Doenças crônicas não transmissíveis e associação com fatores de risco; 2014. *Rev Bras Cardiol*, 27(1): 35-42.
- Vadeyar DA, Yogish HK (2014). Farthest first clustering in links reorganization. *International Journal of Web & Semantic Technology*, 5(3): 17.

Mapeamento de espaço de atributos não-linearmente separável baseado na Golden wavelet: aplicação na regressão logística

Recebido em: 10/09/2020

Aceito em: 29/09/2020

 10.46420/9786588319260cap5

Bruno Rodrigues de Oliveira^{1*} 

Fabício Ely Gossler² 

Marco Aparecido Queiroz Duarte³ 

INTRODUÇÃO

Os termos inteligência artificial e aprendizado de máquina têm aparecido com frequência nos noticiários recentes, devido às incríveis aplicações que a cada dia surgem empregando as técnicas computacionais desenvolvidas nestas disciplinas. As aplicações vão desde diagnóstico de doenças, tal como a COVID-19 por meio de análises de raio-X (Basu et. al, 2020) até carros autônomos que utilizam visão computacional por meio do processamento e reconhecimento de imagens (Schwartz et. al, 2018).

Embora os primeiros estudos sobre estes temas remontem a década de 1950, nos últimos anos ocorreu uma explosão nas aplicações dessas técnicas, em parte devido à enorme massa de dados disponível atualmente e também à evolução dos hardwares utilizados para processamento.

Uma das metodologias mais empregadas atualmente são os algoritmos de aprendizado profundo (*Deep Learning*), que são essencialmente Redes Neurais Artificiais com uma estrutura de aprendizado não-supervisionado na etapa de extração de atributos e aprendizado supervisionado na etapa de classificação, possuindo elas, geralmente, milhares de parâmetros a estimar e várias camadas na rede, daí a denominação “profundo”. Embora os hardwares disponíveis no mercado deem conta do processamento necessário para implementar essas redes, o custo financeiro ainda é elevado. Por isso, muitos métodos clássicos de aprendizado de máquina, e que demandam menor custo computacional, ainda são empregados, principalmente nas situações onde um hardware mais poderoso não está acessível. Além disso, muitos destes métodos clássicos possuem poder explicativo inerente aos modelos aprendidos, enquanto que as redes neurais são vistas como “caixa preta”.

¹ Editora Pantanal.

² Rua Clevelândia 1924, Sibipiruna. Chapadão do Sul-MS. E-mail: fabricioely08@gmail.com.

³ Universidade Estadual de Mato Grosso do Sul (UEMS), Unidade Universitária de Cassilândia. Rodovia MS 306, Cassilândia-MS. E-mail: marco@uems.br.

*Autor de correspondência: bruno@editorapantanal.com.br

Um destes métodos clássicos é a Regressão Logística, que pode ser considerado um dos métodos mais simples de aprendizado de máquina, principalmente por haver poucos hiperparâmetros para serem ajustados. No entanto, uma das desvantagens desse método é que ele não resolve problemas muito complexos, quando é exigida uma fronteira de decisão não-linear. Para driblar essa desvantagem, uma das soluções é encontrar funções que mapeiam o espaço de atributos não-linearmente separável para um espaço linearmente separável.

Sendo assim, o texto ora apresentado tem duplo propósito: primeiro abordar os mecanismos matemáticos envolvidos no método de aprendizado de máquina denominado de Regressão Logística, e segundo, a proposição da utilização de funções wavelets para mapeamento do espaço de atributos, sendo mais especificamente utilizadas as funções Golden wavelets, que foram propostas recentemente e geraram resultados superiores às funções clássicas, relativamente às aplicações de classificação de padrões de arritmias cardíacas (Gossler, 2016).

Os resultados obtidos mostram que as Golden wavelets são adequadas para o mapeamento do espaço de atributos, melhorando significativamente a eficiência na classificação.

APRENDIZADO DE MÁQUINA

Conceitualmente, o aprendizado de máquina é uma das subáreas da inteligência artificial e é constituído de métodos matemático/computacionais que geram modelos de um certo ambiente. A função de modelagem é executada empregando os conjuntos de dados apresentados ao algoritmo de aprendizado de máquina. Esses dados, por sua vez, são informações representativas do ambiente.

Os métodos de aprendizado de máquina podem ser enquadrados em duas grandes categorias: aprendizado supervisionado e aprendizado não-supervisionado. Para a primeira categoria, duas subcategorias principais são: classificação e regressão (Kubat, 2015; Fisher; Lenz, 1996).

Para a classificação ou reconhecimento de padrões, duas etapas essenciais são implementadas, a saber: treinamento/indução e teste/validação. Na primeira etapa, emprega-se um conjunto de dados $T = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, com N instâncias extraídas do ambiente, sendo \mathbf{x}_n o n -ésimo vetor de atributos e y_n um rótulo que designa a qual classe pertence esse vetor. \mathbf{x}_n também recebe a denominação de variável preditora e y_n de variável predita.

Cada algoritmo de aprendizado de máquina utiliza os dados do conjunto T de uma maneira peculiar, entretanto, ao final do processo de indução sobre esse conjunto, o algoritmo fornece um modelo $\hat{h}(\mathbf{x}, \beta) = \hat{y}$, o qual é uma estimativa do modelo real h do ambiente (Haykin, 2007). O parâmetro β é relativo a cada algoritmo empregado, e seu tipo pode ser um simples escalar ou um vetor.

Em linhas gerais, o modelo \hat{h} recebe um vetor de atributos \mathbf{x} desconhecido e retorna um rótulo estimado \hat{y} . Assim sendo, na etapa de teste ou validação um conjunto $T' = \{(\mathbf{x}_n, y_n)\}_{n=1}^{N'}$ é utilizado para verificar o quão próximo o modelo estimado \hat{h} está do modelo real h . Para tanto, pode-se empregar a função (1) para calcular os erros de predição no caso de classificação binária

$$E = \sum_{n=1}^{N'} \begin{cases} 1, & \text{se } y_n \neq \hat{y}_n \\ 0, & \text{se } y_n = \hat{y}_n \end{cases} \quad (1)$$

Deste modo, se $E = N'$ então \hat{h} erra todas as predições, mas, por outro lado, se $E = 0$, então \hat{h} é uma estimativa perfeita do modelo do ambiente. Se $E = N'/2$, então a máquina induzida erra e acerta a mesma quantidade de vezes.

É importante notar nesse ponto que os conjuntos T e T' não foram restringidos a serem diferentes, pois, matematicamente, eles podem ser iguais. No entanto, na prática, para dados reais, aconselha-se que o teste/validação seja implementado tanto sobre o conjunto T quanto sobre o conjunto $T' \neq T$. Tal abordagem é relevante, pois o método de aprendizado pode se superajustar aos padrões. Sendo assim, validar o modelo inferido com dados desconhecidos o força a prever padrões não apresentados na fase de indução.

Outra questão de relevância na verificação da adequação do modelo estimado \hat{h} está relacionada ao conjunto de teste/validação. Quando é verificado que o modelo \hat{h} não é uma boa estimativa do modelo real h devido à quantidade de erros de predição, é comum voltar a fase de ajustes dos hiperparâmetros do modelo. Esses não são os parâmetros β , os quais são estimados a partir dos dados, mas são parâmetros ajustáveis ao nível do projetista, ou seja, são parâmetros fixos que variam de algoritmo para algoritmo. Após ajustar os hiperparâmetros, testa-se novamente o novo modelo encontrado. Quando esta ação é repetida, pode ocorrer que o modelo também fique superajustado aos hiperparâmetros. Neste caso, aconselha-se a utilização de um outro conjunto de validação, distinto do conjunto de teste empregado nas fases de ajustes do modelo.

O modelo \hat{h} , que é uma estimativa do modelo real h do ambiente, também pode ser encarado como uma fronteira de decisão à qual estão associadas superfícies de decisão. Quando os vetores de atributos \mathbf{x}_n do conjunto T estão em \mathbb{R}^2 essa fronteira é uma curva; no \mathbb{R}^3 ela é um plano; e no \mathbb{R}^Q com $Q > 3$, a fronteira de decisão é um hiperplano. Muitos algoritmos de aprendizado de máquina geram fronteiras de decisão apenas lineares, como o Perceptron e a Regressão Logística, que será abordada na próxima seção, enquanto outros têm capacidade de conceber fronteiras também não-lineares, como as Redes Neurais Artificiais e as Máquinas de Vetores Suporte (Haykin, 2007).

Na Figura 1 constam exemplos de superfícies de decisão com fronteiras: linear (Figura 1 (a)) e não-linear (Figura 1 (b)), onde os círculos representam as instâncias $\mathbf{x}_n \in \mathbb{R}^2$ e as cores designam a classe à qual cada instância pertence.

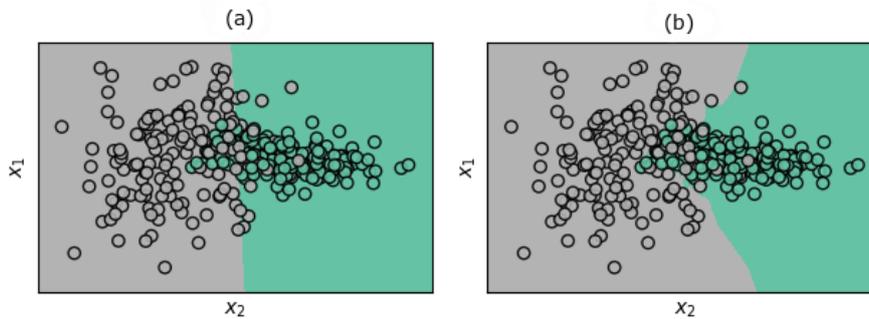


Figura 1. Exemplos de superfícies de decisão: (a) com fronteira linear e (b) com fronteira não-linear. Fonte: os autores.

Fica claro analisando a Figura 1 que a superfície de decisão não-linear resultará em menos erros de predição, pois a superfície verde inclui mais exemplos dessa classe. Muito embora, essa fronteira não-linear possa estar superajustada aos dados de treinamento/indução, acarretando em um resultado inferior sobre o conjunto de teste.

REGRESSÃO LOGÍSTICA

As técnicas de regressão permitem inferir a relação entre uma variável aleatória dependente (predita) e uma ou mais variáveis aleatórias independentes (preditoras) sem pressuposições a respeito dos processos geradores dos dados analisados (Loesch e Hoeltgebaum, 2012), ou seja, os modelos (equações) obtidos por meio destas técnicas consideram apenas as amostras fornecidas e alguns deles, como a regressão linear, alguma restrição sobre a distribuição de probabilidade dos dados.

A regressão logística é uma das técnicas de regressão específicas para aqueles fenômenos onde a variável predita tem uma distribuição de Bernoulli, ou seja, ela é uma variável binária. Os modelos \hat{h} inferidos são utilizados para explicar características dos fenômenos estudados, para classificação de dados e também para previsões. Esta técnica é extensivamente aplicada na área médica para classificação de doenças, inclusive este foi um dos primeiros problemas para o qual esta técnica de regressão teve sucesso, onde foram realizados experimentos para verificação de quais fatores mais influenciavam o surgimento de doenças cardiovasculares (Fávero et al., 2009). A capacidade explicativa da Regressão Logística a torna uma ferramenta importante, por exemplo, na determinação dos fatores que caracterizam um grupo de indivíduos em doentes ou saudáveis (Loesch; Hoeltgebaum, 2012). Nesse contexto, os atributos individuais \mathbf{x}_{kn} podem significar valores aferidos a partir de exames médicos.

Frisa-se que, embora o nome seja “Regressão” Logística, esse método de fato é utilizado para classificação e não regressão. Ficará mais claro adiante porque se utiliza a expressão “regressão”.

A seguir, baseado em Loesch e Hoeltgebaum (2012), Bishop (2006) e Fávero et al. (2009), com as devidas adaptações, a técnica de regressão logística é pormenorizadamente explicada.

Seja $\mathbf{x}_n \in \mathbb{R}^K$ uma realização do vetor aleatório \mathbf{X} com distribuição de probabilidade qualquer e $y_n \in \{0, 1\}$ uma realização da variável aleatória Y com distribuição de Bernoulli, e $T = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$ um conjunto de N amostras aleatórias. Neste contexto, considera-se um problema de reconhecimento de padrões binário, ou seja, há apenas duas classes disponíveis, pois $y_n \in \{0, 1\}$.

Sem perda de generalidade, supõe-se que a probabilidade de ocorrência do valor categórico (rótulo) 1, dado o vetor \mathbf{x}_n , é igual a p_n , e de ocorrência do valor 0, conseqüentemente é $q_n = 1 - p_n$, isto é:

$$P(y_n = 1 | \mathbf{x}_n) = p_n, \tag{2}$$

$$P(y_n = 0 | \mathbf{x}_n) = 1 - p_n. \tag{3}$$

Visto que a variável y_n é de Bernoulli e a função densidade de probabilidade de uma distribuição de Bernoulli para uma variável aleatória y é dada pela expressão $f_y(y) = p^y(1-p)^{1-y}$ dada uma probabilidade p , as equações (2) e (3) podem ser reescritas como uma única função de probabilidade:

$$P(y_n | \mathbf{x}_n) = p_n^{y_n} (1 - p_n)^{1-y_n}. \tag{4}$$

A razão das probabilidades, dada por

$$R(p_n) = \frac{p_n}{1-p_n} = \frac{P(y_n=1|\mathbf{x}_n)}{P(y_n=0|\mathbf{x}_n)}, \tag{5}$$

é denominada chance ou possibilidade, e se traduz como a quantidade de vezes que o valor $y_n = 1$ é mais provável de ocorrer que o valor $y_n = 0$, dado o vetor aleatório \mathbf{x}_n .

Com o intuito de obter uma relação entre as variáveis predictoras e preditas, considera-se que a probabilidade da variável predita y_n ser igual a 1 é dada por p_n . Disto segue que:

$$\begin{aligned} g(y_n) &= p_n \\ &= \frac{1}{1 + \frac{1-p_n}{p_n}} \\ &= \frac{1}{1 + \frac{e^{\ln(1-p_n)}}{e^{\ln p_n}}} \\ &= \frac{1}{1 + e^{-y_n}}, \end{aligned} \tag{6}$$

onde e é o número de Euler e $\ln x$ é o logaritmo de x na base e , também chamado de logaritmo natural.

Partindo do resultado da equação (6) verifica-se que

$$\frac{p_n}{1-p_n} = \frac{\frac{1}{1+e^{-y_n}}}{1-\frac{1}{1+e^{-y_n}}} = \frac{\frac{1}{1+e^{-y_n}}}{\frac{1+e^{-y_n}-1}{1+e^{-y_n}}} = \frac{1}{e^{-y_n}} = e^{y_n}.$$

Computando o logaritmo natural de ambos lados, a variável predita é dada por:

$$\begin{aligned} y_n &= \ln\left(\frac{p_n}{1-p_n}\right), \\ &= \ln R(p_n). \end{aligned} \tag{7}$$

Levando em conta que $R(p_n)$, conforme equação (5), representa o quanto $y_n = 1$ é mais provável de ocorrer que $y_n = 0$, então $R(q_n)$ representa a chance inversa. A equação (7) pode então ser escrita como

$$y_n = \ln\left(\frac{p_n}{q_n}\right), \tag{8}$$

que representa o logaritmo natural da razão entre a probabilidade de uma amostra aleatória (\mathbf{x}_n, y_n) pertencer ao grupo $y_n = 1$ pela probabilidade de pertencer ao grupo $y_n = 0$. Como a função logística multivariada $g(y_n)$, equação (6), varia assintoticamente entre 0 e 1, quando y_n tende a $+\infty$ e $-\infty$, respectivamente, então é uma boa candidata para modelar as probabilidades consideradas.

Uma das premissas da Regressão Logística é que a variável predita y_n seja modelada como uma combinação linear das variáveis predictoras \mathbf{x}_n (Fávero et al., 2009), resultando em um hiperplano:

$$y_n = \beta_0 + \sum_{k=1}^K \beta_k x_{kn}, \tag{9}$$

onde $\boldsymbol{\beta} = [\beta_0, \beta_1, \dots, \beta_K]$ é um vetor de parâmetros desconhecidos, que devem ser aprendidos (estimados) pelo algoritmo. Eles ponderam a importância relativa das variáveis aleatórias na explicação do valor predito y_n , exceto β_0 que determina onde o hiperplano cruza os eixos; x_{kn} representa a k -ésima componente do vetor \mathbf{x}_n .

Essa equação é um modelo de regressão linear que estima o valor y_n a partir da combinação linear dos atributos x_{kn} . Entretanto, e este é um ponto importante, na equação (9), a variável y_n poderá assumir valores diferentes de 0 e 1, pois as componentes do vetor aleatório \mathbf{x}_n são reais. Por isso, o valor predito deve ser modulado pela função logística $g(y_n)$ expressa na equação (6), garantindo assim que se tenha um valor de probabilidade, isto é, um valor que esteja entre 0 e 1,

já que esses valores são o mínimo e o máximo, respectivamente, retornados pela função $g(y_n)$. Desse modo, essa função fornece a probabilidade da variável y_n ser igual a 1, que é equivalente a considerar um corte α nas imagens da função, da seguinte maneira:

$$y_n = \begin{cases} 1, & \text{se } g(y_n) \geq \alpha \\ 0, & \text{se } g(y_n) < \alpha \end{cases} \quad (10)$$

sendo que um valor comum é $\alpha = 0,5$ (Singh, 2019).

Note que, para prever o valor da variável binomial y_n a função logística receberá a combinação linear dos parâmetros desconhecidos β e das componentes do vetor aleatório \mathbf{x}_n , de acordo a equação (9). Ou seja, a função logística implementada é de fato igual a:

$$g\left(\beta_0 + \sum_{k=1}^K \beta_k x_{kn}\right) = \frac{1}{1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}}}. \quad (11)$$

Logo, para um dado corte $\alpha \in]0, 1[$ e um vetor aleatório predictor \mathbf{x}_n , a variável predita y_n é obtida calculando:

$$y_n = \begin{cases} 1, & \text{se } \frac{1}{1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}}} \geq \alpha \\ 0, & \text{se } \frac{1}{1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}}} < \alpha \end{cases}, \quad (12)$$

Isso significa que as variáveis predictoras que retornam valores da função logística acima ou igual a α serão atribuídas à predição 1, e caso contrário a predição será 0.

Na Figura 1 está ilustrada uma função logística univariada, que é do tipo sigmoide, isto é

$$g_u(\beta_0 + \beta_1 x) = \frac{1}{1 + e^{-\beta_0 - \beta_1 x}}. \quad (13)$$

com $\beta_1 > 0$.

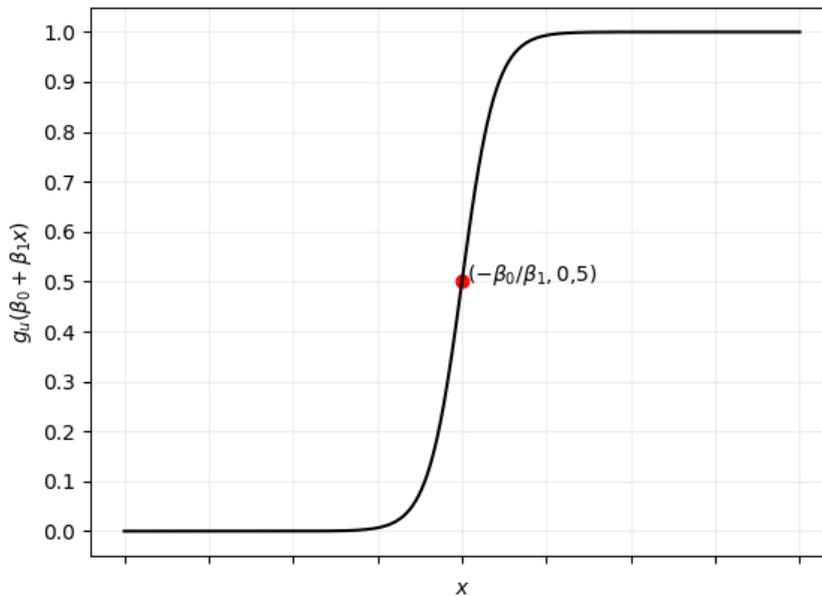


Figura 2. Ilustração de uma função logística univariada. Em vermelho está destacado o ponto de inflexão da curva. Fonte: os autores.

Observa-se que a curva ilustrada na Figura 2 tende assintoticamente a 0 e 1, como já mencionado anteriormente, devido a componente exponencial da função logística. O ponto de inflexão da curva é dado pela derivada segunda da função logística igualada a zero:

$$0 = \frac{d^2 g_u(\beta_0 + \beta_1 x)}{dx^2} = \frac{2\beta_1^2 e^{-2(\beta_0 + \beta_1 x)}}{(1 + e^{-\beta_0 - \beta_1 x})^3} - \frac{\beta_1^2 e^{-\beta_0 - \beta_1 x}}{(1 + e^{-\beta_0 - \beta_1 x})^2} \tag{14}$$

cuja solução real é dada por $\beta_0 + \beta_1 x = 0$. Logo,

$$x = -\frac{\beta_0}{\beta_1} \tag{15}$$

é a abscissa do ponto de inflexão. Substituindo este valor na função logística $g_u(\beta_0 + \beta_1 x)$, resulta que a imagem correspondente é 0,5. Isto significa que a mudança de concavidade da curva logística ocorre no ponto $(-\beta_0/\beta_1; 0,5)$, conforme ilustrado na Figura 1, ponto em vermelho. Assim, o valor de β_0 desloca a curva horizontalmente proporcionalmente a β_1^{-1} , o que é facilmente entendido ao tomar $\beta_1 = 1$. Por outro lado, β_1 influencia na inclinação desta curva, pois a derivada primeira é dada por

$$\beta_1 \frac{e^{-\beta_0 - \beta_1 x}}{(1 + e^{-\beta_0 - \beta_1 x})^2} \tag{16}$$

Afim de estimar os parâmetros β da função logística $g(\beta_0 + \sum_{k=1}^K \beta_k x_{kn})$, com probabilidade p_n , é empregado o método da máxima verossimilhança, considerando que $p_n = 1/(1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}})$ de acordo com equação (7) e que a partir da equação (4) tem-se que:

$$\begin{aligned}
 P(y_n | \mathbf{x}_n) &= p_n^{y_n} (1 - p_n)^{1 - y_n} \\
 &= \left(\frac{1}{1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}}} \right)^{y_n} \left(1 - \frac{1}{1 + e^{-\beta_0 - \sum_{k=1}^K \beta_k x_{kn}}} \right)^{1 - y_n} \\
 &= \frac{e^{y_n(\beta_0 + \sum_{k=1}^K \beta_k x_{kn})}}{e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1}.
 \end{aligned}$$

Portanto, a função de verossimilhança é

$$L(\boldsymbol{\beta} | \mathbf{y}) = \prod_{n=1}^N P(y_n | \mathbf{x}_n, \boldsymbol{\beta}) = \prod_{n=1}^N \frac{e^{y_n(\beta_0 + \sum_{k=1}^K \beta_k x_{kn})}}{e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1}. \quad (17)$$

Aplicando o logaritmo natural com o intuito de simplificar a função de verossimilhança, obtém-se uma função de $\boldsymbol{\beta}$:

$$\begin{aligned}
 E(\boldsymbol{\beta}) &= \ln L(\boldsymbol{\beta} | \mathbf{y}) \\
 &= \ln \prod_{n=1}^N \frac{e^{y_n(\beta_0 + \sum_{k=1}^K \beta_k x_{kn})}}{e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1} \\
 &= \sum_{n=1}^N \ln e^{y_n(\beta_0 + \sum_{k=1}^K \beta_k x_{kn})} - \ln (e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1) \\
 &= \sum_{n=1}^N y_n \left(\beta_0 + \sum_{k=1}^K \beta_k x_{kn} \right) - \ln (e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1)
 \end{aligned} \quad (18)$$

pois os parâmetros $\boldsymbol{\beta}$ que maximizam a função (17) são os mesmos que maximizam (18), já que a função logaritmo é crescente.

Para encontrar o máximo da função $\ln L(\boldsymbol{\beta} | \mathbf{y})$ calcula-se a derivada parcial, o gradiente, em relação ao vetor de parâmetros $\boldsymbol{\beta}$, dada por:

$$\nabla E = \frac{\partial E(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{n=1}^N \left[\left(y_n - \frac{1}{e^{\beta_0 + \sum_{k=1}^K \beta_k x_{kn}} + 1} \right) \sum_{k=1}^K x_{kn} \right]. \quad (19)$$

É importante notar que o gradiente segue a direção dada pelo erro entre o valor real y_n e o valor predito $\hat{y}_n = 1 / (e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn}} + 1)$, vezes a soma dos atributos.

Igualando o gradiente ∇E a zero, os parâmetros $\boldsymbol{\beta}$ são obtidos pela solução da seguinte equação:

$$\sum_{n=1}^N \left[\left(y_n - \frac{1}{e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn}} + 1} \right) \sum_{k=1}^K x_{kn} \right] = 0, \quad (20)$$

onde $\hat{\boldsymbol{\beta}}$ é o vetor de parâmetros estimado.

As soluções da equação (20) são alcançadas utilizando métodos de otimização, tais como gradiente descendente, mínimos quadrados e Newton-Raphson. Dentre as soluções possíveis duas delas são obtidas ao considerar valores muito pequenos ou muito grandes para os parâmetros $\hat{\beta}$, quando as instâncias de treinamento pertencem restritamente a classe positiva ($y_n = 1$) ou negativa ($y_n = 0$), respectivamente.

Em outros termos, se

$$\lim_{\hat{\beta} \rightarrow 0} e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn}} = 0, \quad (21)$$

então:

$$\lim_{\hat{\beta} \rightarrow 0, y_n = 1} y_n - \frac{1}{e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn+1}} + 1} = 0, \quad (22)$$

independentemente dos valores dos atributos x_{kn} .

Considerando as equações (12) e (22), e tomando $\alpha = 0,5$, toda instância x_n será atribuída a classe 1, quando $\hat{\beta} \rightarrow 0$. Assim, obtêm-se um modelo $\hat{h}(x, \beta)$ que é enviesado, porque prevê todos os exemplos como pertencentes a classe 1.

Analogamente, se $\hat{\beta} \rightarrow \infty$, então

$$\lim_{\hat{\beta} \rightarrow \infty} e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn}} = \infty, \quad (23)$$

portanto,

$$\lim_{\hat{\beta} \rightarrow \infty, y_n = 0} y_n - \frac{1}{e^{\hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k x_{kn}} + 1} = 0, \quad (24)$$

também é uma solução da equação (20). Para este caso, a equação (12) sempre retorna a classe 0.

Para evitar que o modelo aprendido preveja apenas uma das classes, utilizam-se técnicas de regularização, as quais restringem os valores que os parâmetros $\hat{\beta}$ podem assumir, empregando para tanto uma penalização aos valores extremos.

As técnicas de regularização L1 e L2 são implementadas adicionando na função $E(\beta)$, equação (18), os termos de penalização $\lambda \sum_{k=0}^K |\hat{\beta}_k|$ e $\lambda \sum_{k=0}^K \hat{\beta}_k^2$, respectivamente, onde $\lambda > 0$ é um hiperparâmetro que relativiza a importância do termo de regularização. As equações resultantes são

$$E_{L1}(\beta) = E(\beta) + \lambda \sum_{k=0}^K |\hat{\beta}_k| \quad (25)$$

$$E_{L2}(\beta) = E(\beta) + \lambda \sum_{k=0}^K \hat{\beta}_k^2. \quad (26)$$

Para exemplificar a utilização do método de Regressão Logística para classificação de padrões, na Figura 3a estão ilustrados os dados de treinamento, onde os pontos verde e púrpura representam

duas classes distintas, e o ponto amarelo é uma instância não rotulada. Neste caso os vetores de atributos \mathbf{x}_n possuem apenas duas componentes, isto é, $\mathbf{x}_n = (x_{1n}, x_{2n})$.

Na Figura 3b a linha branca é a fronteira de decisão obtida ao empregar a Regressão Logística sobre os dados de treinamento da Figura 3a. Ela é modelada pela equação (9) após terem sido aprendidos os parâmetros β . Sobre essa fronteira a probabilidade de que um ponto pertença a classe verde ou a classe púrpura é a mesma, que é 0,5. Portanto, os pontos que estão localizados sobre a fronteira possuem classificação dúbia. Por outro lado, as superfícies de decisão verde e púrpura possuem probabilidade 1 e 0 para classificação dos exemplos das classes verde. Observe que a classe verde é a classe principal, logo, calcula-se a probabilidade p_n da instância pertencer a classe verde e a probabilidade $q_n = 1 - p_n$ de não pertencer a classe verde. Por isso, a cor púrpura está associada a probabilidade 0, na escala de cores da Figura 3b.

Neste exemplo é evidente que algumas instâncias não serão corretamente classificadas.

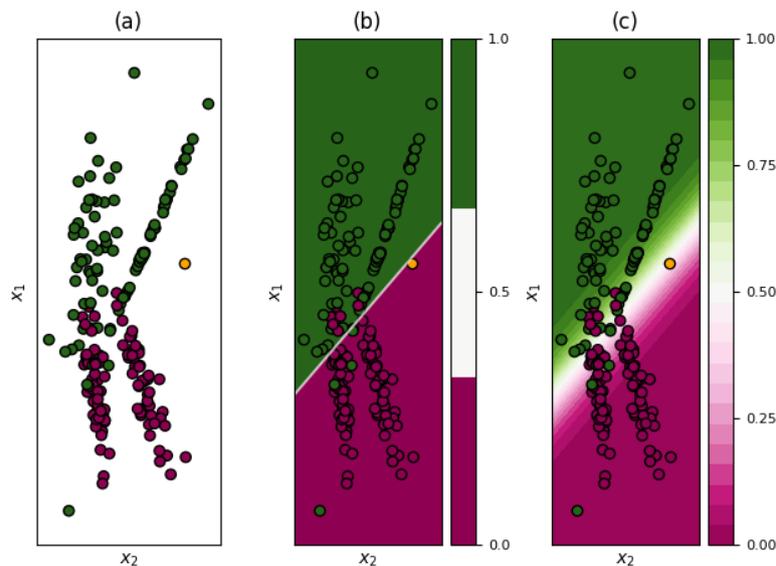


Figura 3. Exemplo de aplicação da Regressão Logística sobre o conjunto de testes (a), gerando as superfícies de decisão (b) e (c). Fonte: os autores.

Na Figura 3c são apresentadas diversas fronteiras de decisão, cada uma com probabilidade maior, quanto mais longe estão da fronteira de decisão com probabilidade de 0,5. Nota-se que quanto mais distante da fronteira de decisão o exemplo estiver, maior será a probabilidade de que esse exemplo pertença à classe que a superfície determina.

Por fim, a instância (ponto amarelo) desconhecida é classificada como pertencente a classe púrpura, pois está localizada abaixo da fronteira de decisão, na superfície correspondente a essa classe.

Uma limitação inerente à Regressão Logística na classificação de padrões é sua capacidade de gerar bons resultados apenas para problemas linearmente separáveis, como aquele ilustrado na Figura

3. Devido a fronteira de decisão gerada ser linear, em ambientes onde as classes se distribuem conforme a Figura 4 (a), por exemplo, a aplicação da Regressão Logística não é apropriada, porque não é possível obter uma fronteira de decisão que separe as classes adequadamente.

Nestas situações uma das abordagens empregadas é a implementação de uma função $\varphi: \mathbf{x}_n \mapsto \mathbf{z}_n$, que mapeia o espaço de atributos não-linearmente separável para um espaço linearmente separável. Nesse novo espaço, a regressão logística pode ser empregada, mas agora sobre os vetores de dados mapeados, isto é, sobre \mathbf{z}_n .

Esta abordagem, aliás, é o foco da pesquisa apresentada neste texto. As funções que serão utilizadas para o mapeamento do espaço de atributos serão apresentadas na próxima seção.

FUNÇÕES WAVELETS

As “funções wavelets” são funções com características especiais utilizadas na Transformada Wavelet para análise e processamento de sinais. Dentre as várias aplicações desta transformada, aquelas que mais ganharam repercussão estão associadas à compressão e remoção de ruído em sinais e imagens digitais (Mallat, 2009). Para remoção de ruído de sinais de eletrocardiograma (ECG), por exemplo, o método de anulamento dos coeficientes menos significantes utilizando a análise multirresolução wavelet supera os métodos comumente empregados na literatura especializada (Oliveira et al., 2018).

A Transformada de Fourier de um sinal $f(t) \in L^2(\mathbb{R})$ é definida por Oppenheim, Willisky e Nawab (1997) como a integral imprópria:

$$F_f(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt, \quad (27)$$

onde j é a unidade imaginária e $L^2(\mathbb{R})$ é o espaço das funções de quadrado integrável.

Uma função $\psi(t) \in L^2(\mathbb{R})$, com norma euclidiana unitária, i.e., $\|\psi(t)\|_2 = 1$, é denominada de **função wavelet** se a condição de admissibilidade dada na equação (28) for satisfeita (Mallat, 2009).

$$C_\psi = 2\pi \int_{-\infty}^{\infty} \frac{|F_\psi(\omega)|^2}{|\omega|} d\omega < \infty \quad (28)$$

A fim de garantir a convergência da integral na equação (28) impõe-se $F_\psi(0) = 0$ (Daubechies, 1992). Esta restrição, tomando a equação (27), implica que

$$0 = F_\psi(0) = \int_{-\infty}^{\infty} \psi(t)e^{-j0t} dt = \int_{-\infty}^{\infty} \psi(t)dt. \quad (29)$$

A equação (29), que impõe média nula as funções wavelets e a restrição de que $\psi(t)$ pertença ao espaço das funções de quadrado integrável evidenciam o comportamento ondulatório característico das funções wavelets: a área da curva da função, acima e abaixo do eixo t , devem ser a mesma e, quando $t \rightarrow \pm\infty$ então $\psi(t)$ decai rapidamente, anulando-se. Portanto, a “ondinha” não oscila

indefinidamente, mas num intervalo curto, sendo este seu intervalo suporte. Isto garante que as wavelets sejam bem localizadas no tempo, implicando que sua energia está concentrada em uma região estreita.

Uma família de funções wavelets é obtida inserindo parâmetros de escala s e deslocamento τ à função $\psi(t)$, obtendo assim versões escalonadas e transladadas no tempo, conforme equação (30), onde $|s|^{-1/2}$ é o termo de normalização da função.

$$\psi_{s,\tau}(t) = |s|^{-1/2} \psi\left(\frac{t-\tau}{s}\right). \quad (30)$$

A função wavelet $\psi(t)$ é denominada de “wavelet mãe” e esta tem energia unitária, pois as “wavelets filhas” são restringidas à normalização $\|\psi_{s,\tau}(t)\|_2 = \|\psi(t)\|_2 = 1$, implicando que:

$$1 = \|\psi(t)\|_2 = \int_{-\infty}^{\infty} |\psi(t)|^2 dt. \quad (31)$$

A equação (30) possibilita localizar, em um vetor, estruturas temporais tão grandes ou tão pequenas quanto se deseja, bastando para isto escolher adequadamente o parâmetro de escala s e transladar a wavelet resultante por toda a extensão do vetor. Esta, aliás, é a característica da análise Wavelet que a torna uma ferramenta mais adequada que a Transformada de Fourier de Tempo Curto para análise de transitórios.

Quando o parâmetro de escala é $0 < s < 1$, $\psi_{s,\tau}(t)$ é uma versão comprimida de $\psi(t)$ ao longo do eixo horizontal. Ao deslocar esta versão ao longo do vetor, multiplicando-a por cada seguimento determinado pela janela wavelet, o resultado é que aqueles seguimentos com maiores oscilações serão melhor localizados do que quando se considera uma versão de $\psi(t)$ mais expandida, que é obtida escolhendo-se $s > 1$. É claro que quando a wavelet é escalonada seu suporte é alterado. Assim, se a wavelet $\psi(t)$ tem um suporte de tamanho N então $\psi_{s,\tau}(t)$ terá um suporte de tamanho $|s|N$. Quando esta wavelet é deslocada, então seu suporte fixa-se próximo a sua localização temporal, ou seja, em $t = \tau$ (Kaiser, 1994).

Além do parâmetro de escala, a quantidade de momentos nulos de uma wavelet também afeta a forma de onda da função. Para uma função wavelet $\psi(t) \in L^2(\mathbb{R})$, o g -ésimo momento nulo é dado pela integral

$$m_g = \int_{-\infty}^{\infty} t^g \psi(t) dt, \quad g = 0, 1, 2, \dots, G - 1. \quad (32)$$

O teorema do momento para a Transformada de Fourier de uma função ψ consiste do seguinte resultado (Vetterli e Kovacevic, 2007):

$$\left. \frac{\partial^g F_\psi(\omega)}{\partial \omega^g} \right|_{\omega=0} = (-j)^g m_g. \quad (33)$$

Considerando o par da Transformada de Fourier, $\psi(t) \leftrightarrow F_\psi(\omega)$ e o valor da transformada na origem, estabelecido anteriormente para a condição de admissibilidade, a saber, $F_\psi(0) = 0$, e substituindo estes na expressão (33), se tem:

$$\int_{-\infty}^{\infty} t^g \psi(t) dt = 0, g = 0, 1, 2, \dots, G - 1. \quad (34)$$

Se o valor da integral na equação (34) se anula para $G - 1$ momentos, então diz-se que a função wavelet ψ tem G momentos nulos. Esse resultado é utilizado pelo Teorema 1 para definir as funções wavelets como resultantes das G -ésimas derivada de uma função de suavização $\theta(t)$.

Teorema 1. Uma wavelet $\psi(t)$, com suporte compacto, tem G momentos nulos se, e somente se, existe uma função $\theta(t)$ com decaimento rápido, tal que

$$\psi(t) = (-1)^G \frac{d^G \theta(t)}{dt^G}. \quad (35)$$

Por meio do Teorema 1 pode-se estabelecer a seguinte notação $\psi_{s,\tau,G}(t) = (-1)^G \frac{d^G}{dt^G} \left[|s|^{-1/2} \psi \left(\frac{t-\tau}{s} \right) \right]$, para uma família de funções wavelets na escala s , deslocamento τ e G momentos nulos. Entretanto, no contexto de classificação de padrões, os vetores de atributos não estão indexados temporalmente, ou seja, a ordem com que os atributos estão inseridos no vetor não é relevante para a tarefa de classificação. Por isso, a partir de agora o parâmetro de deslocamento será desconsiderado. Logo, as funções wavelets serão implementadas apenas em relação aos parâmetros de escala e momentos nulos, isto é, $\psi_{s,G}(t) = (-1)^G \frac{d^G}{dt^G} \left[|s|^{-1/2} \psi \left(\frac{t}{s} \right) \right]$.

Dentre as diversas funções wavelets que são expressas como a derivada de uma função de suavização, conforme equação (35), recentemente uma nova família foi proposta por Lamblém et. al (2016), a qual é denominada Golden wavelets em referência ao famoso número de ouro. Essas funções apresentam resultados superiores às funções wavelets clássicas como a Mexican Hat, na análise da regularidade de sinais com descontinuidades (Gossler et al., 2018), e às wavelets Gaussianas e Daubechies (para alguns momentos nulos) na aplicação de extração de atributos para reconhecimento de arritmias cardíacas (Gossler et al., 2016).

Essas funções são obtidas pela derivada do quociente entre polinômios com coeficientes de Fibonacci, que se expressam por meio da igualdade

$$p_k(t) = t p_{k-1}(t) + H_{k-1}, k \geq 1, \quad (36)$$

onde H_{k-1} é o k -ésimo termo da sequência de Fibonacci $\{1, 1, 2, 3, 5, 8, 13, 21, \dots\}$, que é formada pela soma dos dois termos anteriores.

Um exemplo de Golden wavelet é obtido considerando a função de suavização $\theta(t)$ como o quociente entre os polinômios $p_0(t) = 1$ e $p_2(t) = t^2 + t + 2$ (Gossler et al., 2018), isto é:

$$\theta(t) = \frac{1}{t^2 + t + 2}, \quad (37)$$

e aplicando a derivação, equação (35):

$$\psi(t) = -\frac{2t + 1}{(t^2 + t + 2)^2}, \quad (38)$$

Resultando em uma wavelet com um momento nulo. Para obter wavelets com mais momentos nulos basta derivar sucessivamente a equação (38). Na análise comparativa apresentada em Gossler et al. (2018), estuda-se a função Golden Hat, que possui quatro momentos nulos.

Na seção ulterior as soluções de problemas de classificação não-linearmente separáveis são abordadas e proposta uma técnica baseada nas funções Golden wavelets para mapeamento do espaço de atributos.

RESULTADOS E DISCUSSÕES

O primeiro problema investigado está ilustrado na Figura 4, onde duas classes, amarelo e azul, são representadas por dois atributos x_1 e x_2 , onde a linha vermelha tracejada é uma fronteira de decisão aprendida pela Regressão Logística, cuja expressão é dada por $y_n = 1,11 \times 10^{-5} - 0,012x_1 - 0,002x_2$.

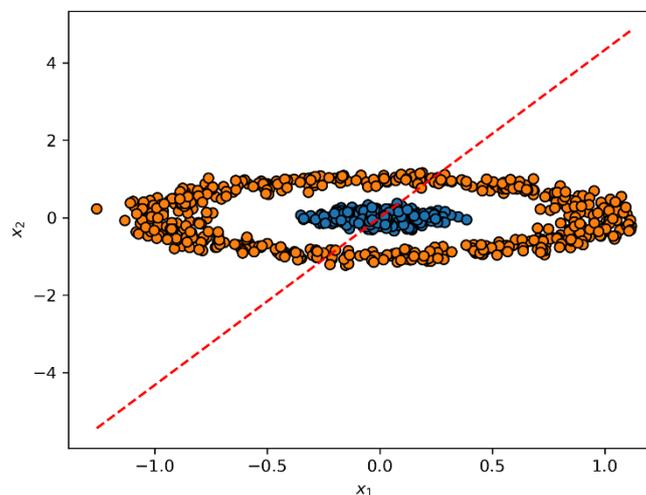


Figura 4. Exemplo de um problema não-linearmente separável. A linha vermelha tracejada representa a fronteira de decisão obtida empregando uma Regressão Logística. Fonte: os autores.

Este é claramente um problema do tipo não-linearmente separável, porque nenhuma fronteira de decisão linear aprendida é capaz de separar as classes adequadamente em superfícies de decisão que contenham a maior quantidade possível de instância da classe relativa a ela, lembrando que essas fronteiras devem particionar o plano em dois outros planos disjuntos. Deste modo, a Regressão

Logística não fornecerá bons resultados de classificação. Mais especificamente, neste exemplo ocorrem 511 erros de predição, de um total de 1.000 instâncias utilizadas para treinamento/teste. Observe que neste caso não foi selecionado outro conjunto para testar o modelo induzido, pois não se faz necessário nesse momento.

Um erro de cerca de 50% em um problema binário balanceado, como o exemplo em discussão, significa que a fronteira de decisão aprendida é irrelevante, porque pode-se simplesmente escolher uma classe ou outra para cada instância apresentada, que o resultado seria análogo.

Com o objetivo de tornar linearmente separável o espaço de atributos ilustrado na Figura 4 é proposto o mapeamento a seguir, equação (39), baseado na Golden wavelet descrita na equação (38):

$$\mathbf{z}_n = \varphi(\mathbf{x}_n) = [\psi_{s,G}(x_{1n}, x_{2n})]^T, \quad (39)$$

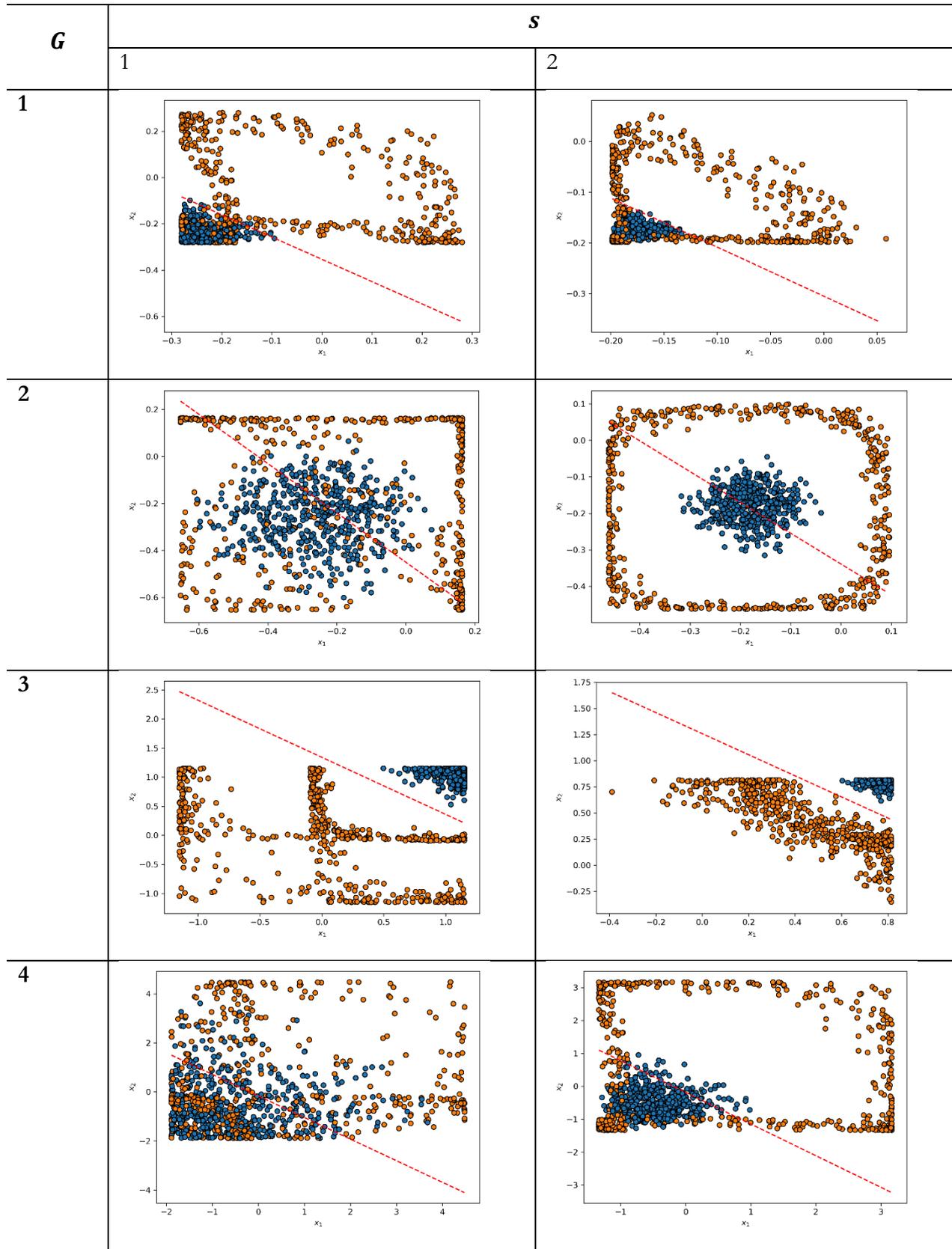
para todo n , recordando que $\mathbf{x}_n = (x_{1n}, x_{2n})$ é o n -ésimo padrão que carrega informações características de um certo ambiente, e \mathbf{z}_n o respectivo padrão mapeado, sendo G e s a quantidade de momentos nulos e a escala considerada, respectivamente, para as funções wavelets; $\psi_{s,G}(x_{1n}, x_{2n})$ significa que a função wavelet $\psi_{s,G}$ é aplicada em cada coordenada, ou alternativamente, podemos também pensar na representação bidimensional dada na equação (40)

$$\psi_{s,G}(x_1, x_2) = \left((-1)^G \frac{d^G}{dt^G} \left[|s|^{-1/2} \psi \left(\frac{x_1}{s} \right) \right], (-1)^G \frac{d^G}{dt^G} \left[|s|^{-1/2} \psi \left(\frac{x_2}{s} \right) \right] \right). \quad (40)$$

No

Quadro 1 estão ilustradas as fronteiras de decisão, linhas vermelhas tracejadas, nos espaços de atributos mapeados pelas Golden wavelets para distintos momentos nulos G e escalas s . Na Tabela 1 estão exibidos os erros de estimação, calculados pela equação (1), gerados ao tomar os espaços de atributos mapeados apresentados no Quadro 1.

Quadro 1. Fronteiras de decisão (linha vermelha tracejada) considerando funções Golden wavelets com diferentes momentos nulos (G) e em distintas escalas (s), para o problema ilustrado na Figura 4. Fonte: os autores.



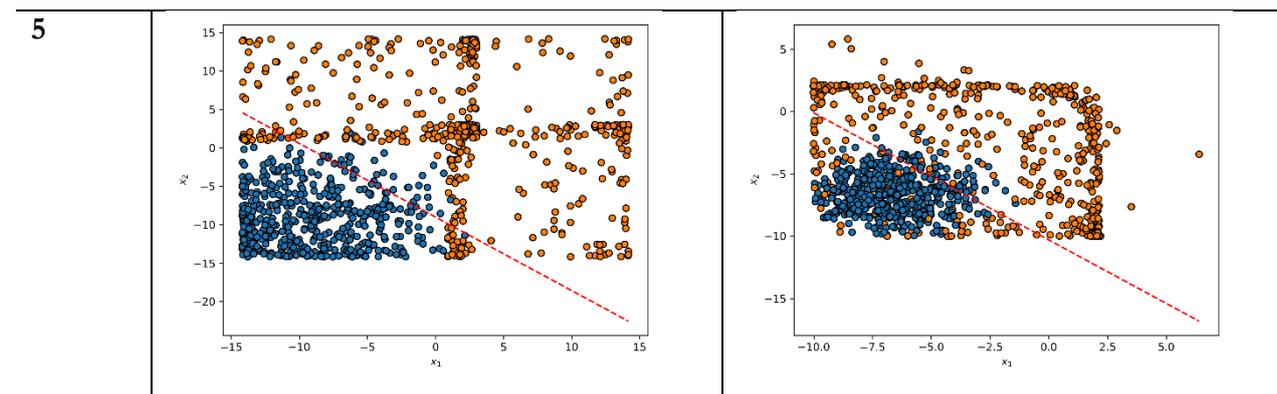


Tabela 1. Erros de estimação considerando funções Golden wavelets com diferentes momentos nulos (G) e em distintas escalas (s). Fonte: Os autores.

G	s	
	1	2
1	168	202
2	393	461
3	0	0
4	392	235
5	92	140

Observa-se na Tabela 1 que apenas quando são implementadas as Golden wavelets $\psi_{1,3}$ e $\psi_{2,3}$ é que os espaços de atributos mapeados se tornam perfeitamente (erro nulo) linearmente separáveis. As demais possibilidades também geram espaços linearmente separáveis, mas com maiores erros de predição, como por exemplo, a utilização das Golden wavelet $\psi_{1,5}$. O pior resultado é obtido implementando a Golden wavelet $\psi_{2,2}$ e, pela fronteira respectiva do

Quadro 1, nota-se que o padrão de dispersão dos atributos neste caso é análogo ao do problema original, Figura 4, exceto que ocorre uma dispersão maior dos atributos.

Em resumo, as Golden wavelets, equação (38), com 3 momentos nulos, são adequadas para transformação do espaço de atributos daqueles problemas similares ao ilustrado na Figura 4. Para compreender porque estas wavelets são adequadas é necessário analisar sua expressão analítica, que é dada por:

$$\psi_{s,3}(t) = -\frac{6\left(2\frac{t}{s} + 1\right)\left[2\left(\frac{t}{s}\right)^2 + 2\frac{t}{s} - 3\right]}{\left[2\left(\frac{t}{s}\right)^2 + \frac{t}{s} + 2\right]^4}. \quad (41)$$

Para exemplificar, tomam-se quatro instâncias: $x_{am1} = (-0,31; 0,67)$, $x_{am2} = (0,38; -0,69)$, $x_{az1} = (-0,35; 0,08)$ e $x_{az2} = (0,38; -0,04)$ das classes amarelo e azul, respectivamente, as quais estão localizadas na mesma superfície de decisão gerada pela Regressão Logística, significando que, pelo modelo aprendido, pertencem a mesma classe, ou seja, x_{am1} e x_{az1} estão no lado superior da fronteira de decisão da Figura 4, e x_{am2} e x_{az2} , no lado inferior (Veja Figura 5 (a)).

O mapeamento realizado pela Golden wavelet $\psi_{1,3}$, mapeia as instâncias x_{am1} , x_{am2} , x_{az1} e x_{az2} nas localizações: $x'_{am1} = (0,76; 0,11)$, $x'_{am2} = (0,50; -0,76)$, $x'_{az1} = (0,63; 1,03)$ e $x'_{az2} = (0,50; 1,51)$, respectivamente. Na Figura 5 estão ilustradas as novas localizações destas instâncias bem como as fronteiras de decisões geradas antes do mapeamento, Figura 5 (a), e após o mapeamento, Figura 5 (b), representadas pela linha tracejada vermelha.

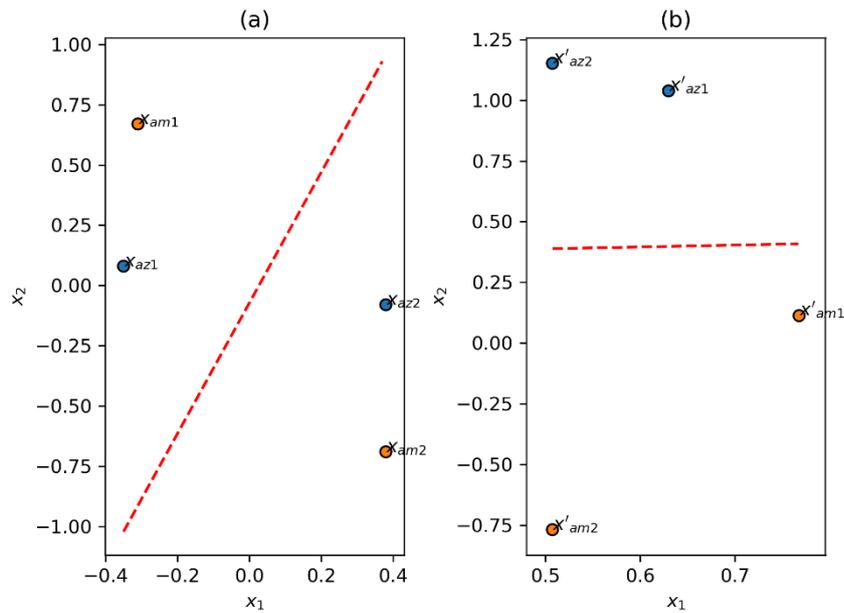


Figura 5. Exemplo de mapeamento para instâncias específicas. Superfícies: (a) não-linearmente separável e (b) linearmente separável após mapeamento utilizando a Golden wavelet da equação (40). As linhas tracejadas vermelhas representam as fronteiras de decisão. Fonte: os autores.

Um segundo problema de classes não-linearmente separáveis está ilustrado na Figura 6. Nesse, embora as instâncias das classes estejam distribuídas em núcleos pouco dispersos, há um núcleo da classe amarela entre dois núcleos da classe azul, o que impede que uma reta seja aprendida para separação das classes. A linha tracejada exibida na Figura 6, separa as classes com um erro de 287 instâncias de um total de 700, correspondendo a 41% de erro. Uma fronteira de decisão adequada neste exemplo seria uma fronteira circular em torno das instâncias da classe amarela. Assim, as instâncias dentro da fronteira pertenceriam a classe amarela e aquelas que estivessem fora a classe azul.

Para resolver este problema pela Regressão Logística, novamente o mapeamento baseado na Golden wavelet, equação (38), é empregado, mas desta vez apenas os resultados para as escalas 1 e 3 são exibidos, no Quadros 2 e Tabela 2.

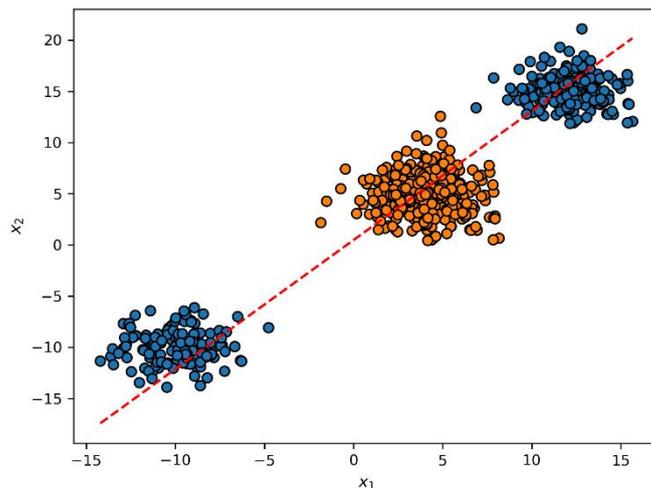
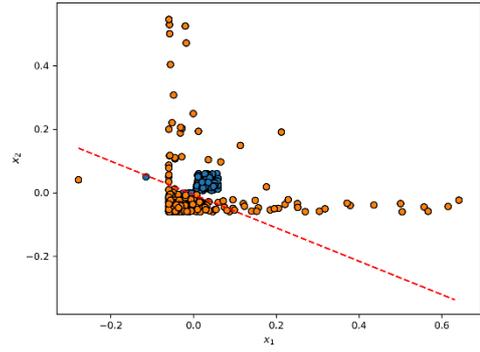
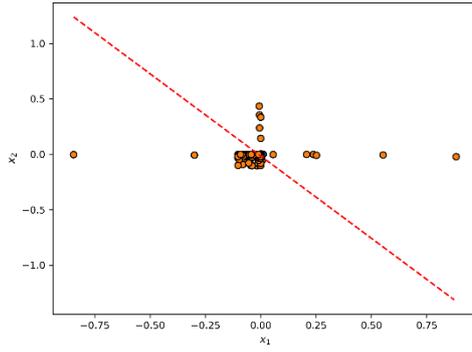


Figura 6. Um problema não-linearmente separável. A classe azul se distribui em dois núcleos, tornando impossível encontrar uma reta que separe ela da classe amarela. Fonte: os autores.

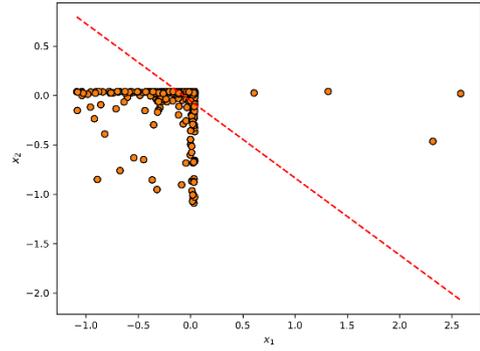
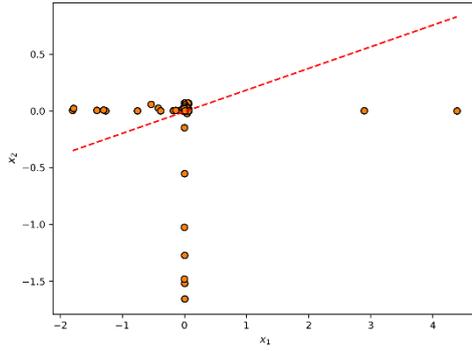
Quadro 2. Fronteiras de decisão (linha vermelha tracejada) considerando funções Golden wavelets com diferentes momentos nulos (G) e em distintas escalas (s), para o problema ilustrado na Figura 6. Fonte: os autores.

G	s	
	1	3
1		
2		

3



4



5

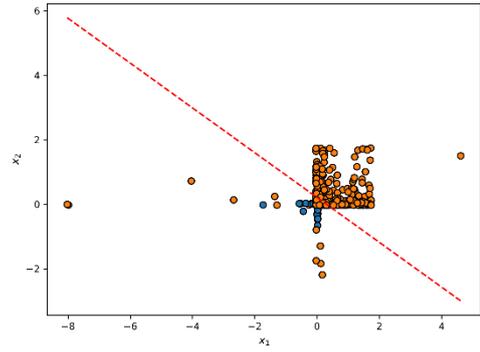
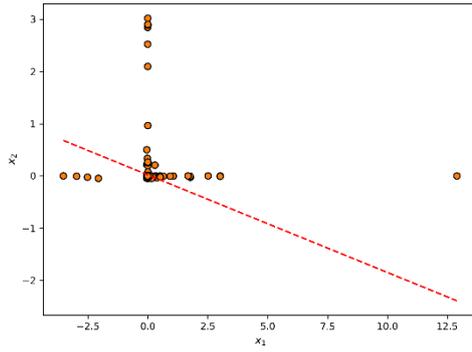


Tabela 2. Erros de estimação considerando funções Golden wavelets com diferentes momentos nulos (G) e em distintas escalas (s). Fonte: os autores.

G	s	
	1	3
1	115	2
2	143	75
3	135	131
4	312	182
5	317	131

Pelos resultados apresentados, é claro que utilizar a Golden wavelet $\psi_{3,1}$, que resulta em um erro de apenas duas instâncias, é a melhor opção. Na imagem correspondente no Quadro 2 vê-se como as classes foram bem separadas ao tomar esse mapeamento.

Nos experimentos anteriores foram fixados apenas alguns valores para a escala e momentos nulos, no entanto, pode-se variar tais parâmetros conforme se queira, tendo em mente que a escala s é um número real positivo e a quantidade de momentos nulos é um inteiro positivo. Além disso, a escala dilata ou comprime a wavelet, tendo conseqüentemente um resultado de dilatação ou contração do espalhamento das instâncias, enquanto que, a quantidade de momentos nulos está relacionada a forma de onda da função, pois esta varia conforme a ordem das derivadas. Pois isso, o resultado no espaço mapeado é muito mais distinto ao variar o parâmetro G , do que ao variar a escala s .

A solução de mapeamento apresentada baseada nas Golden wavelets pode ser entendida como um método de aprendizado de máquina não-supervisionado, analogamente ao método Análise de Componentes Principais, pois tais métodos representam transformações do espaço de entrada de dados.

Devido a essa característica, o método de mapeamento proposto deve empregar um esquema de aprendizagem seletivo, isto é, sobre o conjunto de treinamento testa-se as funções wavelets para transformação do espaço de atributos não-linearmente separável para um espaço linearmente separável. Após, na fase de testes, emprega-se aquela função wavelet que gerou o menor erro.

Além disso, a escolha da função wavelet também é uma tarefa importante, pois pode-se constatar pelos experimentos que a forma de onda da função afeta o resultado do mapeamento. As funções Golden wavelets são adequadas para este propósito, pois muitas formas de ondas podem ser geradas bastando tomar distintas razões entre polinômios com coeficientes de Fibonacci (Gossler, 2016).

Geralmente, espera-se que a performance não seja a mesma sobre os conjuntos de indução e teste, mas tal resultado é comum mesmo nos esquemas de aprendizagem supervisionada.

Em síntese, este ensaio apresentou uma proposta de utilização das funções Golden wavelets para mapeamento do espaço de atributos não-linearmente separáveis para espaços linearmente separáveis, e algumas aplicações utilizando o método Regressão Logística para obtenção das fronteiras de decisão. A abordagem proposta pode, evidentemente, ser aplicada para qualquer algoritmo que gere fronteiras de decisão lineares.

Em trabalhos futuros pretende-se investigar, em problemas reais, como o mapeamento proposto pode melhorar os resultados de performance, e quais das diversas funções Golden wavelets são mais adequadas a resolução destes problemas.

REFERÊNCIAS BIBLIOGRÁFICAS

- Basu S, Mitra S, S N (2020). Deep Learning for Screening COVID-19 using Chest X-Ray Images. *Cold Spring Harbor Laboratory Press*, DOI: 10.1101/2020.05.04.20090423.
- Bishop CM (2006). *Pattern Recognition and Machine Learning*. New York: Springer. 758p.
- Daubechies I (1992). Ten Lectures on Wavelets. Philadelphia: Society for Industrial and Applied Mathematics, 61, 343p.
- Fávero LP, Belfiore P, Silva FL da, Chan BL (2009). *Análise de Dados: Modelagem Multivariada para Tomada de Decisão*. São Paulo: Elsevier, Campus. 544p.
- Fisher M, Lenz HJ (1996). *Learning from Data: Artificial Intelligence and Statistics V*. Springer: New York. 443p.
- Gossler FE, Oliveira BR, Duarte MAQ, Lamblém RL, Villarreal F (2018). A comparative study between mexican and golden hat wavelets. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, 6(2).
- Gossler FE, Oliveira BR, Duarte MAQ, Lamblém RL, Villarreal F (2016). A wavelet generated from Fibonacci-coefficient polynomials and its application in cardiac arrhythmia classification. Proceedings of XIX ENMC - National Meeting on Computational Modeling and VII ECTM - Meeting on Materials Science and Technology, October, João Pessoa.
- Gossler FE (2016). *Wavelets e polinômios com coeficientes de Fibonacci*. Dissertação de Mestrado. Programa de pós-graduação em Engenharia Elétrica (UNESP-Ilha Solteira). Disponível em: <<https://repositorio.unesp.br/handle/11449/148776>>.
- Haykin S (2007). *Redes Neurais: princípios e práticas*. 2 ed. Bookman: São Paulo. 900p.
- Kaiser G (1994). *A Friendly Guide to Wavelets*. Boston: Birkhäuser. 300p.
- Kubat M (2015). *An Introduction to Machine Learning*. Springer: New York. 296p.
- Lamblém RL, Gosler FE, Duarte MAQ, Villarreal F (2016). *Polinômios com coeficientes de Fibonacci e wavelets*. CNMAC, 5(1): 1-6.

- Loesch C., Hoeltgebaum M (2012). *Métodos estatísticos multivariados*. São Paulo: Editora Saraiva. 267p.
- Mallat S (2009). *A Wavelet Tour of Signal Processing: The Sparse Way*. 3 ed. New York: Elsevier. 805p.
- Oliveira BR de, Duarte MAQ, Abreu CCE de, Vieira Filho J (2018). *A wavelet-based method for power-line interference removal in ECG signals*. Res. Biomed. Eng, 34(1): 73-86. DOI: 10.1590/2446-4740.01817.
- Oppenheim AV, Willisky AS, Nawab SH (1997). *Signals and Systems*. 2 ed. New Jersey: Prentice Hall. 957p.
- Schwarting W, Alonso-Mora J, Rus D (2018). *Planning and Decision-Making for Autonomous Vehicles*. Annual Review of Control, Robotics, and Autonomous Systems, (1):187-210. DOI: 10.1146/annurev-control-060117-105157.
- Singh P (2019). *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*. Apress: New York.
- Vetterli M, Kovacevic J (2007). *Wavelets and Subband Coding*. New Jersey: Prentice Hall PTR.

Classificação de Cenas Acústicas Utilizando a Transformada Wavelet Contínua e Aprendizagem de Máquina

Recebido em: 18/09/2020

Aceito em: 01/10/2020

 10.46420/9786588319260cap6

Caio Cesar Enside de Abreu^{1*} 

Edilaine Gomes da Silva² 

Lucas Angelo Mattesco³ 

Bruno Rodrigues de Oliveira⁴ 

Carlos Alex Sander J. Gulo⁵ 

Lucas Kriesel Sperotto⁶

INTRODUÇÃO

De acordo com Halliday et al. (2002), o som é a propagação de uma frente de compressão mecânica ou onda mecânica; o som é uma onda longitudinal, que se propaga de forma circuncêntrica, apenas em meios materiais, como os meios sólidos, líquidos ou gasosos, não podendo propagar-se no vácuo. Os sons audíveis pelo ouvido humano têm uma frequência entre 20Hz e 20.000Hz, portanto, os sons que estão abaixo ou acima desta faixa de frequência são denominados, respectivamente, de infrassom e ultrassom.

O som que percebemos ao caminhar por um parque, uma rua movimentada ou ao andar de ônibus, por exemplo, contém várias informações sobre o ambiente e os eventos que estão ocorrendo naquele ambiente. Os seres humanos utilizam estas informações para compreenderem o ambiente onde estão, como por exemplo, uma pessoa que está se preparando para sair de casa, ouvindo o que está

^{1,2,3} Universidade do Estado de Mato Grosso – UNEMAT, Rua Santa Rita, 128, Alto Araguaia-MT. CEP 78780-000.

E-mail: ¹caioenside@unemat.br, ²edilaine.gomes@unemat.br, ³lucas.mattesco@unemat.br

⁴ Pantanal Editora. E-mail: bruno@editorapantanal.com.br

^{5,6} Grupo de Pesquisa PIXEL - Ciência da Computação – UNEMAT. Rua Santa Rita, 128, Centro - Alto Araguaia-MT.

CEP 78.780-000. E-mail: ⁵sander@unemat.br, ⁶sperotto@unemat.br

* Autor correspondente: caioenside@unemat.br

acontecendo do lado de fora pode colocar um casaco (se ouvir sons de ventos), utilizar um guarda-chuva (sons da chuva) ou também pode deduzir como está a situação do trânsito em sua rua. A combinação de vários tipos de sons em um determinado ambiente caracteriza uma cena acústica (Barchiesi et al., 2015; Stowell et al., 2015).

Segundo Chachada e Kuo (2013), a análise de áudio realizada de forma automática tem sido empregada em diversas áreas, tais como: robótica, inteligência artificial, dispositivos móveis, tecnologia assistiva, dentre outras. O estudo e o aperfeiçoamento de técnicas de análise de áudio podem fornecer, juntamente com sua posterior utilização em dispositivos tecnológicos, o desenvolvimento de dispositivos inteligentes e a criação de sistemas de tomada de decisão cada vez mais precisos. Dessa forma, a análise não somente dos sons individuais, mas da cena acústica que eles formam, pode trazer melhorias antes impensáveis para o ser humano.

Segundo Virtanen et al. (2018), a habilidade da audição é vista como algo importante para o ser humano pois, além de fazer parte do sistema sensorial, é por meio dela que se tornou possível o desenvolvimento da linguagem falada e escrita que conhecemos, sem as quais não seria possível ter evoluído tanto em termos tecnológicos. O desafio enfrentado na área de Classificação de Cenas Acústicas (CCA) é o de construir algoritmos e métodos computacionais, capazes de reconhecer os diversos sons pertencentes à um determinado ambiente ou localização.

O desenvolvimento de métodos de processamento de sinais capazes de extrair automaticamente informações do som, apresentam grande potencial em diversas aplicações do dia a dia, dentre aquelas mencionadas anteriormente. Porém, ainda é um grande desafio encontrar formas eficazes de realizar o reconhecimento de cenas acústicas, seja pela busca de novos descritores capazes de extrair informações relevantes do sinal de áudio, ou pela melhoria constante nos algoritmos de aprendizagem de máquina. Isto se deve ao fato de que em qualquer ambiente real, o arquivo de áudio capturado contém sons gerados por fontes e intensidades aleatórias, e que normalmente são sobrepostos de formas distintas.

Neste capítulo, serão apresentados os principais métodos de CCA e os principais atributos utilizados para esta classificação, assim como exemplos de aplicações. Uma investigação adicional centrada na utilização da Transformada Wavelet Contínua (TWC) será apresentada. Para isso, simulações serão conduzidas utilizando um algoritmo padrão de *Machine Learning*, o tradicional Máquina de Vetores de Suporte (SVM), em conjunto com um critério de decisão tradicional, baseado no voto majoritário. Uma base de dados será fixada e apenas os descritores de entrada para o classificador serão alternados. O objetivo é fazer uma comparação justa dos descritores avaliados. Por fim, pretende-se discutir se a TWC compete, em relação a performance, com os descritores tradicionalmente empregado nesse problema, denominados de Coeficientes Cepstrais de Frequência Mel (MFCCs) e Mel-

Espectrograma, que são amplamente utilizados para análise de áudio e voz (Barchiesi et al., 2015; Mesaros et al., 2018).

CLASSIFICAÇÃO AUTOMÁTICA DE CENAS ACÚSTICAS

A cena acústica, de acordo com Lignelli (2011), é o som que é formado quando vários sons, de diversas fontes, são combinados, formando o áudio que é ouvido num ambiente como um todo. Por exemplo, a cena acústica de uma grande avenida pode ser formada pelo som emitido por pessoas conversando, vendedores gritando, carros passando e/ou buzinando, sirenes de ambulâncias, entre outros. Assim, o ser humano consegue compreender qual é o ambiente, somente por ouvir a cena acústica.

Em Nguyen e Pernkopf (2018), a CCA é definida como o reconhecimento de um determinado ambiente, tomando como base uma gravação de áudio, e assumindo que a cena acústica é uma representação efetiva de um local ou situação. O objetivo da classificação automática de cenas acústicas é identificar o ambiente onde em que o um arquivo de áudio foi gravado, utilizando como base classes predefinidas de áudio que caracterizam o cenário original, por exemplo, “parque”, “rua de pedestres”, “estação de metrô”; ou para indicar que a cena é de um ambiente diferente ou desconhecido (Mesaros et al., 2018). A Figura 1 apresenta a ideia geral do problema.

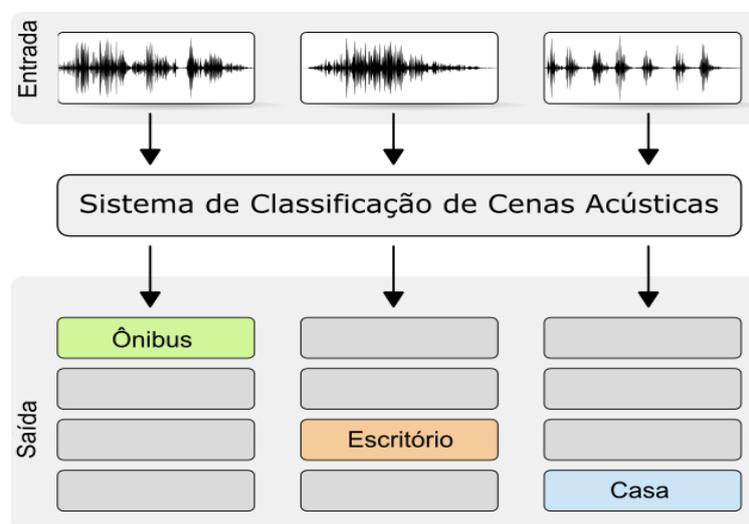


Figura 1. Esquema da classificação de cenas acústicas. Fonte: Adaptado de <http://dcase.community/challenge2019/task-acoustic-scene-classification>.

ESTADO DA ARTE

De acordo com Barchiesi et al. (2015), o primeiro método computacional que aborda especificamente o tema CCA foi publicado em 1997, em Sawhney e Maes (1997). Apesar de já haver um tempo considerável desde então, a área de pesquisa só ganhou força e notoriedade a partir de 2013, com o surgimento de desafios internacionais consecutivos. Denominado *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events* (DCASE), este desafio internacional, que é apoiado pela *IEEE Signal Processing Society*, proporcionou um avanço significativo em um curto espaço de tempo, e tem conseguido agregar pesquisadores ao redor do mundo, entusiastas e representantes da indústria. A primeira edição do desafio ocorreu em 2013. Os organizadores desenvolveram uma base de dados sobre a qual os participantes desenvolveram e testaram seus algoritmos. Os participantes foram encorajados a submeter um artigo acompanhando o código fonte dos sistemas propostos, para que todos os detalhes das propostas pudessem ficar disponíveis para a comunidade. Após o recebimento das propostas, os organizadores testaram todos os sistemas sobre uma base de dados oculta.

Os resultados, juntamente com todo o material, estão disponíveis no *website* do evento. Além disso, os resultados, incluindo o estado da arte da área até aquela data, foram publicados em Barchiesi et al. (2015) e Stowell et al. (2015). A segunda edição do DCASE ocorreu em 2016 e, a partir daquele ano, o desafio passou a ser acompanhado por um *workshop*. Todas as edições do DCASE podem ser encontradas no endereço <http://dcase.community/events>.

A fim de caracterizar todos os sistemas submetidos às últimas edições do DCASE, a Tabela 1 apresenta uma visão geral dos atributos e dos algoritmos de aprendizagem de máquina utilizados.

Tabela 1. Visão geral dos atributos e algoritmos de aprendizagem de máquina utilizados em todos os desafios DCASE realizados entre 2013 e 2019, contabilizando 324 sistemas verificados.

Atributos	Edição DCASE				
	2013	2016	2017	2018	2019
Espalhamento espectral profundo	-	-	-	4	3
Filtro Gammatone + log-mel-espectrograma + transformada Q-Constante (CQT)	-	-	-	-	2
Log-mel-espectrograma + outros	-	-	25	32	72
Log-mel-espectrograma	-	-	9	5	-
MFCCs + outros	7	17	13	5	5
Espectro de potência ponderado perceptualmente	-	-	-	3	4
Forma de onda bruta	-	-	-	3	2
Forma de onda bruta + log-mel-espectrograma	-	-	-	-	7
Forma de onda bruta + espectrograma + i-vetor	-	-	-	4	-
Espectrograma + outros	-	5	11	-	1
Vários	8	27	38	4	2
Wavelet + outros	3	-	2	-	1
Algoritmos de aprendizagem de máquina					
Rede Neural Convolutacional	-	10	49	32	76
Rede Neural Profunda	-	7	4	-	1

<i>Ensemble</i>	-	4	2	4	3
SVM e Modelo de Mistura de Gaussianas	1	3	4	-	5
Rede Neural de Regressão Generalizada	-	1	2	-	-
I-vetor + outros	1	2	2	1	-
Perceptron multicamadas (MLP) / MLP + outros	-	-	11	1	-
Fatoração de Matrizes Não-Negativas (NMF) /NMF + outros	-	1	2	-	-
Máquina de Vetores de Suporte	11	10	10	1	1

Nota-se, a partir da Tabela 1, que as pesquisas recentes e que constituem o estado-da-arte têm focado na utilização de atributos como mel-espectrograma e modelos de *machine learning* com complexidade elevada, tais como Redes Neurais Profundas e Redes Neurais Convolucionais. Em relação à Transformada Wavelet, apenas 6 dos 324 sistemas submetidos em todos os eventos utilizaram esse método alguma versão dessa ferramenta. Em nenhum destes sistemas foi utilizada a Transformada Wavelet Contínua (TWC) no processo de extração de atributos. Este fato reforça a necessidade de investigação a respeito de tais atributos.

Analisando a Tabela 1 de uma forma mais minuciosa, pode-se observar também dois momentos importantes da CCA: 1º) utilização predominante de algoritmos clássicos de aprendizagem de máquina, principalmente o classificador SVM juntamente com os atributos MFCCs, até a edição de 2016; 2º) popularização de abordagens de Aprendizagem Profunda, na língua inglesa *Deep Learning*, caracterizada na Tabela 1 pela utilização de Redes Neurais Convolucionais tendo como entrada o mel-espectrograma.

A fim de obter o mel-espectrograma de um arquivo de áudio, primeiro obtém-se o espectrograma do sinal, que consiste em uma matriz cujas colunas armazenam o espectro de amplitude ou de potência do conteúdo de cada janela, e, posteriormente, converte-se o eixo de frequência para a escala mel (Sakashita e Aono, 2018). É comum, também, converter o mel-espectrograma para uma escala logarítmica, denotado por log-mel-espectrograma. A escala Mel está relacionada à capacidade do ouvido humano em distinguir sons com frequências parecidas, uma vez que o ouvido humano não percebe as frequências em uma escala linear, como fornecida pelo espectrograma. Já os MFCCs são obtidos a partir do cálculo da Transformada de Cosseno Discreta sobre o mel-espectrograma (Barchiesi et al. 2015).

MFCCs constituem uma representação mais compressiva do sinal de áudio, quando comparado ao mel-espectrograma e são frequentemente utilizados com modelos lineares, como Modelos de Misturas Gaussianas e SVM. Para utilização com algoritmos de *Deep Learning*, que em geral utilizam modelos mais complexos e necessitam de uma quantidade maior de dados para treinamento, o mel-espectrograma fornece melhores resultados (Sakashita e Aono, 2018).

Aplicações

Nesta seção, alguns exemplos de aplicações que a CCA pode possibilitar são apresentados. De forma geral, um sistema de CCA possibilita a aplicação de computação sensível ao contexto, dispositivos portáteis inteligentes, sistemas de navegação robótica, gerenciamento de arquivos de áudio, sistemas de segurança, entre outras (Barchiesi et al., 2015).

De maneira mais específica, pode-se destacar a possibilidade de desenvolvimento de aplicativos que permitam que *smartphones* mudem seu modo de operação (mudando para o modo silencioso, por exemplo) dependendo da localização do usuário; cadeiras de rodas inteligentes que ajustam seu funcionamento baseado no reconhecimento de ambientes abertos ou fechados; ajuste automático de implantes cocleares (IC) e aparelhos auditivos baseado nas características acústicas do ambiente em que o usuário se encontra (Hansen et al., 2019); em robótica, a integração da CCA com a visão computacional pode fornecer informações adicionais, principalmente onde informações visuais não são suficientes (Chu et al., 2006); em aplicações de segurança, a classificação de eventos sonoros suspeitos, como por exemplo, detecção de gritos e batidas pode integrar o sistema (Radhakrishnan et al., 2005).

MATERIAL E MÉTODOS

Transformada Wavelet Contínua

A Transformada Wavelet Contínua (TWC) é uma transformação de um espaço real unidimensional para um espaço real bidimensional. Dado um sinal $f(t) \in L^2(\mathbb{R})$, onde $L^2(\mathbb{R})$ é o espaço real das funções de quadrado integrável, a TWC é o mapeamento $Wf(s, u): L^2(\mathbb{R}) \mapsto L^2(\mathbb{R}^+ \times \mathbb{R})$ (Daubechies, 1992; Mallat, 2009).

$$Wf(s, u) = \int_{-\infty}^{\infty} f(t) \psi_{s,u}^*(t) dt, \quad (1)$$

onde $s \in \mathbb{R}^+$ e $u \in \mathbb{R}$ são os parâmetros de escala e translação, respectivamente, e $*$ denota o conjugado complexo da família de funções *wavelets* $\psi_{s,u}$, dada por

$$\psi_{s,u}(t) = s^{-0.5} \psi\left(\frac{t-u}{s}\right). \quad (2)$$

A norma Euclidiana de tais funções devem ser unitária, ou seja, $\|\psi_{s,u}\| = 1$. Além disso, uma função *wavelet* deve satisfazer a condição de admissibilidade $C_\psi = 2\pi \int_{-\infty}^{\infty} |\hat{\psi}(\omega)|^2 / \omega d\omega < \infty$, onde $\hat{\psi}(\omega)$ é a Transformada de Fourier de $\psi(t)$.

Algumas funções *wavelets* têm suporte compacto, como a *wavelet* de Haar e outras. Apesar de algumas nunca convergirem para zero, têm um suporte que pode ser considerado compacto, pois o decaimento é rápido, por exemplo, a *wavelet* Ricker. A condição de admissibilidade acima mencionada, implica que as funções *wavelets* têm média nula, logo, as áreas abaixo e acima do eixo temporal se anulam,

$\int_{-\infty}^{\infty} \psi(t)dt = 0$, e isto caracteriza sua forma em “ondinhas”, que não oscila indefinidamente como uma senóide, mas apenas no seu intervalo suporte.

Dado seu caráter ondulatório e seu suporte compacto, as funções *wavelets* podem ser vistas como janelas de análise. Neste sentido, a TWC consiste na obtenção da similaridade entre o sinal $f(t)$ e a função analisadora *wavelet* $\psi(t)$, que é transladada sobre o sinal. Considerando uma *wavelet* com suporte igual a M , ao transladar essa função pelo sinal, alterando o parâmetro u , e realizar a multiplicação da função transladada por segmento de comprimento M do sinal, aqueles segmentos que possuem maior similaridade com a forma de onda da *wavelet* resultarão em coeficientes *wavelets* com maior amplitude. Ao alterar a escala s , a função wavelet é comprimida quando $0 < s < 1$ e dilatada quando $s > 1$, implicando que seu intervalo suporte também é modificado e, conseqüentemente, a sua forma de onda, acarretando na detecção de outras similaridades entre o sinal $f(t)$ e a função *wavelet* escalonada.

A Transformada de Fourier das funções *wavelets* mostra que as *wavelets* têm características de filtros passa-banda, cuja banda de passagem depende da escala escolhida. Assim, se no domínio de Fourier uma função *wavelet* é nula fora intervalo $]\alpha, \beta[$, ou seja, somente as frequências dentro deste intervalo passam pelo filtro *wavelet*, então, ao alterar o parâmetro de escala s , este intervalo é modificado para $]\frac{\alpha}{s}, \frac{\beta}{s}[$. Isso implica que, quanto maior a escala, menor será a resolução em frequência.

Uma das ferramentas mais empregadas da TWC é o Escalograma, que é obtido tomando o quadrado do valor absoluto dos coeficientes *wavelets*, isto é $|Wf(s, u)|^2$. Ele representa o espectro de energia dos coeficientes *wavelets* em uma certa escala s em função do tempo u , utilizando intensidade de cores para designar a energia de cada coeficiente. A fim de entender a distribuição de energia por escala, recorre-se ao Espectro Wavelet Global (*Global Wavelet Spectrum* - GWS) que é calculado como a média temporal dos coeficientes para uma escala selecionada, de acordo com a equação (4), onde consideramos uma versão discreta da TWC, dada pela equação (3). Para um sinal $g[n]$ de comprimento N , obtido a partir da amostragem de $f(t)$, sendo a a escala e b o parâmetro de translação discretizados, temos (Alessio, 2016):

$$Wg(a, b) \approx \sum_{n=0}^{N-1} g[n] \psi^* \left[\frac{n-b}{a} \right], \quad (3)$$

$$GWS(a) = \frac{1}{N} \sum_{n=0}^{N-1} |Wg(a, b)|^2. \quad (4)$$

Na equação (4), o GWS pode ser calculado utilizando a potência, da forma como está definido, ou utilizando a amplitude. Em outras palavras, pode-se utilizar apenas o valor absoluto dos coeficientes *wavelets*. Neste trabalho, após a realização de algumas simulações, optou-se por utilizar apenas a magnitude.

Implementação do sistema

O sistema principal implementado para avaliar os atributos é baseado no voto majoritário, que pode ser descrito da seguinte forma: para cada sinal de áudio a ser classificado, realiza-se um janelamento de Hanning com 1024 pontos; para cada janela, os descritores são extraídos, normalizados pela média e variância, e então utilizados para alimentar um SVM com *kernel* linear; para cada janela, uma classificação é emitida. A classificação final do arquivo é emitida observando a classe que recebeu a maior quantidade de votos pelo classificador. Vale destacar que o sistema implementado não foi desenvolvido para competir com os sistemas vencedores do evento, pelo contrário, o intuito é utilizar uma abordagem padrão de aprendizagem de máquina, que favoreça e simplifique a comparação dos atributos avaliados.

A base de dados escolhida é a fornecida para o DCASE-2013. Esta base de dados contém 100 arquivos de áudio divididos em dez classes, no formato .wav, com taxa de amostragem de 44.100 Hz, 16 bits, estéreo. Bases de dados fornecidas para edições mais recentes do DCASE contém milhares de arquivos de áudios, sendo desenvolvidas principalmente para avaliação de algoritmos de *Deep Learning*. Utilizar tais bases de dados fogem ao escopo do presente trabalho.

Com relação aos protocolos da simulação, a validação cruzada com 5 pastas é utilizada a fim de que todos os sinais da base de dados sejam classificados. Para cada pasta, oito arquivos de áudio por classe são utilizados para treino e dois arquivos para teste. Para implementação da Transformada Wavelet Contínua, a biblioteca para Python denominada PyWavelets (Gregory et al., 2019) foi utilizada. A biblioteca librosa (Mcfee et al., 2015) foi utilizada para extração dos atributos MFCCs e do mel-espectrograma.

A Tabela 2 apresenta algumas variações do sistema principal, destacando suas principais configurações, escolhidas para fornecer as análises pretendidas. Resultados das simulações com diferentes parâmetros dos algoritmos são apresentados na próxima seção.

Tabela 2. Características dos sistemas implementados. Fonte: os autores.

Sistema	Atributos extraídos	Família Wavelet	Pós-processamento dos atributos
1	MFCCs	-	Filtragem cepstral
2	GWS	Gauss de ordem 8	Filtragem recursiva
3	Escalograma + Hog (9 orientações)	Gauss de ordem 8	-
4	Mel-espectrograma	-	-
5	Log-mel-espectrograma	-	-

A Filtragem cepstral, mencionada na Tabela 2, é realizada da seguinte forma:

$$S[n, m] = S[n, m] \left(1 + \text{sen} \left(\frac{\pi(n+1)}{\beta} \right) \right) \frac{\beta}{2}, \tag{4}$$

onde S é a matriz que armazena a sequência de MFCCs em suas colunas, $n = 1, 2, \dots, N_{MFCCs}$ e $m = 1, 2, \dots, M$, com N_{MFCCs} e M sendo, respectivamente, o número de MFCCs extraídos e a quantidade de janelas analisadas. Utilizou-se $\beta = 26$. Já a filtragem recursiva utilizada para suavizar a sequência de GWS, extraída de cada janela, é realizada como segue:

$$GWS_m = \alpha GWS_{m-1} + (1 - \alpha) GWS_m, \tag{5}$$

α é o parâmetro que controla o quão suave é a transição do vetor de atributos entre janelas consecutivas. Definiu-se empiricamente $\alpha = 0,6$.

Os Sistemas 3 e 4 utilizam o descritor HOG, sigla padrão para denotar Histograma de Gradientes Orientados, calculado sobre o escalograma gerado pela TWC. O descritor HOG é muito utilizado para localização de objetos em uma imagem digital, pela sua capacidade em detectar bordas e contornos. Quando aplicado sobre o escalograma, extrai informação relacionada a distribuição da energia do sinal pelas escalas (Mattesco e Abreu, 2019).

Na Tabela 3 constam mais detalhes sobre a análise *wavelet* realizada pela CWT sobre a base de dados DCASE-2013, sendo apresentadas a sequência de escalas utilizadas para a decomposição e as respectivas frequências alcançadas. As frequências correspondentes para cada escala são estimadas levando em consideração a função *wavelet* escolhida e a frequência de amostragem do sinal, de forma teórica, podendo ser utilizada para orientar a escolha das escalas (Oliveira et al., 2018).

Tabela 3. Escalas utilizadas para o cálculo da CWT e respectivas frequências correspondentes para a função *wavelet* Gaussiana de ordem 8 e frequência de amostragem do sinal 44.100 Hz. Fonte: os autores.

Escalas	1,2 1,5 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 82,5 83, 83,5 84, 84,5 85, 85,5 86, 86,5 87, 87,5 88, 88,5 89, 89,5 90, 90,5 91, 91,5 92, 92,5 93, 93,5 94, 94,5 95, 95,5 96, 96,5 97, 97,5 98, 98,5 99, 99,5 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510,
Frequências (Hz)	22050, 17640, 13230, 6615, 4410, 3307,5 2646, 2205, 1890, 1653,75 1470, 1323, 1202,73 1102,5 1017,69 945, 882, 826,88 778,24 735, 696,32 661,5 630, 601,36 575,22 551,25 529,2 508,85 490, 472,5 456,21 441, 426,77 413,44 400,91 389,12 378, 367,5 362,47 357,57 352,8 348,16 343,64 339,23 334,94 330,75 326,67 322,68 320,73 318,8

	316,89	315,	313,14	311,29	309,47	307,67	305,9	304,14	302,4	300,68
	298,98	297,3	295,64	294,	292,38	290,77	289,18	287,61	286,05	284,52
	282,99	281,49	280,	278,53	277,07	275,62	274,2	272,78	271,38	270,
	268,63	267,27	265,93	264,6	240,55	220,5	203,54	189,	176,4	165,38
	155,65	147,	139,26	132,3	126,	120,27	115,04	110,25	105,84	101,77
	98,	94,5	91,24	88,2	85,35	82,69	80,18	77,82	75,6	73,5
	69,63	67,85	66,15	64,54	63,	61,53	60,14	58,8	57,52	56,3
	54,	52,92	51,88							

Complementarmente à Tabela 3, a Figura 2 apresenta um gráfico com a sequência de escalas escolhidas no eixo das abscissas e as respectivas frequências no eixo das ordenadas.

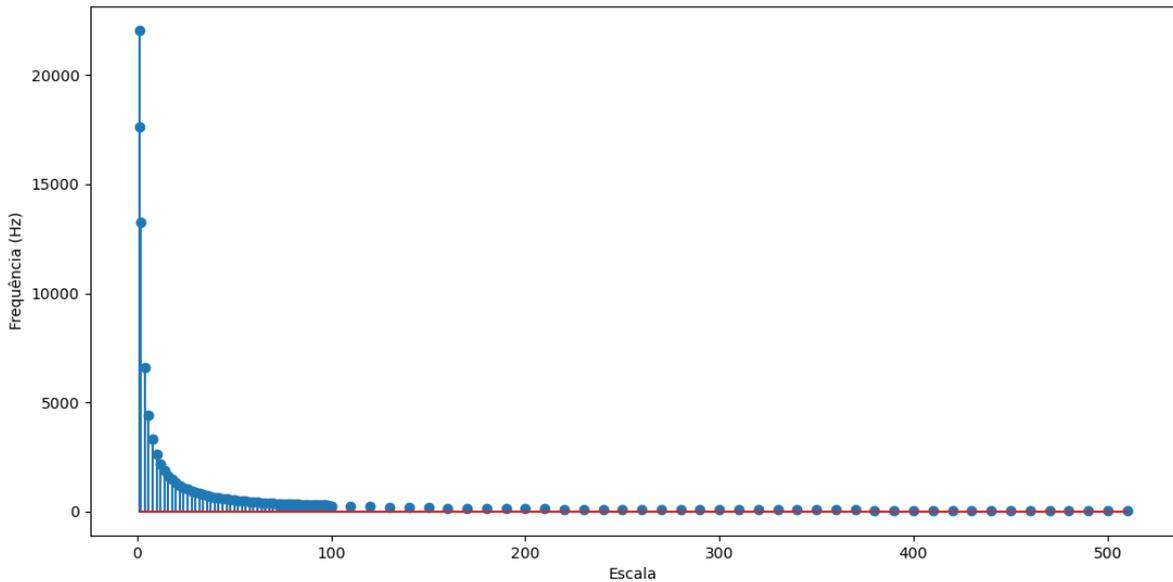


Figura 2. Escalas utilizadas para o cálculo da CWT e respectivas frequências alcançadas. Fonte: os autores.

Note na Tabela 2 que a escala 1,2 foi escolhida para que se contemple a maior frequência do sinal, que neste caso é de 22050 Hz, conforme o Teorema de Nyquist. As demais escalas foram escolhidas de forma que se tenha uma distribuição suave das frequências analisadas e uma maior resolução em baixas frequências, conforme pode ser visto na Figura 2.

A Figura 3 apresenta a sequência de GWS obtida durante a análise de 5 segundos de um arquivo de áudio da classe ônibus. O GWS obtido em cada janela foi armazenado em uma coluna da matriz de saída. Para fins de visualização, a matriz obtida foi plotada como imagem.

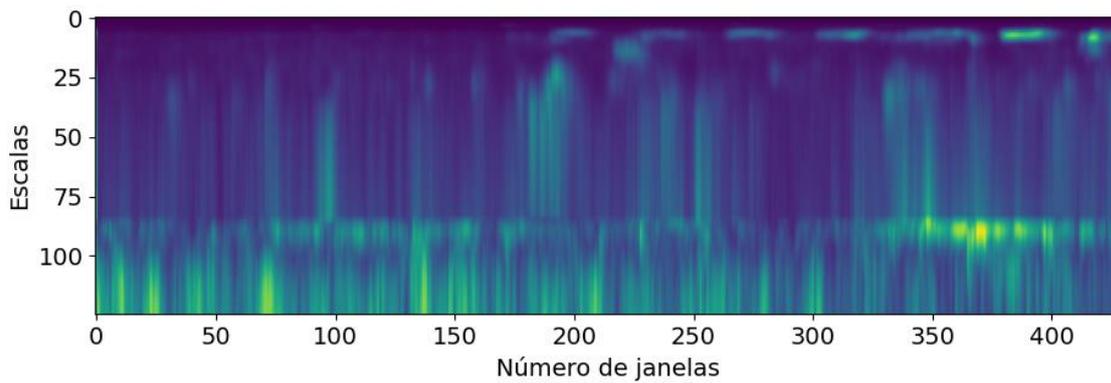


Figura 3. Sequência de GWS obtidos após análise de uma porção de cinco segundos de um arquivo de áudio da classe ônibus. Fonte: os autores.

RESULTADOS E DISCUSSÃO

Como mencionado anteriormente, o principal objetivo deste trabalho é apresentar uma comparação justa entre os descritores fornecidos pela TWC, os MFCCs e o mel-espectrograma. Estes dois últimos são considerados descritores padrão para análise de áudio. Dessa forma, a análise apresentada se limita a dimensão dos descritores e não da abordagem de aprendizagem em si. Por este motivo, uma abordagem tradicional que utiliza um classificador padrão foi implementada.

É importante destacar também que os resultados apresentados nesta seção são naturalmente inferiores aos resultados encontrados pelos competidores dos eventos DCASE, visto que, de forma geral, os competidores buscam a construção de uma abordagem de aprendizagem complexa e inovadora, normalmente utilizando arquiteturas de redes neurais avançadas em conjunto com os descritores padrão apresentados anteriormente.

Tabela 3. Resultados das simulações sobre a base de dados DCASE-2013.

Sistema	Dimensão do vetor de atributos	Porção do sinal utilizado (s)	Acurácia (%)
1	19	0,5	37
1	19	5	49
2	125	0,5	45
2	125	5	47
3	4032	0,5	41
3	4032	5	43
4	32	0,5	39
4	32	5	41
4	128	0,5	37
4	128	5	37
5	32	0,5	36
5	32	5	47

Na Tabela 3, os resultados apresentados na coluna Acurácia representam a média das acurácias obtidas em cada uma das cinco pastas da validação cruzada. Os três melhores resultados estão destacados em cinza escuro e cinza claro, respectivamente. De maneira geral, nota-se que o Sistema 1, que utiliza os MFCCs como atributos, alcançou o melhor resultado, sendo ligeiramente melhor que o Sistema 2 e 5, que são baseados no atributo *wavelet* GWS e log-mel-espectrograma, respectivamente. Em seguida, aparece o Sistema 3, que utiliza o atributo *wavelet* baseado no descritor HOG. O sistema que apresentou o pior resultado foi o Sistema 4, que é baseado no mel-escalograma.

Com relação ao Sistema 1, utilizou-se 19 MFCCs, já considerando que o coeficiente de ordem zero foi descartado. Este número foi escolhido empiricamente e forneceu os melhores resultados. Nota-se que os resultados são melhores quando se utiliza uma porção maior do sinal, alcançando uma acurácia média de 49%.

De forma similar, o Sistema 2 alcançou seu melhor resultado quando utilizado 5 segundos do sinal, alcançando 47% de acurácia. Este sistema utiliza o GWS calculado utilizando a função *wavelet* de Gauss de ordem 8. Várias simulações foram realizadas alternando a função *wavelet*, porém a Gaussiana de ordem 8 se mostrou mais adequada para analisar os sinais de áudio da base DCASE-2013. Uma explicação sugerida é o fato de sua forma de onda apresentar mais similaridades com o sinal analisado do que as demais funções testadas.

O descritor HOG utilizado no Sistema 3 fornece um vetor de atributos com alta dimensionalidade, quando comparado aos demais descritores. Apesar de ser superado pelos MFCCs, log-mel-espectrograma e pelo GWS, gerou resultados ligeiramente superiores ao mel-espectrograma, alcançando 43% de acurácia.

Por fim, analisando os resultados dos Sistemas 4 e 5, observa-se que utilizar um número elevado de bandas mel não é adequado. A melhor configuração é encontrada utilizando apenas 32 bandas, alcançando 41% e 47% de acurácia, respectivamente. Observando os resultados utilizando o mel-espectrograma e o log-mel-espectrograma, nota-se que para 0,5 segundos de áudio analisado, o primeiro forneceu um melhor desempenho. No entanto, utilizando um período maior do sinal, a versão em escala logarítmica apresentou um desempenho superior. Comparando ambas as versões com os MFCCs, nota-se um melhor desempenho para o último. Este resultado vem de encontro com o discutido na seção anterior: tanto o mel-espectrograma, quanto o log-mel-espectrograma, geram melhores resultados quando utilizado em metodologias de *Deep Learning*.

Do ponto de vista de aplicações de tempo real, é desejável que se utilize um período curto para emitir uma classificação do ambiente acústico. Nesse sentido, o atributo que gerou o melhor desempenho quando utilizado uma porção de 0,5 segundos do arquivo de áudio foi o GWS.

Em termos gerais, pode-se concluir que a TWC é uma ferramenta que pode fornecer resultados semelhantes, ou até mesmo superiores aos descritores popularmente utilizados para a CCA. Este fato indica que trabalhos futuros devem se concentrar na proposição de uma abordagem de aprendizagem mais complexa e profunda, que possa explorar de forma mais adequada e significativa o escalograma.

REFERÊNCIAS BIBLIOGRÁFICAS

- Alessio SM (2016). *Digital Signal Processing and Spectral Analysis for Scientists: Concepts and Applications*. New York: Springer, 909p.
- Chachada S, Kuo J (2014). Environmental Sound Recognition: A Survey. *Apsipa Transactions On Signal And Information Processing*, 3. DOI: 10.1017/ATSIP.2014.12.
- Chu S et al. (2006). Where Am I? Scene Recognition For Mobile Robots Using Audio Features. Toronto, Canada: 885–888. DOI: 10.1109/Icme.2006.262661.
- Daubechies I (1992). *Ten Lectures on Wavelets*. Philadelphia: Society for Industrial and Applied Mathematics, 61, 343p.
- Domingues MO et al (2016). Explorando A Transformada Wavelet Contínua. *Rev. Bras. Ensino Fís.*, São Paulo, 38(3). DOI: 10.1590/1806-9126-RBEF-2016-0019.
- Gregory RL et al. (2019). PyWavelets: A Python package for wavelet analysis. *Journal of Open Source Software*, 4(36). DOI: 10.21105/joss.01237.
- Jhl Hansen et al. (2019). Cci-Mobile: Design And Evaluation Of A Cochlear Implant And Hearing Aid Research Platform For Speech Scientists And Engineers. In: *IEEE Embs International Conference On Biomedical & Health Informatics (BHI)*, Chicago, Il, Eua: 1-4, DOI: 10.1109/BHI.2019.8834652.
- Mattesco L, Abreu CCE (2019). O Uso do Descritor HOG no Problema da Classificação de Cenas Acústicas sobre o Domínio Wavelet. *Anais da X Escola Regional de Informática de Mato Grosso*. SBC. DOI: 10.5753/eri-mt.2019.8618.
- Mallat S (1989). A Theory For Multiresolution Signal Decomposition - The Wavelet Representation. *Ieee On Pattern Analysis And Machine Intelligence*, 11(7): 674-693. DOI: 10.1109/34.192463.
- Mallat S (2009). *A Wavelet Tour Of Signal Processing: The Sparse Way*. 3 ed. New York: Elsevier. DOI: 10.1016/B978-0-12-374370-1.X0001-8.
- Mesaros A, Heittola T, Virtanen T (2018). Acoustic Scene Classification: An Overview Of Dcase 2017 Challenge Entries. *2018 16th International Workshop On Acoustic Signal Enhancement (Iwaenc)*, Tokyo: 411-415. DOI: 10.1109/Iwaenc.2018.8521242.
- McFee B et al (2015). *Librosa: Audio and music signal analysis in python* (2015). In: *Proceedings of the 14th python in science conference*: 18-25. DOI: 10.25080/Majora-7b98e3ed-003.

- Nguyen TKT, Pernkopf F (2018). Acoustic Scene Classification Using A Convolutional Neural Network Ensemble And Nearest Neighbor Filters. In: Workshop On Detection And Classification Of Acoustic Scenes And Events.
- Oliveira BR et al. (2018). A wavelet-based method for power-line interference removal in ECG signals. *Res. Biomed. Eng*, 34(1): 73-86. DOI: 10.1590/2446-4740.01817.
- Radhakrishnan R et al. (2005). Audio Analysis For Surveillance Applications. *IEEE Workshop On Applications Of Signal Processing To Audio And Acoustics*: 158-161. DOI: 10.1109/ASPAA.2005.1540194.
- Sakashita Y, Aono M (2018). Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions. *Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge*.

ÍNDICE REMISSIVO

A

agrupamento, 49, 53, 54
aprendizado de máquina, 8, 25, 28, 29, 59, 60, 61, 80
Aprendizagem de Máquina, 50, 57, 83

C

caso de uso, 41, 42, 44, 46
CCA, 84, 85, 86, 87, 88, 95
cenário, 39, 43, 44, 45, 85
Cenas Acústicas, 83, 84, 95

Ch

checklist, 34, 39, 41

C

Ciência de Dados, 33
classificação, 31, 39, 50, 56, 59, 60, 61, 62, 63, 68, 69, 72, 73, 74, 84, 85, 88, 90, 94
clusterização, 50, 51, 54
coeficiente, 10, 11, 89, 94
comportamento, 8, 9, 22, 26, 30, 32, 37, 56, 70

D

Deep Learning, 31, 32, 59, 81, 87, 90, 94
desenvolvimento de SI, 35, 36, 39
domótica, 25, 26

E

Engenharia de Requisitos, 33, 47, 48
escalograma, 89, 90
espaço de atributos, 59, 60, 70, 73, 74, 76, 80, 81
espectro, 86, 89
espectrograma, 85, 86

F

Farthest First, 49, 50, 53, 54, 56

filtro, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22, 89

G

GWS, 89, 90, 91, 92, 93, 94

I

Inteligência Artificial, 25

M

machine learning, 30, 31, 84, 87
mapeamento, 48, 60, 70, 73, 74, 77, 78, 80, 81, 88
metodologia, 33, 34, 35, 36, 37, 39, 41, 42, 47
MFCCs, 84, 86, 87, 90, 91, 93, 94

O

obesidade, 49, 50, 55, 56, 57, 58

P

polinômio, 10, 11, 16, 20, 21
prototipação, 33, 35, 39, 41, 47

Q

qualidade, 33, 34, 35, 39, 41, 47, 57

R

Redes Neurais
 Artificiais, 25, 26
 LSTM, 26
regressão logística, 59, 62, 63, 70
requisitos, 28, 33, 34, 35, 39, 41, 48

S

Savitzky-Golay, 8, 9, 10, 15, 16, 17, 18, 19, 20, 21, 22, 23
sensores, 26, 27, 29
Série Temporal, 8, 9
Sistema de Informação, 47, 48

Software, 47, 48, 95

superfície de decisão, 62, 77

T

tendência, 8, 9, 10, 14, 15, 16, 17, 18, 19, 21,
22, 57, 58

Transformada Wavelet, 8, 9, 10, 13, 70, 83, 84,
87, 88, 90, 95



Pantanal Editora

Rua Abaete, 83, Sala B, Centro. CEP: 78690-000
Nova Xavantina – Mato Grosso – Brasil
Telefone (66) 99682-4165 (Whatsapp)
<https://www.editorapantanal.com.br>
contato@editorapantanal.com.br

ISBN 978-658831926-0



9

786588

319260