

LILIAN COELHO DE FREITAS  
(ORGANIZADORA)

---

*Collection:*

# APPLIED COMPUTER ENGINEERING 2

---

Atena  
Editora  
Ano 2022

LILIAN COELHO DE FREITAS  
(ORGANIZADORA)

---

*Collection:*

**APPLIED COMPUTER  
ENGINEERING  
2**

**Editora chefe**

Profª Drª Antonella Carvalho de Oliveira

**Editora executiva**

Natalia Oliveira

**Assistente editorial**

Flávia Roberta Barão

**Bibliotecária**

Janaina Ramos

**Projeto gráfico**

Camila Alves de Cremo

Daphynny Pamplona

Gabriel Motomu Teshima

Luiza Alves Batista

Natália Sandrini de Azevedo

**Imagens da capa**

iStock

**Edição de arte**

Luiza Alves Batista

2022 by Atena Editora

Copyright © Atena Editora

Copyright do texto © 2022 Os autores

Copyright da edição © 2022 Atena Editora

Direitos para esta edição cedidos à Atena Editora pelos autores.

Open access publication by Atena Editora



Todo o conteúdo deste livro está licenciado sob uma Licença de Atribuição Creative Commons. Atribuição-Não-Comercial-Não-Derivativos 4.0 Internacional (CC BY-NC-ND 4.0).

O conteúdo dos artigos e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores, inclusive não representam necessariamente a posição oficial da Atena Editora. Permitido o *download* da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Todos os manuscritos foram previamente submetidos à avaliação cega pelos pares, membros do Conselho Editorial desta Editora, tendo sido aprovados para a publicação com base em critérios de neutralidade e imparcialidade acadêmica.

A Atena Editora é comprometida em garantir a integridade editorial em todas as etapas do processo de publicação, evitando plágio, dados ou resultados fraudulentos e impedindo que interesses financeiros comprometam os padrões éticos da publicação. Situações suspeitas de má conduta científica serão investigadas sob o mais alto padrão de rigor acadêmico e ético.

**Conselho Editorial****Ciências Exatas e da Terra e Engenharias**

Prof. Dr. Adélio Alcino Sampaio Castro Machado – Universidade do Porto

Profª Drª Alana Maria Cerqueira de Oliveira – Instituto Federal do Acre

Profª Drª Ana Grasielle Dionísio Corrêa – Universidade Presbiteriana Mackenzie

Profª Drª Ana Paula Florêncio Aires – Universidade de Trás-os-Montes e Alto Douro

Prof. Dr. Carlos Eduardo Sanches de Andrade – Universidade Federal de Goiás

Profª Drª Carmen Lúcia Voigt – Universidade Norte do Paraná



Prof. Dr. Cleiseano Emanuel da Silva Paniagua – Instituto Federal de Educação, Ciência e Tecnologia de Goiás  
Prof. Dr. Douglas Gonçalves da Silva – Universidade Estadual do Sudoeste da Bahia  
Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná  
Profª Drª Érica de Melo Azevedo – Instituto Federal do Rio de Janeiro  
Prof. Dr. Fabrício Menezes Ramos – Instituto Federal do Pará  
Profª Dra. Jéssica Verger Nardeli – Universidade Estadual Paulista Júlio de Mesquita Filho  
Prof. Dr. Juliano Bitencourt Campos – Universidade do Extremo Sul Catarinense  
Prof. Dr. Juliano Carlo Rufino de Freitas – Universidade Federal de Campina Grande  
Profª Drª Luciana do Nascimento Mendes – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Prof. Dr. Marcelo Marques – Universidade Estadual de Maringá  
Prof. Dr. Marco Aurélio Kistemann Junior – Universidade Federal de Juiz de Fora  
Prof. Dr. Miguel Adriano Inácio – Instituto Nacional de Pesquisas Espaciais  
Profª Drª Neiva Maria de Almeida – Universidade Federal da Paraíba  
Profª Drª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte  
Profª Drª Priscila Tessmer Scaglioni – Universidade Federal de Pelotas  
Prof. Dr. Sidney Gonçalo de Lima – Universidade Federal do Piauí  
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista



**Diagramação:** Daphynny Pamplona  
**Correção:** Bruno Oliveira  
**Indexação:** Amanda Kelly da Costa Veiga  
**Revisão:** Os autores  
**Organizadora:** Lilian Coelho de Freitas

**Dados Internacionais de Catalogação na Publicação (CIP)**

C697 Collection: applied computer engineering 2 / Organizadora Lilian Coelho de Freitas. – Ponta Grossa - PR: Atena, 2022.

Formato: PDF

Requisitos de sistema: Adobe Acrobat Reader

Modo de acesso: World Wide Web

Inclui bibliografia

ISBN 978-65-258-0044-8

DOI: <https://doi.org/10.22533/at.ed.448221603>

1. Computer engineering. I. Freitas, Lilian Coelho de (Organizadora). II. Título.

CDD 621.39

Elaborado por Bibliotecária Janaina Ramos – CRB-8/9166

**Atena Editora**

Ponta Grossa – Paraná – Brasil

Telefone: +55 (42) 3323-5493

[www.atenaeditora.com.br](http://www.atenaeditora.com.br)

contato@atenaeditora.com.br



## DECLARAÇÃO DOS AUTORES

Os autores desta obra: 1. Atestam não possuir qualquer interesse comercial que constitua um conflito de interesses em relação ao artigo científico publicado; 2. Declaram que participaram ativamente da construção dos respectivos manuscritos, preferencialmente na: a) Concepção do estudo, e/ou aquisição de dados, e/ou análise e interpretação de dados; b) Elaboração do artigo ou revisão com vistas a tornar o material intelectualmente relevante; c) Aprovação final do manuscrito para submissão.; 3. Certificam que os artigos científicos publicados estão completamente isentos de dados e/ou resultados fraudulentos; 4. Confirmam a citação e a referência correta de todos os dados e de interpretações de dados de outras pesquisas; 5. Reconhecem terem informado todas as fontes de financiamento recebidas para a consecução da pesquisa; 6. Autorizam a edição da obra, que incluem os registros de ficha catalográfica, ISBN, DOI e demais indexadores, projeto visual e criação de capa, diagramação de miolo, assim como lançamento e divulgação da mesma conforme critérios da Atena Editora.



## DECLARAÇÃO DA EDITORA

A Atena Editora declara, para os devidos fins de direito, que: 1. A presente publicação constitui apenas transferência temporária dos direitos autorais, direito sobre a publicação, inclusive não constitui responsabilidade solidária na criação dos manuscritos publicados, nos termos previstos na Lei sobre direitos autorais (Lei 9610/98), no art. 184 do Código Penal e no art. 927 do Código Civil; 2. Autoriza e incentiva os autores a assinarem contratos com repositórios institucionais, com fins exclusivos de divulgação da obra, desde que com o devido reconhecimento de autoria e edição e sem qualquer finalidade comercial; 3. Todos os e-book são *open access*, *desta forma* não os comercializa em seu site, sites parceiros, plataformas de *e-commerce*, ou qualquer outro meio virtual ou físico, portanto, está isenta de repasses de direitos autorais aos autores; 4. Todos os membros do conselho editorial são doutores e vinculados a instituições de ensino superior públicas, conforme recomendação da CAPES para obtenção do Qualis livro; 5. Não cede, comercializa ou autoriza a utilização dos nomes e e-mails dos autores, bem como nenhum outro dado dos mesmos, para qualquer finalidade que não o escopo da divulgação desta obra.



## APRESENTAÇÃO

A série de *e-books* intitulada “*Collection: Applied computer engineering 2*” está organizada em 10 capítulos e apresenta diversas aplicações da engenharia de computação, com foco especial à aplicação de inteligência computacional em várias áreas do conhecimento, como mercado financeiro, transporte, saúde, jogos digitais, entre outros.

Dessa forma, esta coleção permitirá aos leitores uma ampla visão das potencialidades da engenharia da computação e dos avanços da pesquisa nesta área.

Os organizadores da Atena Editora agradecem aos autores, por viabilizaram a construção deste trabalho, e desejam a todos, uma leitura proveitosa.

Lilian Coelho de Freitas

## SUMÁRIO

### **CAPÍTULO 1..... 1**

#### **AVALIAÇÃO DE TÉCNICAS DE APRENDIZADO DE MÁQUINA APLICADAS À ANÁLISE DE RISCO DE CRÉDITO**

Jane Thais Soares de Oliveira

Rogério Alves Santana

Honovan Paz Rocha

 <https://doi.org/10.22533/at.ed.4482216031>

### **CAPÍTULO 2..... 21**

#### **FLUXO DE CARGA LINEARIZADO – UM ESTUDO COMPARATIVO USANDO A LINGUAGEM AMPL**

Hugo Andrés Ruiz Flórez

Gloria Patricia Lopez Sepulveda

Jose Airton Azevedo dos Santos

Cristiane Lionço de Oliveira

Leandro Antonio Pasa

 <https://doi.org/10.22533/at.ed.4482216032>

### **CAPÍTULO 3..... 33**

#### **IMPLEMENTAÇÃO DE REDE NEURAL CONVOLUCIONAL PARA PREDIÇÃO DE COVID-19 ATRAVÉS DE IMAGENS DE RAIOS X**

Erik Gabriel Cruz Sena

Honovan Paz Rocha

 <https://doi.org/10.22533/at.ed.4482216033>

### **CAPÍTULO 4..... 51**

#### **JOGOS DIGITAIS DE ENTRETENIMENTO E O ESTÍMULO DA INTELIGÊNCIA LÓGICO-MATEMÁTICA DE GARDNER**

Carlos Alberto Paiva

Regina Melo Silveira

 <https://doi.org/10.22533/at.ed.4482216034>

### **CAPÍTULO 5..... 74**

#### **RASTREAMENTO DE MOUSE PARA AVALIAÇÃO DE EXPERIÊNCIA DO USUÁRIO EM PORTAIS DE NOTÍCIAS: UM ESTUDO DE CASO**

Danilo Teixeira Lima

Flavio Rafael Trindade Moura

Kennedy Edson Silva de Souza

Rita de Cássia Romeiro Paulino

Marcos Cesar da Rocha Seruffo

 <https://doi.org/10.22533/at.ed.4482216035>

### **CAPÍTULO 6..... 87**

#### **ROTEAMENTO DE VEÍCULO GUIADO AUTONOMAMENTE PARA ARMAZÉNS**

## INTELIGENTES

Wesley Marques Lima

Honovan Paz Rocha

 <https://doi.org/10.22533/at.ed.4482216036>

### **CAPÍTULO 7..... 105**

UTILIZANDO GAN E REDES NEURAIS ARTIFICIAIS MLP PARA SUPORTE AO DIAGNÓSTICO PRECOCE DA DOENÇA DE ALZHEIMER: UM ESTUDO ACERCA DO POTENCIAL DA EXPANSÃO ARTIFICIAL DOS DADOS

Jonathan da Silva Bandeira

Renan Costa Alencar

Mêuser Jorge Silva Valença

 <https://doi.org/10.22533/at.ed.4482216037>

### **CAPÍTULO 8..... 121**

UTILIZAÇÃO DE UM PERCEPTRON MÚLTIPLAS CAMADAS NA APROXIMAÇÃO DE FUNÇÕES CONTÍNUAS

Dhiego Loiola de Araújo

 <https://doi.org/10.22533/at.ed.4482216038>

### **CAPÍTULO 9..... 133**

COMPUTAÇÃO EVOLUTIVA APLICADA AO MERCADO FINANCEIRO: UM SISTEMA DE RECOMENDAÇÃO ESTRATÉGICO PARA OS USUÁRIOS INICIANTE

Benjamin Luiz Franklin

 <https://doi.org/10.22533/at.ed.4482216039>

### **CAPÍTULO 10..... 147**

ESTUDO DA REPROVAÇÃO NO CURSO DE SISTEMAS DE INFORMAÇÃO DA UNIMONTES NO PERÍODO DE 2014-1 A 2019-2

Marilée Patta

Reginaldo Moraes de Macedo

 <https://doi.org/10.22533/at.ed.44822160310>

### **SOBRE A ORGANIZADORA..... 161**

### **ÍNDICE REMISSIVO..... 162**

## AValiação de Técnicas de Aprendizado de Máquina Aplicadas à Análise de Risco de Crédito

*Data de aceite: 01/03/2022*

### Jane Thais Soares de Oliveira

Graduanda em Engenharia Física Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri  
Janaúba  
MG, Brasil

### Rogério Alves Santana

Instituto de Engenharia, Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Janaúba  
MG, Brasil

### Honovan Paz Rocha

Instituto de Engenharia, Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Janaúba  
MG, Brasil

**RESUMO:** O serviço de concessão de crédito ao consumidor final tem crescido de forma contundente nos últimos anos, sendo esta uma tendência se levarmos em consideração os juros mais baixos praticados com um cenário político/econômico cada vez mais conservador. Neste cenário, torna-se ainda mais relevante o processo de análise de risco de crédito, que ainda utiliza muitas técnicas arcaicas, com enfoque no tratamento individual e subjetivo dos dados do candidato à concessão. Visando otimizar a tarefa de análise de risco de crédito o presente trabalho tem como objetivo o estudo, implementação e avaliação de 10 dentre os principais algoritmos de classificação presentes na literatura aplicados

à classificação de candidatos à concessão de crédito. Adicionalmente, a etapa de pré-processamento da base de dados utilizada incluiu a tarefa de seleção de características através de filtros univariados e multivariados, com o intuito de encontrar os atributos mais relevantes bem como a eliminação de redundâncias. Os experimentos realizados demonstraram a importância da seleção de características para melhoria dos algoritmos de classificação, além disso, foi possível verificar que os métodos do tipo ensemble obtiveram os melhores resultados de maneira geral considerando-se a base de dados utilizada.

**PALAVRAS-CHAVE:** Aprendizado de máquina, algoritmos de classificação, seleção de características, ensemble, análise de crédito

**ABSTRACT:** The service of granting credit to the final consumer has grown sharply in recent years, and this is a trend if we take into account the lower interest rates practiced with a scenario ethical/economic is increasingly conservative. In this scenario, the credit risk analysis process becomes even more relevant, which still uses many archaic techniques, focusing on the individual and subjective treatment of the grant candidate's data. Aiming to optimize the task of analysis of credit risk The present work aims at the study, implementation and evaluation of 10 among the main classification algorithms present in the applied literature to the classification of candidates for the concession of credit. In addition, the pre-processing stage of the database used included the characteristic task of characteristic attractions of univariate and multivariate filters, in

order to find the most relevant attributes as well as the elimination action of redundancies. The experiments carried out demonstrated the importance of the selection of characteristics for the improvement of the classification algorithms, in addition, it was possible to verify that the methods of the ensemble type obtained the best results. general way considering the database used.

**KEYWORDS:** Machine learning, classification algorithms, feature selection, ensemble, credit analysis

## 1 | INTRODUÇÃO

Atualmente, as empresas têm encontrado um mercado cada vez mais competitivo, e a busca por sobrevivência se torna desleal com empresas que não conseguem se adaptar a novas tecnologias. Isso não seria diferente para as instituições financeiras, que lidam com grande quantidade de solicitações de concessão de crédito, onde deve-se levar em consideração que os potenciais clientes são oriundos de diferentes classes sociais, faixa etária, dentre outros diversos fatores socioeconômicos. Portanto, possuir modelos para análise de crédito que sejam rápidos e eficazes, poderia colocar este tipo de organização numa situação favorável em relação às concorrentes.

Com o crescimento da área de Data Mining nas últimas décadas, devido ao aumento da disponibilidade de dados, informações de alto valor agregado geradas a partir de dados brutos começaram a ser descobertas e utilizadas para gerar mais valor nas organizações [1]. A área financeira não foge à regra, pois organizações deste segmento são detentoras de grande quantidade de dados ignorados ao longo dos anos, como por exemplo, dados fornecidos por um potencial cliente no ato de solicitação de crédito.

O processo de gerenciamento baseado em risco de crédito nas instituições financeiras vem passando por uma evolução ao longo dos últimos anos, pois os métodos usados na tomada de decisão tradicional, que é baseada exclusivamente em critérios julgamentais e subjetivos, têm perdido espaço dentro das instituições. Os novos cenários demandam a busca por instrumentos mais eficazes para o gerenciamento da exposição ao risco de crédito, visando minimizar perdas. [2]

As técnicas de classificação de padrões surgem nesse meio como uma alternativa para suprir a necessidade das instituições financeiras para tratar essas informações de maneira mais probabilística e generalista, evitando julgamentos individuais e subjetivos que desconsideram o histórico da empresa com relação a concessões de crédito. é importante salientar que, com base nas regras de decisão tradicionais, as chances de eventuais candidatos à inadimplência serem classificados de maneira distinta são relativamente altas. Entretanto, quando se efetua uma análise multivariada das variáveis junto à classificação automática baseada no histórico de indivíduos em situações semelhantes, o risco de classificar erroneamente pode diminuir consideravelmente. Neste contexto, o intuito deste trabalho é implementar, analisar e avaliar alguns dos principais algoritmos de aprendizado

de máquina aplicados à análise de risco de crédito, considerando-se as tarefas de seleção de características e classificação.

A principal contribuição deste trabalho é fornecer a um gestor uma ampla análise com indicações dos métodos potencialmente mais adequados à automatização da classificação dos clientes na tarefa de análise de risco de crédito. O enfoque desta proposta é o auxílio à tomada de decisão, dada a diversidade de algoritmos avaliados quanto à eficiência para classificação de clientes. Vale ressaltar que, considerando-se a amplitude das nossas pesquisas, percebemos uma escassez de trabalhos na literatura que realizem a análise e avaliação desta quantidade de combinações de métodos de seleção de características e classificação, aplicados à análise de risco de crédito.

Este trabalho está organizado como descrito a seguir. A Seção II apresenta os trabalhos relacionados. A Seção III apresenta uma breve descrição dos métodos de aprendizado de máquina utilizados. A Seção IV aborda o pré-processamento dos dados. A Seção V contém a metodologia utilizada nos experimentos. A Seção VI apresenta os resultados e discussões a cerca dos experimentos. Por fim, na Seção VII é apresentada a conclusão deste trabalho.

## 2 | TRABALHOS RELACIONADOS

Trabalhos direcionados à análise de risco de crédito através da aplicação de técnicas de inteligência artificial não são novidade. Entretanto, devido ao conservadorismo inerente às instituições financeiras, continua havendo certa resistência por parte das mesmas para aplicação destas pesquisas em ambiente real. Estas dificuldades acabam por desacelerar algumas pesquisas, o que de certa forma abre campo para exploração. A seguir, serão abordados trabalhos aplicados nesta área.

A dissertação em [3] aborda o desenvolvimento de 3 algoritmos para classificação de clientes na área de crédito, ao qual são usadas as técnicas de Regressão Logística, Redes Neurais e Algoritmos Genéticos, com a justificativa de que qualquer avanço nas técnicas de classificação que resulte numa melhora de precisão para um modelo de previsão, acarretará em ganhos financeiros para a instituição.

A pesquisa proposta por [4] buscou Avaliar o Risco de Crédito utilizando alguns grupos de classificadores, os modelos utilizados foram: Regressão Logística, rede Bayesiana Ingênua, SVM, árvore de decisão e aprendizagem baseada em instâncias. Caracterizou-se as principais vantagens e desvantagens dos modelos, concluiu que o grau de acerto das predições usando um grupo de classificadores ao invés de um único modelo de classificação propicia resultados superiores ao melhor classificador individual.

O artigo em [5] apresentou um novo modelo de classificação para avaliação de risco de crédito, baseado em pesquisas das quais afirmavam que conjuntos de classificadores são superiores a classificadores únicos, o artigo propõe um novo modelo de classificação

usando da técnica de sobreamostragem (SMOTE) ao qual é usada para mitigar os efeitos negativos de conjuntos de dados desequilibrados ao rebalancear o conjunto de dados de treinamento, combinado com o algoritmo de otimização enxame de partículas, que tem como função pesquisar os pesos e desvios junto com a rede neural.

O trabalho proposto por [6] demonstra que a classificação voltada para problemas de análise de crédito apresentam muitos infortúnios, dado que os resultados sofrem interferência de fatores como desbalanceamento de uma das classes do conjunto de dados, desequilíbrio de dados nas amostras, dentre outros empecilhos. Portanto, para maximizar os resultados da classificação, o mesmo propôs a utilização de um algoritmo de reamostragem combinado baseado em um modelo de mistura gaussiano. Funcionando da seguinte maneira: A partir do fator de amostragem, determina o número de amostras da classe majoritária e da classe minoritária, em seguida, o agrupamento da mistura gaussiana é usado para subamostragem da maioria das amostras, e a técnica de sobreamostragem minoritária sintética é usada para o resto das amostras, a fim de eliminar qualquer problema de desequilíbrio.

O artigo em [7] propõe uma análise do método vizinho mais próximo ponderado, em que o mesmo foi aplicado na avaliação de crédito. Para os experimentos utilizaram uma base de dados de um banco privado da Indonésia. Com o intuito de analisar o desempenho de algumas funções do kernel, tais como: retangular, triangular, gaussiano, epanechnikov, triweight e inversão. A pesquisa obteve como resultado que a utilização do kernel gaussiano é superior as demais.

A pesquisa proposta por [8] tem o objetivo de fazer o uso combinado de pontuação de crédito e da pontuação de lucro para aumentar a eficácia do processo de concessão de empréstimos em cooperativas de crédito. O artigo revela que uso de métodos estatísticos melhoram significativamente a previsibilidade do padrão quando comparado ao uso de técnicas subjetivas. Além de demonstrar a superioridade do modelo random forest em estimar pontuação de crédito e de lucro em comparação com o método de regressão logística. Salienta-se que para fazer essa análise o trabalho utilizou uma base de dados de uma cooperativa de crédito brasileira.

Foram organizados na Tabela I alguns resultados encontrados em trabalhos relacionados a análise de risco de crédito.

Contudo, só foram listados os resultados que utilizavam a base "German Credit, que é a base utilizada neste trabalho.

ordem	autor	método	Acurácia (%)
1	[5]	RNA+PSO	78,70
2	[9]	SVM (RBFK)	73,60
3	[10]	MLP	73,93
4	[11]	MLP	75,35
5	[12]	ifair	73,00
6	[6]	mistura gaussiana	75,21

Tabela I: comparação de resultados da literatura para previsão de crédito para a base german credit.

### 3 I CLASSIFICAÇÃO DE PADRÕES

A tarefa de classificar padrões é realizada baseada em um conjunto de dados contendo observações (atributos) e cujas categorias (classes) são conhecidas. Existem diversos algoritmos para classificação de padrões, os classificadores utilizados neste trabalho serão abordados nas próximas subseções.

#### A. Naive Bayes

O teorema de Bayes criado por Thomas Bayes no séc. XVIII foi a inspiração para elaboração do algoritmo *Naive Bayes*, que se tornou um classificador probabilístico popular na área de Aprendizado de máquina. O método *Naive Bayes* é um algoritmo de aprendizagem supervisionada baseados na aplicação do teorema de *Bayes* com a suposição “ingênua” de independência condicional entre cada par de características dado o valor da variável de classe [13].

#### B. Logistic Regression

A Regressão Logística (inglês: *Logistic regression*) é um dos algoritmos de *machine learning* mais conhecido e utilizado no mundo, este método trabalha com os conceitos de estatística e probabilidade. De forma técnica a Regressão Logística mede a relação entre a variável dependente categórica e uma ou mais variáveis independentes, estimando as probabilidades com base em uma função logística [14].

#### C. K-nearest-neighbors

O algoritmo Kvizinhos mais próximos (inglês: *K-nearestneighbors* ou *KNN*) é um algoritmo de aprendizado de máquina supervisionado simples e fácil de se implementar que pode ser usado para resolver problemas de classificação e regressão. O aprendizado baseado em instâncias é um método não-paramétrico, que consiste em: Dado um banco de dados, com exemplos já rotulados e um novo elemento desconhecido, esse elemento assumi as características dos K exemplos que estejam mais próximos a ele. Ou seja, quanto mais próximos os exemplos do elemento, maiores são as semelhanças, consequentemente a vizinhança do elemento passa a descreve-lo [14].

#### D. *Support vector machine*

Uma máquina de vetores de suporte (inglês: *support vector machine* ou *SVM*) é um classificador linear binário não probabilístico que possui um aprendizado supervisionado que analisa os dados e reconhece padrões. O algoritmo de treinamento do SVM constrói um modelo que atribui novos exemplos a uma categoria ou outra. Portanto o modelo representa os exemplos como pontos no espaço, mapeados de maneira que os exemplos de cada categoria sejam divididos por um espaço claro que seja tão amplo quanto possível. O que uma SVM faz é encontrar uma linha de separação, mais comumente chamada de hiperplano entre dados de duas classes. Essa linha busca maximizar a distância entre os pontos mais próximos em relação a cada uma das classes [15].

#### E. *Decision Tree*

Árvore de decisão (inglês: *Decision tree*) é um método de aprendizado supervisionado não paramétrico usado em classificações. Tem como objetivo criar um modelo que preveja o valor de uma variável de destino, aprendendo regras de decisão simples inferidas dos recursos de dados. Uma árvore pode ser vista como uma aproximação constante por partes. Ou seja, ela aprende com os dados ao se aproximar de uma curva senoidal com um conjunto de regras de decisão *if-thenelse*. Quanto mais profunda a árvore, mais complexas são as regras de decisão e mais adequado é o modelo [16].

#### F. *Bagging Decision Tree*

*Bagging Decision Tree* é um algoritmo de aprendizagem supervisionada, que cria um conjunto de árvores de decisão usando *Bagging*. Sendo que *Bagging* (é um acrônimo para agregação de bootstrap), é entendido como um meta-algoritmo da área de aprendizado de máquina, proposto por Breiman [17] em 1996 com o intuito de melhorar a precisão da predição.

O algoritmo *Bagging* gera várias amostras de *bootstraps* diferentes e de forma aleatória, baseado em amostragem que substitui o conjunto de dados original. Ou seja, cada instância selecionada pode ser repetida várias vezes na mesma amostra, consequentemente aumenta-se os dados de treinamento. Sendo a previsão final uma média de todos os modelos preditivos. muito utilizado quando o objetivo é reduzir a variância dos classificadores individuais.

#### G. *Boosting Decision Tree*

*Boosted Decision Tree* é um algoritmo de aprendizagem supervisionada, que cria um conjunto de árvores de decisão usando *boosting*. Portanto, é um procedimento que agrega muitos classificadores “fracos” para alcançar um alto desempenho de classificação. Além disso, o *boosting* ajuda a estabilizar a resposta dos classificadores em relação às mudanças na amostra de treinamento. Já que o algoritmo aprende ajustando o resíduo das

árvores que o precederam [18].

#### H. *Random Forest*

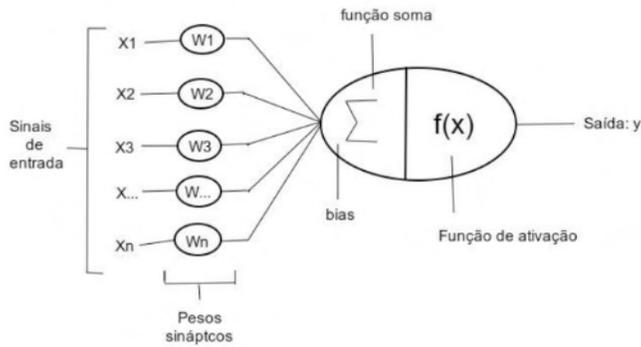
Criado por Tim Kan Ho [19] em 1995 as florestas aleatórias (inglês: *Random forest*) é um algoritmo que cria muitas árvores de decisão, de maneira aleatória, formando uma floresta, onde cada árvore será utilizada na escolha do resultado final. Florestas aleatórias surgem como meio de calcular a média de várias árvores de decisão profundas, treinadas em diferentes partes do mesmo conjunto de treinamento com o objetivo de reduzir a variância, essa redução ocorre com um custo que é um pequeno aumento do viés e perda de interpretabilidade, mas geralmente aumenta significativamente o desempenho no modelo final. Cabe ressaltar que esse modelo surgiu pois árvores de decisão que crescem muito fundo tendem a aprender padrões altamente irregulares, conseqüentemente elas se ajustam demais ao conjunto de treinamento, ou seja, tem baixa polarização e alta variância.

#### I. *Voting Classification*

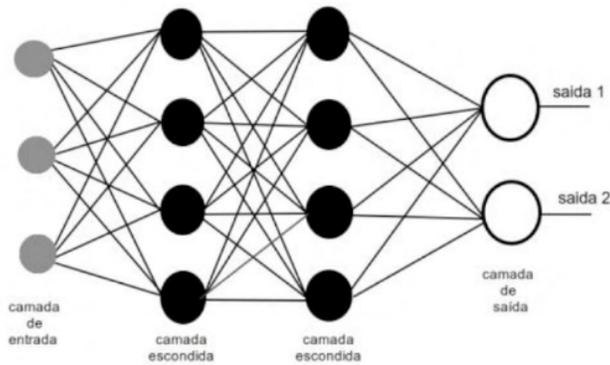
O classificador de votação (inglês: *Voting classification*) é um modelo de aprendizado de máquina que treina um conjunto de vários modelos e prevê uma saída (classe) com base em sua probabilidade mais alta da classe escolhida como a saída. Ou seja, ele combina classificadores de aprendizado de máquina conceitualmente diferentes e usa um voto majoritário ou as probabilidades médias previstas (voto suave) para prever os rótulos das classes. Tal classificador pode ser útil para um conjunto de modelos de desempenho igualmente bom, a fim de equilibrar suas fraquezas individuais [20]. Portanto, optou-se neste trabalho por utilizar a combinação de 3 classificadores, sendo estes: KNN, SVM e *Random Forest*.

#### J. *Neural Network*

Redes neurais Artificiais (inglês: *Neural Network*) são modelos computacionais inspirados no sistema nervoso humano e que o intuito é aprender, podendo ser utilizadas em tarefas de classificação. Em outras palavras, dado um problema, a RNA aprende quando acha uma possível solução geral (chamada de generalização), que é capaz de prever novos dados [21]. Uma rede neural é composta por vários neurônios como da Fig. 1 (a), conectados em camadas, que calculam determinadas funções matemáticas (conhecidas como funções de ativação) são organizadas de modo a ter camadas de entrada, ocultas e de saída, lembrando que uma RNA pode possuir uma ou múltiplas camadas ocultas, modelo de RNA na Fig. 1 (b) .



(a) Esquema de neurônio.



(b) Esquema de uma rede neural

Figura 1. Rede neural.

1) *Multilayer perceptron*: Existem diversos modelos de Redes Neurais, para este trabalho utilizou-se as redes de múltiplas camadas (inglês: *Multilayer Perceptron — MLP*). Sendo um modelo que apresenta uma ou mais camadas de neurônios entre as camadas de entrada e de saída da rede, também conhecidas como intermediárias ou ocultas, e o seu treinamento é feito utilizando o algoritmo de Retropropagação (*Backpropagation*). Por conseguinte, o treinamento de uma rede MLP consiste em ajustar os pesos e os viés ou bias de suas unidades para que a classificação desejada seja obtida.

#### 4 | PRÉ-PROCESSAMENTO DOS DADOS

O trabalho proposto em [22] mostra que o “Pré processamento de dados” consiste na área que trabalha os dados para maximizar as possibilidades em se encontrar padrões úteis, sendo de grande importância no processo de mineração, pois um algoritmo aplicado

a uma base de dados que contenha valores discrepantes e que não tenha passado por um pré-processamento pode identificar padrões de dados incongruentes.

A pesquisa em [23] complementa ao afirmar que para se obter todo o conhecimento disponível dentro de uma base de dados, o primeiro passo é preparar essa base, a fim de eliminar redundâncias. A normalização dos dados, por exemplo, mantém os dados padronizados dentro de uma mesma faixa de valores, facilitando tarefas como aquelas pertencentes à área de aprendizado de máquina. A seguir serão abordados dois assuntos importantes na área de pré-processamento, que são: Normalização e Seleção de características.

#### A. Normalização

A normalização trata-se de um dos passos mais básicos quando se quer trabalhar com bases de dados em que seus atributos numéricos estejam em faixas numéricas diferentes, e conseqüentemente, um atributo possa sobrepor outro, gerando valores discrepantes ou até mesmo impossibilitando que o classificador obtenha resultados precisos. Existem diversas métricas de normalização para uma base de dados, a escolhida neste projeto é dada por (1) que mantém a faixa de todos os atributos entre [-1, 1].

$$x_{ij} = 2 * \left( \frac{x_{ij} - x_{jmin}}{x_{jmax} - x_{jmin}} \right) - 1, \quad (1)$$

onde  $x_{ij}$  é o valor do atributo  $j$  para o cliente  $i$ ,  $x_{jmin}$  e  $x_{jmax}$  respectivamente são: o valor mínimo e máximo encontrado no atributo  $j$ .

#### B. Seleção de Características

Selecionar características é uma tarefa extremamente necessária quando se lida com bases de dados, já que possui inúmeras informações e muitas delas podem ser redundantes. O trabalho proposto por [24] aborda o fato de que as bases de dados por muitas vezes podem conter atributos irrelevantes bem como um número reduzido de amostras, provocando um aumento de complexidade computacional e a perda de exatidão na tarefa de classificação, a partir disto, a seleção de características, surge para minimizar o problema, e tornar o algoritmo mais eficiente, dado que remove atributos irrelevantes para a classificação.

Neste trabalho foram utilizado 3 métodos de seleção de características: *F-score*, Coeficiente de correlação de Pearson, e um Algoritmo Genético.

1) *F-score*: O método de seleção *F-Score* (*Fisher score*) é simples de implementar e também eficiente, pois consegue medir a relevância de cada atributo para as classes, sendo uma análise univariada dos atributos. Considerando um problema de classificação binário, onde as classes são  $C1$  e  $C2$ , a fórmula é definida por:

$$f(i) = \frac{(\mu_i^{C1} - \mu_i) + (\mu_i^{C2} - \mu_i)}{\sigma_i^{C1} + \sigma_i^{C2}} \quad (2)$$

onde  $\mu_i^C$  e  $\sigma_i^C$  correspondem, respectivamente, à média e o desvio padrão para a classe C com relação i-ésima à característica.

2) *Coefficiente de correlação de Pearson*: O coeficiente de Correlação de Pearson pode ser definido como um filtro univariado para seleção de características, dado que ele gera um ranking que pontua cada atributo de acordo com sua capacidade individual de discriminar duas classes. Com base no ranking gerado, pode-se eliminar os atributos com as piores pontuações do ranking, através da definição de um limiar. O coeficiente é dado pela fórmula:

$$C(j) = \frac{\sum_{i=1}^p (x_{ij} - \bar{x}) \cdot (y_i - \bar{y})}{\sigma_{x_j} \cdot \sigma_y} \quad (3)$$

onde  $x_{ij}$  é o valor do atributo na j-ésima característica e no i-ésimo padrão de entrada,  $\bar{x}$ ,  $\bar{y}$ ,  $\sigma_x$  e  $\sigma_y$  são respectivamente as médias e desvios-padrão de x e y.

### C. Algoritmo Genético

O algoritmo Genético (*inglês: genetic algorithm GA*) foi proposto na década de 1960 por John H. Holland um pesquisador da Universidade de Michigan, com o intuito de otimizar sistemas complexos. O trabalho [25] descreve o GA como sendo um algoritmo matemático inspirado no princípio Darwiniano, onde os mais aptos tendem a sobreviver e reproduzir. Com base nisso, o algoritmo é um mecanismo de busca adaptativa, que leva em consideração os princípios de seleção natural e recombinação genética.

Por ser um algoritmo muito versátil tem inúmeras utilidades totalmente numérica desse conjunto de dados que também foi disponibilizada no mesmo local. Para facilitar a classificação foi utilizada a base de dados modificada que contem apenas atributos numéricos e não há dados faltantes.

Cada iteração do algoritmo genético corresponde à aplicação de um conjunto de quatro operações básicas: cálculo de aptidão, seleção, cruzamento e mutação. A implementação das duas últimas operações citadas anteriormente, podem ser vistas na Fig. 2. Para a operação de seleção foi utilizado o método da roleta. Ao final de cada iteração, também chamada de geração, cria-se uma nova população que tende a representar uma melhor aproximação da solução do problema de otimização do que a população anterior.

## 5 I METODOLOGIA

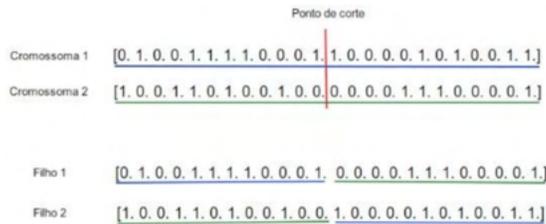
Os algoritmos avaliados neste trabalho foram elaborados em linguagem Python,

utilizando o IDE (ambiente de desenvolvimento integrado) Spyder 3.7, sistema operacional Ubuntu 18.04 Lts e processados em um computador com as seguintes configurações: Intel(R) Core(TM) i5-7200U CPU 2.50GHz, 8Gb de RAM, 64-bit.

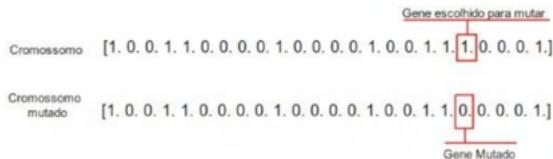
### A. Base de dados

A base de dados utilizada nesse estudo foi a *German Credit*. Esta base foi obtida no repositório UCI [26] elaborada pelo professor Hofmann [27] na cidade de Hamburgo e consiste em uma base de dados real com informações de 1000 clientes de uma instituição bancária. Esta base de dados, foi originalmente disponibilizada contendo atributos categóricos e numéricos, contudo, a Universidade Strathclyde produziu uma versão totalmente numérica desse conjunto de dados que também foi disponibilizada no mesmo local. Para facilitar a classificação foi utilizada a base de dados modificada que contém apenas atributos numéricos e não há dados faltantes.

O perfil de cada um destes clientes é definido através de 24 variáveis numéricas e uma variável de rótulo, indicando se este cliente se tornou inadimplente ou não. Vale ressaltar que esta é uma base desbalanceada, contendo 30% de inadimplentes e 70% adimplentes.



(a) Demonstração de um cruzamento.



(b) Demonstração de uma mutação.

Figura 2. Algoritmo Genético implementado.

### B. pré-processamento e seleção de características aplicado a base

Inicialmente, com intuito de padronizar todas as variáveis da base de dados para uma escala comum, utilizou-se (1) para efetuar a normalização da base, deixando todas as variáveis na faixa  $[-1, 1]$ . Com relação à variável rótulo a base de dados foi alterada da seguinte forma: uma das classes (classe de inadimplentes) que era representada pelo valor 2, foi alterada para o valor -1.

1) *Seleção de Características*: Nesta etapa, utilizou-se dois filtros univariados (F-score e Pearson) e um método multivariado (Algoritmo Genético) para efetuar a seleção de características.

Ao aplicar o filtro *F-score* na base de dados, foi obtida uma pontuação, que pode ser observada na Fig. 3. No topo de cada coluna encontra-se o número de identificação do atributo, que está listado na segunda coluna da Tabela III. Pode-se observar na Fig. 3 que os 6 melhores atributos se destacam, dado esse fato, os mesmos foram escolhidos de forma empírica para serem utilizados na classificação.

O método Coeficiente de Pearson foi aplicado na base de dados e gerou um ranking, que pode ser visto na Fig. 4. Com o auxílio da segunda coluna da Tabela III é possível identificar todos os atributos, pois no topo de cada coluna encontra-se a numeração que identifica o mesmo. Nota-se que, para o Coeficiente de Pearson os 4 primeiros atributos se destacam dos demais, portanto, os mesmos foram escolhidos de forma empírica para serem utilizados pelos classificadores.

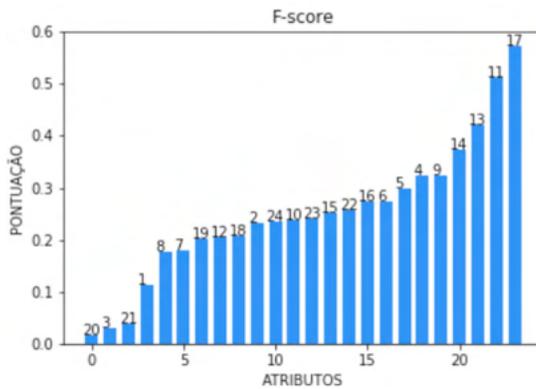


Figura 3. Pontuação gerada pelo F-score.

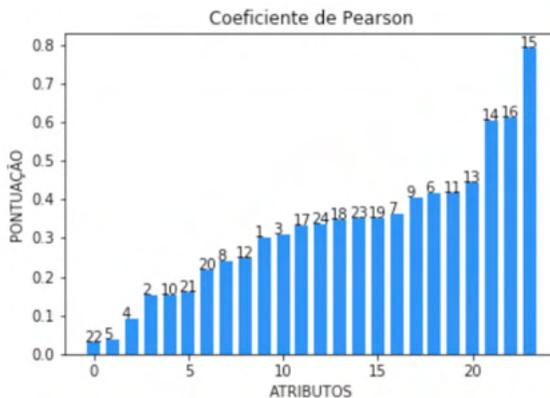


Figura 4. Pontuação gerada pelo Coeficiente de Pearson.

Para o presente trabalho, o GA foi projetado para encontrar a melhor combinação de atributos para cada método aplicado, fazendo uma análise multivariada, a fim de otimizar a função objetivo. Neste caso, temos: O vetor de atributos como sendo o indivíduo, a função objetivo assumindo a acurácia de cada modelo e foram utilizados 3 critérios de convergência para o GA (Atingir um dado numero de gerações, diferença entre melhor indivíduo, e a média da população, percentual 95%da população convergir para um mesmo valor), um deles sendo satisfeito, a melhor solução foi encontrada. O algoritmo genético foi aplicado utilizando os parâmetros definidos na Tabela II que foram baseados no trabalho [28].

A maioria dos métodos de aprendizado de máquina necessitam do ajuste de parâmetros ao serem implementados. A escolha dos parâmetros de vários dos métodos implementados neste trabalho foi feita com base na literatura. A segunda coluna das Tabelas VI, VII e VIII contem as referências aos trabalhos de onde os parâmetros foram obtidos para cada algoritmo, excluindo-se os classificadores *Naive Bayes* e *Voting* que não possuem parâmetros a serem ajustados. Este último, é composto pelos classificadores KNN, SVM e Random Forest, já parametrizados com base em outros trabalhos da literatura.

População	Chance de Mutação	Chance de Cruzamento
30	0,01	0,80

Tabela II : parâ metros escolhidos para o GA.

## 6 | EXPERIMENTOS

Os experimentos apresentados nesta seção englobam a combinação dos métodos de seleção de características e os 10 algoritmos de classificação implementados.

### A. Resultados

A divisão dos dados de treinamento e teste foi executada mediante a utilização do método de validação cruzada *k-fold*. O valor escolhido foi  $k = 10$ , seguindo diversos trabalhos da literatura.

Com o intuito de se ter uma visão prévia de quais variáveis possuem maior relevância para classificação na base "*German Credit*", os filtros univariados *F-score* e Coeficiente de Pearson foram aplicados. Gerando uma pontuação capaz de auxiliar na seleção de características. Contudo, isso nos traz apenas uma visão inicial e individualizada, uma vez que estes filtros não levam em consideração a informação gerada pela combinação de atributos.

Para que seja possível uma análise da relevância dos atributos considerando-os de maneira combinada, um Algoritmo Genético foi implementado e ajustado para realizar a seleção de características de forma multivariada. A Tabela III apresenta todos os atributos

selecionados pelo GA para cada modelo de classificação. Diferente dos filtros *F-score* e Coeficiente de Pearson o Algoritmo Genético não gera um ranking, consequentemente, não é possível determinar quais os melhores atributos. Para definir os atributos selecionados pelo GA, ele foi executado 10 vezes para cada classificador, de forma que, a solução final foi definida pelos atributos encontrados em pelo menos 50% das execuções. A Tabela III está organizada da seguinte maneira: a coluna 1 contem o ID do atributo, a coluna 2 descreve o atributo, e as colunas de 3 a 12 contêm um valor binário, onde o valor 1 informa que o atributo foi selecionado pelo GA considerando-se o classificador indicado no cabeçalho da coluna e 0 caso o atributo não esteja entre os selecionados.

A Tabela IV mostra a quantidade de atributos selecionados por cada método de seleção de características para cada classificador utilizado no trabalho.

Com o intuito de facilitar a visualização e identificação dos resultados contidos nas Tabelas V, VI, VII e VIII destacou-se em **negrito** os melhores resultados.

A Tabela V foi montada com o intuito de propiciar uma análise comparativa entre os filtros univariados, bem como a combinação dos dois filtros utilizados. O melhor resultado entre eles será confrontado posteriormente com o Algoritmo Genético, que é um método multivariado.

nº	Atributos	KNN	Bayes	R.Logistica	A.decisão	SVM	R forest	bagging	boosting	Voting	MLP
1	Verificação de saldo	1	1	1	0	1	1	1	1	1	1
2	Nº meses do empréstimo	1	1	1	0	1	0	1	0	1	0
3	Historico de credito	1	0	0	1	0	1	0	1	0	0
4	Crédito de imposto mín. alternativo-AMT	0	0	0	1	0	1	1	1	1	1
5	Saldo poupança	1	1	0	1	0	0	1	0	1	0
6	Trabalho Atual	1	0	0	0	0	0	1	0	1	0
7	Sexo	1	0	1	1	0	0	1	0	1	0
8	Tempo de residencia atual	1	1	1	0	1	0	1	0	1	0
9	Propriedade	0	0	0	0	1	1	1	1	1	1
10	Idade	0	1	1	1	0	0	0	0	0	0
11	Outros parcelamentos	0	0	1	0	0	0	1	0	1	0
12	Crédito em outro banco	0	0	0	1	0	1	1	1	1	1
13	Conta individual/ conjunta	0	0	1	0	0	1	0	0	0	0
14	Telefone	0	0	1	1	1	1	0	1	0	1
15	Trabalho estrangeiro	1	0	1	1	1	1	1	1	1	1
16	Compracar novo	0	1	1	1	0	0	1	0	1	0
17	Compracar usado	0	0	1	1	1	1	1	1	1	1
18	devedors/avalista	0	0	1	0	1	1	1	1	1	1
19	Devedorc/avalista	0	0	0	0	0	1	0	0	0	0

20	Aluguelcasa	0	1	1	0	1	0	0	0	0	0
21	Possuicasa	1	1	1	0	0	1	0	1	0	1
22	Desempregado	0	1	1	1	0	1	1	1	0	1
23	Trabalhoinformal	0	0	0	0	1	0	1	1	1	0
24	Trabalhoformal	1	0	1	1	1	0	1	1	1	0

Tabela III: Atributos selecionados pelo GA para cada método utilizado.

Modelo	Nº atributos
Sem seleção (todos os modelos)	24
<i>F-score</i> (todos os modelos)	6
Pearson (todos os modelos)	4
<i>F-score</i> + Pearson (todos os modelos)	8
KNN(GA)	10
Bayes(GA)	9
R.Logística(GA)	16
SVM(GA)	11
Árvore de decisão (GA)	12
Random forest (GA)	13
Bagging (GA)	17
Boosting (GA)	13
Voting (GA)	16
MLP (GA)	10

Tabela IV: Número de atributos selecionados por cada método de seleção de características.

Com o objetivo de se analisar os possíveis benefícios da seleção de características, todos os classificadores foram aplicados à base de dados sem nenhuma seleção. A Tabela VI mostra os resultados obtidos para os 10 classificadores.

A Tabela VII mostra os resultados obtidos para os 10 classificadores, usando na seleção de características a combinação dos dois filtros univariados, pois esta combinação se mostrou mais eficiente que cada filtro aplicado individualmente, como pode ser observado através da Tabela V.

Modelo	Acurácia		
	<i>F-score</i>	C. Pearson	<i>F-score</i> + C. Pearson
KNN	0,631	0,677	0,691
Bayes	0,687	0,645	0,689
R. logística	0,692	0,691	0,730
SVM	0,699	0,692	0,728
A. decisão	0,683	0,680	0,686
R. Forest	0,714	0,714	0,720
Bagging	0,715	<b>0,716</b>	0,740
Boosting	0,711	0,715	<b>0,758</b>
Voting	<b>0,721</b>	0,713	0,729
MLP	0,706	0,701	0,735

Tabela V: resultados obtidos entre os filtros univariados.

Modelo	parâmetros	Acurácia (%)	Desvio padrão	Máximo (%)	Mínimo (%)	Tempo médio(s)
KNN	[7]	0,711	0,032	0,719	0,688	0,012
Bayes	-	0,688	0,041	0,740	0,650	<b>0,003</b>
R. logística	[8]	0,752	0,033	0,780	0,699	0,010
SVM	[9]	0,738	0,029	0,760	0,700	0,129
A. decisão	[29]	0,699	<b>0,027</b>	0,719	0,650	0,004
R. Forest	[8]	0,767	0,037	0,810	<b>0,740</b>	3,312
Bagging	[30]	<b>0,785</b>	0,036	<b>0,839</b>	<b>0,740</b>	3,529
Boosting	[31]	0,739	0,034	0,798	0,710	0,827
Voting	-	0,746	0,042	0,800	0,709	0,191
MLP	[11]	0,740	0,031	0,794	0,700	0,879

Tabela VI: resultados obtidos sem seleção de características.

## B. Discussões

Analisando-se a Tabela V pode-se verificar que ao com binar os dois filtros obteve-se uma melhoria significativa na classificação com relação à utilização individualizada deles.

Modelo	parâmetros	Acurácia (%)	Desvio padrão	Máximo (%)	Mínimo (%)	Tempo médio(s)
KNN	[7]	0,691	0,042	0,720	0,649	<b>0,001</b>
Bayes	-	0,689	0,031	0,711	0,644	0,002
R. logística	[8]	0,730	0,041	0,778	0,698	0,004
SVM	[9]	0,728	0,036	0,799	0,701	0,067
A. decisão	[29]	0,686	0,054	0,721	0,630	0,002
R. Forest	[8]	0,731	0,034	0,806	0,701	0,770
Bagging	[30]	0,740	0,043	0,811	0,699	1,760
Boosting	[31]	<b>0,758</b>	0,046	<b>0,825</b>	<b>0,711</b>	0,740
Voting	-	0,729	0,048	0,801	0,699	0,070
MLP	[11]	0,735	<b>0,029</b>	0,798	0,703	0,739

Tabela VII : resultados obtidos com seleção de características (f-score + coeficiente de pearson).

Modelo	parâmetros	Acurácia (%)	Desvio padrão	Máximo (%)	Mínimo (%)	Tempo médio(s)
KNN	[7]	0.731	0.038	0.800	0.670	0,003
Bayes	-	0.728	0,039	0,780	0,650	<b>0,002</b>
R. logística	[8]	0.785	0,032	0,839	0,719	0,032
SVM	[9]	0.759	<b>0,028</b>	0,810	0,709	0,069
A. decisão	[29]	0.706	0,050	0,770	0,640	0,003
R. Forest	[8]	0.766	0,039	<b>0,869</b>	0,729	1,467
Bagging	[30]	<b>0,792</b>	0,034	0,819	<b>0,731</b>	2,34
Boosting	[31]	0.765	0,038	0,829	0,728	0,807
Voting	-	0.771	0,033	0,801	0,719	0,095
MLP	[11]	0.762	0,044	0,831	0,711	0,369

Tabela VIII: resultados obtidos com seleção de características (Algoritmo genético).

Pegando como exemplo o *Boosting Decision Tree*, é possível verificar que o mesmo aumentou sua acurácia em aproximadamente 4,3% ao combinar os filtros.

Entretanto, ao compararmos as Tabelas VI e VII podemos observar que a seleção de características com os filtros univariados gerou maiores valores de acurácia em poucos casos, quando efetuamos a comparação com os resultados obtidos sem qualquer seleção de atributos. Contudo, observase que mesmo selecionando somente os atributos de maior destaque para combinação dos filtros, sendo relativamente um número pequeno de atributos, foi possível a obtenção de resultados satisfatórios. Portanto, podemos considerar que existem benefícios ao utilizar estes filtros, pois são resultados satisfatórios obtidos com baixo custo computacional. Além disso os filtros são de fácil implementação e reduzem o custo computacional na etapa de classificação, consistindo em uma alternativa viável para seleção de características em bases de dados da área análise de risco de crédito. O algoritmo de *Boosting Decision Tree*, por exemplo, aumentou sua acurácia em 1,9% e reduziu seu tempo médio de execução em 0,087 segundos quando comparamos os resultado sem seleção de características e com a seleção por combinação dos filtros Fscore e Coeficiente de Pearson.

Fazendo uma análise dos resultados da Tabela VIII notase que o algoritmo genético foi um excelente método de seleção de características. Mesmo tendo um custo computacional relativamente alto, vale a pena sua implementação para selecionar os melhores atributos da base de dados de acordo com cada classificador. Por culminar na melhoria significativa dos classificadores, seja na avaliação da acurácia quanto na redução do custo computacional na etapa de classificação.

Analisando por exemplo: o algoritmo Bagging Decision Tree, nota-se, que o mesmo aumentou sua acurácia em 0,7% e reduziu seu tempo médio de execução em 1,18 segundos quando se compara o resultado sem seleção de características e com seleção utilizando o GA. Observa-se o mesmo comportamento para os demais métodos, como o algoritmo *Boosting Decision Tree* que aumentou sua acurácia em 2,6% e reduziu seu tempo médio de execução em 0,020 segundos.

Ao comparar as Tabelas VI, VII e VIII verifica-se que os métodos tendem a melhorar seu desempenho a medida que se retira atributos redundantes da base de dados e que o comportamento dos classificadores tendem a se manter constantes no ranking, sofrendo variações mínimas de posição. Tanto que os 3 piores métodos (KNN, bayes, árvore de decisão), mesmo tendo melhorias nos resultados continuam ruins se comparados aos demais, permanecendo nas últimas 3 posições no decorrer dos experimentos. Observa-se que os métodos do tipo ensemble foram bastante regulares mantendo-se sempre entre os primeiros colocados. Nota-se que o melhor resultado geral (ver Tabela VIII) foi obtido pelo método *Bagging Decision Tree* com 79,2% de acurácia, seguido pela Regressão Logística com 78,5%, os demais métodos de *ensemble*, a rede MLP com 76,2%, e a SVM 75,9%.

Contudo, cabe ressaltar que mesmo os métodos do tipo ensemble obtendo os melhores resultados, os mesmos possuem os maiores custos computacionais. Enquanto que outros métodos com custo computacionais relativamente baixos também classificam bem. Consequentemente, para analisar a viabilidade do método, fatores como esse precisam ser considerados, a fim de determinar qual método melhor se encaixa as necessidades do gestor.

## 7 | CONCLUSÃO

As técnicas de aprendizado de máquina utilizadas neste trabalho demonstraram ser de grande valia para a tarefa de análise de risco de crédito. Sendo que todos os 10 classificadores implementados mostraram precisão semelhante a trabalhos da literatura, considerando-se a base de dados utilizada.

Foi possível verificar que, mesmo sendo um classificador simples, a Regressão Logística obteve um bom desempenho nos experimentos, o que pode indicar benefícios da seleção de características. Os métodos de *ensemble* obtiveram os melhores resultados de forma geral para o problemas em questão, seguidos pela rede MLP e SVM. Cabe ressaltar que, todos os métodos de modo geral obtiveram resultados semelhantes a vários trabalhos da literatura (ver Tabela I).

Os experimentos também demonstraram a efetividade dos métodos de seleção de características, quanto a redução do custo computacional e a melhoria significativa da classificação, onde foi possível verificar que o Algoritmo Genético foi o melhor método para seleção dos atributos.

O trabalho demonstrou que, as técnicas avaliadas são capazes de contribuir para uma maior compreensão do processo de concessão de crédito, auxiliando assim o processo de tomada de decisão do gestor. Desta forma, o gestor pode reduzir tempo e custos para traçar políticas ou estratégias que reduzam o nível de inadimplência dentro da instituição credora, através da utilização de técnicas de aprendizado de máquina.

Para trabalhos futuros, exploraremos o refinamento de hiper-parâmetros e da

arquitetura da rede MLP e dos métodos *ensemble*, a fim de encontrar a melhor estrutura e configuração dos métodos para o problema descrito, bem como, implementação de métodos que tratem os problemas de desbalanceamento da base de dados. Adicionalmente, estenderemos os experimentos, adicionando outros classificadores baseados em aprendizado profundo (em inglês, *Deep Learning*) e bases de dados do mesmo domínio, além de implementar testes estatísticos para fundamentar melhor as comparações.

## AGRADECIMENTOS

Os autores agradecem à UFVJM por todo o suporte prestado no desenvolvimento deste trabalho.

## REFERÊNCIAS

1. E. P. Lemos, M. T. A. Steiner, and J. C. Nievola, "Análise de crédito bancário por meio de redes neurais e árvores de decisão: uma aplicação simples de data mining," *Revista de Administração-RAUSP*, vol. 40, no. 3, pp. 225–234, 2005.
2. G. A. S. Brito, A. Assaf Neto, and L. J. Corrar, "Sistema de classificação de risco de crédito: uma aplicação a companhias abertas no brasil," *Revista contabilidade & finanças*, vol. 20, pp. 28–43, 2009.
3. E. B. Gonçalves, "Análise de risco de crédito com o uso de modelos de regressão logística, redes neurais e algoritmos genéticos," Ph.D. dissertation, Universidade de São Paulo, 2005.
4. R. H. de Andrade, "Avaliação de risco de crédito utilizando grupo de classificadores," Master's thesis, Universidade Federal de Minas Gerais, 2008.
5. F. Shen, X. Zhao, Z. Li, K. Li, and Z. Meng, "A novel ensemble classification model based on neural networks and a classifier optimisation technique for imbalanced credit risk evaluation," *Physica A: Statistical Mechanics and its Applications*, vol. 526, p. 121073, 2019.
6. X. Han, R. Cui, Y. Lan, Y. Kang, J. Deng, and N. Jia, "A gaussian mixture model based combined resampling algorithm for classification of imbalanced credit data sets," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 12, pp. 3687–3699, 2019.
7. M. Mukid, T. Widiari, A. Rusgijono, and A. Prahutama, "Credit scoring analysis using weighted k nearest neighbor," in *Journal of Physics: Conference Series*, vol. 1025, no. 1. IOP Publishing, 2018, p. 012114.
8. D. A. V. de Paula, R. Artes, F. Ayres, and A. M. A. F. Minardi, "Estimating credit and profit scoring of a brazilian credit union with logistic regression and machine-learning techniques," *RAUSP Management Journal*, 2019.
9. Z. Zhang, J. He, G. Gao, and Y. Tian, "Sparse multi-criteria optimization classifier for credit risk evaluation," *Soft Computing*, vol. 23, no. 9, pp. 3053–3066, 2019.
10. L. Nanni and A. Lumini, "An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring," *Expert systems with applications*, vol. 36, no. 2, pp. 3028–3033, 2009.
11. C.-F. Tsai and J.-W. Wu, "Using neural network ensembles for bankruptcy prediction and credit scoring," *Expert systems with applications*, vol. 34, no. 4, pp. 2639–2649, 2008.

12. P. Lahoti, K. P. Gummadi, and G. Weikum, "ifair: Learning individually fair data representations for algorithmic decision making," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 1334–1345.
13. A. McCallum, K. Nigam et al., "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, vol. 752, no. 1. Citeseer, 1998, pp. 41–48.
14. M. Svensén and C. M. Bishop, "Pattern recognition and machine learning," 2007.
15. R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *the Journal of machine Learning research*, vol. 9, pp. 1871–1874, 2008.
16. L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Routledge, 2017.
17. L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.
18. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, pp. 3146–3154, 2017.
19. T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.
20. D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.
21. S. Haykin, *Redes neurais: princípios e prática*. Bookman Editora, 2007.
22. F. N. Bonifácio, "Comparação entre as redes neurais artificiais mlp, rbf e lvq na classificação de dados," *Paraná: Universidade Estadual do Oeste do Paraná*, 2010.
23. E. Sobrinho, J. Araújo, L. A. Guedes, and R. Francês, "Descoberta de conhecimento em uma base de dados de bilhetes de tarifaç o: Estudo de caso em telefonia celular," 2005.
24. H. P. Rocha and A. P. Braga, "Seleç o clonal de caracter sticas rankeadas por filtros univariados para classificaç o de tipos de leucemia aguda," *Proceedings Semin rio Interno da disciplina de T cnicas Cl ssicas de Reconhecimento de Padr es*, p. 64, 2010.
25. M. A. C. Pacheco et al., "Algoritmos gen ticos: princ pios e aplicaç es," *ICA: Laborat rio de Intelig ncia Computacional Aplicada. Departamento de Engenharia El trica. Pontif cia Universidade Cat lica do Rio de Janeiro. Fonte desconhecida*, p. 28, 1999.
26. D. Dua and C. Graff, "*UCI reposit rio de aprendizado de m quina*," 2017. [Online]. Available: <http://archive.ics.uci.edu/ml>
27. H. Hofmann, "German credit data," *UCI Reposit rio de aprendizado de m quina*, 2000.
28. G. L. PAPP, "Seleç o de atributos utilizando algoritmos gen ticos multiobjetivos," Ph.D. dissertation, Pontif cia Universidade Cat lica do Paran , 2002.
29. S. Y. Sohn and J. W. Kim, "Decision tree-based technology credit scoring for start-up firms: Korean case," *Expert Systems with Applications*, vol. 39, no. 4, pp. 4007–4012, 2012.
30. D. Zhang, X. Zhou, S. C. Leung, and J. Zheng, "Vertical bagging decision trees model for credit scoring," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7838–7843, 2010.
31. G. Wang and J. Ma, "Study of corporate credit risk prediction based on integrating boosting and random subspace," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13 871–13 878, 2011.

## FLUXO DE CARGA LINEARIZADO – UM ESTUDO COMPARATIVO USANDO A LINGUAGEM AMPL

*Data de aceite: 01/03/2022*

*Data da submissão: 06/01/2022*

### **Hugo Andrés Ruiz Flórez**

Universidade Tecnológica Federal do Paraná  
UTFPR  
Medianeira – PR  
<http://lattes.cnpq.br/1231733785708599>

### **Gloria Patricia Lopez Sepulveda**

Faculdade de Ensino Superior de São Miguel  
do Iguaçu Uniguaçu  
São Miguel do Iguaçu – PR  
<http://lattes.cnpq.br/8402032745107959>

### **Jose Airton Azevedo dos Santos**

Universidade Tecnológica Federal do Paraná  
UTFPR  
Medianeira – PR  
<http://lattes.cnpq.br/1208427854093144>

### **Cristiane Lionço de Oliveira**

Universidade Tecnológica Federal do Paraná  
UTFPR  
Medianeira – PR  
<http://lattes.cnpq.br/0915594224998333>

### **Leandro Antonio Pasa**

Universidade Tecnológica Federal do Paraná  
UTFPR  
Medianeira – PR  
<http://lattes.cnpq.br/4095187669845473>

**RESUMO:** Existem diferentes formulações propostas para resolver o problema de fluxo de carga nos Sistemas Elétricos de Potência,

dentre as quais é possível destacar o fluxo de carga CA e o fluxo de carga linearizado. A formulação do fluxo de carga CA é a mais usada para determinar com exatidão o ponto de operação de um Sistema Elétrico de Potência em regime permanente. Enquanto, a formulação do fluxo de carga linearizado permite calcular o ponto de operação de uma forma mais simples e aproximada. Geralmente, para realizar análises de fluxo de carga CA ou fluxo de carga linearizado é necessário realizar a implementação computacional em uma linguagem de programação específica ou podem ser usados softwares comerciais desenvolvidos exclusivamente para realizar estudos em Sistemas Elétricos de Potência. No entanto, existe uma alternativa de solução recentemente usada que consiste em representar o problema de fluxo de carga como sendo um problema de programação matemática. Esta alternativa de solução permite resolver o problema de fluxo de carga usando softwares de modelagem matemática amplamente usados nas análises de Sistemas de Elétricos de Potência, como é o caso da linguagem AMPL. Portanto, neste trabalho será usada a linguagem AMPL junto com o solver Knitro para resolver o problema de fluxo de carga linearizado. São apresentados dois modelos de programação matemática do fluxo de carga linearizado. O primeiro desconsidera as perdas de potência ativa e o outro incorpora as perdas de potência ativa na formulação matemática. Desta forma, será considerado um estudo de caso que consiste na determinação do ponto de operação de um sistema-teste. Finalmente, será realizada uma comparação dos resultados

obtidos pelos modelos de programação matemática do fluxo de carga linearizado com os resultados obtidos através do modelo de programação matemática do fluxo de carga CA. Os resultados obtidos mostram que a linguagem de programação AMPL permite solucionar satisfatoriamente o problema de fluxo de carga em Sistemas Elétricos de Potência.

**PALAVRAS-CHAVE:** Sistemas Elétricos de Potência, Fluxo de Carga Linearizado, Otimização Matemática, AMPL, Solver Knitro.

## LINEARIZED LOAD FLOW – A COMPATATIVE STUDY USING THE AMPL LANGUAGE

**ABSTRACT:** There are different ways to solve the power flow problem in Electrical Power Systems, among which stand out: the AC power flow and the linearized power flow. The AC power flow equation is the most used to accurately estimate the power system operation point in steady-state. Meanwhile, the linearized power flow formulation allows to calculate the power system operation point in a quick and simple way. In general, to perform AC power flow or linearized power flow analysis it is necessary to carry out the computational implementation in a specific programming language or can be used commercial softwares developed to perform Electrical Power System analysis. However, there is an alternative solution that has been recently used to solves the load flow problem as a mathematical optimization problem. This alternative approach allows to solve the load flow problem using mathematical optimization softwares widely used in the Electrical Power System analysis, such as the AMPL language. Therefore, in this work the AMPL language will be used together with the Knitro solver to obtain the linearized power flow solution. For this purpose, two mathematical programming models to solve the linearized power flow are presented. The first model ignores the active power losses and the second one includes the active power losses in the mathematical formulation. Thus, a case study that consists in determining the operating point of a test-system is considered. Finally, a comparison among the results obtained by the linearized power flow mathematical models and by the AC power flow mathematical model is carried out. The results show that the AMPL language can be used to successfully solve the power flow problem in Electrical Power Systems.

**KEY-WORDS:** Electrical Power System, Linearized Load flow, Mathematical Optimization, AMPL, Knitro Solver.

## 1 | INTRODUCCIÓN

O fluxo de carga é uma das principais ferramentas usadas para determinar o ponto de operação de um Sistema Elétrico de Potência (SEP) em regime permanente. Seu objetivo consiste em calcular os valores das variáveis de estado (módulos e ângulos das tensões), a partir dos valores dos componentes e dos valores de demanda de um SEP, de tal forma que, possa ser verificado se as grandezas do SEP operam dentro dos valores operativos permitidos (STEVENSON, 1994).

Na literatura é possível encontrar diferentes formulações para resolver o problema de fluxo de carga em SEP, dentre as quais é possível destacar o fluxo de carga AC e o fluxo

de carga linearizado (MONTICELLI, 1983).

O fluxo de carga AC é resolvido tradicionalmente através do método de Newton-Raphson e as suas versões desacopladas. Este método obtém os valores das variáveis de estado de forma iterativa através da resolução de um conjunto de equações não-lineares. Por sua vez, o fluxo de carga linearizado (também conhecido como fluxo de carga cc) surgiu como uma alternativa de resolver o problema de fluxo de carga CA de uma forma simplificada, uma vez que, **é possível achar** de forma aproximada os valores das variáveis de estado ao resolver um sistema linear de equações (GLOVER, 2011). Em suma, a principal diferença entre os dois métodos radica no tipo de análise que pretende ser realizado, já que, enquanto no fluxo de carga AC é possível analisar o comportamento das variáveis nodais para um cenário específico, no método linearizado só é possível analisar o comportamento das variáveis de forma aproximada.

Dada a sua importância na área dos SEP é possível encontrar alguns softwares tais como ANAREDE, PowerWorld, Digsilent, PSAT, entre outros, que permitem achar a solução do problema de fluxo de carga AC e/ou linearizado (TAMASHIRO, 2016). No entanto, existe uma alternativa de solução recentemente explorada que consiste na possibilidade de representar o problema de fluxo de carga como sendo um problema de programação matemática (FLOREZ, et. al 2021). A principal vantagem desta abordagem, diferente dos softwares comerciais usados para resolver o problema de fluxo de carga, é a possibilidade de poder realizar alterações no modelo de otimização matemática, de tal forma que, seja possível resolver outros problemas de otimização matemática dentro da área dos SEP. Adicionalmente, é importante salientar que existem softwares de modelagem algébrica amplamente usados na área dos SEP, tais como, GAMS, Lingo, Pyomo, AMPL, entre outros, que podem ser usados para resolver os modelos de programação matemática propostos para resolver o problema de fluxo de carga.

Neste trabalho, são apresentados dois modelos matemáticos para obter a resposta do problema do fluxo de carga linearizado (desconsiderando e considerando as perdas de potência). Para determinar a solução dos modelos matemáticos de otimização será usada a linguagem de programação AMPL junto com o solver Knitro (FOURIER, 2003). Será usado um sistema-teste de 5 barras como estudo de caso (GLOVER, 2011). Além disso, será realizada uma comparação dos resultados obtidos com os modelos matemáticos de otimização do problema de fluxo de carga linearizado com os resultados obtidos através do modelo matemático de otimização do fluxo de carga convencional. Os resultados mostram que a linguagem AMPL é uma ferramenta confiável para realizar estudos de fluxo de carga em SEP.

## 2 | MODELO MATEMÁTICO FLUXO DE CARGA AC

O problema de fluxo de carga pode ser representado através do seguinte modelo de

otimização não-linear (FLOREZ et. al, 2021):

$$\min P_L = \sum_{(i,j) \in \Omega L} g_{ij} \left( t_{ij}^2 V_i^2 + V_j^2 - 2t_{ij} V_i V_j \cos(\theta_{ij} + \varphi_{ij}) \right) \quad (1)$$

s. a.

$$P_i^G - P_i^D - \sum_{(i,j) \in \Omega L} P_{ij} = 0, \quad \forall_i \in \Omega B \quad (2)$$

$$Q_i^G - Q_i^D + Q_i^{sh} - \sum_{(i,j) \in \Omega L} Q_{ij} = 0 \quad \forall_i \in \Omega B \quad (3)$$

$$\underline{Q_i^G} \leq Q_i^G \leq \overline{Q_i^G} \quad \forall_i \in \Omega B \quad (4)$$

Sendo que:

- A função objetivo mostrada em (1) representa as perdas de potência ativa do SEP.
- As restrições (2) e (3) representam o balanço de potência ativa e reativa do SEP, onde os fluxos de potências são calculados da seguinte forma:

$$P_{ij} = (t_{ij} V_i)^2 g_{ij} - t_{ij} V_i V_j \left( g_{ij} \cos(\theta_{ij} + \varphi_{ij}) + b_{ij} \sin(\theta_{ij} + \varphi_{ij}) \right) \quad (5)$$

$$Q_{ij} = -V_i^2 (t_{ij}^2 b_{ij} + b_{ij}^{sh}) + t_{ij} V_i V_j \left( b_{ij} \cos(\theta_{ij} + \varphi_{ij}) - g_{ij} \sin(\theta_{ij} + \varphi_{ij}) \right) \quad (6)$$

$$P_{ji} = V_j^2 g_{ij} - t_{ij} V_i V_j \left( g_{ij} \cos(\theta_{ij} + \varphi_{ij}) - b_{ij} \sin(\theta_{ij} + \varphi_{ij}) \right) \quad (7)$$

$$Q_{ji} = -V_j^2 (b_{ij} + b_{ij}^{sh}) + t_{ij} V_i V_j \left( b_{ij} \cos(\theta_{ij} + \varphi_{ij}) + g_{ij} \sin(\theta_{ij} + \varphi_{ij}) \right) \quad (8)$$

A restrição (4) corresponde os limites de geração de potência reativa nas barras do SEP.

Ao solucionar o modelo matemático anterior é possível obter diretamente os valores de  $V_i$  e  $\theta_i$  em todas as barras do SEP, assim como, os fluxos de potência e nas Linhas de Transmissão e as perdas de potência ativa do SEP.

### 31 MODELO MATEMÁTICO FLUXO DE CARGA LINEARIZADO DESCONSIDERANDO AS PERDAS DE POTÊNCIA NAS LINHAS DE TRANSMISSÃO

De acordo com Monticelli (1983), faz-se necessário incorporar as seguintes aproximações para linearizar a formulação do problema de fluxo de carga:

- Considerando que nos parâmetros série das Linhas de Transmissão se cumpre que  $r_{ij} \ll x_{ij}$ , pode ser considerado que  $r_{ij} = 0$ . Portanto,  $g_{ij} = 0$  e  $b_{ij} = -1/x_{ij}$ . Adicionalmente, os parâmetros shunt das Linhas de Transmissão também são desconsiderados.
- Assume-se que os módulos das tensões nas barras do SEP têm valores unitários

$$(V_i \approx V_j \approx 1).$$

- Dado que as aberturas angulares  $\theta_{ij}$  são pequenas, pode-se considerar que (com base na formulação no fluxo de carga AC)  $\cos(\theta_{ij} + \varphi_{ij}) \approx 1$  e  $\text{sen}(\theta_{ij} + \varphi_{ij}) \approx \theta_{ij} + \varphi_{ij}$ .

Baseado nas aproximações expostas anteriormente, a solução do fluxo de carga linearizado pode ser representado através do seguinte modelo de otimização:

$$\min P_L = \sum_{(i,j) \in \Omega_L} (P_{ij} + P_{ji}) \quad (9)$$

$$s. a. \quad (10)$$

$$P_i^G - P_i^D - \sum_{(i,j) \in \Omega_L} P_{ij} = 0, \quad \forall i \in \Omega_B$$

Sendo que:

- A função objetivo mostrada em (9) representa as perdas de potência ativa do SEP (que neste caso seriam nulas).
- A restrição mostrada em (10) representa o balanço de potência ativa nas barras do SEP, onde os fluxos de potência ativa são calculados como:

$$P_{ij} = -t_{ij} b_{ij} (\theta_{ij} + \varphi_{ij}) \quad (11)$$

$$P_{ji} = t_{ij} b_{ij} (\theta_{ij} + \varphi_{ij}) \quad (12)$$

Ao solucionar o modelo matemático anterior é possível obter diretamente os valores de  $\theta_i$  em todas as barras do SEP, assim como, os fluxos de potência ativa  $P_{ij}$  nas Linhas de Transmissão.

#### 4 | MODELO MATEMÁTICO FLUXO DE CARGA LINEARIZADO CONSIDERANDO AS PERDAS DE POTÊNCIA NAS LINHAS DE TRANSMISSÃO

De acordo com Monticelli (1983) é possível incorporar as perdas de potência ativa na formulação do fluxo de carga linearizado. Desta forma, é possível melhorar os resultados do fluxo de carga linearizado quando comparados com os resultados obtidos pela formulação AC do problema de fluxo de carga.

Assim sendo, para obter a formulação desta versão do fluxo de carga linearizado é necessário considerar as seguintes aproximações:

- O parâmetro série  $r_{ij}$  das Linhas de Transmissão é considerado na formulação. Portanto, define-se  $g_{ij} = r_{ij}/(r_{ij}^2 + x_{ij}^2)$ . No entanto, o parâmetro  $b_{ij}$  continua sendo calculado de forma aproximada ( $b_{ij} \approx -1/x_{ij}$ ) e os parâmetros shunt são desconsiderados.
- Baseado na expansão em séries de Taylor das funções  $\text{sen}(\theta)$  e  $\text{cos}(\theta)$ , então pode

ser considerado que  $\cos(\theta_{ij} + \varphi_{ij}) \approx 1 - (\theta_{ij} + \varphi_{ij})^2/2$  e  $\text{sen}(\theta_{ij} + \varphi_{ij}) \approx \theta_{ij} + \varphi_{ij}$ .

A partir das aproximações anteriores e com base nos modelos de otimização matemática apresentados acima, a solução do fluxo de carga linearizado considerando as perdas de potência ativa pode ser obtida mediante o seguinte modelo matemático de otimização:

$$\min P_L = \sum_{(i,j) \in \Omega L} g_{ij} \left( t_{ij}^2 - 2t_{ij} \left( 1 - \frac{(\theta_{ij} + \varphi_{ij})^2}{2} \right) + 1 \right) \quad (13)$$

s. a.

$$P_i^G - P_i^D - \sum_{(i,j) \in \Omega L} P_{ij} = 0, \quad \forall_i \in \Omega B \quad (14)$$

Sendo que:

- A função objetivo mostrada em (13) representa as perdas de potência ativa do SEP.
- A restrição (13) representa o balanço de potência ativas nas barras do SEP, onde os fluxos de potência ativa são calculados como:

$$P_{ij} = t_{ij}^2 g_{ij} - t_{ij} g_{ij} \left( 1 - \frac{(\theta_{ij} + \varphi_{ij})^2}{2} \right) - t_{ij} b_{ij} (\theta_{ij} + \varphi_{ij}) \quad (15)$$

$$P_{ji} = g_{ij} - t_{ij} g_{ij} \left( 1 - \frac{(\theta_{ij} + \varphi_{ij})^2}{2} \right) + t_{ij} b_{ij} (\theta_{ij} + \varphi_{ij}) \quad (16)$$

Ao solucionar o modelo matemático anterior é possível obter diretamente os valores de  $\theta_i$  em todas as barras do SEP, assim como, os fluxos de potência ativa  $P_{ij}$  nas Linhas de Transmissão e as perdas de potência ativa do SEP.

## 5 | ESTUDO DE CASO PROPOSTO

Para realizar o estudo comparativo dos modelos de programação matemática do problema de fluxo de carga linearizado e do modelo de programação matemática do fluxo de carga CA será usado como referência o sistema-teste da Figura 1 (GLOVER, 2011).

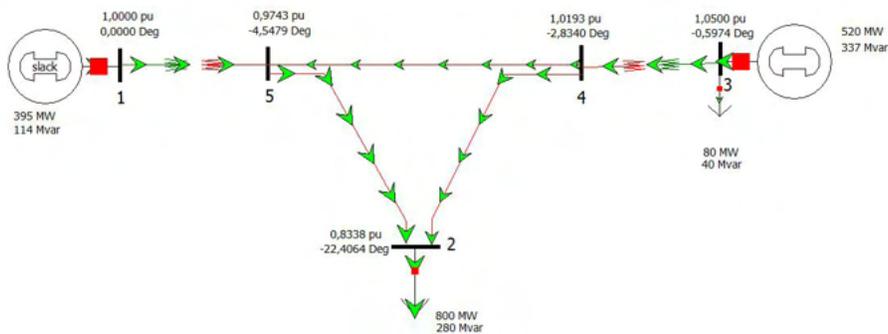


Figura 1 – Sistema-teste usado na validação dos modelos de otimização propostos.

Fonte: Autoria própria.

Os dados de entrada do sistema-teste necessários para realizar as simulações são apresentados nas Tabelas 1 e 2.

$i$	Tipo	$P_i^{Gesp}$	$Q_i^{Gesp}$	$P_i^{Desp}$	$Q_i^{Desp}$	$V_i^{esp}$	$\theta_i^{esp}$	$Q_i^{Gmin}$	$Q_i^{Gmax}$	$B_i^{sh}$
1	$V\theta$	--	--	0,00	0,00	1,00	0,00	--	--	0,00
2	$PQ$	0,00	0,00	8,00	2,80	--	--	--	--	0,00
3	$PV$	5,20	--	0,80	0,40	1,05	--	-2,80	4,00	0,00
4	$PQ$	0,00	0,00	0,00	0,00	--	--	--	--	0,00
5	$PQ$	0,00	0,00	0,00	0,00	--	--	--	--	0,00

Tabela 1 – Dados de entrada referente às barras do sistema-teste.

Fonte: Modificado de Glover, 2011.

$i$	$j$	$r_{ij}$	$x_{ij}$	$b_{ij}$	$t_{ij}$
1	5	0,00150	0,0200	0,0000	1,0000
3	4	0,00075	0,0100	0,0000	1,0000
2	4	0,00900	0,1000	1,7200	1,0000
2	5	1,00450	0,0500	0,8800	1,0000
4	5	0,00225	0,0250	0,4400	1,0000

Tabela 2 – Dados de entrada referente às LT e transformadores do sistema-teste.

Fonte: Modificado de Glover, 2011.

Por simplicidade, as informações anteriores podem ser inseridas em um único arquivo de texto comum para os três modelos matemáticos, como mostrado na Figura 2.

```

param: OB: TB  PG_esp  QG_esp  PD_esp  QD_esp  V_esp      Th_esp  Qmin  Qmax  Bsh:=
      1  1  0      0      0      0      1      0      -999  999  0
      2  0  0      0      8      2.8    1      0      0      0  0
      3  2  5.2    0      0.8    0.4    1.05    0      -2.8  4.0  0
      4  0  0      0      0      0      1      0      0      0  0
      5  0  0      0      0      0      1      0      0      0  0
;

param: OL: r      x      bsh      t      phi:=
      1  5  0.00150  0.0200  0.0000  1.0000  0.0000
      3  4  0.00075  0.0100  0.0000  1.0000  0.0000
      2  4  0.00900  0.1000  0.8600  1.0000  0.0000
      2  5  0.00450  0.0500  0.4400  1.0000  0.0000
      4  5  0.00225  0.0250  0.2200  1.0000  0.0000
;

```

Figura 2 – Dados de entrada do sistema-teste.

Fonte: Próprio autor.

Por outro lado, para solucionar os modelos de otimização matemática do problema de fluxo de carga através da linguagem AMPL é usado o solver Knitro amplamente usado em aplicações na área dos SEP.

Com o intuito de fazer uma comparação do desempenho do solver Knitro usado na resolução dos modelos de programação matemática do problema de fluxo de carga, na Tabela 3 é mostrado um resumo dos valores da função objetivo, o erro relativo, o número de iterações e o tempo de cômputo obtidos após realizar as simulações.

	<i>Fluxo CA</i>	<i>Fluxo linearizado (sem perdas)</i>	<i>Fluxo linearizado (com perdas)</i>
<i>Valor da função</i>			
<i>objetivo</i>	0,3484	$2,2204e^{-16}$	0,2366
<i>(Perdas)</i>			
<i>Erro relativo</i>	$6,43e^{-9}$	$7,11e^{-15}$	$2,41e^{-12}$
<i>Número de iterações</i>	6	2	3
<i>Tempo [s]</i>	0,004	0,003	0,003

Tabela 3 – Comparação do desempenho do solver Knitro.

Fonte: Próprio autor.

Salienta-se que, os valores da função objetivo mostrados na Tabela 3 correspondem às perdas de potência ativa do sistema-teste. Adicionalmente, nas Figuras 3-5 são mostrados os resultados obtidos nas simulações realizadas usando a linguagem de programação AMPL.

```

Locally optimal solution.
objective 0.3483875784; feasibility error 1.04e-14
6 iterations; 7 function evaluations
*****
***** Resultados do fluxo de carga (p.u) *****
*****
i      TB      PGi      QGi      PDi      QDi      Pi      Qi      Ui      Thi
1      1      3.9484    1.1428    0.0000    0.0000    3.9484    1.1428    1.0000    0.0000
2      0      0.0000    0.0000    0.0000    2.8000    -8.0000    -2.8000    0.8338    -22.4064
3      2      5.2000    3.3748    0.8000    0.4000    4.4000    2.9748    1.0500    -0.5973
4      0      0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    1.0193    -2.8340
5      0      0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.9743    -4.5479

i      j      Pij      Qij      Pji      Qji      PLij      QLij
1      5      3.9484    1.1428    -3.9220    -0.8049    0.0253    0.3379
3      4      4.4000    2.9748    -4.3888    -2.7189    0.0192    0.2559
2      4      -2.9184    -1.3911    3.0368    1.2154    0.1184    1.3157
2      5      -5.0816    -1.4089    5.2566    2.6302    0.1750    1.9448
4      5      1.3440    1.5035    -1.3336    -1.8253    0.0104    0.1157

Perdas do sistema (Soma das perdas nos trechos): Ploss= 0.3484   Qloss= 3.9699
Perdas do sistema (PG menos PD do sistema):      Ploss= 0.3484   Qloss= 3.9699

```

Figura 3 – Resultados do modelo matemático do fluxo de carga CA.

Fonte: Próprio autor.

```

Locally optimal solution.
objective 2.220446049e-16; feasibility error 8.88e-16
2 iterations; 3 function evaluations
*****
***** Resultados do fluxo de carga (p.u) *****
*****
i      TB      PGi      PDi      Pi      Ui      Thi
1      1      3.6000    0.0000    3.6000    1.0000    0.0000
2      0      0.0000    0.0000    -8.0000    1.0000    -18.6948
3      2      5.2000    0.8000    4.4000    1.0000    0.5238
4      0      0.0000    0.0000    0.0000    1.0000    -1.9972
5      0      0.0000    0.0000    0.0000    1.0000    -4.1253

i      j      Pij      Pji      PLij
1      5      3.6000    -3.6000    0.0000
3      4      4.4000    -4.4000    0.0000
2      4      -2.9143    2.9143    0.0000
2      5      -5.0857    5.0857    0.0000
4      5      1.4857    -1.4857    0.0000

Perdas do sistema (Soma das perdas nos trechos): Ploss= 0.0000
Perdas do sistema (PG menos PD do sistema):      Ploss= 0.0000

```

Figura 4 – Resultados do modelo matemático do fluxo de carga linearizado sem perdas.

Fonte: Próprio autor.

```

Locally optimal solution.
objective 0.2365522991; feasibility error 5.06e-12
3 iterations; 4 function evaluations

***** Resultados do fluxo de carga (p.u) *****

i      TB      PGi      PDi      Pi      Ui      Thi
1      1      3.8366   0.0000   3.8366   1.0000   0.0000
2      0      0.0000   8.0000   -8.0000   1.0000  -19.1763
3      2      5.2000   0.8000   4.4000   1.0000   0.1535
4      0      0.0000   0.0000   0.0000   1.0000  -2.3634
5      0      0.0000   0.0000   0.0000   1.0000  -4.3839

i      j      Pij      Pji      PLij
1      5      3.8366  -3.8147   0.0218
3      4      4.4000  -4.3856   0.0144
2      4      -2.8960   2.9728   0.0769
2      5      -5.1040   5.2230   0.1190
4      5      1.4128  -1.4083   0.0044

Perdas do sistema (Soma das perdas nos trechos): Ploss= 0.2366
Perdas do sistema (PG menos PD do sistema):      Ploss= 0.2366

```

Figura 5 – Resultados do modelo matemático do fluxo de carga linearizado com perdas.

Fonte: Próprio autor.

Adicionalmente, nas Tabelas 4 e 5 são comparados os valores das tensões nas barras e os valores dos fluxos de potência obtidos através do fluxo de carga linearizado e os resultados obtidos através do fluxo de carga CA, respectivamente.

Barra	Fluxo CA		Fluxo Linearizado (sem perdas)		Fluxo Linearizado (com perdas)	
	$V_i$ [p.u.]	$\theta_i$ [graus]	$V_i$ [p.u.]	$\theta_i$ [graus]	$V_i$ [p.u.]	$\theta_i$ [graus]
1	1,0000	0,0000	1,0000	0,0000	1,0000	0,0000
2	0,8338	-22,4064	1,0000	-18,6948	1,0000	-19,1763
3	1,0500	-0,5973	1,0000	0,5238	1,0000	0,1535
4	1,0193	-2,8340	1,0000	-1,9972	1,0000	-2,3634
5	0,9743	-4,5479	1,0000	-4,1253	1,0000	-4,3839

Tabela 4 – Comparação dos valores das variáveis de estado do sistema-teste.

Fonte: Próprio autor.

<i>i</i>	<i>j</i>	<i>Fluxo AC</i>		<i>Fluxo linearizado (sem perdas)</i>		<i>Fluxo linearizado (com perdas)</i>	
		<i>P<sub>ij</sub></i>	<i>P<sub>ji</sub></i>	<i>P<sub>ij</sub></i>	<i>P<sub>ji</sub></i>	<i>P<sub>ij</sub></i>	<i>P<sub>ji</sub></i>
1	5	3,9484	-3,9230	3,6000	-3,6000	3,8366	-3,8147
3	4	4,4000	-4,3808	4,4000	-4,4000	4,4000	-4,3856
2	4	-2,9184	3,0368	-2,9143	2,9143	-2,8960	2,9728
2	5	-5,0816	5,2566	-5,0857	5,0857	-5,1040	5,2230
4	5	1,3440	-1,3336	1,4857	-1,4857	1,4128	-1,4083

Tabela 5 – Comparação dos valores dos fluxos de potência ativa nas Linhas de Transmissão.  
Fonte: Próprio autor.

A partir dos resultados anteriores e tomando como referência os resultados do fluxo de carga CA é possível mensurar o impacto das aproximações incorporadas nos modelos de fluxo de carga linearizados.

## 6 I CONCLUSÃO

Neste trabalho os resultados obtidos mostram que é possível representar o fluxo de carga linearizado considerando e desconsiderando as perdas de potência ativa através de um modelo de programação matemática, respectivamente. Estes modelos matemáticos do problema de fluxo de carga são implementados na linguagem de programação matemática AMPL e resolvidos satisfatoriamente através do solver Knitro.

O modelo matemático de otimização do fluxo de carga linearizado que desconsidera as perdas de potência ativa corresponde a um modelo de programação linear, o que poderia ser considerado como uma condição favorável na hora de resolver um problema de otimização matemática de maior complexidade usando essa formulação. No entanto, pode ser visto que há uma diferença considerável entre os valores das grandezas quando comparados com os valores obtidos pelo modelo do fluxo de carga CA.

Por sua vez, o modelo matemático de otimização do fluxo de carga linearizado que considera as perdas de potência ativa, embora apresente melhores resultados do que o modelo que desconsidera as perdas de potência, pelo fato de ser um modelo de programação não-linear não seria uma vantagem o seu emprego para realizar aplicações nas análises de SEP.

De acordo com o anterior, em um trabalho futuro seria interessante propor um modelo matemático de otimização do fluxo de carga linearizado que possa ser representado como um modelo de programação linear, que incorpora as perdas de potência do sistema e cujos valores das variáveis de estado sejam bem próximos aos fornecidos pelo fluxo de carga CA.

## REFERÊNCIAS

FLOREZ, H. A. R.; MARUJO, D.; LOPEZ, G. P.; MOLINA, A. Fluxo de carga em sistemas elétricos de potência – Um estudo de caso usando a linguagem AMPL. **Coleção Desafios das Engenharias: Engenharia Elétrica**, p. 47-61, 2021.

FOURIER, R.; GAY, D. M.; KERNIGHAN, B. W. AMPL: A modeling language for mathematical programming. Thomson Publisher, 2003.

GLOVER, J. D.; SARMA, M. S.; OVERBYE, T. J. Power system analysis and design. Ed. Cengage Learning, 2011.

MONTICELLI, A. Fluxo de carga em rede de energia elétrica. São Paulo, SP: Ed. Edgard Blücher, 1983.

STEVENSON, W. D.; GRAINGER, J. J.; Power system analysis. New York, Ed. Mc Graw Hill, 1994.

TAMASHIRO, M. A.; GUIMARÃES, G. C.; RODRIGUES, A. R.; SILVA, A. M. B.; CAIXETA, D. A.; MONTEIRO, V. A. Features of present computer tools for load flow calculation and experience acquired with the Improvement of UFUFlow program. **IEEE Latin America Transactions**, v. 14, n. 4, 2016.

## IMPLEMENTAÇÃO DE REDE NEURAL CONVOLUCIONAL PARA PREDIÇÃO DE COVID-19 ATRAVÉS DE IMAGENS DE RAIOS X

*Data de aceite: 01/03/2022*

### **Erik Gabriel Cruz Sena**

Graduando em Engenharia Mecânica Instituto de Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Diamantina Minas Gerais

### **Honovan Paz Rocha**

Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Janáuba Minas Gerais

**RESUMO:** A COVID-19 é uma das grandes ameaças do século, desta forma, tornou-se uma necessidade prevenir a propagação do vírus para diminuir a carga nos leitos de UTI. Desta maneira, torna-se relevante o desenvolvimento de ferramentas para auxílio ao diagnóstico de casos positivos da doença. Neste contexto, este trabalho tem como objetivo o estudo, implementação e ajuste de uma Rede Neural Convolutiva para classificação de pacientes com a doença. O modelo implementado foi inicialmente aplicado ao banco de dados MNIST com o propósito de ajustar e validar a arquitetura da rede, em seguida, foi ajustado e aplicado a uma base de dados composta por imagens de radiografia da região torácica de pacientes saudáveis e com COVID-19. Os experimentos realizados mostraram que a abordagem proposta obteve desempenho superior a alguns métodos do estado da arte, classificando corretamente

mais de 97% dos pacientes com COVID-19. Os resultados demonstram que a abordagem proposta gera boa separabilidade entre classes, podendo ser usado como ferramenta de auxílio ao diagnóstico.

**PALAVRAS-CHAVE:** Redes Neurais Convolutivas, Reconhecimento de Padrões, COVID-19, Imagens de raios X.

## 1 | INTRODUÇÃO

Segundo [1], as espécies de coronavírus são um grupo de vírus de RNA altamente diversos, envelopados, de sentido positivo e de fita simples. Eles causam várias doenças envolvendo os sistemas respiratório, entérico, hepático e neurológico. Considerando-se os sintomas físicos, envolvem frequentemente tosse, febre e dificuldades respiratórias [2].

Com a rápida escalada da doença a nível global, a COVID-19 logo se tornou uma pandemia. Alguns países têm adotado medidas, tais como isolamento de casos suspeitos, fechamento de escolas e universidades, distanciamento social e lockdown a fim de reduzir os impactos da pandemia, diminuindo o pico de incidência e o número de mortes [3].

Dessa forma, considerando a grande disponibilidade de X da região torácica de pacientes com suspeita de COVID-19, alavancou pesquisas na área de aprendizado de máquina aplicada, destacando-se algumas técnicas com casos recentes de sucesso no

campo de reconhecimento/classificação de padrões em imagens. Dentre estas técnicas estão as conhecidas como Redes Neurais Profundas (em inglês, *Deep Neural Networks* DNN), Transferência de Aprendizado (em inglês, *Transfer Learning*) e Redes Neurais Convolucionais (em inglês, *Convolutional Neural Networks* CNN).

As CNN's, com origens no final da década de 1970 através do trabalho pioneiro de Kunihiko Fukushima [4] e consolidadas pelo trabalho de *Yann Lecun* [5] em 1998, é um tipo de modelo de aprendizado profundo que se tornou dominante em tarefas de visão computacional. O que torna as CNN's eficientes são os filtros convolucionais capazes de extrair características importantes para classificação, além de levarem em consideração a estrutura espacial das imagens [6].

Nesse contexto, este trabalho tem o objetivo o estudo, implementação e ajuste de uma CNN para classificação de pacientes com COVID-19. A arquitetura da CNN foi ajustada e validada utilizando-se uma base de dados de benchmark, em seguida o modelo foi aplicado a uma base de dados contendo imagens de raio X da região torácica de pacientes, com o objetivo de classificá-los indicando se existem ou não lesões pulmonares, e por conseguinte apontando se provavelmente estão infectados pela COVID-19.

Os resultados obtidos através dos experimentos realizados, demonstraram eficácia do modelo proposto de 97,58%, considerando-se a detecção de pacientes com a doença.

Este trabalho está organizado da seguinte maneira: A Seção II apresenta alguns trabalhos recentes relacionados à detecção de COVID-19 utilizando-se Redes Neurais. A Seção III, dados relativos à pacientes suspeitos de COVID-19, vários apresenta estudos relacionados à classificadores e aprendizado trabalhos nas áreas de Ciência dos Dados e Inteligência de máquinas. A Seção V, descreve a metodologia utilizada Artificial começaram a ser desenvolvidos com o objetivo de auxiliar no entendimento e contenção desta doença. A disponibilidade de bases de dados contendo imagens de raio para ajuste e validação da arquitetura da CNN, bem como a adequação e aplicação do modelo final para a base de dados de raios X. Ademais, a Seção VI apresenta o experimento proposto, como também os resultados e discussões relativos à aplicação da CNN à base de dados, objeto de estudo. Por fim, a Seção VII demonstra as conclusões que obtivemos com este trabalho, além de listarmos possíveis trabalhos futuros.

## 2 | TRABALHOS RELACIONADOS

Considerando-se que a detecção de COVID-19 é um assunto extremamente recente, é possível perceber uma grande quantidade de trabalhos aplicados ao tema, em parte isso é devido à emergência para estudo desta doença que demonstra letalidade e alta taxa de espalhamento global. Desta forma, elencamos alguns dos trabalhos mais recentes relacionados à aplicação de técnicas de *machine learning* à detecção de COVID-19 com base em imagens de raio X.

No trabalho de [7] propõe usar *Transfer Learning* para detecção de casos positivos de COVID-19. O aprendizado por transferência é um método de aprendizado que reutiliza uma rede pré-treinada como ponto de partida. Dito isso, o modelo foi treinado com 224 imagens positivas da doença, junto com 714 imagens de pneumonia bacteriana e viral e 504 imagens em condições normais. As imagens, por sua vez, foram redimensionadas para um tamanho de 200 x 266 pixels e treinadas em diferentes modelos de Aprendizagem por Transferência. Conclui que o melhor modelo é usando MobileNet v2, obtendo uma sensibilidade de 98,66%.

Em sua pesquisa, [8] utilizou imagens de radiografias de tórax prontamente disponíveis com achados clínicos de casos positivos de COVID-19 (154 imagens). O modelo de rede convolucional selecionado foi a ResNet-101, entretanto, esse modelo já foi pré-treinado para reconhecer objetos de outras várias imagens. Dessa forma, para completar o conjunto de testes, foi simulado um cenário cuja taxa de prevalência, calculada como o número de casos de uma doença em população durante um período de tempo, é de 2,57%. Assim, extraiu-se 5.828 imagens não rotuladas do conjunto de imagens ChestXray14. Os autores concluíram que usando a CNN pré-treinada obteve uma sensibilidade de 77,3%.

Em [9] aborda um estudo utilizando 3 modelos de CNN, Inception V3, Xception e ResNeXt a fim de selecionar a melhor dentre elas. Para análise do desempenho, foram coletadas 6.432 amostras do repositório *Kaggle*, todas elas reduzidas para um tamanho de 128 x 128 pixels. Por fim, a conclusão deste trabalho é que dos 3 modelos, a Xception tem o melhor desempenho, podendo ser usada para classificar pacientes.

Para [10], foram usados cinco técnicas de aprimoramento de imagem para detecção de COVID-19 utilizando *dataset* com 3 classes diferentes. Essas técnicas como correção gamma e complemento, tem por objetivo melhorar a percepção da patologia. Foram usadas 6 redes convolucionais para o treinamento, testando para imagens segmentadas e não segmentadas do pulmão. Finalmente, conclui que o melhor modelo é aquele que emprega a correção gamma como aprimoramento juntamente com imagem não segmentada.

### 3 | CLASSIFICADORES

Nesta seção, abordaremos os principais algoritmos para classificação implementados neste trabalho, incluindo-se algoritmos clássicos utilizados apenas para efeitos de comparação durante o processo de validação da CNN proposta.

#### A. *Multilayer Perceptron*

Segundo [11], as redes neurais artificiais (RNAs) são muito úteis quando utilizadas para a resolução de problemas, tais como, reconhecimento de padrões, classificação e previsão. Nesse contexto, a Perceptron de Múltiplas Camadas (em inglês, *Multilayer Perceptron* MLP) destaca-se pela sua popularidade e aplicabilidade. Uma rede MLP é constituída por uma camada de entrada (*input layer*), uma ou mais camadas escondidas

(hidden layer) e uma camada de saída (*output layer*).

O funcionamento de um neurônio é dado pelo ponderamento dos pesos sinápticos com um sinal recebido e o resultado dessa operação, é processado por uma função matemática chamada de função de ativação. Essas funções de ativação são essenciais, pois elas fazem transformações não-linear nos dados de entrada, tornando a rede neural capaz de aprender e executar tarefas mais complexas [12]. Dentre as funções de ativação que são usadas, destacam-se a Sigmóide, a Tangente Hiperbólica, a ReLU e a Softmax.

A função Sigmóide é amplamente utilizada, já que simula uma saída binária, mas com geração de uma não linearidade obtida através da geração de valores reais que tem limites definidos na faixa entre 0 e 1. A função sigmóide ou logística e a sua respectiva derivada estão representadas em (1).

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad \sigma'(x) = \sigma(x)(1 - \sigma(x)) \quad (1)$$

Outra importante função de ativação é a Tangente Hiperbólica. Sendo definida a partir do seno e cosseno hiperbólicos, funciona como uma versão escalada da função logística, atuando no intervalo  $[-1, 1]$ . A função e a sua respectiva derivada podem ser vistas em (2).

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad \tanh'(x) = \frac{1}{\cosh^2(x)} \quad (2)$$

A função Unidade Linear Retificada (em inglês, Rectified Linear Unit ReLU) é uma função fácil de otimizar, pelo fato de ser similar a função linear. Além disso, ela tende a ter mais efetividade comparada com as funções sigmoidais, principalmente na questão de execução. Entretanto, a sua eficácia varia com o problema. A função ReLU e a sua derivada estão representadas em (3).

$$\begin{aligned} ReLU(x) &= \max(0, x) \\ ReLU'(x) &= \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{caso contrário} \end{cases} \end{aligned} \quad (3)$$

Por fim, a função Softmax força a saída de uma rede neural a representar a probabilidade dos dados serem de uma das classes definidas. Ela não costuma ser aplicada nas camadas escondidas da rede, mas sim na camada de saída, onde realmente há interesse em gerar probabilidade para definir a classe. A função Softmax pode ser representada em (4).

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \text{ para } j = 1, \dots, K. \quad (4)$$

### B. Redes Neurais Convolucionais

No contexto de redes MLP, quando passamos a utilizar duas ou mais camadas ocultas para construção de sua arquitetura, temos o surgimento do termo aprendizado profundo (em inglês, *Deep Learning* DP). Dentre os diversos tipos de DP, considerando-se os mais conhecidos e utilizados estão as chamadas Redes Neurais Convolucionais, que trouxeram avanços em processamento de imagens, voz e vídeo. Esses avanços são obtidos através de modelos computacionais compostos por várias camadas de processamento, e isso faz com que aprendam representações de dados com múltiplos níveis de abstrações [13].

De forma geral, pode-se tentar compreender como uma rede neural convolucional funciona, através da Fig. 1. Nela podemos observar uma imagem passando por um processo de convolução, *pooling* e *fully connected*.

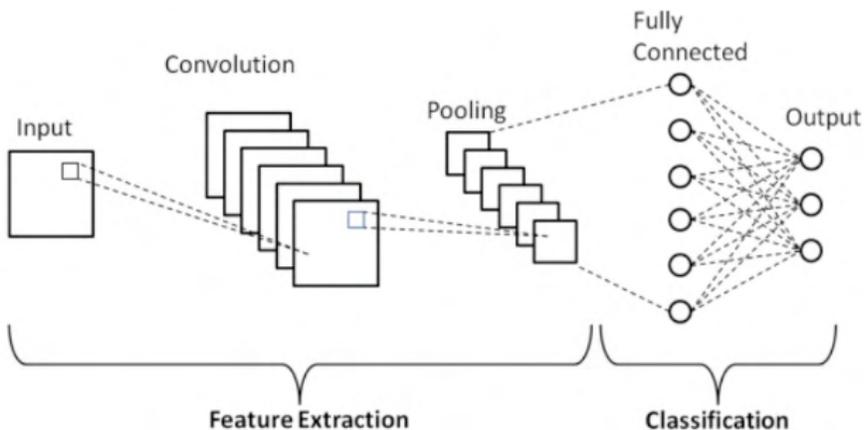


Figura 1. Exemplo de rede neural convolucional [14].

1) *Convolução*: A convolução é um processo em que utilizam-se filtros, também chamado de *kernels*, para captar características importantes para a rede. Esses filtros são formados por pesos inicializados aleatoriamente e atualizando-os através do processo de retropropagação do erro.

Além disso, segundo [15] os filtros são usados em locais diferentes da imagem. Isto acaba fazendo com que os padrões que ocorrem com frequência, situado em qualquer ponto da matriz, possam ser aprendidos. Essa característica, é denominada compartilhamento de parâmetros.

A operação de convolução, representada pelo símbolo  $*$ , é descrita a partir de (5)

$$g(i, j) = (x*k)[i, j] = \sum_m \sum_n x[m, n]k[i - m, j - n], \quad (5)$$

em que,  $g(i, j)$  é a saída ativada para a próxima camada,  $x$  é a entrada da imagem e  $k$  é a matriz de filtros.

O passo seguinte após a convolução, é a aplicação de uma função de ativação para geração de uma não linearidade na saída desta camada. Ela pode ser usada para ajustar ou cortar o resultado, sem ela o modelo se torna apenas um modelo de regressão linear.

2) *Pooling*: A camada de *pooling* nada mais é que uma camada para simplificar a informação da camada anterior, a fim de reduzir a complexidade para outras camadas. O *maxpooling* é o método mais utilizado, ele divide a imagem em uma sub-região determinada, por exemplo  $2 \times 2$ , e retorna o valor máximo dentro dessa sub-região [12]. Esse tipo de método serve para diminuir a quantidade de pesos a serem aprendidos e também para evitar *overfitting*, quando o algoritmo se ajusta bem ao conjunto de dados, mas é ruim para generalização, e alto custo operacional.

3) *Fully Connected*: Nesta etapa, a camada totalmente conectada (em inglês, *fully connected*) é semelhante como os neurônios são organizados em uma rede *Multilayer Perceptron*. Dessa forma, cada nó em uma camada totalmente conectada está diretamente conectado a todos os nós na camada anterior e na próxima.

### C. Support Vector Machines

Máquina de Suporte Vetorial (em inglês, *Support Vector Machines SVM*) é um algoritmo cujo objetivo é classificar um determinado conjunto de pontos de dados que são mapeados para um espaço de características usando uma função denominada de *kernel* [16]. Nesse sentido, essas funções realizam transformações de dados complexas fazendo com que os atributos passem a ser linearmente separáveis.

De forma mais simplificada, o SVM encontra uma reta de separação, chamada de hiperplano, em que se busca maximizar a distância entre os atributos mais próximos de cada classe.

Outro importante parâmetro a definir é o parâmetro de custo  $C$ . Para [17], a margem é maior para  $C = 0,01$  do que  $C = 10.000$ . Portanto, valores maiores de  $C$  focam em pontos, classificados corretamente, próximos ao limite de decisão, enquanto que valores menores envolvem dados mais distantes.

### D. Random Forest

Criado em 1995 por Tim Kan Ho [18], as florestas aleatórias (em inglês, *Random Forest RF*) são classificadores de conjuntos de árvores de decisão, com cada árvore

cultivada usando algum tipo de randomização. As RFs têm a capacidade de processar grandes quantidades de dados com altas velocidades de treinamento, com base em uma árvore de decisão.

De acordo com [17], as *Random Forests* são uma forma de calcular a média de várias árvores de decisão, treinadas em diferentes partes do mesmo conjunto de treinamento, com o objetivo de reduzir a variância e o viés.

Uma vantagem desse algoritmo é a sua aplicação, podendo ser utilizado tanto para tarefas de classificação quanto para regressão.

### E. Gradient Boosting

O Gradient Boosting (GBM) é um algoritmo de Boosting bastante popular devido a sua grande adaptabilidade e é usado principalmente em competições de machine learning. Essa técnica usada no GBM, constrói um conjunto de árvores, com cada árvore aprendendo e melhorando a anterior, diferentemente das *Random Forests* que constroem um conjunto de árvores independentes [19]. Além disso, o GBM treina iterativamente novos modelos com um enfoque nos erros dos modelos anteriores, tornando a predição mais resistente ao viés.

Segundo [20] uma das maiores motivações de usar o GBM é que ele permite otimizar uma função de custo especificada pelo usuário, em vez de uma função de perda que geralmente oferece menos controle e não corresponde a algumas aplicações do mundo real.

## 4 | MÉTRICAS DE AVALIAÇÃO

Dentre as principais formas de avaliação de modelos preditivos, temos a matriz de confusão, que permite avaliação consistente do modelo. A matriz de confusão é uma matriz que contém informações relacionadas à classificações reais e previstas. Utilizada em problemas de classificação binária, ela busca entender a relação entre acertos e erros que o modelo apresenta [21]. Uma matriz de confusão é representada pela Fig. 2.

		Real	
		Sim	Não
Predito	Sim	Verdadeiro Positivo (VP)	Falso Positivo (FP)
	Não	Falso Negativo (FN)	Verdadeiro Negativo (VN)

Figura 2. Matriz de confusão para problema de classificação binária. Adaptado de [22]

Em que, VP significa que a classe prevista e real originalmente fazem parte da classe positiva, FP significa que a classe predita retornou positiva, porém a classe original observada era negativa, VN expressa que tanto os valores preditos e reais fazem parte da classe negativa e FN representa que o valor predito resultou na classe negativa, mas a classe original era positiva.

Após encontrar esses valores e obter a matriz de confusão, é possível calcular métricas de avaliação para estabelecer o quão preciso é o modelo [22]. Existem várias métricas, e cada uma delas tem peculiaridades que se encaixa melhor em um determinado estudo.

A Acurácia, por exemplo, indica o desempenho em geral do modelo, ou seja, representa a taxa de acerto de todo o classificador. A Precisão, indica quantas classes positivas o modelo acertou dentre todas as classificações de classe positiva. Em outras palavras, quanto maior a precisão, menor o erro de falsos positivos. Outra métrica é a Sensibilidade ou Recall, no qual representa a taxa de valores classificadas como positivo, comparada com quantos deveriam ser. Um recall alto indica que o classificador produziu poucos falso negativos. Por fim, o F1-Score tem como objetivo unir precisão e sensibilidade a fim de ter um único valor que determine a qualidade geral do modelo. Dessa forma, quando essa métrica tiver um valor alto, o modelo terá uma boa precisão quanto uma boa sensibilidade. O cálculo dessas métricas de avaliação podem ser encontrados na Tabela I.

Acurácia	Precisão	Sensibilidade	F1-Score
$\frac{VP+VN}{VP+VN+FP+FN}$	$\frac{VP}{VP+FP}$	$\frac{VP}{VP+FN}$	$\frac{2 \cdot VP}{2 \cdot VP + FP + FN}$

Tabela I: métricas de avaliação.

A curva AUC-ROC como [23] descreve, é uma métrica de avaliação no qual é medida o desempenho de um modelo através de gráfico. ROC (em inglês, Receiver Operating Characteristics) é uma curva de probabilidade, que mede o número de vezes que o classificador errou a predição. Já o AUC (em inglês, Area Under the Curve), representa quanto o modelo é capaz de distinguir entre as classes.

Um bom algoritmo de classificação tem um AUC próximo de 1, o que simboliza que o algoritmo separa bem as classes. AUC próximo de 0, tem a pior medida de separabilidade. E quando o valor é 0,5 significa que o modelo não tem capacidade de separação.

## 5 I METODOLOGIA

A fim de se realizar um estudo sobre o ajuste de parâmetros em Redes Neurais, foi utilizado o banco de dados da MNIST de dígitos manuscritos, que é comumente usado por pesquisadores que desejam obter técnicas de aprendizado em reconhecimento de padrões.

Os modelos de redes neurais e operações de convolução utilizados neste trabalho

foram implementados em linguagem Python versão 3.7, com o auxílio apenas de bibliotecas básicas de operações matemáticas como a Numpy. Considerando-se os outros métodos implementados (SVM, RF e GBM) o pacote scikit-learn [24] na versão 0.24.1 foi utilizado como framework base. As configurações gerais da máquina utilizada para os experimentos são: CPU i5-9300H 2.4GHz, RAM 8GB e sistema operacional Windows 10 64-bit.

Dessa forma, para essa base de dados precisou-se primeiramente ajustar a estrutura da MLP. Sendo assim, o algoritmo implementado foi configurado da seguinte forma: gradiente descendente como método de otimização, um tamanho de lote de 50, escolhido empiricamente através de uma busca linear e um número máximo de 100 épocas. Além disso, foi definido uma taxa de aprendizagem adaptativa, cujo o objetivo é fazer grandes mudanças no início do treinamento e ir diminuindo no decorrer do procedimento, para que pequenas mudanças fossem feitas a fim de otimizar o treinamento. O algoritmo foi executado 10 vezes para cada quantidade de neurônios na camada escondida a fim de colher métricas de estatística descritiva para a acurácia obtida, como o desvio padrão, a média e também o melhor e pior caso possível.

Variando a quantidade de neurônios na camada escondida, um resultado com bom tradeoff entre acurácia e tempo de treinamento foi obtido utilizando 300 neurônios. Com esse parâmetro encontrado, foram feitos vários outros testes variando as funções de ativação. Os resultados desse experimento podem ser vistos na Tabela II.

Funções		Acurácia	Melhor	Pior	Tempo de
Camada escondida	Camada de saída	(%)	caso (%)	caso (%)	treinamento médio (s)
ReLU	Softmax	97,78 ± 0,12	97,94	97,54	610,46
ReLU	Sigmóide	96,71 ± 0,14	97,00	96,53	567,89
ReLU	Tanh	96,18 ± 0,08	96,28	96,01	613,60
Sigmóide	Sigmóide	94,77 ± 0,17	95,15	94,50	692,39
Sigmóide	Tanh	95,63 ± 0,20	96,01	95,19	658,95
Tanh	Sigmóide	94,16 ± 0,16	94,49	93,96	641,77
Tanh	Tanh	95,18 ± 0,10	95,40	95,08	695,67

Tabela II : dados obtidos utilizando 300 neurônios junto com a taxa de aprendizagem adaptativa.

Pode-se perceber que a combinação das funções ReLU (x) e Softmax(x) geraram o melhor resultado quanto a acurácia, além disso, o tempo de treinamento médio sofreu pouca variação neste comparativo.

Após a implementação e ajuste da rede neural, os parâmetros da MLP foram incluídos na CNN que por sua vez, foi configurada para ter 2 camadas de convolução, sendo que cada uma dessas camadas passa por um processo de maxpooling e esta foi aplicada para um conjunto de dados de COVID-19.

A base de dados COVID-19 é formada de diferentes conjuntos de dados disponíveis publicamente e disponibilizado no repositório Kaggle [25]. Esse conjunto possui imagens de raios X de tórax para casos positivos de COVID-19, pneumonia viral e casos saudáveis.

Foi utilizado essa de base de dados na sua ultima versão, contendo 3.616 imagens de COVID-19 no qual a maioria das imagens (2.743) são coletadas do conjunto de dados BIMCVCOVID19 [26], 10.192 imagens de casos normais e 1.345 de pneumonia viral. Porém, para esse estudo foi utilizado somente imagens de COVID-19 e saudáveis, mostrado na Fig. 3.

O número de imagens aproveitadas de casos COVID-19 e saudáveis foram ambas de 3.520, visto que, o conjunto de dados possuía alguns ruídos que poderiam afetar o resultado final, como mostra a Fig. 4. Sendo assim, foi empregado undersampling como procedimento para diminuição dos dados, que consiste em reduzir de forma aleatória os exemplos da classe majoritária.

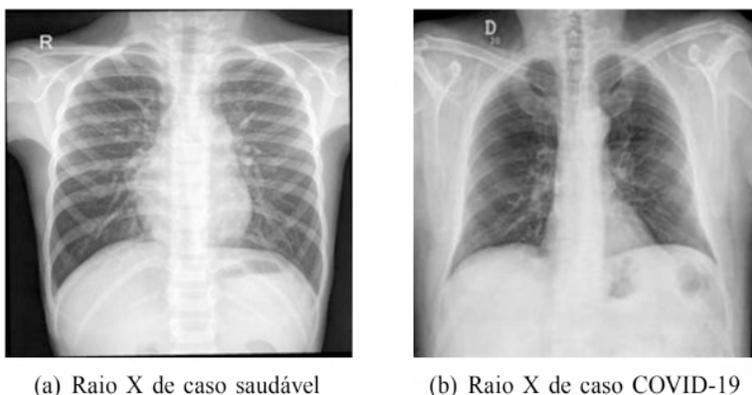


Figura 3. Imagens de caso positivo de COVID-19 e saudáveis retiradas da base de dados [25].

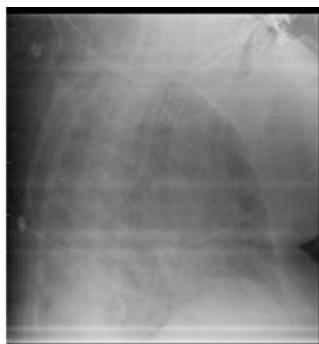


Figura 4. Radiografia indefinida [25].

As imagens de radiografias foram redimensionadas para serem aplicadas na rede

com intuito de reduzir parâmetros. Elas foram redimensionadas para 150x150 pixels, além de serem normalizadas através de (6)

$$X = \frac{X - \text{mean}(X)}{\text{std}(X)}, \quad (6)$$

onde,  $X$  representa o vetor de dados da imagem,  $\text{mean}()$  é média aritmética e  $\text{std}()$  é o desvio padrão.

Dito isso, o treinamento foi feito utilizando ADAM como método de otimização com intuito de melhorar a performance do algoritmo e um tamanho de lote, escolhido empiricamente, de 32. Ainda foi empregado 15 filtros convolucionais de tamanho 3x3 em cada camada de convolução e um número máximo de 4 épocas.

Para aplicação dos outros métodos utilizados (SVM, RF e GBM) a base de dados também foi redimensionada para 150x150 pixels além de ser normalizada e vetorizada. Os hiperparâmetros selecionados para esses algoritmos foram obtidos a partir de trabalhos presentes na literatura sobre classificação de imagens, uma vez que esses métodos servirão apenas de comparação com a CNN implementada.

Por fim, foi usado o método estatístico validação cruzada com  $k$  partições (em inglês, *k-fold cross validation*) para todos os outros métodos utilizados. Esse método é comumente usado na literatura para estimar a capacidade de modelos de aprendizado de máquina, medindo o desempenho do modelo de forma geral.

## 6 | RESULTADOS

Para o experimento com a base de dados COVID-19, utilizamos algumas métricas de avaliação, considerando-se que, para este problema em específico, a principal métrica a ser avaliada é a sensibilidade (*recall*). Essa métrica, é usada em situações

em que os falsos negativos são mais importantes do que os falsos positivos, ou seja, o modelo é treinado com enfoque especial na classificação correta de pacientes com COVID19, visando errar o mínimo possível. Isso foi definido tendo em vista uma interpretação na vida real de uma instância de falso negativo, onde iria resultar em uma suposição equivocada de que o paciente não está infectado com a doença, o que poderia acarretar um diagnóstico tardio além de contribuir para a propagação do vírus.

Dessa forma, a precisão é menos relevante para este trabalho, visto que, a gravidade em classificar pacientes saudáveis como se tivessem a doença é algo menos crítico, uma vez que os procedimentos para tratamento da doença não são debilitantes considerando-se o que a literatura demonstra até a data deste estudo. Entretanto, do ponto de vista estatístico, um resultado ruim quanto à métrica precisão pode representar um baixo valor de acurácia para o modelo, indicando falsamente que o modelo não classifica bem pacientes com a doença.

A Tabela III, mostra o desempenho do algoritmo usando *k-fold cross validation*. Neste trabalho, utilizou-se  $k = 10$  em todos os experimentos, a fim de reduzir o viés.

Métricas	Média	Desvio Padrão	Mínimo	Máximo
Acurácia	89,60%	3,72%	84,09%	93,76%
Precisão	81,19%	6,79%	69,60%	88,35%
<i>Recall</i>	97,58%	1,83%	94,23%	99,35%
F1-Score	88,38%	4,55%	81,00%	93,00%
AUC	90,15%	3,11%	86,06%	93,75%
Tempo de execução (h)	71,76	0,06	71,69	71,83

Tabela III : desempenho obtido pela cnn proposta.

Através da Tabela III, verifica-se que os resultados do experimento são promissores, considerando-se a métrica avaliada e usando um método de validação robusto e consolidado na literatura.

O banco de dados também foi submetido aos classificadores *Random Forest*, *Gradient Boosting* e SVM, com o intuito de relacionar a capacidade dos algoritmos em encontrar resultados que sejam comparáveis à CNN implementada.

Para o RF, os parâmetros foram coletados a partir do trabalho de [27], em conjunto com uma rede pré treinada, a Resnet 50, para extração de características. Entretanto, o classificador RF neste estudo usou a técnica de *bootstrap aggregating bagging*, que é projetado para melhorar a estabilidade e a precisão dos algoritmos de aprendizado de máquina, além de reduzir a variância. Dessa forma, usando uma profundidade de árvores igual a 20 e o número de árvores na floresta igual a 100, este método produziu o melhor desempenho com relação à métrica precisão, além de apresentar o menor custo computacional.

A Tabela IV mostra os resultados obtidos pelo Random Forest considerando-se as métricas utilizadas.

Métricas	Média	Desvio Padrão	Mínimo	Máximo
Acurácia	86,59%	5,79%	78,83%	93,89%
Precisão	84,81%	8,69%	75,43%	98,43%
<i>Recall</i>	90,33%	3,28%	83,52%	96,02%
F1-Score	87,29%	5,18%	79,78%	93,75%
AUC	86,60%	5,80%	78,83%	93,89%
Tempo de execução (s)	68,15	1,61	67,13	69,48

Tabela IV : desempenho obtido pelo *Random Forest*.

A Tabela V mostra os resultados encontrados utilizando-se a SVM. Os parâmetros para este classificador foram obtidos do trabalho de [28] onde a SVM foi implementada juntamente com uma rede convolucional pré-treinada, empregada para classificar casos positivos de COVID-19. O kernel linear foi usado na SVM.

Métricas	Média	Desvio Padrão	Mínimo	Máximo
Acurácia	79,93%	5,54%	70,17%	88,30%
Precisão	79,51%	6,61%	70,05%	89,84%
Recall	81,09%	5,40%	70,45%	88,63%
F1-Score	80,31%	5,42%	70,25%	88,25%
AUC	79,91%	5,55%	70,17%	88,25%
Tempo de execução (min)	32,61	3,22	29,67	35,98

Tabela V: desempenho obtido pela SVM.

A Tabela VI mostra os resultados obtidos utilizando-se o GBM. Os parâmetros utilizados para este classificador são os mesmos empregados no trabalho de [29], no qual foi usado o algoritmo *Stochastic Gradient Boosting* para melhorar o desempenho de um modelo de classificação. O GBM foi aplicado à base usando uma profundidade de árvores igual a 20, número de árvores na floresta igual a 100 e uma taxa de aprendizagem de 0,1.

Métricas	Média	Desvio Padrão	Mínimo	Máximo
Acurácia	75,76%	5,44%	67,61%	82,67%
Precisão	76,16%	4,82%	70,45%	86,93%
Recall	76,16%	4,82%	70,45%	86,93%
F1-Score	75,95%	4,94%	68,61%	82,20%
AUC	75,76%	5,44%	67,61%	82,67%
Tempo de execução (h)	5,23	0,09	5,31	5,15

Tabela VI : desempenho obtido pelo *Gradient Boosting*.

Ao analisar a Tabela VI, nota-se que o tempo de execução do algoritmo é alto em comparação ao SVM e *Random Forest*.

Diante disso, como o principal objetivo desse método é reduzir o viés dos modelos simples, o GBM acaba levando bastante tempo de execução, visto que, o treinamento dos modelos são feitos de maneira gradual, aditiva e sequencial.

Observando as tabelas IV e V, percebe-se que o modelo usando SVM obteve um resultado inferior a *Random Forest* e CNN. Isso pode ter ocorrido devido à dependência da SVM com relação à escolha do kernel, além de possivelmente precisar de um ajuste mais refinado para a base de dados utilizada. Em contrapartida, a RF busca a melhor característica em um subconjunto aleatório das características, criando uma grande diversidade e gerando melhores modelos, obtendo resultados mais satisfatórios [17].

Entretanto, ao comparar os resultados obtidos utilizando a CNN e a RF pode-se perceber que a RF converge com um baixo tempo de execução, contudo, obtém resultados inferiores à CNN. Isso ocorre pois as redes convolucionais tendem a aprenderem certas características importantes para a classificação. As Random Forests tendem a levar em consideração a classe majoritária com um peso maior, devido à forma classificação das árvores.

Nesse sentido, as redes neurais convolucionais predominam em classificar e distinguir casos saudáveis e casos positivos de COVID-19 em imagens de raio x. A Fig. 5 e a Fig. 6 mostram, respectivamente, a matriz de confusão e a curva ROC do melhor resultado obtido pela CNN.

		Real	
		Sim	Não
Predito	Sim	VP = 311	FP = 41
	Não	FN = 3	VN = 349

Figura 5. Matriz de confusão do melhor resultado.

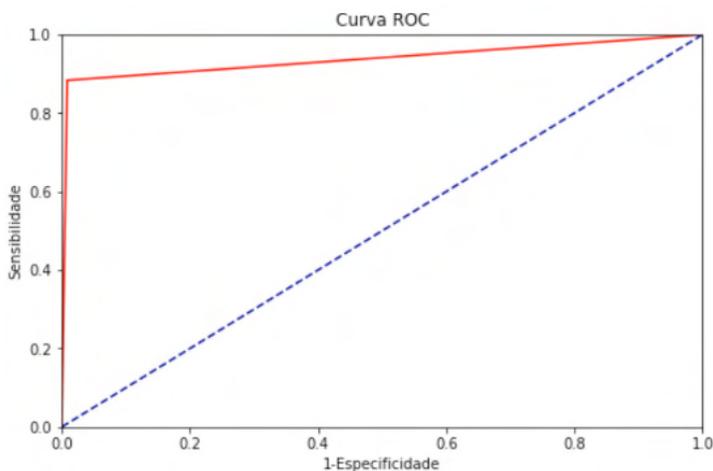


Figura 6. Curva ROC do melhor resultado.

Na Fig. 6, a linha pontilhada azul representa uma resposta aleatória do modelo e a linha vermelha indica o modelo gerado pelo algoritmo proposto. É possível perceber através desse gráfico que o modelo consegue ter uma boa separabilidade devido ao seu valor

de AUC. Isto mostra que o algoritmo obtém boa separabilidade entre casos saudáveis e casos positivos de COVID-19.

A Tabela VII mostra o desempenho obtido em trabalhos recentes da literatura para classificar imagens de COVID-19 com diferentes CNN's e diferentes bancos de dados.

Artigos	Técnicas	Banco de dados	Performance
[10]	CNN (CheXNet) + Correção gamma	COVID-19 <i>dataset</i> (3616 COVID-19, 8851 Saudáveis e 6012 Não-COVID)	Acurácia: 96,29%, precisão: 96,28%, sensibilidade: 96,29%, F1-Score: 96,28% e especificidade: 97,27%
[7]	CNN (MobileNet v2)	<i>dataset</i> com 224 COVID-19, 714 Pneumonia (bacteriana e viral) e 504 Saudáveis	Sensibilidade: 98,66%, precisão: 96,78% e especificidade: 96,46%
[8]	CNN (ResNet-101)	154 COVID-19 e 5.828 imagens não rotuladas extraído do conjunto ChestX-ray14	AUC: 0,82 sensibilidade: 77,3%, acurácia: 71,9% e especificidade: 71,8%
[9]	CNN (Inception V3, Xception e ResNeXt)	<i>dataset</i> com 1583 imagens Saudáveis, 576 COVID-19 e 4273 de Pneumonia	Acurácia: 97,97% e sensibilidade: 92,00%
<b>CNN pro-posta</b>	<b>CNN com 2 camadas de convolução</b>	<b>COVID-19 <i>dataset</i> (3520 COVID-19 e 3520 Saudáveis)</b>	<b>AUC: 90,15%, sensibilidade: 97,58%, precisão: 81,19%, acurácia: 89,60% e F1-Score: 88,38%</b>

Tabela VII: comparação com o estado-da-arte.

O trabalho de [25] é o único trabalho que utiliza um banco de dados com um grande número de imagens. Todavia, mesmo com uma rede convolucional profunda e com aprimoramento de imagem, a sensibilidade alcançada foi de 96,29%, sendo inferior ao resultado obtido no presente estudo. Ademais, o método estatístico usado para validar a eficiência foi o *k-fold cross validation* com 5 *folds*, o que mostra um resultado menos robusto comparado ao realizado neste trabalho.

Analisando o estudo de [7], pode-se observar que o Recall atingindo foi de 98,66%, um valor maior ao que foi obtido neste trabalho. É possível observar também que o método de treinamento utilizado é bastante robusto, visto que o método de *Transfer Learning* armazena o conhecimento adquirido em outras aplicações. Por outro lado, o banco de dados usado conta com uma pequena amostra de dados, contribuindo para o desempenho

dos modelos que sofrem menos com relação ao custo computacional.

Os trabalhos de [9] e [8] também utilizam poucas imagens positivas para COVID-19, isto acarreta em um desbalanceamento de dados, e conseqüentemente, maior chance de o treinamento sofrer *overfitting* e priorizar a classe majoritária, conforme podemos perceber pelos valores obtidos para a métrica sensibilidade, que foram nitidamente inferiores aos valores obtidos pela nossa abordagem.

Apesar de não utilizar as imagens de casos positivos para pneumonia não oriunda de COVID-19, a CNN proposta consegue diferenciar bem radiografias de pacientes com COVID-19 e pacientes saudáveis. Além disso, a COVID-19 pode evoluir para pneumonia, provocando síndrome respiratória aguda grave, sendo assim, mesmo o modelo identificando alguma anomalia presente nas radiografias e classificando como caso positivo de COVID-19, permite que o paciente seja isolado para uma checagem mais minuciosa, tendo uma probabilidade baixa de infectar outra pessoa.

Por fim, é certo dizer que um diagnóstico não é feito somente a partir de uma radiografia do tórax, mas o modelo proposto pode influenciar na tomada de decisão, propiciando medidas de contenção da COVID-19, dado que a rede convolucional permite uma rápida resposta.

## 7 | CONCLUSÃO

É possível concluir que o uso de *deep learning* com CNNs, é capaz de ter efeitos significativos na extração de características e detecção automática de imagens de raios X relacionadas à pacientes com COVID-19.

Dessa forma, com base nos resultados obtidos usando uma Rede Neural Convolucional, foi possível classificar casos positivos de COVID-19, tendo uma sensibilidade de 97,58%. Este modelo pode ser útil como ferramenta para classificação, auxiliando na triagem rápida do paciente e prevenindo a propagação do vírus.

Os resultados obtidos neste estudo são confiáveis e generalizáveis, uma vez que foram treinados e validados com um grande conjunto de dados, sendo comparáveis a diversos estudos recentes e outros algoritmos de classificação do estado-da-arte.

Para trabalhos futuros, é relevante empregar técnicas de segmentação, onde a região de interesse poderia ser destacada, eliminando partes irrelevantes para o treinamento e assim, simplificando a imagem para uma melhor análise. Além disso é interessante melhorar a velocidade de treinamento da CNN utilizando paralelismo, com o objetivo de diminuir o tempo de execução do algoritmo, uma vez que desta forma o esforço computacional pode ser dividido para cada núcleo do processador ou unidade de processamento gráfico (em inglês, *Graphics Processing Unit GPU*).

## AGRADECIMENTOS

Os autores agradecem à UFVJM por todo o suporte prestado no desenvolvimento deste trabalho.

## REFERÊNCIAS

- [1] F. He, Y. Deng, and W. Li, “Coronavirus disease 2019: What we know?” *Journal of medical virology*, vol. 92, no. 7, pp. 719–725, 2020.
- [2] P. M. de Medeiros Carvalho, M. M. Moreira, M. N. A. de Oliveira, J. M. M. Landim, and M. L. R. Neto, “The psychiatric impact of the novel coronavirus outbreak,” *Psychiatry research*, vol. 286, p. 112902, 2020.
- [3] S. K. Brooks, R. K. Webster, L. E. Smith, L. Woodland, S. Wessely, N. Greenberg, and G. J. Rubin, “The psychological impact of quarantine and how to reduce it: rapid review of the evidence,” *The lancet*, vol. 395, no. 10227, pp. 912–920, 2020.
- [4] K. Fukushima, S. Miyake, and T. Ito, “Neocognitron: A neural network model for a mechanism of visual pattern recognition,” *IEEE transactions on systems, man, and cybernetics*, no. 5, pp. 826–834, 1983.
- [5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [6] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional neural networks: an overview and application in radiology,” *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.
- [7] I. D. Apostolopoulos and T. A. Mpesiana, “Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks,” *Physical and Engineering Sciences in Medicine*, vol. 43, no. 2, pp. 635–640, 2020.
- [8] M. Z. Che Azemin, R. Hassan, M. I. Mohd Tamrin, and M. A. Md Ali, “Covid-19 deep learning prediction model using publicly available radiologist-adjudicated chest x-ray images as training data: preliminary findings,” *International Journal of Biomedical Imaging*, vol. 2020, 2020.
- [9] R. Jain, M. Gupta, S. Taneja, and D. J. Hemanth, “Deep learning based detection and analysis of COVID-19 on chest x-ray images,” *Applied Intelligence*, vol. 51, no. 3, pp. 1690–1700, Oct. 2020. [Online]. Available: <https://doi.org/10.1007/s10489-020-01902-1>
- [10] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. A. Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan *et al.*, “Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images,” *Computers in biology and medicine*, vol. 132, p. 104319, 2021.
- [11] L. G. L. Fernandes et al., “Previsão de séries de tempo: redes neurais artificiais e modelos estruturais,” 1995.
- [12] M. A. Nielsen, *Neural networks and deep learning*. Determination press San Francisco, CA, 2015, vol. 25.
- [13] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

- [14] Phung and Rhee, "A high-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets," *Applied Sciences*, vol. 9, p. 4500, 10 2019.
- [15] L. G. Hafemann, L. S. Oliveira, P. R. Cavalin, and R. Sabourin, "Transfer learning between texture classification tasks using convolutional neural networks," in 2015 *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2015, pp. 1–7.
- [16] A. Eloyr and A. Nunes, "Support vector machines," 2020. [Online]. Available: <https://lamfo-unb.github.io/2020/07/04/SVM/>
- [17] J. Friedman, T. Hastie, R. Tibshirani *et al.*, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.
- [18] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.
- [19] G. Duarte, "Gradient boostings parte 1: Métodos de ensemble learning," 2020. [Online]. Available: <https://datarisk.io/gradient-boostings-parte-1-metodos-de-ensemble-learning>
- [20] H. Singh, "Understanding gradient boosting machines," 2018. [Online]. Available: <https://towardsdatascience.com/understandinggradient-boosting-machines-9be756fe76ab>
- [21] A. Santra and C. J. Christy, "Genetic algorithm and confusion matrix for document clustering," *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 1, p. 322, 2012.
- [22] D. Nogare, "Performance de machine learning matriz de confusão," 2020. [Online]. Available: <http://diegonogare.net/2020/04/performance-de-machine-learning-matriz-de-confusao>
- [23] S. Narkhede, "Understanding auc-roc curve," *Towards Data Science*, vol. 26, pp. 220–227, 2018.
- [24] F. e. a. Pedregosa, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [25] T. Rahman, "Covid-19 radiography database," 2020. [Online]. Available: <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>
- [26] BIMCV-COVID19, "Datasets related to covid19's pathology course," 2020. [Online]. Available: <https://bimcv.cipf.es/bimcv-projects/bimcvcovid19>
- [27] R. Mostafiz, M. S. Uddin, N.-A. Alam, M. M. Reza, and M. M. Rahman, "Covid-19 detection in chest x-ray through random forest classifier using a hybridization of deep cnn and dwt optimized features," *Journal of King Saud University-Computer and Information Sciences*, 2020.
- [28] P. K. Sethy, S. K. Behera, P. K. Ratha, and P. Biswas, "Detection of coronavirus disease (covid-19) based on deep features and support vector machine," 2020.
- [29] L. Lin, W. Yue, and Y. Mao, "Multi-class image classification based on fast stochastic gradient boosting," *Informatica*, vol. 38, no. 3, 2014.

# CAPÍTULO 4

## JOGOS DIGITAIS DE ENTRETENIMENTO E O ESTÍMULO DA INTELIGÊNCIA LÓGICO-MATEMÁTICA DE GARDNER

Data de aceite: 01/03/2022

**Carlos Alberto Paiva**

PCS Escola Politécnica  
Universidade de São Paulo  
São Paulo, SP, Brasil

**Regina Melo Silveira**

PCS Escola Politécnica  
Universidade de São Paulo  
São Paulo, SP, Brasil

**RESUMO:** Existem hoje inúmeros trabalhos que abordam o uso de jogos digitais de forma pedagógica. Embora seja uma mídia largamente utilizada, poucos trabalhos oferecem profundidade em estudos que relacionem jogos digitais de entretenimento às metodologias que estimulem o desenvolvimento da Inteligência Lógico-Matemática (ILM), apresentada por Howard Gardner [1] em sua Teoria das Inteligências Múltiplas. Como objetivo, a partir dos domínios cognitivos apresentados por Benjamin S. Bloom [2], este trabalho apresenta um estudo sobre o jogo The Legend of Zelda: Breath of the Wild [3], relacionando seu uso como estímulo ao desenvolvimento da ILM. A metodologia utilizada foi a pesquisa bibliográfica com natureza aplicada e os resultados encontrados por esta pesquisa, buscam apresentar dados, análises e discussões que possam nortear e subsidiar professores no uso de jogos digitais de entretenimento e a aprendizagem baseada em jogos.

**PALAVRAS-CHAVE:** Jogos Digitais de Entretenimento, Teoria das Inteligências

Múltiplas, Inteligência Lógico-Matemática, Bloom.

### 1 | INTRODUÇÃO

Profissionais do ensino que fazem uso de jogos digitais (JD) de forma pedagógica, trabalham, na maioria das vezes, de forma empírica, baseando-se apenas em sua experiência profissional e pessoal. Grande parte desses profissionais são também desenvolvedores ou se apoiam em profissionais acadêmicos dentro da área de desenvolvimento de jogos e, raras exceções, acabam utilizando na maioria das vezes, apenas jogos classificados como educacionais.

Parte dos problemas enfrentados por estes profissionais encontram-se justamente nos jogos educacionais. Autores como [4] apontam que encontrar e utilizar bons jogos continua sendo um desafio. Muitos dos jogos educacionais utilizam poucos princípios pedagógicos e são descartados pois não agregam valor à aula. Jogos desenvolvidos por educadores, com viés acadêmico, mostram-se pouco divertidos e não conseguem atrair a atenção dos estudantes. Becta [5] e Kirriemuir [6] apontam possíveis razões para os jogos educacionais não atingirem as expectativas esperadas:

- A maioria dos jogos educacionais é simples e não atende às expectativas dos estudantes, acostumados com a sofisticação dos jogos de

entretenimento.

- As tarefas propostas por esses tipos de jogos são repetitivas, o que o torna enfadonho num curto período.
- As atividades dentro do jogo são limitadas, concentrando o aprendizado em uma única habilidade, ou no acúmulo de conteúdos homogêneos.

Segundo Prensky [7], existem profissionais que fazem uso dos jogos de entretenimento (desenvolvidos a princípio para esse fim) na aprendizagem pois, além de possuírem características de superproduções (*game design* e *gameplay* desenvolvidos por estúdios e produtoras profissionais), possuem larga aderência dos estudantes, sendo amplamente usados em seus dispositivos pessoais (consoles, *smartphones*, computadores, entre outros). O desafio, nesse caso, é apresentar a ponte entre uma mídia com finalidade exclusivamente lúdica e os objetivos de aprendizagem.

Pesquisas sobre o uso de jogos digitais de entretenimento na aprendizagem são incipientes e o interesse por essa mídia levanta questões sobre sua eficácia e aplicabilidade. Esta pesquisa apresenta um estudo que discorre sobre as formas de uso do jogo digital [3] como ferramenta didática que estimule o aprendizado da Inteligência Lógico-Matemática.

#### A. Objetivo

este artigo possui como objetivo principal: A partir do conceito de um *framework* conceitual, relacionar o desenvolvimento da ILM com domínios cognitivos, por meio do uso de jogo digital de entretenimento.

Como objetivos específicos, este trabalho visa apresentar os elementos de *design* do jogo The Legend of Zelda: Breath of the Wild [3], propondo as seguintes possibilidades de relacionamentos: 1ª) O uso de mecânicas e missões propostas no jogo, como estímulo ao desenvolvimento de competências lógico-matemáticas. 2ª) A utilização dos desafios e elementos característicos disponibilizados pelo jogo, como estímulo ao desenvolvimento dos DCs propostos por Bloom [8].

A escolha de The Legend of Zelda: Breath of the Wild (ZBotW) para esse trabalho, ocorreu pela necessidade da amplitude da amostragem a ser estudada. ZBotW é um *case* de sucesso mercadológico, com grande número de jogadores, além de ampla amostragem etária. Embora seja um jogo do gênero aventura, possui diversos elementos que podem estimular o aprendizado, como aspectos de RPG com mundo aberto, *puzzles*, mecânicas de *fighting games*, entre outros.

#### B. Justificativa

Marc Prensky [7] e João Mattar [9] apresentam em suas obras, estudos sobre o uso de jogos e a aprendizagem; Lúcia Santaella [10] traz reflexões a respeito de questões emergentes sobre jogos e a cultura digital; Alves e Coutinho [11] expõem em seu trabalho, o uso prático de jogos e aprendizagem baseadas em evidências empíricas. É significativo

o número de pesquisas que estabelecem a conexão entre esses domínios cognitivos e o desenvolvimento da aprendizagem.

O uso de jogos digitais como ferramenta pedagógica tem proporcionado o estímulo ao desenvolvimento de competências como interação, colaboração, competitividade e, principalmente, a cooperação entre os estudantes. Grande parte das pesquisas identificadas para elaboração deste trabalho, baseiam-se nesses domínios de aprendizagem, relacionados diretamente às inteligências Interpessoal e Intrapessoal, apresentadas também em [1]. As pesquisas encontradas que relacionam diretamente o uso de jogos e o desenvolvimento da ILM foram proporcionalmente menores, quando não nulas. Uma das hipóteses para essa diferença é que, devido à natureza lúdica dos jogos, estimula-se de forma natural as características específicas da área de humanidades inerentes à pedagogia, em contrapartida ao desenvolvimento do raciocínio e da lógica, competências diretamente ligadas ao desenvolvimento da ILM.

Face a essa dicotomia, este trabalho se justifica pela proposta de oferecer, a partir do uso de jogos digitais, a possibilidade do relacionamento entre duas áreas distintas: o lazer proporcionado pela imersão em um jogo digital de entretenimento, que estimulam naturalmente às Inteligências Interpessoal e Intrapessoal; e o estímulo do aprendizado lógico-matemático a partir das atividades lúdicas de um jogo.

### *C. Metodologia*

Foi adotada pesquisa a partir de fontes bibliográficas existentes sobre o tema, especificando autores que indiquem o uso de *games* que estimulem o desenvolvimento da ILM na aprendizagem, elencando como o inter-relacionamento dos DCs ocorre na assimilação de conteúdo didático.

Quanto à natureza da pesquisa, este estudo classifica-se como sendo pesquisa aplicada, cujo objetivo é permitir a aquisição de novos conhecimentos para o desenvolvimento ou aprimoramento de produtos, processos e sistemas. Rodrigues [12] define que a pesquisa aplicada, se resume por tentar resolver um problema real por meio científico em que haja aplicação prática. Este trabalho encontra-se nessa classificação de natureza, porque busca oferecer a aquisição de novos conhecimentos que possibilitem o desenvolvimento da ILM, através do uso de jogos digitais de entretenimento.

Os dados obtidos foram analisados e apresentados no formato de tabelas, a fim de apontar a equivalência entre os domínios cognitivos e as atividades lúdicas presente em um jogo especificamente selecionado.

Este trabalho está detalhado em três seções: Seção II. INTELIGÊNCIAS MÚLTIPLAS DE GARDNER (IMGs), na qual são explorados a origem e os principais elementos que compõem a teoria desse autor; a definição e estímulos da ILM; o uso das IMGs na aprendizagem; as atividades e estratégias para o seu estímulo da ILM; a relação entre

*design* em jogos digitais e Gardner; e os oito tipos de diversão em jogos e os estímulos a partir deles. Seção III. RELAÇÃO ENTRE JD DE ENTRETENIMENTO E AS IMGs, na qual é discutida a possibilidade do relacionamento desta ferramenta na aprendizagem da ILM; uma possível classificação de jogos quanto ao seu gênero; a relação entre a taxonomia de Bloom, JD e as IMGs; a apresentação do jogo [3] e os motivos que levaram à sua escolha para pesquisa; e a apresentação dos domínios cognitivos de Bloom relacionados ao jogo [3]. Encerrando o trabalho, na seção IV. CONCLUSÕES, discorre-se sobre o uso de JD de entretenimento no estímulo da ILM e possibilidades de pesquisa sobre essa ferramenta.

## 2 | INTELIGÊNCIAS MÚLTIPLAS DE GARDNER

A inteligência, originada do latim *intellectus* significa entender, compreender. Já foi definida de diferentes formas e o que antes era visto como algo restrito aos que possuíam habilidades em cálculos e memória de maneira surpreendente, atualmente é vista de maneira abrangente. Em [13] define-se como a faculdade de entender, compreender e conhecer; a capacidade para resolver problemas e criar ideias, assim como apresenta que inteligência também é juízo, discernimento e a capacidade de se adaptar e de conviver.

Em Paris, no começo do século 20, o psicólogo Alfred Binet [14] publicou um teste de raciocínio verbal e matemático para testar o potencial para resolução de problemas de lógica e memória. Pouco tempo depois, Wilhelm Stern criou um sistema de pontuação-padrão para o teste com o nome de Intellingenz-Quotient, o teste de QI.

Uma importante ruptura veio do psicólogo Howard Gardner [15]. Ele se inspirou no modo como a neurociência vê o cérebro hoje: um conjunto de vários módulos distintos, que evoluíram separadamente e, hoje, funcionam como processadores para funções específicas. Com isso em mente concluiu que a inteligência não é um conceito único, indivisível, mas uma soma de várias habilidades.

Segundo Gardner [1], em sua Teoria das Inteligências Múltiplas (TIM), cada indivíduo não é dotado de um mesmo conjunto de competências, conseqüentemente, nem todos aprendem da mesma forma. O trabalho com jogos ganha espaço como ferramenta de aprendizagem, na medida em que propõe o estímulo ao interesse do aluno, desenvolve níveis diferentes de experiência pessoal e social, ajuda a construir novas descobertas, desenvolve e enriquece sua personalidade e simboliza um instrumento pedagógico, levando o educador à condição de condutor e estimulador da aprendizagem.

A ideia de [1] se tornou popular desde que foi publicada. Hoje é senso comum achar que está certa e que o teste de QI tradicional ficou ultrapassado. A TIM estabelece nove tipos de inteligências, com vários ramos, nos quais uma inteligência complementa a outra. São elencadas como:

- **Lógico-matemática:** uma das inteligências mais conhecidas e, por muito tempo, considerada a mais importante. Manifesta a habilidade de confrontar e avaliar objetos e abstrações, comparar dados, utilizar estratégias de raciocínio, pode

ser motivada através de quebra-cabeças, desafios numéricos e lógicos, ou ainda situações-problemas com cálculos. É o tipo de inteligência múltipla desenvolvida por cientistas.

- **Linguística:** é a capacidade de se expressar através da escrita, leitura e toda a situação que envolva o uso da palavra. Pode ser motivada através da escrita propriamente dita: poemas, dissertações; através da fala: leituras, debates, discussões; além de habilidade em ouvir as palavras que são geradas por outros. É encontrada em escritores, locutores e advogados.

- **Musical:** assim como outras inteligências inter-relacionadas, a inteligência musical busca complementos na inteligência linguística para compor letras de música e outros fatores, porém sua expressividade está na presença do ritmo. Pode ser motivada através do canto, da composição, da apreciação de uma melodia, entre outros. É muito desenvolvida em compositores, cantores, dançarinos e maestros.

- **Naturalista:** É a habilidade de distinguir, classificar e utilizar os elementos de diferentes ambientes da natureza, como a diferença entre plantas quase idênticas. Pode ser motivada pelo contato com a natureza. É desenvolvida em biólogos e membros de tribos indígenas.

- **Espacial:** capacidade de compreender o mundo visual, tendo uma percepção de espaço de maneira específica, saber lidar com ambientes complexos e senso de localização, além de pintar, modelar, desenhar e construir modelos em diferentes dimensões. É útil a quem trabalha com coordenação motora e precisa compreender o mundo visual. É bem desenvolvida em arquitetos e engenheiros.

- **Físico-Cinestésica:** também conhecida como corporal é a habilidade de expressar-se através dos movimentos do corpo através da dança, teatro ou esportes. Pode ser usada para resolver problemas e executar movimentos complexos com o próprio corpo. É encontrada em dançarinos, mímicos e esportistas

- **Interpessoal:** capacidade de liderança, saber mediar relações humanas determinando humores e sentimentos entre outros estados mentais. Necessária a quem coordena e executa trabalhos em grupo. É ricamente desenvolvida em vendedores, políticos, professores e atores.

- **Intrapessoal:** É a habilidade de olhar para dentro de si e entender as próprias intenções, objetivos e emoções. Destaca a capacidade de autocontrole, saber lidar com o próprio estado emocional, reconhecendo sentimentos e humores. As inteligências Interpessoal e Intrapessoal propostas por Gardner é vista também como uma Inteligência Emocional proposta por Daniel Goleman. Está presente entre psicólogos, filósofos e cientistas.

- **Existencial:** foi uma das últimas inteligências classificadas por Gardner e corresponde à habilidade de entender e buscar respostas às questões fundamentais sobre a existência humana, como por exemplo, “Que é Deus?”, “O que vai acontecer conosco?”, “O que é o amor?”. Está presente em representantes religiosos e espirituais como padres, pastores, xamãs, entre outros.

As inteligências não são exclusivas e se complementam. Ainda segundo [15] podem ser estimuladas de diferentes formas; todas são importantes e, em determinado momentos da história, uma ganha destaque em relação à outra.

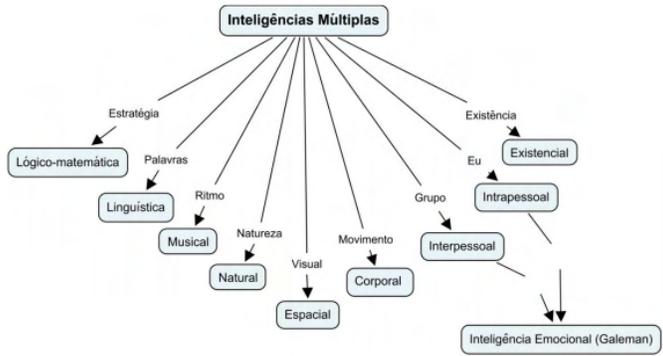


Fig. 1. Inteligências Múltiplas e seus aspectos [16].

Na Fig. 1 pode-se observar a relação e os principais processos indicados por cada Inteligência Múltipla (IM):

#### A. O uso das IMs na aprendizagem

Ao se atirar uma bola a uma pessoa, solicitando que a segure, são colocadas em ação diferentes inteligências do interlocutor. Quando esse envia um comentário sobre o sucesso ou fracasso na captura da bola, é colocada em prática a inteligência linguística; ao prever o espaço percorrido entre a bola e seu corpo, fez uso da inteligência lógico-matemática; sua inteligência espacial permitiu precisar o momento exato em que sua inteligência cinestésico-corporal determinou a amplitude de seu movimento.

Quando um educador em sala de aula desenvolve uma atividade com seus estudantes, de natureza especificamente motora, suscita o empenho prioritário da ILM, uma vez que esta é ligada a percepção de distâncias, possibilitando a interação com as demais inteligências já citadas [17].

Conseqüentemente para Antunes [13], “estimular com atividades, jogos e estratégias as diferentes inteligências de nossos alunos é possível, [...] e pode ser desenvolvido para qualquer faixa etária e nível de escolaridade e em qualquer disciplina do currículo escolar”. O uso específico de atividades que estimulem a ILM, requer do educador, conhecimentos específicos sobre a ação inteligente do cérebro e, conseqüentemente, estratégias para estimulá-lo através de jogos e desafios.

## *B. A inteligência lógico-matemática*

A ILM está ligada à competência em compreender elementos da linguagem numérica e algébrica, permitindo a ordenação de símbolos, noções gerais sobre quantidades e reflexões que envolvem questões de espaço e tempo. Presente intensamente nas áreas de Engenharia, Física e na Matemática, também se manifesta na Contabilidade, em tarefas que envolvem programação de computadores e profissões que fazem uso de números e processos lógicos.

Confundida por Piaget com a ideia de “inteligência” geral, seu estímulo inicia-se cedo, desde quando o bebê conquista a “permanência do objeto”, no simples ato de brincar e procurar um brinquedo [18]. Por volta dos seis anos, a inserção da matemática no cotidiano da criança permite que ela aprenda a comparar e decifrar objetos em suas características primárias: tamanho, espessura, distância (da criança), iguais ou diferentes. As operações matemáticas e fundamentos da geometria são apresentados na forma de materiais concretos como moedas, tampinhas, blocos, cordas e cordões, estimulando a criança em seu raciocínio abstrato.

Segundo [17], quando se pretende estimular a inteligência lógico-matemática, deve-se propor desafios que:

- Envolvam o reconhecimento de diferentes objetos, que permitam a associação, comparação, padrões e relacionamento entre eles;
- Exploreem conceitos de métrica, quantidade, volumes, tempo, causa e efeito;
- Permitam o uso de símbolos abstratos para representar objetos concretos;
- Ofertem sugestões sobre o uso de habilidades na resolução de problemas lógicos;
- Levantem e testem hipóteses;
- Instiguem o uso de habilidades matemáticas como estimativas, probabilidades, interpretação de estatísticas, representações gráficas e outras;
- Proponham operações complexas como programação de computadores;
- Envolvam a construção de objetos, estimulando o pensamento matemático e a formulação de modelos.

## *C. Atividades e estratégias para estimular a ILM*

Antunes [17] constitui as seguintes atividades como exercícios lógico-matemáticos: cálculo numérico, o esforço imaginativo para materializar corpos e formas geométricas no espaço, o uso de conceitos de quantidade, causa e efeito, a aplicação de símbolos abstratos na representação de objetos concretos, tarefas que envolvam raciocínios de proporção, grandezas, volume, massa, peso e outros cuja expressão simbólica sejam números ou palavras que derivem deles, constituem exercícios lógico-matemáticos.

Seguem especificidades sobre algumas das atividades e estratégias indicadas pelo autor:

1. **Uso de Linguagem de Códigos:** Um decifrador de códigos é alguém que possui elevado desenvolvimento da habilidade lógico-matemática. A capacidade de perceber padrões de linguagem em símbolos aparentemente indecifráveis é algo desejável e deve ser estimulada sempre que possível. O educador que deseja estimular essa habilidade, pode se valer de recursos e atividades em sala, bastando para isso possuir uma fórmula que se apresente através de um código lógico, como por exemplo a troca de vogais por consoantes em uma frase.
2. **Desafios Lógicos:** Consiste na apresentação de problemas lógicos ao estudante, a fim de estimular sua capacidade de raciocínio. Por exemplo: Um homem, ao olhar para uma fotografia, dizia “Não tenho nem irmãos, nem irmãs, mas o pai deste homem é filho de meu pai”. De quem é o retrato? Resposta: De seu filho
3. **Interpretação de Sinais:** A leitura de um documento com símbolos e sinais, como uma carta geográfica usada em aulas de geografia, representa uma ação lógico-matemática integral. Sua leitura permite que o estudante execute uma tarefa de decodificação simbólica e, na sequência, o leva a projetar no espaço os limites desses símbolos. Esse tipo de atividade é essencial para a fluência de pensamento lógico, uma vez que envolve a substituição compreensiva de sinais convencionais.
4. **Exploração e Resolução de Problemas com o uso de Padrões:** A palavra “padrão” significa “base para comparação”. A partir dessa definição, pode-se explorar a diversidade dos padrões em disciplinas curriculares tão diversas, como biologia, química ou mesmo astronomia. Pode-se levar estudantes a explorarem padrões na distribuição de astros interplanetários, elementos de um ecossistema ou átomos em uma molécula. A partir desse processo, pode-se propor atividades que desafiem o raciocínio lógico do estudante.
5. **Uso de Puzzles:** Uma maneira interessante de atrair o estudante para a descoberta ocorre quando o educador fragmenta uma ilustração, um quadro-síntese ou esquemático, transformando-o num jogo de encaixe. Estudantes que observam uma pintura, estudam o uso das formas, a harmonia do conjunto e distribuição das cores, com o objetivo de possuir maior competência para montar essa tela, caso esteja fragmentada.
6. **Análise e Interpretação de Gráficos:** Gráficos são essenciais para a interpretação estatística e é parte significativa no mecanismo de compreensão. Usar gráficos corretamente representa para o estudante um exercício mental tão expressivo, quanto aprender dois idiomas, expondo seu conhecimento através de ambos. Sua análise e interpretação facilita a compreensão das informações e representa uma linguagem que transforma informações numéricas em imagens facilmente legíveis.
7. **Trabalhando com Fórmulas:** Uma das definições de fórmula é “descrição científica sob a forma de símbolos ou de figuras que revela uma maneira de proceder” [17]. Onipresentes em disciplinas de física ou matemática, pode ser usada também

em conteúdos curriculares como Geografia ou mesmo Língua Portuguesa. Pode-se inculcar no estudante o domínio de regras e fórmulas como, por exemplo, o não uso de “n” antes de “p” ou “b” na Língua Portuguesa. Deve-se observar que, ainda mais importante que conhecer uma fórmula ou regra, é conhecer e entender os processos que levaram à sua criação, instigando a descoberta e raciocínio por trás de cada uma. Como observa [17], “Quando o aluno identifica na disciplina a existência de fórmulas e quando é levado pelo professor a identificar sua essência, seguramente aprende melhor e mais intensamente desperta sua inteligência lógico-matemática”.

#### **8. Uso de Medidas e Médias Estatísticas nas Reflexões Lógico-Matemáticas:**

Uma das formas de se estimular a ILM de um estudante é destacar a importância e a significação de medidas e médias estatísticas, fazendo com que sua imersão no conteúdo didático se manifeste através dessa segunda linguagem. Se o estudante consegue considerar os elementos numéricos associados ao conteúdo de aprendizado, está aprimorando sua habilidade lógico-matemática, pois é capaz de mensurar as grandezas métricas envolvidas no processo.

**9. Reversibilidade entre Linguagem Gráfica e Linguagem Textual:** Estudantes que respondem uma questão textual, podem utilizar outros meios para oferecer uma resposta, como por exemplo: colagem, paródia, gestos mímicos ou, como em específico na ILM, gráficos e ilustrações pictóricas. O efeito inverso desse processo é igualmente positivo. A transformação de textos em gráficos e a reversibilidade, colaboram na compreensão de conteúdos e a atividade lógico-matemática

**10. Atividades que envolvam Análises de Probabilidades:** Questões simples do dia a dia como “Será que vai fazer sol ou chuva?” ou “Levarei mais tempo no trânsito indo de carro ou de ônibus”, são exemplos de como a probabilidade está imersa na rotina diária das pessoas de maneira quase imperceptível. Considerar possibilidades e solucionar problemas, sintetiza uma das mais importantes áreas da competência lógico-matemática. Embora possa-se pensar que a aprendizagem de probabilidade envolva necessariamente a suposição, deve-se aferir a sua concretização ou não. Estudantes habituados a considerar as “chances” de ocorrência de um evento, abandonando a probabilidade intuitiva em busca da probabilidade lógica, exercitam elementos de sua criatividade e estimulam, em qualquer conteúdo disciplinar, a ILM.

Antunes [18] sugere o uso de jogos direcionados à essa inteligência com o intuito de estimulá-la desde cedo: jogos para assimilação de conceitos simbólicos nas relações numéricas e geométricas, facilitando sua compreensão espacial (grande, pequeno, fino, grosso, largo e estreito, alto e baixo); jogos que despertem a consciência operatória, propondo a assimilação de conceitos de grandeza dos sistemas de numeração; jogos operatórios com ferramentas de avaliação lógico-matemática e jogos que estimulem o raciocínio lógico, que possam ofertar também o vínculo da matemática com a filosofia e música.

O autor em sua obra [18], apresenta 56 jogos analógicos, que podem ser utilizados por um educador a fim de estimular a ILM, utilizando as dez estratégias citadas

anteriormente. Cada jogo indica a habilidade lógico-matemática a ser desenvolvida, além de outras inerentemente conectadas às demais IMG. Na Tabela 1 são elencadas, a partir da indicação de Antunes [18], as habilidades e seu relacionamento com a ILM. O autor indica também o desenvolvimento de outras habilidades, não necessariamente ligadas à ILM:

<b>HABILIDADES LÓGICO-MATEMÁTICA</b>	<b>OUTRAS HABILIDADES</b>
Noção de tamanho (grande e pequeno)	Percepção espacial
Noção de tamanho (alto e baixo)	Coordenação motora
Noção de tamanho (maior e menor)	Contagem
Noção de tamanho (fino e grosso)	Percepção visual
Noção de tamanho (largo e estreito)	Atenção
Noção de conjunto; Formas Geométricas	Ideia de conjuntos
Sistema de numeração	Noção de Cores e Posição
Raciocínio lógico	Noção de sequência
Pensamento lógico	Sociabilidade
Percepção de sistemas de numeração	Classificação
Identificação de Símbolos	Operações simples
Associação de Quantidades	Classificação
Associação de ideias	Coordenação tátil
Operações e conjuntos	Criatividade
Instrumentos de Medida	Alfabetização cartográfica
Compreensão de sistemas de numeração	Orientação espacial
Ordenação	Ordenação
Compreensão de números ordinais	Ordenação crescente
Compreensão de sinais	Uso de sinais
Operações aritméticas	Observação; Atenção
Operações fracionárias	Cálculo mental
Noção de horas	
Percepção de distâncias lineares	Noção de distância
Representação em escala	Aplicação de escalas
Noção de Abstrações	Motricidade

Tabela 1. Habilidades lógico-matemáticas.

#### *D. Piaget e os estímulos lógico-matemáticos*

Segundo Piaget, o desenvolvimento cognitivo lógico-matemático de uma criança, antes dos seis anos de idade, pode ser sensivelmente estimulado com o uso de atividades lúdicas [1]. Além da atividade cognitiva, o brincar se apresenta também como atividade social, na qual as crianças exercitam suas habilidades físicas e interagem entre si.

Nesse período, são valiosos os estímulos que ofertem a ideia de conjuntos e grandezas que permitam a manipulação numérica eficiente, transformando a percepção do símbolo como grandeza que sustenta seu valor. Uma vez que esses conceitos são compreendidos, pode-se adicionar elementos simples de adição e subtração.

Em [1], [18] (p. 71) são apresentadas as cinco condições indicadas por Piaget, que regem esses sistemas de operações aritméticas, denominadas por Piaget como “operações concretas”, que possibilitam efetuar:

- **Composições:** combinando dois ou mais elementos de um conjunto, objetivando formar outro da mesma espécie. Exemplo: Todos os animais podem ser classificados como vertebrados ou invertebrados, logo um animal é invertebrado ou invertebrado ( $1 + 1 = 2$ ).
- **Reversões:** aceitando que é possível reverter transformações através da operação inversa. Exemplo: Se excluir invertebrados, todos os animais que sobram são vertebrados ( $2 - 1 = 1$ )
- **Associações:** pode-se efetuar diferentes associações em um sistema de operações, permitindo que seu resultado continue o mesmo. Exemplo: um grupo de meninos e velhos são homens; e meninas e velhas são mulheres, logo meninos + homens ou meninas + mulheres constituem uma só associação  $15 (7 + 8) = 15 (4 + 11)$ .
- **Anulação:** A combinação de uma operação com seu inverso, resulta em operação idêntica ou nula ( $+5 - 5 = 0$ )
- **Tautologia:** ocorre quando uma classificação é acrescida a si mesma, permanecendo a mesma, ou seja, não se transforma em valor quantitativo. Exemplo: vertebrados + vertebrados = vertebrados.

#### E. *Design em jogos digitais e gardner*

O trabalho de Hunicke [19] envolvendo desenvolvimento, *design* e pesquisa em jogos, pode se relacionar diretamente às IMGs uma vez que, em sua pesquisa, os autores apontam a influência dos elementos de *design* implícitos em jogos. Esses elementos, trabalhados durante o desenvolvimento do *software* com a finalidade de oferecer entretenimento e diversão, abordam aspectos que ativam respostas emocionais, motoras e dinâmicas do jogador e, conseqüentemente, podem estimular o desenvolvimento das inteligências, interpessoal, intrapessoal, motora e lógico-matemática, sendo essa última, sob a ótica do desenvolvimento estratégico e lógico, o objeto de estudo deste trabalho.

Mais do que apenas estimular essas inteligências, a aprendizagem ocorre de maneira espontânea e natural, uma vez que o jogador não percebe, de maneira consciente, que está recebendo estímulo, pois está imerso e focado na atividade, no processo conhecido como Flow, identificado por [20]. Pode-se destacar ainda o conceito de “diversão” oferecida

ao jogador que, de maneira prazerosa, completa e retroalimenta a aprendizagem.

#### *F. Oito tipos de diversão*

Os 8 tipos de diversão foram citados pela primeira vez em um dos artigos que mais influenciaram a teoria de *game design* nos últimos anos: “MDA: A Formal Approach to Game Design and Game Research” [19]. Neste artigo os autores apresentam a estrutura Mechanics, Dynamics, and Aesthetics (MDA) que em português pode ser traduzido como Mecânicas, Dinâmicas e Estética.

As mecânicas são as regras e conceitos que formalmente especificam o jogo como sistema. As dinâmicas são os comportamentos que emergem do jogo enquanto jogado, estimulados diretamente pela mecânica. A estética são as respostas emocionais evocadas pela dinâmica do jogo. A partir do questionamento, o que faz um jogo ser “divertido”, [19] conceitua uma definição para além do significado genérico da palavra “diversão”, chamada de 8 tipos de diversão [21] [22]. Segundo o autor, não se trata de dizer que existem apenas 8 tipos de diversão, pois pode-se encontrar a nona ou a décima. Trata-se de um sistema que ajuda a olhar a diversão de forma mais aprofundada, conectada ao propósito estético e da experiência que se deseja causar no jogador.

Os oito tipos de diversão apontados por [19] e [22], são (tradução nossa):

- **Sensação Jogo como sensação de prazer:** Jogos que evocam emoção no jogador, seja pelo som, pelo visual, vibração do controle, ou esforço físico. Pode funcionar como objeto de arte, para olhar, ouvir ou sentir a beleza ou outras sensações prazerosas.
- **Fantasia Jogo como faz-de-conta:** Jogo como meio de levar o jogador para outro mundo. Pode oferecer a experiência de ser um outro alguém, com habilidades possíveis ou impossíveis.
- **Narrativa Jogo como desdobramento de história:** Jogo como um meio para contar uma história ou narrativa para o jogador. · Pode ser um filme ou uma simples sequência de fatos, interessantes para o jogador.
- **Desafio Jogo como percurso de obstáculos:** Jogo que fornece ao(s) jogador(es) alto valor competitivo ou com desafios cada vez mais difíceis. Além de obstáculos para serem superados, pode oferecer problemas para serem resolvidos ou planos para serem planejados.
- **Companheirismo Jogo como ferramenta de socialização:** Jogo que tem interações sociais como base ou como uma característica marcante. Pode ajudar a tecer socialização e interações humanas.
- **Descoberta Jogo como território desconhecido:** Jogos em que os jogadores exploram o mundo que estão. Você poderá ser um turista andando em uma cidade desconhecida, ou um viajante espacial, explorando por necessidade ou pura curiosidade.

- **Expressão Jogo como autodescoberta:** Jogos que permitem a autoexpressão do jogador por meio do jogo. Pode ser uma expressão por meio do personagem, da roupa que usa por exemplo, ou até mesmo na forma de jogar o jogo. O jogo é o veículo para expressar-se.

- **Submissão Jogo como um passatempo:** Jogos que possuem “cultivo” ou “tédio” como elemento base. Podem ser considerados passatempos “estúpidos”. Há pessoas que jogam Paciência porque acham um problema interessante, outros pelo puro prazer de manipular as cartas. Este segundo é submissão.

Apesar da definição detalhada, dificilmente um jogo possui apenas um tipo de diversão. O mais comum é que se tenha uma ou duas principais e outras secundárias. Os aspectos apontados em cada tipo de diversão no trabalho dos autores citados, referem-se à experiência de entretenimento que um jogo pode oferecer. Pensando-se o *design* do jogo, que prioritariamente objetiva trazer entretenimento, observar-se que pode ocorrer paralelamente, o estímulo às IMGs.

Um jogo como Paciência, classificado como diversão do tipo Submissão-Passatempo, possui *design* que estimula o raciocínio lógico sobre a disposição da sequência de cartas, a fim de que supere o desafio no menor número de movimentos possíveis. Outro jogo, objeto de estudo deste trabalho, The Legend of Zelda: Breath of the Wild (ZBotW) [3] poderia ser classificado pelos autores supracitados, embasado em seu *design*, como diversão do tipo Fantasia, Descoberta e Desafio. Essas características classificatórias podem proporcionar o aprendizado da lógica, uma vez que o jogo traz elementos como gerenciamento de recursos e “cálculo” de possibilidades, já que o jogador necessita criar estratégias e alianças para superar inimigos mais fortes; ou gerenciar alimentos e armas, para superar condições climáticas sazonais no jogo (frio e calor extremos).

### 3 I RELAÇÃO ENTRE JOGOS DIGITAIS DE ENTRETENIMENTO E AS INTELIGÊNCIAS MÚLTIPLAS DE GARDNER

Aranha e Araújo [23] definem *framework* conceitual como um conjunto de conceitos que se inter-relacionam, com o objetivo de orientar uma investigação, determinando o escopo e a lógica do uso desses conceitos para a solução de problemas. Para este trabalho, os conceitos pesquisados neste *framework*, foram: o relacionamento entre um jogo digital de entretenimento, os domínios cognitivos (DCs) relacionados por Bloom [2] e o estímulo da aprendizagem da ILM.

Para apresentação deste relacionamento, foram cogitados jogos de entretenimento que possuem larga aderência pelos estudantes, assim como franquias que estejam no mercado há longa data e sejam um *case* de sucesso mercadológico. Esses critérios foram pensados objetivando o maior número de estudantes, além da amostragem etária ampla. O *framework* tencionado se baseia no uso do jogo ZBotW, relacionando:

- As possíveis classificações e gêneros de um jogo;
- Os DCs relacionados aos gêneros de jogos.
- A seleção de habilidades e competências lógico-matemáticas que se pretende estimular a partir desse jogo;

a. Classificações de jogos quanto aos seus gêneros

Na Tabela 2, adaptada de [24], são destacados os principais gêneros (nas linhas verdes) e seus subgêneros (nas linhas brancas), em jogos digitais de entretenimento:

GÊNEROS	DESCRIÇÃO
<b>AÇÃO</b>	Jogos caracterizados por ocorrerem em tempo real. Requerem destreza e coordenação para se obter sucesso na partida. Sua principal qualidade está na injeção de adrenalina que proporcionam, motivada por exigir reações rápidas e precisas. Essa classificação pode ser dividida em 5 sub-gêneros:
<b>Aventura e Ação</b>	Estilo que mescla elementos de aventura com elementos de narrativa, em histórias de longa duração. Exige movimentos rápidos e precisos para derrotar inimigos. Exemplo: God of War e Dante's Inferno
<b>Plataforma (Arcade)</b>	Jogos de perspectiva <i>side-scrolling</i> (lateral), na qual o jogador se desloca de um lado a outro da tela, enfrentando desafios que surgem na forma de obstáculos ou inimigos, enquanto coleta itens. Exemplo: SuperMario Bros, Sonic, Donkey Kong
<b>Fighting Games (Combate)</b>	Jogos de disputa (em geral) entre dois <i>players</i> , de perspectiva lateral, no qual o jogador usa combinação pré-definida de teclas ( <i>combo</i> ), para atingir o oponente. Exemplo: Mortal Kombat e Street Fighter
<b>Shooter (Tiro)</b>	Jogos nos quais o <i>player</i> irá combater inimigos com armamentos variados, gerenciando munição, evitando ser atingido. Se dividem em <i>First Person Shooter</i> (FPS), nos quais a câmera se posiciona no campo de visão do personagem, ou em <i>Third Person Shooter</i> (TPS), nos quais a câmera oferece a visão do jogador a certa distância (posicionada nas costas do personagem). Exemplo: Call of Duty e Battlefield
<b>Corrida</b>	Jogos em que o <i>player</i> pilota carros ou outros veículos, objetivando vencer disputas contra outros pilotos. Exige destreza e coordenação motora para manter o veículo em seu percurso, evitando danos, mantendo velocidade elevada, a fim de chegar em primeiro lugar. Exemplo: Gran Turismo e Mario Kart.
<b>AVENTURA</b>	Estilo no qual o jogador resolve <i>puzzles</i> (quebra-cabeças), descobre tesouros, coleta itens e resolve mistérios ao longo de uma história. Uma das suas principais qualidades é a exigência de estratégias para solução dos problemas apresentados ao <i>player</i> . Essa classificação pode ser dividida em 4 sub-gêneros:
<b>Survival Horror (Horror)</b>	Um dos subgêneros mais popular, o objetivo do jogo é fazer com que o personagem consiga sobreviver a um ambiente inóspito, geralmente com enredo no sobrenatural, escapando de inúmeros perigos para se manter vivo. O elemento <i>jump scare</i> é bem explorado nesse gênero. Exemplo: Amnesia, e Layers of Fear

GÊNEROS	DESCRIÇÃO
<b>Estratégia</b>	Possui como principal característica o gerenciamento e obtenção de recursos para se alcançar determinado objetivo. Uma grande qualidade nesse sub-gênero é a exigência de raciocínio e planejamento para cada jogada. Divide-se em <i>Turn-Based Strategy</i> (TBS), com ritmo lento, no qual o jogador precisa elaborar a estratégia eficaz maximizando, em seu turno, danos ao oponente enquanto minimiza prejuízos durante o turno do oponente; e <i>Real-Time Strategy</i> (RTS), com ritmo acelerado. Ex: Age of Empires e Civilization IV.
<b>RPGs (Role-Playing Games)</b>	Conhecidos como jogos de representação de papéis, são focados em uma narrativa, na qual o papel assumido pelo <i>player</i> será decisivo para enfrentar os desafios propostos. As narrativas de forma geral oferecem o papel de mago, guerreiro e ladrão, no qual o jogador irá evoluir suas habilidades de acordo com sua escolha inicial. Exemplo: Skyrim e Dragon Age.
<b>MMOGs (Massive Multiplayer Online Games)</b>	Jogos que ocorrem em ambiente <i>online</i> , com grande número de jogadores conectados, em disputas compartilhadas. Englobam outros sub-gêneros como: MMORPGs, MMOFPSs, MMORTSs (RPGs, Tiro e Estratégia em Tempo Real, respectivamente)
<b>PUZZLES</b>	Tipo de jogo que não oferece narrativa ou personagens. Apresenta desafios que precisam ser solucionados. Exigem o uso da lógica, matemática, ou a repetição de padrões para sua resolução. Exemplo: Tetris e The Incredible Machine
<b>ESPORTES</b>	Jogos que simulam um determinado esporte, geralmente sob o ponto de vista do jogador. Tem como principal característica o realismo em relação às regras dos esportes, assim como em suas definições gráficas, eventualmente caracterizando atletas famosos. Exemplo: PES e NBA
<b>SIMULA DORES</b>	Games cuja principal característica é o realismo, simulando o comportamento complexo de máquinas como aviões, trens e automóveis, ou ainda simulando uma estrutura social ou econômica. Objetiva a fidelidade dos detalhes, trazendo maior imersão ao jogador. Exemplo: SimCity, Flight Simulator

Tabela 2. Gêneros e subgêneros de jogos digitais.

Uma vez que a seleção do jogo a ser aplicado depende do conhecimento prévio do educador, é recomendado que esse utilize um *game* ao qual se sinta confortável em seu uso. No exemplo apresentado por este trabalho, ZBotW pode ser classificado como jogo de Aventura, no subgênero Estratégia, com mecânicas de *Fighting Games* e elementos de um RPG de mundo aberto.

#### b. Bloom, Jogos Digitais e a IMGs

Bloom, em sua teoria [2], define as taxonomias de acordo com os objetivos de aprendizagem, a partir de domínios específicos do desenvolvimento cognitivo, psicomotor e afetivo. Essa proposta de divisão vai ao encontro dos principais elementos cognitivos de aprendizagem propostos por expoentes da área como Vygotsky [25]. Pode-se concluir que esses domínios possuem relacionamento direto com a TIMs, uma vez que as IMs podem ser estimuladas com o uso de jogos digitais classificados, quanto ao seu uso, nas categorias revisadas de conhecimento e cognição de Bloom. Os autores em [23] tecem uma relação direta entre a taxonomia revista de Bloom e algumas das possíveis classificações de jogos. Na Tabela 3 pode-se observar a relação indicada pelos autores [23]:

Dimensão dos Processos Cognitivos	Gênero do Jogo Digital
<b>6. CRIAR:</b> Reunir dados para formar algo novo ou reconhecer os componentes de uma nova estrutura (Gerar, Planejar, Produzir)	Simulação
<b>5. AVALIAR/SINTETIZAR:</b> Fazer julgamentos com base em critérios e padrões (Verificar, Criticar)	Estratégia, RPG, Adventure, ...
<b>4. ANALISAR:</b> Dividir um conceito em partes e descrever como elas se relacionam com o todo (Diferenciar, Organizar, Atribuir).	Estratégia, RPG, Puzzle, ...
<b>3. APLICAR:</b> Usar um procedimento (Executar, Implementar)	Simulação, Esportes, ...
<b>2. ENTENDER:</b> Dar um significado ao material ou experiências educacionais (Interpretar, Exemplificar, Classificar, Resumir, Concluir, Comparar, Explicar)	Adventure
<b>1. LEMBRAR:</b> Produzir a informação certa a partir da memória (Reconher, Relembrar)	Casuais

Tabela 3. Processos cognitivos e gêneros digitais.

Segundo os autores, a competência é o emprego da consciência de modo rápido e criativo a fim de enfrentar situações que surgem. Para enfrentar essas situações, é preciso a mobilização de habilidades, atitudes e conhecimentos em uma ação inter-relacionada, como por exemplo a utilizada em jogos digitais. As habilidades são elementos pelos quais se pretende atingir as competências. Estudantes expressam o que sabem e podem aprender, por meio de habilidades adquiridas em conteúdos estudados. A prática dessas habilidades é capaz de apresentar competências para a solução de desafios nos domínios da matemática, da linguagem e dos fenômenos naturais, bem como o confronto de situações-problema e a elaboração de argumentações consistentes.

Em [23] aponta-se que jogos digitais podem ser mediadores de aprendizagem e podem contribuir para o despertar de competências, habilidades e melhor compreensão dos conteúdos em diversas áreas, indo ao encontro dos processos apresentados por Bloom. A atenção e o raciocínio lógico, principalmente em jogos de estratégia, são também elementos estimulados pelo uso de jogos digitais.

### c. *The Legend of Zelda*

Para realização do estudo proposto por este trabalho, a franquia selecionada foi a série *The Legend of Zelda*, lançada globalmente há mais de 35 anos [26], em 1986. Embora a série seja primariamente do gênero aventura, possui diversos elementos que podem estimular o aprendizado. Em um de seus últimos lançamentos, o título *The Legend of Zelda: Breath of the Wild* [26] traz aspectos de RPG com mundo aberto, *puzzles*, mecânicas de *fighting games*, entre outros. Possui grande aderência e um público fiel, inclusive grupos

de *fanfics*, fãs que criam histórias fora dos cânones da franquia e raramente autorizadas pelos proprietários do jogo.

A franquia possui algumas das maiores vendas no mercado de *games*, totalizando mais de 90 milhões de cópias vendidas. *The Legend of Zelda: Breath of the Wild* é o jogo mais vendido da franquia com mais 21 milhões de cópias [27]. Além do sucesso de vendas é um jogo premiado. Foi o grande vencedor do DICE Awards em 2018: jogo do ano, melhor *game design*, melhor jogo de aventura e melhor direção [28]. Exceto o prêmio de melhor *game design*, essas também foram as premiações conquistadas no ano anterior no The Game Awards em 2017 [29].

#### d. Elementos do Jogo ZBotW relacionados à ILM

ZBotW se trata de um jogo de mundo aberto, com um mapa de 360 km<sup>2</sup> de extensão (aproximadamente a área da Alemanha) e possui o tempo de término previsto entre 70, 125 e 185 horas de jogo (tempo mínimo, médio e máximo, respectivamente). Face a essa extensão, pode-se identificar diversos elementos do jogo que apresentam possibilidades de relacionamentos com as ILM.

Na sequência são descritos alguns dos principais elementos do jogo e seu *design* de entretenimento. Um ambiente virtual interativo, com vídeos desses elementos, está disponível em: <https://framevr.io/zelda-ilm-gardner>

1. **Shrines (Santuários):** Os Shrines em ZBotW são *dungeons* (termo usado em jogos de RPG para designar caverna, masmorra ou calabouço subterrâneo) distribuídas por todo o mapa do jogo. Nessa versão da franquia, são apresentados 120 Shrines ao jogador. Cada um consiste em um ambiente subterrâneo separado do mapa principal, portanto seguro quanto a todas as ameaças externas ao Shrine, no qual o protagonista é apresentado a um elaborado *puzzle*. No início do jogo os Shrines são introduzidos com o objetivo de fornecer as chamadas “runas”, que dão habilidades especiais ao jogador, para que possa manipular objetos e resolver os desafios crescentes durante seu progresso. Após solucionar os *puzzles* dos quatro primeiros Shrines, os prêmios passam a surgir na forma de armas e Spirits Orbs. A cada quatro Spirits Orbs, pode-se efetuar o acréscimo de tamanho no medidor de *stamina* ou no *heart container*, respectivamente indicadores de força ou indicadores de vida. Alguns Shrines, com prêmios mais significativos, se encontram ocultos no mapa e somente após a solução de uma missão secundária, as *sides quests*, pode-se obter a sua localização. Apesar de serem opcionais, os prêmios recebidos após a solução dos *puzzles* de cada Shrine, torna a jornada do protagonista mais fácil e, não raro, apresentam alguns dos momentos mais desafiantes do jogo. Cada um dos 120 Shrines apresenta forte estímulo ao desenvolvimento das habilidades lógico-matemáticas, pois sua solução envolve noções de tempo, compreensão de símbolos, raciocínio lógico, abstração, percepção e orientação espacial, entre outros.

**2. Mapa e Inventário:** ZBotW foi desenvolvido com um mapa que representasse uma região fictícia da forma mais realista possível. Esse mapa, além de representar uma área geograficamente grande, com 360 km<sup>2</sup> de extensão, é complexo e completo, agrupando diferentes acidentes geográficos como ilhas, florestas, montanhas e rios, até os diferentes climas que permeiam essa geografia, como o extremo frio presente em montanhas até o árido e quente clima desértico ou mesmo vulcânico. Esse mapa possui uma rica representação cartográfica, o que facilita a identificação do jogador, além de estimular as habilidades lógico-matemáticas envolvidas no processo de interpretação de símbolos, distâncias e percepção e orientação espacial. Além disso, ZBotW apresenta um inventário organizado ao jogador. Por meio desse inventário, que se completa à medida que o jogador explora o ambiente, é possível coletar e armazenar armas, munição, roupas, diversos tipos de alimentos e outros itens. A partir desses itens, é possível efetuar combinações de alimentos corretos para exploração de ambientes hostis, como o frio em montanhas ou extremo calor de um deserto. É possível (e necessário) agrupar armas com materiais diferenciados, como por exemplo o uso de armas construídas somente com madeira, para ambientes chuvosos. O uso de armas construídas com metais, em um ambiente sob condição de chuva, atrai raios ou mesmo causa maiores danos ao protagonista, caso esse esteja enfrentando inimigos que possuem armas elétricas. A estratégia a ser usada em cada ambiente possui relacionamento com as habilidades lógico-matemáticas, como as apresentadas na Tabela 4.

**3. Sementes Korocs:** São personagens com o *design* visual híbrido humano-planta infantilizado. São do tipo *Non Playable Character* (NPC), ou seja, personagens não jogáveis, e se encontram escondidas em todo o mapa do jogo. No mapa interativo disponível em [30], pode-se visualizar a localização de todas as 900 sementes. As sementes são úteis pois servem para aumentar o espaço de inventário do protagonista no jogo, mas para serem coletadas, apresentam pequenos *mini puzzles* que exigem diversas habilidades lógico-matemáticas para solução. Associação de formas geométricas, dimensão de objetos e raciocínio lógico e sequencial, são algumas delas.

Além desses elementos, incluídos no jogo de forma a compor a história narrada, é importante destacar as missões secundárias estratégicas (*side-quests*), apresentadas muitas vezes por NPCs dispersos pelo extenso mapa do jogo, que também envolvem raciocínio lógico para solução.

Na Tabela 4, encontram-se sintetizados alguns dos possíveis relacionamentos entre os principais elementos do jogo e o estímulo ao aprendizado da ILM:

HABILIDADES LÓGICO-MATEMÁTICA	ELEMENTOS DO JOGO
Associações de Formas	Sementes Korocs (diversas)
Associações de Ideias	Usar armas corretas em regiões sazonais
Associações Quantidades	Sementes Korocs (diversas)

HABILIDADES LÓGICO-MATEMÁTICA	ELEMENTOS DO JOGO
Classificação	Identificar Armas (e seu uso)
Compreensão de Símbolos e Sinais	Labirintos do Jogo
Formas Geométricas	Korocs (diversos)
Identificação Símbolos	Shrines: Shee Vaneer e Venath
Noções de Abstrações	Uso de armas inadequadas em certos ambientes (ex.: metal, chuva e raios)
Dimensões de Objetos	Korocs (diversos)
Noções de Tempo Cronológico	Shrine Owa Daim (Stasis) e Midraoh (Lua de Sangue)
Operações aritméticas	Sementes Korocs (diversas)
Operações com Conjuntos	Preparar alimentos diversificados
Operações fracionárias	Preparar alimentos apropriados
Ordenação	Sementes Korocs (diversas)
Pensamento e Raciocínio Lógico	Sides-quests e Shrines distribuídas no jogo.
Percepção de Distâncias Lineares	Transportes disponíveis no Jogo
Percepção e Orientação Espacial	Puzzles presentes nas Divine Beasts
Representação Cartográfica (Símbolos e Distâncias)	Pontos de Identificação no Mapa do Jogo
Representação e Aplicação de Escala	Pontos de Identificação no Mapa do Jogo
Sistemas de Numeração	Sementes Korocs (diversas)

Tabela 4. Relação entre habilidades lógico-matemáticas e Zelda Botw (autoria própria).

e. *Domínios Cognitivos relacionados ao jogo*

A nova versão da Taxonomia de Bloom do Domínio Cognitivo [8], indicada na Fig. 2, manteve os aspectos verbais da taxonomia original, considerando a possibilidade de interpolação entre as categorias.

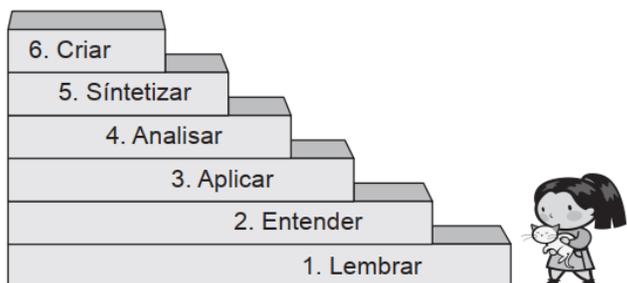


Fig. 2. Categorização Atual da Taxonomia de Bloom [31].

Na sequência esses processos são elencados, relacionando-os ao jogo [3]:

- **LEMBRAR Produzir a informação certa a partir da memória (Reconhecer, Relembrar):** Embora [23] tenham apresentado esse DC como estimulado por jogos

casuais, em ZBotW ele se faz presente a cada desafio enfrentado, uma vez que inimigos possuem características diferenciadas que requerem seu reconhecimento e estratégias específicas para superá-los.

- **ENTENDER – Dar significado ao material ou experiência educacional (Interpretar, Classificar, Concluir, Explicar):** A interpretação de situações específicas no jogo, que levam à classificação e conclusão de determinados objetivos, é vital para o sucesso de sua realização. Pode-se usar as *side quests* como exemplo, uma vez que são propostos desafios ao jogador. Esses desafios (muitas vezes lógicos) exigem interpretação correta para realização, ou jamais poderão ser concluídos.

- **APLICAR – Usar um procedimento (Executar, Implementar):** O uso desse DC é executado dentro de ZBotW em diversas situações. Os quatro primeiros grandes desafios no jogo, oferecem como recompensa as runas: Crionis, Statis, Magnesis e Bomb. Mais do que apenas recompensas, essas ferramentas são mais bem aplicadas se forem implementadas de forma pensada. Assim Magnesis, por exemplo, que em sua execução atrai elementos de metal, pode ser aplicada para rastrear objetos metálicos escondidos e fora do campo de visão. Crionis cria blocos de gelo sob superfícies líquidas, os quais podem ser usados como pontes de gelo temporárias, para alcançar locais de difícil acesso.

- **ANALISAR – Dividir um conceito em partes e descrever como elas se relacionam com o todo (Diferenciar, Organizar, Atribuir):** Domínio Cognitivo ligado diretamente à estratégia, em ZBotW o jogador é estimulado a pensar na solução dos muitos problemas apresentados durante o jogo. Saber organizar a melhor estratégia para um problema específico, diferenciar os desafios de acordo com o ambiente em que se está, atribuir as melhores armas ou as melhores vestimentas (que nesse jogo não possuem apenas finalidades estéticas), uma vez que o ambiente influencia diretamente no jogo, é vital para a sobrevivência do protagonista.

- **AVALIAR-SINTETIZAR – Fazer julgamentos com base em critérios e padrões (Verificar, Criticar):** Esse DC se faz presente em praticamente todos os processos dentro do jogo ZBotW. Pode-se citar como exemplos: avaliar a quantidade do indicador de *stamina* (força) para conseguir chegar ao topo de montanha; avaliar a durabilidade de um tipo de arma, de acordo com o objeto de uso (um machado resiste mais ao ser usado em árvores do que em pedras); verificar se é mais vantajoso evitar (usando modo furtivo *stealth*) um grupo de inimigos ou enfrentá-los, dependendo dos recursos (armas, alimentos para recuperação de perdas) que se possui no momento.

- **CRIAR – Reunir dados para formar algo novo ou reconhecer os componentes de uma nova estrutura (Gerar, Planejar, Produzir):** O último DC apontado na nova versão de Bloom, resulta na conclusão de todos os 5 primeiros. Não se pode chegar a ele sem passar pelos demais. Essa condição *sine qua non* se faz presente também em ZBotW. A criação é algo inerente em todos os momentos do jogo. Isso ocorre desde a elaboração e produção de armas mais eficientes, passando pelo

planejamento dos itens coletados e armazenados para criar alimentos específicos, necessários para determinadas partes do jogo, como por exemplo, produzir alimentos “quentes” para que o protagonista possa enfrentar áreas frias do jogo (como montanhas e neve). Além dessa criação básica, é necessário que o jogador planeje sua forma de agir contra os inúmeros tipos de inimigos que surgem no jogo. Planejar e criar estratégias diferenciadas para cada um deles, é vital para a sobrevivência do protagonista. Observa-se que para que isso ocorra, é necessário que o jogador tenha passado por cada uma das etapas acima, adquirindo experiência e competência nesses DCs, a fim de gerar novas estratégias.

## 6 | CONCLUSÕES

A Teoria das Inteligências Múltiplas de Gardner, e as Inteligências Emocionais apresentadas por Goleman, revolucionaram o modo de se pensar o conhecimento, o aprendizado e o próprio conceito sobre “o que é inteligência”. Criados em 1905, os testes de QI foram aprimorados com o passar das décadas e a necessidade humana. Ainda assim, esse elaborado modelo, não abrange outras habilidades intelectuais que um ser humano pode ter. Basicamente fazem mensurar a cognição básica e, nesse ponto, é inegável que as IMs abrangem um espectro muito mais amplo.

Pensar a relação do desenvolvimento das IMs a partir do uso de jogos digitais de entretenimento e, em específico, a ILM que trabalha conceitos de estratégia, à primeira vista parece algo natural, uma vez que raciocínio lógico é algo inerente aos elementos de que definem o próprio conceito de jogo: desafio, estratégia e o lúdico. Contextualizar o desenvolvimento da ILM a partir de um jogo é um pouco mais complexo, pois envolve a escolha de quais aspectos devem ser estudados e, acima de tudo, quais os domínios cognitivos que indicam o possível desenvolvimento desses aspectos.

Neste caso, a contribuição deste trabalho foi a de apresentar um relacionamento direto entre o uso de ZBotW, com os DCs de Bloom e as competências lógico-matemáticas que podem ser estimuladas, ilustradas nas tabelas e tópicos apresentados. Obviamente existem restrições sobre a possibilidade de uso de jogos de entretenimento para esse fim, como por exemplo a amplitude e aderência dos estudantes ao jogo selecionado, ou o domínio do educador sobre os aspectos a serem estimulados e desenvolvidos. Em ambos os casos a solução reside na escolha e conhecimento do jogo pelo educador.

O jogo digital de entretenimento, mais do que uma forma de lazer de sucesso, pode ser pensado como uma ferramenta muitas vezes prática e acessível também em outras áreas. É possível pensar em trabalhos futuros que envolvam pesquisas, nas quais pode-se relacionar também o estímulo de outras inteligências múltiplas. Há pesquisas que já mostram o uso de jogos de entretenimento em áreas tão diversas como a saúde, segurança ou mesmo aprendizagem social. Aprofundar essas pesquisas e relacioná-las ao trabalho de Gardner, permite ampliar ainda mais o uso de uma ferramenta tão múltipla que é o jogo

digital de entretenimento.

## REFERÊNCIAS

- [1] H. Gardner, “Estruturas da Mente: A Teoria das Inteligências Múltiplas,” Porto Alegre, RS: Artes Médicas Sul, 1994.
- [2] B. Bloom and M. F. Englehart *et al.*, “Taxonomy of educational objectives,” New York: David Mckay, 1956. 262 p. (v. 1).
- [3] Nintendo. The Legend of Zelda: Breath of the Wild. Nintendo Switch. Quioto (Japão): Nintendo Co., Ltd., 03 mar. 2017. [Online] Available: <https://www.nintendo.pt/Jogos/Nintendo-Switch/The-Legend-of-Zeld-a-Breath-of-the-Wild-1173609.html#gameDetails>. Accessed: Mar. 15, 2021.
- [4] N. Balasubramanian and B. Wilson, “Games and Simulations,” in Soc. Information Technology and Teacher Education. 2006.
- [5] Becta, Computer Games in Education Project. Coventry: Becta, 2001. Available: <https://cibermemo.files.wordpress.com/2015/12/edujoc2004.pdf>. Accessed: Mai. 10, 2017.
- [6] J. Kirriemuir and A. McFarlane, “Literature Review in Games and Learning,” Bristol: Futurelab, 2004. 39 p.
- [7] M. Prensky, “Aprendizagem Baseada em Jogos Digitais,” São Paulo: Senac, 2012
- [8] L. Anderson and D. Krathwohl, “A taxonomy for learning, teaching, and assessing: a revision of Bloom’s Taxonomy of Educational Objectives,” Nova York: Addison Wesley Longman, 2001. 336 p
- [9] J. Mattar, “Games em Educação – Como os nativos digitais aprendem,” São Paulo: Editora Pearson. 2010
- [10] L. Santaella and M. Feitoza, “Mapa do Jogo: a diversidade cultural dos games,” São Paulo: Cengage Learning, 2009
- [11] L. Alves and I. Coutinho, “Jogos digitais e aprendizagem: Fundamentos para uma prática baseada em evidências,” Campinas: Papirus, 2016
- [12] M. Rodrigues, “Metodologia da pesquisa: elaboração de projetos, trabalhos acadêmicos e dissertações em ciências militares,” Rio de Janeiro: EsAO, 2005
- [13] C. Antunes, “Inteligências Múltiplas e seus Jogos: introdução,” vol. 1. 3ª ed. Petrópolis, RJ: Vozes, 2012
- [14] R. Sternberg, and L. Jarvin, “Alfred Binet’s contributions as a paradigm for impact in psychology,” in R. J. Sternberg (Ed.), The anatomy of impact: What makes the great works of psychology great (p. 89–107). American Psychological Association, 2003.
- [15] H. Gardner, “Inteligência – Um Conceito Reformulado,” Rio de Janeiro, RJ: Objetiva, 2001.
- [16] H. Delfino, “Inteligências Múltiplas Segundo Howard Gardner,” 2013.
- [17] C. Antunes, “Inteligências Múltiplas e seus Jogos: inteligência lógico-matemática,” vol. 6. 4ª ed. Petrópolis, RJ: Vozes, 2012.

- [18] C. Antunes, "Jogos para a estimulação das múltiplas inteligências," 20ª ed. Petrópolis, RJ: Vozes, 2014.
- [19] R. Hunicke, and M. Leblanc and R. Zubek, "MDA: A Formal Approach to Game Design and Game Research," San Jose, 2004.
- [20] M. Csikszentmihalyi, "Play and intrinsic rewards," Journal of Humanistic Psychology, Vol 15(3), 1975, 41-63.
- [21] S. Rehm, "Gaming for Fun (Part 1): Eight Kinds of Fun," Chicago. 2014.
- [22] C. Nutt, "GDC, Game Design Workshop: Mechanics, Dynamics, Aesthetics," 2008.
- [23] E. Aranha and G. Araujo, "Avaliação formativa das competências e habilidades: instrumentação para jogos digitais," CINTED-UFRS v. 11, n. 3, dez. 2013.
- [24] SENAC. "Planejamento de Jogos Digitais para Multiplataformas: Gêneros de jogos eletrônicos," in Rede Fecomércio RS de Educação. 2020.
- [25] L. Vygotsky and A. Luria, "Estudos da história do comportamento: símios, homem primitivo e criança," Tradução: Lólio Lourenço de Oliveira. Porto Alegre: Artes Médicas, 1996.
- [26] N. Wagner, "The Legend of Zelda e seus 35 anos de história Parte I: estabelecendo padrões para a indústria dos games," in Nintendo Blast. 2021. [Online] Available: <https://www.nintendoblast.com.br/2021/02/the-legend-of-zelda-e-seus-35-anos-de.html> Accessed: Mar. 05, 2021.
- [27] W. Wakka, "Nintendo Switch se aproxima de 80 milhões de unidades vendidas," in CanalTech. 2021. [Online] Available: <https://canaltech.com.br/games/nintendo-switch-se-aproxima-de-80-milhoes-de-unidades-vendidas-178269/> Accessed: Mar. 08, 2021
- [28] A. Osborn, "Zelda: Breath of the Wild vence prêmio de Melhor Jogo do Ano no DICE Awards 2018," in IGN Brasil. 2018.
- [29] B. Penilhas, "Zelda: Breath of the Wild vence prêmio de Melhor jogo de 2017 no The Game Awards" in IGN Brasil. 2017.
- [30] Breath of the Wild Interactive Maps. Zelda Maps, 2021. [Online] Available: <https://zeldamaps.com/?game=BotW/>. Accessed: Jul. 20, 2021.
- [31] A. Ferraz, and R. Belhot, "Taxonomia de Bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais," Gest. Prod., São Carlos, v. 17, n. 2, p. 421-431, 2010.

## RASTREAMENTO DE MOUSE PARA AVALIAÇÃO DE EXPERIÊNCIA DO USUÁRIO EM PORTAIS DE NOTÍCIAS: UM ESTUDO DE CASO

Data de aceite: 01/03/2022

Data de submissão: 07/01/2022

### Danilo Teixeira Lima

UFPA, Programa de Pós-Graduação em Estudos Antrópicos da Amazônia (PPGEAA)  
Castanhal – Pará  
<http://lattes.cnpq.br/4881833295260812>

### Flavio Rafael Trindade Moura

UFPA, Faculdade de Engenharia da Computação e Telecomunicações  
Belém – Pará  
<http://lattes.cnpq.br/4713619033514788>

### Kennedy Edson Silva de Souza

Castanhal – Pará  
UFPA, Programa de Pós-Graduação em Estudos Antrópicos da Amazônia (PPGEAA)  
<http://lattes.cnpq.br/1106866186533455>

### Rita de Cássia Romeiro Paulino

Florianópolis – Santa Catarina  
UFSC, Programa de Pós-Graduação em Jornalismo (PPGJOR)  
<http://lattes.cnpq.br/7783801662264275>

### Marcos Cesar da Rocha Seruffo

UFPA, Programa de Pós-Graduação em Estudos Antrópicos da Amazônia (PPGEAA)  
Belém – Pará  
<http://lattes.cnpq.br/3794198610723464>

**RESUMO:** Portais de notícias são um dos principais meios de obtenção de informação atualmente, segundo Digital New Report, 83%

dos brasileiros consomem notícias nomeio online, incluindo sites de mídias sociais (Reuters, 2021). Entender como os usuários consomem informação é de fundamental importância, visto que pode permitir a melhora da experiência de uso nos portais, tais como os relatados neste estudo. Assim, pesquisas acadêmicas a respeito da avaliação da Experiência do Usuário (UX) e suas relações estruturais, fazem-se necessárias. Técnicas para rastreamento de *mouse* geram dados de interação que podem ser usados para melhorar a UX de portais. Nesse contexto, este artigo apresenta um estudo de caso, a partir de uma metodologia para capturar e analisar o rastreamento do mouse de usuário, a fim de identificar tarefas executadas em cinco portais de notícias. Os resultados apontam que a realização das quatro tarefas definidas, permitem identificar que propagandas e outros recursos de publicidade são mais atrativos do que a notícia. O estudo identificou que há excesso de informações visuais em alguns portais e a existência de funcionalidades importantes, mas com pouco destaque.

**PALAVRAS-CHAVE:** Portais de notícias; Experiência do usuário; Rastreamento.

### MOUSE TRACKING FOR ASSESSMENT OF USER EXPERIENCE IN NEWS WEBSITES: A CASE STUDY

**ABSTRACT:** News websites are one of the main means of obtaining information currently, according to Digital New Report, 83% of Brazilians consume news online, including social media sites (Reuters, 2021). Understanding how users consume information is of fundamental

importance, as it can improve the user experience in the portals reported in this study. Thus, academic research regarding the evaluation of User Experience (UX) and its structural relationships are necessary. Techniques for mouse tracking generate interaction data that can be used to improve the UX of portals. In this context, this article presents a case study, based on a methodology to capture and analyze user mouse tracking, in order to identify tasks performed in five news portals. The results indicate that the accomplishment of the four defined tasks allows to identify which advertisements and other advertising resources are more attractive than the news. The study identified that there is an excess of visual information in some portals and as a result, we verified the existence of important features, but with little emphasis, among others.

**KEYWORDS:** News portals; User experience; Tracking.

## 1 | INTRODUÇÃO

Pesquisas apontam o protagonismo dos portais<sup>1</sup> de notícias entre as formas de consumo de informação pelos brasileiros. De acordo a Poder Data (2020), cerca de 41% do brasileiros se informam pelos veículos jornalísticos na Internet, visando se manterem informados, ficando à frente da televisão (25%), jornais/revistas impressos (13%) e rádio (2%). Outro dado a ser considerado é o da pesquisa realizada pela Provokers (2020), apontando que em decorrência da pandemia da Covid-19, o consumo entre leitores de notícias em veículos digitais no Brasil aumentou 65%. De acordo com o *Digital New Report* (Reuters, 2021), 83% dos brasileiros consomem notícias no meio online, incluindo sites de mídias sociais.

O levantamento da Provokers (2020) revelou que 92% do público que se informa por meios digitais, leem essas notícias pelo menos duas vezes por semana, sendo que 83% acessam notícias ao menos uma vez ao dia. Contudo, apesar do protagonismo dos Portais de Notícias, existe uma falta de conhecimento no que diz respeito à experiência do usuário (*User Experience/UX*) e suas relações estruturais (Arahyi e Schaik, 2015) nestes sistemas interativos.

Arahyi e Schaik(2015, p. 2472)afirmam que há “uma falta de pesquisas acadêmicas em sites de notícias, em termos de construções de UX e suas relações estruturais. Há ainda uma falta de conhecimento sobre como os sites de notícias podem ser projetados, visando promover uma experiência do usuário de alta qualidade”. Contudo, sistemas de avaliação e monitoramento são capazes de coletar dados para análises estatísticas que permitem traçar padrões de uso, tornando-os uma referência para desenvolvimento de sites e sistemas. É o caso do estudo de Paulino et al. (2021) que objetivou identificar as interações no site transmídia “Debarriosomos”,utilizando umsistema quantitativo de rastreamento de interações intitulado *Artificial Intelligenceand Mouse Tracking-Baseduser Experience Evaluation Tool*(AIMT-UXT).

---

<sup>1</sup> O termo portal de notícia e site de notícia tem o mesmo significado neste artigo, representando o endereço de domínio para acesso aos veículos de divulgação de notícias

Dessa forma, esses sistemas de avaliação e monitoramento possibilitam recomendações de uso e até mesmo a predição de consumo de dados. Assim, sendo possível desenvolver soluções que sejam capazes de mensurar a atenção do usuário, além de identificar e avaliar o comportamento destes usuários, permitindo adaptações em tempo real, por exemplo, e consequentemente melhoria de UX.

Este artigo visa auxiliar na resposta da seguinte pergunta: Como usuários navegam e se apropriam de portais de notícias em uma busca de conteúdos específicos? Para responder tal pergunta, é apresentado um estudo de caso com cinco portais, onde os usuários realizaram tarefas pré-definidas, em ambiente controlado, e os dados dessa interação são rastreados por meio da ferramenta *TrackingTechniquesBasedUser Experience Evaluation Tool (T2-UXT)*<sup>2</sup>. O estudo se concentra no rastreamento de *mouse*, contudo a ferramenta possibilita o rastreo teclado e movimentos dos olhos, além da gravação de voz e análise de sentimentos.

O artigo está organizado na seguinte forma: a seção 2 apresenta os trabalhos mais relevantes sobre avaliação de UX em Portais de Notícias, bem como, pesquisas que utilizam ferramentas de rastreios para análise da interação de usuários em sistemas interativos; a seção 3 detalha a ferramenta escolhida para subsidiar esse estudo de caso; a seção 4 detalha o estudo, com o cenário utilizado e os usuários envolvidos; a seção 5 apresenta os achados obtidos e faz uma discussão sobre UX nos portais investigados; seguido a seção 6 deste artigo, que apresenta as conclusões sobre o estudo.

## 2 | TRABALHOS CORRELATOS

Nesta seção serão apresentados estudos relacionados à avaliação de UX em Portais de Notícias e pesquisas que já utilizaram ferramentas de rastreamento para análise de comportamento de usuários. Para isso, as bases de dados utilizadas foram: Periódico Capes<sup>3</sup>, Scielo<sup>4</sup> e ResearchGate<sup>5</sup>. Esse estudo apresenta características inovadoras quando comparadas a outras pesquisas de UX, voltadas para portais de notícias publicadas nos últimos anos. O trabalho de Al-Radaiehet al. (2011), por exemplo, realiza uma avaliação de sites de jornais online usando duas avaliações medidas: usabilidade e conteúdo da web. Contudo, esse trabalho é limitado a uma avaliação por meio de questionários baseados na definição de usabilidade e de conteúdo da web.

Outro trabalho que aborda UX e portais de notícias é o estudo de Silva e Petrucelli (2018), que trata-se de uma pesquisa com conceitos teóricos e um caso real, fazendo uma breve análise de um portal de notícias da área esportiva. O trabalho se concentra em discutir o conceito de UX *Design* com foco na experiência do usuário durante a navegação.

O trabalho de Arahji e Schaik (2015) também se dedica a avaliar a experiência do

2 Disponível em <https://github.com/KennedyWaves/T2-UXT>

3 Endereço eletrônico: [periodico.capes.gov.br](http://periodico.capes.gov.br)

4 Endereço eletrônico: [scielo.br](http://scielo.br)

5 Endereço eletrônico: [researchgate.net](http://researchgate.net)

usuário em portais de notícias. Este estudo identifica características do site, o atendimento de necessidades, reações afetivas e construções de aceitação de tecnologia. O instrumento para a obtenção de indicadores de variáveis que medem aspectos da UX também foi um questionário.

Dessa forma, diferente dos estudos que se concentram em avaliar a UX em Portais de Notícias por meio de questionários e análises visuais, o presente estudo apresenta como inovação o uso de uma ferramenta que permite rastreamento das interações do usuário em sistemas. A utilização dessa ferramenta, gratuita e de código aberto, possibilita abordagens diferenciadas de questionários, como a geração de mapas de calor, mapas de tempo que os usuários se concentram em determinadas áreas de um site, entre outras possibilidades que serão discutidas neste trabalho.

Apesar de não terem sido encontrados estudos que utilizem ferramentas de rastreamento de dados para avaliação de UX e suas relações com portais de notícias, diversos trabalhos têm abordado a utilização de ferramentas de rastreamento para avaliar a UX, com a finalidade de adaptar e/ou personalizar sistemas interativos, recomendar ou, até mesmo, prever conteúdos, tais como os trabalhos de (Diego-Mas et al., 2019), (Chen et al., 2017) e (Bastug et al., 2018) que utilizam dados de rastreamento ocular e/ou do *mouse*, para melhoria da UX.

Assim, a partir da breve revisão de literatura realizada nesta seção, verifica-se que técnicas de monitoramento de mouse e de olhar têm sido amplamente investigadas para UX. Na próxima seção serão abordados os diferenciais e funcionamento da ferramenta utilizada por este estudo.

### 3 | FERRAMENTA T2-UXT

A ferramenta T2-UXT ao longo dos últimos anos segue se adaptando as necessidades que surgem que vem surgindo. Desenvolvida por pesquisadores da Universidade Federal do Pará (UFPA), a ferramenta possui os seguintes diferenciais: (i) execução da ferramenta de avaliação em um navegador (tornando-a extensiva, simples e adaptativa); (ii) rastreamento de mouse, teclado, ocular, gravação de voz e análise de sentimentos olhar com o mouse; (iii) o código-aberto, de fácil implementação e manuseio; (iv) propor uma metodologia baseada em técnicas de inteligência artificial, especificamente algoritmo de agrupamento; (v) possibilidade de aplicação em ambientes multimídia (Souza et al., 2019). Abaixo é apresentado o processo de interação do usuário com o sistema, bem como o passo a passo para coleta de dados realizado pela ferramenta:

#### 3.1 Interação com Sistema

Durante a fase de interação, o usuário navega entre as páginas do portal de notícia/sistemas, buscando as informações que os interessam, deixando rastros por meio do

mouse. Estes rastros são coletados na próxima etapa.

### 3.2 Coleta de dados

A ferramenta T2-UXT é composta de três módulos: *Single-View*, *Heatmap* e *DataFuzzy*. O software permite a coleta, organização e processamento de dados. A arquitetura da aplicação sustenta-se em um arranjo de tecnologias que garante flexibilidade através do uso de PHP e *JavaScript*, permitindo a captura e armazenamento de dados independentemente de plataformas específicas, e o alto desempenho no processamento de dados através da utilização de C#. A T2-UXT é desenvolvida a fim de registrar as interações de usuários, a partir de coleta de dados de rastreamento de olho, mouse, navegação, digitação, gravação de voz e análise de sentimentos.

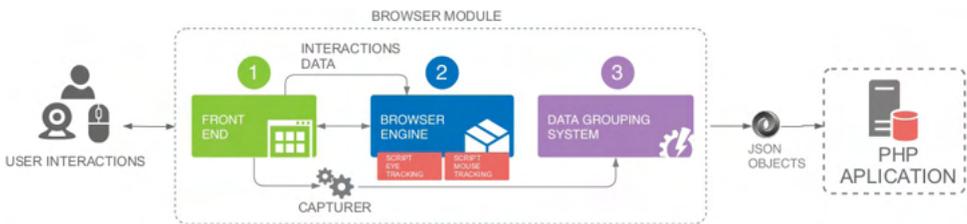


Figura 1. Metodologia para coleta de dados de interação do usuário.

A Figura 1 ilustra a metodologia de coleta de dados. O processo inicia com a aferição das interações do usuário por meio do *mouse* na interface do portal de notícia ou outros *website*. O módulo Navegador da T2-UXT, é responsável pela coleta de dados de interação durante a navegação em páginas *Web*, na forma de uma extensão desenvolvida para o navegador de *Internet Google Chrome*. Os dados capturados são agrupados em objetos JSON (*JavaScriptObjectNotation*) e enviados ao servidor de armazenamento, onde são convertidos para o formato XML (*ExtensibleMarkupLanguage*) e armazenados para utilização na etapa posterior.

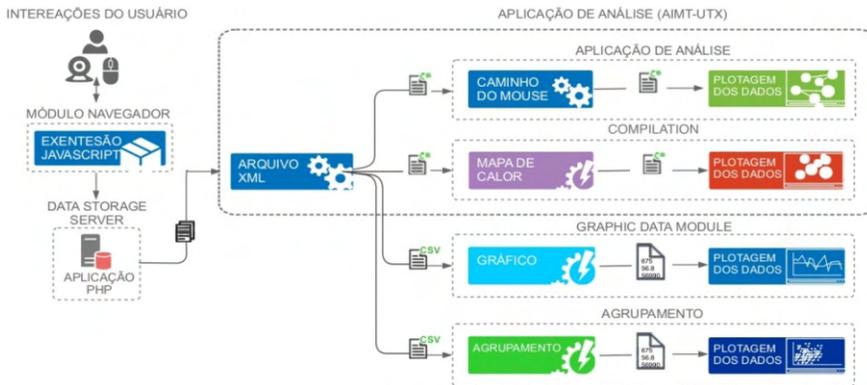


Figura 2. Metodologia para análise dos dados de interação do usuário.

A Figura 2 mostra a arquitetura do modelo de análise e gráficos. Após os dados de interação serem enviados para o servidor de armazenamento, os mesmos são analisados pela T2-UXT que, em seguida, gera como saídas o mapa de calor e o rastro de mouse. A ferramenta permite ainda a criação de arquivos utilizados para elaboração de Gráficos, onde, os dados armazenados em XML (*eXtensibleMarkupLanguage*) são convertidos para CSV (*Comma-separatedValues*) e analisados utilizando o *Microsoft Excel*, no qual são gerados gráficos.

#### 4 | ESTUDO DE CASO

Nesta seção, serão apresentados os detalhes da pesquisa realizada. Esse estudo utilizou 5 portais de notícias de diferentes regiões do Pará. Sendo 3 portais da Região Metropolitana de Belém, com mais de 2 milhões de habitantes (IBGE, 2018); 1 portal da Região Sudeste do Pará, com mais de 600 mil habitantes; e 1 portal da Região Oeste do Pará, contendo mais de 700 mil habitantes. Respectivamente: O Liberal, DoI, Belém Trânsito, Correio: o portal de Carajás e Portal Santarém. Esses portais foram escolhidos para este estudo por possuírem grandes números de acessos em suas respectivas regiões.

A partir de análises iniciais nos portais de notícias selecionados, foi planejado um roteiro, que contém tarefas a serem executadas pelo usuário em todos portais analisados, sendo estas

- (I) Abrir o Portal de Notícias;
- (II) Rolar o portal de forma completa (Scroll Lock);
- (III) Após rolar todo o portal, voltar ao topo (Scroll Lock);
- (IV) Buscar notícia pelo título “vacinação”;

As características a serem avaliadas são apoiadas nos estudos de Al-Radaiehet al. (2011) e Nielsen (1994). A Tarefa (I) objetiva verificar como se dá a primeira interação do

usuário, no que diz respeito a aparência da interface do portal; A Tarefa (II) visar analisar questões como organização, legibilidade e interação com as possíveis mídias disponíveis; A Tarefa (III) foi pensada para identificar a facilidade de navegação e disponibilidade de links; A Tarefa (iv) objetiva verificar eficiência, facilidade de navegação e acessibilidade na busca pelo termo “vacinação”.

Os caminhos que podem ser utilizados para cada uma das Tarefas são diversos, necessitando inclusive que o usuário retorne páginas, caso não encontre o item da Tarefa proposta. O teste foi executado com apenas um usuário, visto que esta proposta se concentra em validar o estudo de caso em si e não estabelecer um padrão médio de comportamento de usuário. Este usuário é membro do laboratório de pesquisa e não possuía familiaridade com nenhum dos portais, como intuito de fazer as Tarefas de forma não viciada. Entretanto, o usuário recebeu as instruções necessárias antes da realização do teste, para que não houvesse dúvidas ou intervenção durante a execução das Tarefas. Os testes foram realizados em um computador com sistema operacional *Microsoft Windows* 10 e navegador *Google Chrome*.

## 5 | RESULTADOS

Nesta seção serão apresentados os resultados das análises nos portais de notícias avaliados. Foram gerados mapas de calor para realizar as análises de comportamento do usuário. Os mapas de calor são uma representação gráfica sobre quais pontos (em pixels) de um portal de notícia houve maior atividade por parte do usuário. Sendo um resultado a partir das maiores interações com o mouse, considerando também cliques e rolagem da página (Ferreira, 2019).

Na Tabela 1 apresentamos os portais de notícias avaliados e o respectivo ID atribuído:

Portal de notícias / Endereço eletrônico	ID
O Liberal / <a href="http://oliberal.com.br">oliberal.com.br</a>	1
Dol / <a href="http://dol.com.br">dol.com.br</a>	2
Belém Trânsito / <a href="http://belemtransito.com.br">belemtransito.com.br</a>	3
Correio / <a href="http://correirodecaraajas.com.br">correirodecaraajas.com.br</a>	4
Santarém / <a href="http://portalsantarem.com.br">portalsantarem.com.br</a>	5

Tabela 1. Portais de notícias e respectivos ID.

A página inicial dos portais 1 e 3, conforme Figuras 3 (a) e 4, possuem propagandas e outros recursos de publicidade mais atrativos do que a notícia em destaque. Na Figura 3(a) é possível ver a página inicial do portal 1 que, por meio do mapa de calor, foi verificado

que a publicidade localizada na área central e superior do site recebeu mais interações<sup>6</sup> do que a notícia em destaque. A concentração de interações na parte superior foi motivada pelo fato de que o usuário estava tentando remover a propaganda. Na Figura 3 (b) é possível verificar página inicial do portal 2. O mapa de calor apontou que o usuário primeiramente interagiu com as funcionalidades e vídeo em destaque pelo portal. Resultado de uma página inicial mais limpa e com menos propagandas em destaque.

Na Figura 4, o portal 3 apresenta uma página inicial com anúncios muito grandes, inclusive um anúncio do tipo *AdSense*<sup>7</sup> cobrindo a propaganda de um anunciante principal do portal. Por meio dos mapas gerados, foi possível identificar que as publicidades comprometeram as interações com as notícias, inclusive, chegando a ocultar em alguns casos.



Figura 3 (a). Mapa de calor da tela inicial do portal 1.



Figura 3 (b). Mapa de calor da tela inicial do portal 2.



Figura 4. Mapa de calor da tela inicial do portal 3.

Na Figura 5 (a), é possível verificar que o portal 4 inseriu as funcionalidades e as

<sup>6</sup> Considera-se, a partir da entrevista feita com o usuário, que a atenção do mesmo estava onde o cursor do mouse estava localizado.

<sup>7</sup> AdSense é o serviço de publicidade oferecido pelo Google Inc, é uma maneira de os editores ganharem dinheiro com conteúdo on-line. Relacionando anúncios a um site com base nos visitantes e no conteúdo disponível.

notícias como destaque, na parte inferior do site ficou destinado o espaço para anúncio. O resultado é que os usuários interagem prioritariamente com as notícias e não tem a necessidade de tentar remover a notícia. Na Figura 5 (b), a disposição da página inicial do portal 5, apresenta vários tipos de *layouts* na mesma linha, dessa forma, o site deixa de ser perceptível pelo usuário. Dessa forma, a disposição que foi distribuída os conteúdos na página dificulta a interação do usuário.

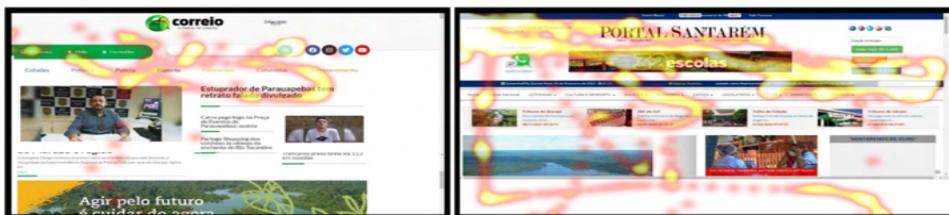


Figura 5 (a). Mapa de calor da tela inicial do portal 4.

Figura 5 (b). Mapa de calor da tela inicial do portal 5.

Por meio da Tarefa (iv) foi possível identificar que uma funcionalidade importante, a de busca de termos no portal de notícias, ficou comprometida nos portais 3 e 5. Assim, torna-se dificultoso que usuários, sobretudo os com menos experiência nesse tipo de site, possam realizar a leitura de uma notícia específica. Em contrapartida, os portais 2 e 4 deram destaque para essa funcionalidade em seus portais.

Conforme a Tarefa (iv), as notícias buscadas em todos os portais de notícias foram a partir do termo “vacinação”, dessa forma ao fim das notícias era comum que contivesse uma lista de locais disponíveis para vacinar contra a COVID-19. Contudo, boa parte dos portais de notícias se limitou a publicar relação de locais, sem explorar os recursos multimídias (fotos, imagens, galerias, etc.), diminuindo a possibilidade de uma leitura mais confortável e de interesse que o usuário permanecesse por mais tempo no ambiente.

Nas Figuras 6 (a) e 6 (b), é possível realizar uma comparação entre os portais 1 e 2, respectivamente, onde a primeira matéria que continha imagem recebeu mais interação do usuário que a segunda, sem imagem ou outros recursos multimídias. O portal 1, de forma estratégica, inseriu ao longo da matéria uma funcionalidade que o usuário pudesse se conectar com uma rede social. Assim, aumenta a possibilidade que o público possa acompanhar, por meio de suas redes sociais, a rotina de informações publicadas no portal.



Figura 6 (a). Mapa de calor em matéria alusiva a “vacina” do portal 1.



Figura 6 (b). Mapa de calor em matéria alusiva a “vacina” do portal 2.

Na Figura 7, no portal 3, a matéria apresenta somente texto. Um ponto a ser destacado é que os anúncios chegam a dificultar a leitura do conteúdo veiculado no referido portal. Os anúncios também não contêm imagens ilustrativas, dessa forma, a interação do usuário com a matéria foi prioritariamente para tentar remover os anúncios.



Figura 7. Mapa de calor de matéria alusiva a “vacina”, portal 3.

O portal 4 utilizou recursos de imagens ao longo da matéria, conforme apresentado na Figura 8 (a). Por meio do mapa de calor, foi possível ver que isso proporcionou maior interação do usuário com a publicação veiculada pelo portal. Entretanto, apesar de uma quantidade considerável de anúncios ao lado direito do portal, também foi utilizada na área central inferior do site, uma publicidade que proporcionou dificuldades na leitura em determinados trechos.



Figura 8 (a). Mapa de calor em matéria alusiva a “vacina” do portal 4. Figura 8 (b). Mapa de calor em matéria alusiva a “vacina” do portal 5.

A respeito da matéria sobre vacinação realizada pelo portal 5, Figura 8 (b), foi possível identificar o uso de recursos visuais. Contudo, o tamanho reduzido da fonte utilizada na escrita dificulta a leitura. Outro ponto que merece atenção é que algumas palavras não reconhecidas pela plataforma de escrita do portal ficavam sendo exibidas com um sublinhado vermelho, algo similar ao que o *Microsoft Word* e demais plataformas de escritas fazem. Assim, passa a impressão para o usuário que a matéria foi veiculada com erros ortográficos.

## 6 | CONCLUSÃO

Apesar do brasileiro muito utilizar portais de notícias como meio de obtenção de informação, ainda é necessário ampliar as pesquisas acadêmicas a respeito das construções de UX nesse portais. Dessa forma, este artigo apresentou um estudo de caso que aponta desafios na UX portais de notícias, resultados gerados a partir de utilização de uma ferramenta de rastreamento de dados, indo além de questionários de avaliação de usuários.

Nesse estudo foi realizada de quatro tarefas, apontando desafios a UX dos portais avaliados: anúncios e/ou outros recursos de publicidade localizados de forma mais atrativa do que a notícia a ser destacada; alguns anúncios dificultam a interação dos usuários com a notícia; funcionalidades importantes tendo pouco destaque; excesso de informações visuais; vários tipos de *layout* na mesma linha, dificultando que o site seja perceptível pelo usuário.

Enquanto limitações desta pesquisa, o estudo se concentrou na utilização de dados rastreados por *mouse*, contudo a ferramenta ainda possibilita a obtenção de dados rastreados pelo teclado, olhos, gravação de voz e análise de sentimentos. Os resultados foram discutidos a partir dos mapas de calor, contudo também seria possível análises por meio de gráficos, quantidade de cliques, tempo de execução de cada atividade, entre outros.

Nesse sentido, novos estudos estão em andamento e a pesquisa deverá ser complementada. Como trabalhos futuros, pretende-se: (a) realizar análises a partir das outras funcionalidades permitidas pela T2-UXT; (b) comparar os achados obtidos pela ferramenta com resultados de questionários; (c) apresentar discussões relacionadas a portais de notícias e consumo de informação; (d) contextualizar o *lôcus* da pesquisa, construindo um diálogo sobre a inclusão digital e a relação com a internet vivida no estado do Pará.; (e) testar novas Tarefas com uma quantidade mais representativa de usuários, visando estabelecer mapas de calor com médias de uso representativas.

## REFERÊNCIAS

- AL-RADAIDEH, Q. A.; ABU-SHANAB, E.; HAMAM, S.; ABU-SALEM, H. **Usability Evaluation of Online News Websites: A User Perspective Approach**. International Journal of Human and Social Sciences, 2011.
- ARANYI, G.; SCHAİK, P. V. **Modeling User Experience With News Websites**. Vol. 66. Journal of the Association for Information Science and Technology, 2015. Doi: 10.1002/asi.23348.
- BANUENUMAH, W.; SEKYERE, F.; DOTCHE, K. A. **Field survey of smart metering implementation using a simple random method: A case study of new juaben municipality in ghana**. IEEE PES Power Africa, 2017.
- BASTUG, R. S.; YESILKAYA, B.; UNAY, M.; AKAN, A. **Virtual mouse control by webcam for the disabled**. P. 1–4. Medical Technologie National Congress (TIPEKNO), 2018.
- CHEN, Y.; LIU, M.; ZHANG, M.; MA, S. **User Satisfaction Prediction with Mouse Movement Information in Heterogeneous Search Environment**. Vol. 29, no. 11, p. 2470-2483. IEEE Transactions on Knowledge and Data Engineering, 2017. Doi: 10.1109/TKDE.2017.2739151.
- Consumo de notícias digitais no Brasil aumenta na pandemia. Meio e Mensagem. 2020. Disponível em: <https://www.meioemensagem.com.br/home/midia/2020/09/25/consumo-de-noticias-digitais-no-brasil-aumenta-na-pandemia.html>. Acesso em: 10 nov. 2021.
- DIEGO-MAS, J. A.; GARZON-LEAL, D.; POVEDA-BAUTISTA, D.; ALCAIDE-MARZAL, J. **User-interfaces layout optimization using eye-tracking, mouse movements and genetic algorithms**. Vol. 78, p. 197–209. Applied Ergonomics, 2019.
- Ferreira, K. **Mapa de calor: o que é, qual a funcionalidade e 3 ferramentas de heatmap**. Rock content, 2019. Disponível em: <https://rockcontent.com/br/blog/mapa-de-calor/>. Acesso em: 10, jan. 2022.
- IBGE – INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Censos 2018. Projeção da população do Brasil e das Unidades da Federação. Rio de Janeiro: IBGE, 2018.
- NIELSEN, J. **Heuristic evaluation**. Em Nielsen, J., and Mack, R.L. (Eds.), Usability Inspection Methods, John Wiley & Sons, New York, NY, 1994.
- PAOLO, V. **Eye movements in visual search: cognitive, perceptual and motor control aspects**. P. 353-93. Reviews of óculo motor research, 1990.
- PODER DATA. Principais meio para se informar. 2020. Disponível em: <https://www.poder360.com.br/midia/41-usam-veiculos-jornalisticos-na-internet-para-ler-noticias/>. Acesso em 16 dez. 2021.

REUTERS INTITUTE. **Digital News Report**. 10 edição, 2021. Disponível em: [https://reutersinstitute.politics.ox.ac.uk/sites/default/files/2021-06/Digital\\_News\\_Report\\_2021\\_FINAL.pdf](https://reutersinstitute.politics.ox.ac.uk/sites/default/files/2021-06/Digital_News_Report_2021_FINAL.pdf). Acesso em: 10 dez. 2021.

ROMEIRO PAULINO, RITA DE CÁSSIA; DA ROCHA SERUFFO, MARCOS CÉSAR; LISBOA EMPINOTTI, MARINA; SILVA DE SOUZA, KENNEDY EDSON; PIMENTA, ANA CARLA. **Análise da Experiência do Usuário (UX) de Narrativa Transmídia através de Mouse-Tracking**. COMUNICAÇÃO & INOVAÇÃO (ONLINE), v. 22, p. 41-61, 2021.

SILVA, G. A.; PETRUCELLI, E. E. **PRINCÍPIOS DE UX DESIGN NO DESENVOLVIMENTO DE WEBSITES**. Vol. 15, p. 28-38. Interface tecnológica, 2018.

SOUZA, K. E. S; SERUFFO, M. C. R; DE MELLO, H. D; SOUZA, D. D. S; VELLASCO, M. M. B. R. **User Experience Evaluation Using Mouse Tracking and Artificial Intelligence**. IEEE, 2019. Doi: 10.1109/ACCESS.2019.2927860.

SOUZA, K. E. S; ZACARIAS, R. O; SERUFFO, M. C. R; SANTOS, R. P. **T2-UXT: A Tool to Support Transparency Evaluation in Software Ecosystems Portals**. Association for Computing Machinery, New York, NY, USA, 415–420. Proceedings of the 34th Brazilian Symposium on Software Engineering, 2020. DOI: 10.1145/3422392.3422502.

# CAPÍTULO 6

## ROTEAMENTO DE VEÍCULO GUIADO AUTONOMAMENTE PARA ARMAZÉNS INTELIGENTES

Data de aceite: 01/03/2022

### Wesley Marques Lima

Graduando em Engenharia Física Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Janaúba MG, Brasil

### Honovan Paz Rocha

Instituto de Engenharia, Ciência e Tecnologia Universidade Federal dos Vales do Jequitinhonha e Mucuri Janaúba MG, Brasil

**Resumo:** A área de logística, presente em grande parte da indústria, comércio e serviços, é uma das que mais necessitam de investimento em tecnologia para otimização de recursos, de maneira que o produto ou serviço final da empresa se mantenha no mercado de maneira estável e mais competitiva. Com o intuito de aperfeiçoar a logística vários trabalhos da literatura trazem estudos com Veículos Guiados Autonomamente (*Automated Guided Vehicles-AGVs*) com a intenção de tornar autônomo o transporte interno de mercadorias e conseqüentemente mais eficiente. Um dos desafios dos *AGVs* é a roteirização, que é a busca por caminhos mínimos entre um estado inicial e os objetivos. Com O objetivo de estudar e solucionar este problema, o presente trabalho busca modelar computacionalmente um projeto real de armazém inteligente e implementa 3 algoritmos, Busca de Custo Uniforme, A\* e Colônia de formigas, que

são aplicados ao modelo representado através de um grafo. Eles são implementados para várias instâncias do problema, e em seguida seus resultados são avaliados e comparados.

**PALAVRAS-CHAVE:** Busca de Custo Uniforme, A\*, Colônia de formigas, Roteirização, Armazém inteligente, AGVs

## 1 | INTRODUÇÃO

Considerando-se o rápido avanço tecnológico e a constante busca por redução de custos, a necessidade de tornar os grandes centros de armazenamento e distribuição de produtos mais autônomos é cada vez maior. Embora o Brasil caminhe a passos curtos nesta direção, países da Europa, Ásia e América do Norte já possuem um alto nível de automação aliado à utilização das mais recentes tecnologias para a otimização [1].

O setor industrial no Brasil possui atualmente certa defasagem tecnológica, e devido a isso as companhias brasileiras tendem a perder competitividade frente às concorrentes estrangeiras e podem até deixar de existir em um tempo não tão distante. Em um estudo em 2020 foram realizadas entrevistas em várias organizações em que foi possível constatar que 78% delas ainda estão no estágio 1 e 2 da indústria, enquanto alguns países caminham para a quarta revolução industrial [2].

A logística na indústria abrange vários processos, como a escolha do local pra se

armazenar determinado produto, o fluxo de materiais, dentre outros. A geração de um fluxo otimizado de produtos é de grande importância para que a logística seja eficiente. Desta maneira, investimentos em melhorias operacionais que visam aumentar a velocidade dos processos através de um fluxo eficiente de materiais, são cada vez mais frequentes [3].

Assim, pode-se perceber que otimizar o transporte de materiais é essencial para uma logística eficiente. O ganho em eficiência ocorre porque uma quantidade significativa de tempo é despendida com o transporte desses materiais, além disso é necessário o uso de mão-de-obra nesta tarefa, o que dificulta a utilização de alguns recursos baseados no uso de computadores, como em sistemas de monitoramento e controle [1]. Nas tarefas de transporte de materiais em ambientes como armazéns, portos e distribuidores de produtos a mão de obra é sobrecarregada com operações de carga e descarga, monitoração de recursos e planejamento de rotas de transporte. Essa sobrecarga somada ao uso de maquinário pesado nas atividades e a necessidade de concentração nas tarefas, leva os funcionários ao cansaço físico e mental, diminuindo seu desempenho e aumentando a chance de ocorrerem acidentes [4].

A automação do transporte de materiais em ambientes internos é essencial para aumentar a produtividade e reduzir custos. Por isso a utilização de Veículos Guiados Autonomamente (*Automated Guided Vehicles-AGVs*) tem sido cada vez mais frequente. Visando aprimorar sistemas que utilizam *AGVs* para o transporte de materiais, vários trabalhos tem sido desenvolvidos nessa área com o objetivo de melhorar o sistema de roteamento desses veículos. Nesses sistemas são usados diversos métodos para encontrar a melhor rota para cada *AGV*, desde algoritmos clássicos na resolução de problemas por meio de busca até meta-heurísticas inspiradas no comportamento de elementos da natureza.

Tendo em vista que automatizar o transporte de materiais é fundamental para aumentar a produção e reduzir custos, o presente trabalho tem como objetivo modelar computacionalmente um projeto real de armazém inteligente e implementar métodos da literatura para encontrar rotas mínimas para um *AGV* que atua nesse armazém, com o intuito de analisar o funcionamento desses métodos em um ambiente real.

Este trabalho é organizado da seguinte forma: na Seção II são apresentados alguns trabalhos que resolvem problemas semelhantes ao problema proposto. Na Seção III os algoritmos usados para encontrar rotas mínimas são descritos. Na seção IV é descrito como o armazém inteligente é modelado e representado através de um grafo, mostra também como os algoritmos usados para a busca de rotas mínimas são implementados. A Seção V apresenta e discute os principais resultados obtidos. Por fim a Seção VI apresenta as considerações finais.

IV é descrito como o armazém inteligente é modelado e representado através de um grafo, mostra também como os algoritmos usados para a busca de rotas mínimas são implementados. A Seção V apresenta e discute os principais resultados obtidos. Por fim a

Seção VI apresenta as considerações finais.

## 2 | TRABALHOS RELACIONADOS

O problema da roteirização de veículos em ambientes com AGVs é objeto de estudo em diversos trabalhos nas últimas décadas, alguns deles são brevemente descritos a seguir.

Em [5] os autores propõem a otimização de caminhos no transporte interno de ferramentas. Para resolver o problema o armazém é representado através de um grafo e os algoritmos Busca de Custo Uniforme, A\*, Floyd-Warshall e BellmanFord são utilizados para otimização das rotas dos veículos autônomos. Os autores apontam que os métodos geram resultados similares, com o Busca de Custo Uniforme sendo ineficiente quanto ao custo computacional.

Um problema semelhante é abordado em [1], onde um armazém também é representado como um grafo e o algoritmo de Dijkstra é usado para encontrar o caminho mínimo para os veículos autônomos. Outros algoritmos como A\* e Campos Potenciais são utilizados para diminuição do número de curvas do caminho encontrado. O trabalho mostrou a viabilidade dos algoritmos, que com custo computacional baixo se mostraram uma boa opção para resolução deste tipo de problema.

No trabalho [6] o problema é formulado como um sistema multiagente. Nessa abordagem o principal problema é a associação de tarefas, que foi resolvida utilizando quatro estratégias diferentes: CNET, *Fuzzy*, DynCNET e FiTA. Os autores utilizaram um Algoritmo Genético para o ajuste de parâmetros dos métodos e os resultados obtidos mostraram que o DynCNET obteve o melhor desempenho, seguido de perto pela estratégia *Fuzzy*.

Para gerar rotas livres de colisões para um veículo autônomo, em [7] os autores utilizam um algoritmo de planejamento de caminho livre de colisões, o algoritmo foi testado para vários layouts, pontos iniciais e objetivos. Os resultados mostram que o algoritmo obtém rotas livres de colisões em todos os casos. O trabalho considera um único robô móvel, necessitando de adaptação para ambientes com múltiplos AGVs.

O algoritmo A\* é utilizado para determinar o caminho mínimo para dois AGVs no trabalho [8]. Para o tratamento de colisões, um sistema dinâmico com janelas de tempo é utilizado, assim quando houver colisão um AGV espera para que o outro passe, de maneira que a colisão seja evitada.

Em [9] o sistema utilizado armazena informações de varias fontes em nuvem, e com isso é possível a detecção antecipada de obstáculos. Após a detecção, uma estratégia conhecida como *Local Path Planning* é utilizada para criação de desvios para as rotas com obstáculos. Nos sistemas atuais de AGVs desvios de rota normalmente não são permitidos, sendo mais comum a espera pela desobstrução do caminho, entretanto,

a funcionalidade *Local Path Planning* permite ao *AGV* continuar sua tarefa mesmo com obstáculos, diminuindo uma possível dependência temporal no sistema.

Para resolver o problema de roteamento de veículos, o trabalho [10] utiliza o algoritmo de Dijkstra com uma modificação que permite ao método armazenar não apenas a primeira solução ótima encontrada, mas também as demais soluções, de maneira que estas são utilizadas para resolver problemas com colisões entre os *AGVs*. Ao detectar conflitos em uma rota, é analisada a viabilidade de se usar uma das demais soluções encontradas pelo Dijkstra modificado. Caso nenhuma das rotas encontradas pelo Dijkstra modificado seja livre de conflitos, outras formas para tratar o problema são: os *AGVs* aguardam a desobstrução do caminho por um período de tempo, as rotas são refeitas considerando as mudanças no ambiente, e por fim, caso tudo tenha falhado, as tarefas são reenviadas. Esta abordagem mostrou mais eficiência no quesito tempo do que método tradicional de tempo de espera por colisão.

No trabalho [11] o problema de otimização de rotas para veículos autônomos em armazém inteligente é resolvido utilizando algoritmos como o  $A^*$  para encontrar caminhos mínimos entre os veículos e as tarefas. Devido a possíveis congestionamentos e obstáculos, essa rota é constantemente otimizada, assim as rotas são replanejadas dinamicamente de acordo com o tráfego local, condições e paradas devido a eventos inesperados. O sistema melhora a eficiência de uma frota de veículos autônomos em termos de entrega e tem custo computacional viável para aplicações no setor industrial, isso foi mostrado através de testes em ambiente real de pequena escala.

Uma estratégia de despacho dinâmico é utilizada em [12], ela se baseia em alocar tarefas aos veículos autônomos mais próximos, porém pode ser muito custoso encontrar um caminho mais curto livre de conflitos e de loops, por isso é proposta uma estratégia de roteamento que adota o algoritmo  $k$ -ésimo caminho mais curto, assim se obtendo uma rota ideal e viável computacionalmente. O trabalho mostra a criação de um sistema para encontrar rotas para veículos autônomos sem conflitos ou loops. Os algoritmos são executados em tempo polinomial e podem ser aplicados em sistemas de larga escala, os experimentos foram simulados e mostram a eficiência dos algoritmos.

No trabalho [13] o problema de possíveis conflitos é resolvido usando um método de roteamento dinâmico para controle de múltiplos veículos autônomos. Os conflitos são evitados usando espaço de estados discretos em mapa tridimensional para identificar e resolver os conflitos. O desempenho do algoritmo é avaliado comparando-o com o algoritmo BFS.

O algoritmo tem baixo esforço computacional se comparado a métodos convencionais e é muito bom para ambientes dinâmicos onde a flexibilidade dos veículos autônomos tenha que ser alta. Os testes de validação foram feitos via simulação mostrando a aplicabilidade do algoritmo.

A abordagem usada em [14] é bem diferente das citadas anteriormente, pois

utiliza em seu trabalho a meta-heurística colônia de formigas para resolver o problema de roteamento de veículos com janela de tempo. Para resolução do problema o trabalho utiliza o *Multiple Ant Colony System (MACS)* que é a utilização de dois *Ant Colony System (ACS)*, onde cada um otimiza um objetivo, porém os resultados de um dos ACS influencia na solução do outro, assim as duas colônias trabalham de forma distinta, mas simultânea na busca de melhores soluções. O *MACS* foi desenvolvido em *Matlab* e foi aplicado na resolução de problemas em grafos com 25 e 50 nós, os resultados para esses problemas foram satisfatórios, pois as soluções foram próximas das ótimas. Para problemas com mais de 100 nós o algoritmo se mostrou ineficiente pois o tempo de execução foi muito alto, isso devido aos scripts criados no trabalho não terem sido otimizados no quesito performance.

Inspirado pelos trabalhos apresentados, no presente trabalho serão implementados alguns dos métodos citados acima, são eles: Busca de Custo Uniforme,  $A^*$  e *ACS*.

### 3 | ALGORITMOS IMPLEMENTADOS

Nesta seção detalharemos os métodos utilizados neste trabalho para resolução do problema de busca pela rota mínima para um *AGV*.

#### *A. Busca de Custo Uniforme*

A Busca de Custo Uniforme é um algoritmo de busca sem informação, ou seja, ele só tem acesso às informações contidas no grafo. A estratégia de busca adotada aqui é expandir sempre o nó com menor custo de caminho  $g(n)$  na árvore de busca, por isso sua borda é uma fila de prioridades ordenada segundo o custo  $g$ .

Usando o problema de buscar o caminho mais curto entre dois vértices de um grafo como exemplo, o funcionamento do algoritmo ocorre da seguinte forma: a lista de prioridades é inicializada com o nó que corresponde ao estado inicial, caso ele não seja o objetivo é retirado da lista de prioridades, expandido e colocado na lista de explorados. Para isso todas as ações possíveis são consideradas (cada ação leva a um vértice adjacente ao que foi expandido), e todos os nós filhos exceto os presentes no conjunto de explorados são colocados na lista de prioridades de acordo com o custo do caminho até o momento. Os nós cujo estado já existe em algum outro nó da lista de prioridades, será substituído pelo nó com menor custo. O algoritmo é finalizado quando a lista de prioridades não possuir mais nenhum nó [15].

O algoritmo é completo e ótimo, ou seja, sempre que houver solução para um dado problema, a solução ótima será encontrada. Ao observar o funcionamento dessa busca é fácil perceber essa característica, pois o nó expandido é sempre o que tem caminho de menor custo até esse nó, sendo assim, quando o objetivo for encontrado o caminho que chegou a ele será o menor existente.

Outros aspectos importantes do algoritmo são as complexidades de tempo e espaço,

que para o pior caso é  $O(b^{1+\lceil C^*/\epsilon \rceil})$  para ambos,  $C^*$  representa o custo da solução ótima,  $\epsilon$  é uma constante positiva que delimita o custo mínimo de uma ação, e  $b$  é o fator de ramificação. Complexidade de tempo e espaço são os grandes problemas enfrentados por esse método, apesar de sempre achar a solução ótima, é inviável para problemas grandes [15].

Na seção seguinte um algoritmo um pouco mais eficiente será apresentado, ele funciona de forma muito semelhante ao Busca de Custo Uniforme, o que os difere é a função de custo.

### B. A\*

O A\* (pronuncia-se “A estrela”) é um algoritmo de busca informada, ou seja, uma informação heurística  $h(n)$  é utilizada para direcionar a busca. Assim como na Busca de Custo Uniforme, no A\* a borda também é uma lista de prioridades, porém a mesma é ordenada de acordo com uma função de custo  $f(n) = g(n) + h(n)$ .

O funcionamento do algoritmo é semelhante ao da Busca de Custo Uniforme, já que a única diferença entre eles é a função de custo que ordena a lista de prioridades.

A busca A\* é completa, porém sua otimalidade depende da heurística adotada. Uma das condições para que o algoritmo seja ótimo é a heurística ser admissível, que significa que ela nunca deve superestimar o custo até o objetivo. Podemos supor um problema onde se queira encontrar o menor caminho entre dois vértices quaisquer de um grafo que represente um mapa, uma heurística admissível nesse caso seria a distância em linha reta entre os vértices e o vértice objetivo. É perceptível que  $h(n)$  nesse caso não superestima o custo até o objetivo, pois a menor distância entre dois pontos é sempre uma reta.

A segunda condição é apenas para problemas de busca em grafos, e diz que a heurística deve ser consistente, essa sendo inclusive uma condição mais forte que a primeira, pois toda heurística consistente é também admissível. Ela será consistente se, o valor de  $h(n)$  não for maior que  $f(n') - g(n)$ , sendo  $n'$  um nó gerado por  $n$ , ou seja,  $h(n) \leq g(n') - g(n) + h(n')$ .

As complexidades de tempo e espaço são um problema também para o A\*, com o uso de uma boa heurística ele ganha eficiência com relação ao tempo se comparado com algoritmos de busca cegos. Para esse algoritmo a complexidade de espaço é o maior problema, pois geralmente atinge um limite de memória bem antes de atingir um dado limite de tempo, isso torna o algoritmo impraticável pra problemas grandes, heurísticas não consistentes podem ser utilizadas tornando o A\* mais eficiente, porém não ótimo.

Em termos dos erros relativo e absoluto da heurística podemos estimar a complexidade deste algoritmo. O erro absoluto é dado por  $\Delta \equiv h^* - h$ , onde  $h^*$  é o custo real a partir da raiz para o objetivo, e o erro relativo é definido como  $\epsilon \equiv (h^* - h)/h^*$ . Para

problemas simples com apenas um objetivo a complexidade de tempo do  $A^*$  é exponencial no erro máximo absoluto, ou seja,  $O(b^d)$ , que para os custos de passo constante, pode ser escrito como  $O(b^e d)$ , onde  $d$  é a profundidade da solução [15].

Para próxima seção será apresentando um algoritmo bem diferente dos vistos até então, o Colônia de formigas, que é um algoritmo estocástico muito utilizado na resolução de problemas combinatórios.

### C. Colônia de formigas

O algoritmo Colônia de formigas usado no trabalho é o *Ant Colony System* (ACS), ele foi proposto por Dorigo e Gambardella (1997) [16], e aplicado inicialmente para a resolução do problema do caixeiro viajante (TSP), que é um problema clássico da matemática e faz parte da classe dos NPdifíceis, que são uma série de problemas que não podem ser resolvidos em tempo polinomial por algoritmos exatos.

As formigas inicialmente caminham de forma aleatória, porém nas iterações subsequentes tendem a seguir por caminhos com feromônio (o feromônio é a substância que as formigas usam para se comunicarem). Essa substância evapora a uma certa taxa, fazendo com que caminhos onde passam poucas formigas sejam cada vez menos atrativos e assim elas começam a convergir para caminhos onde passam a maior parte das formigas.

No ACS, as formigas ao fazerem sua trilha, podem ser mais conservadoras e seguir o melhor caminho conhecido por elas, ou mais desbravadoras se baseando em uma probabilidade que as permite encontrar novos caminhos. Para auxiliar na escolha de uma das opções um parâmetro  $q_0$  que assume um valor entre 0 e 1 foi criado, em seguida um número aleatório  $q$  entre 0 e 1 é sorteado, caso ele seja menor que  $q_0$  o caminho escolhido pela formiga será o melhor conhecido por ela, caso contrário ela escolherá seu caminho baseada em uma probabilidade.

Com o parâmetro  $q_0$  podemos deixar as formigas mais desbravadoras ao diminuir o valor de  $q_0$  ou mais conservadoras ao aumentar o valor de  $q_0$ , i.e., aumentando a diversidade no algoritmo. Portanto temos que a formiga no ACS decide para qual nó seguir de acordo com (1).

$\text{argmax}[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta$ , se  $q \leq q_0$

$$J = \begin{cases} \text{argmax}_{j \in N^k} [\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta, & \text{se } q \leq q_0 \\ S, & \text{caso contrário} \end{cases} \quad (1)$$

Onde  $t$  representa a iteração corrente,  $\tau_{ij}$  e o feromônio presente na aresta  $ij$ ,  $\eta_{ij}$  é o inverso da distância entre os pontos  $ij$ ,  $N_i^k$  é um conjunto de nós adjacentes ao nó  $i$  que ainda não foram visitados pela formiga  $k$ ,  $\alpha$  e  $\beta$  ditam a relevância do feromônio e da distância respectivamente e  $S$  é uma variável discreta com distribuição de probabilidade

dada por (2):

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)] \cdot [\eta_{il}]^\beta}, \quad \forall j \in N_i^k \quad (2)$$

Em que  $P_{ij}^k$  é a probabilidade da formiga  $k$  percorrer a aresta  $ij$ . Após a formiga trilhar seu caminho até o objetivo ocorre a atualização do feromônio nas rotas, no ACS acontece uma atualização local e uma global. Na local toda vez que A. uma formiga passa por um arco o valor de seu feromônio é alterado de acordo com (3).

$$\tau_{ij}(t) \leftarrow (1 - \rho)\tau_{ij}(t) + \rho\tau_0 \quad (3)$$

Em que  $\tau_0$  é o feromônio inicial nas rotas e  $\rho$  é o coeficiente de evaporação do feromônio. O objetivo dessa atualização é diminuir a influência de rotas iniciais tornando as formigas capazes de fugir de caminhos nãoótimos. Já a atualização global do feromônio nas rotas ocorre de acordo com (4).

$$\tau_{ij}(t) \leftarrow (1 - \rho)\tau_{ij}(t - 1) + \rho\Delta\tau(t) \quad (4)$$

A atualização aqui ocorre em todos os arcos do grafo, porém apenas as arestas que compõem a melhor solução encontrada pelas formigas recebem feromônio, as demais sofrem apenas evaporação, i.e.,  $\Delta\tau(t)$  é zero para arestas que não fazem parte da melhor solução. A quantidade de feromônio que será depositado depende da qualidade da solução, pois  $\Delta\tau(t)$  é o inverso da distância do melhor caminho encontrado na iteração  $t$ . O algoritmo foi implementado com uma adaptação, ela é descrita na Seção IV.

## 4 | METODOLOGIA

Nesta seção é descrito como o trabalho foi desenvolvido, desde a modelagem de um projeto de armazém até a aplicação de algoritmos para encontrar rotas mínimas.

Com o objetivo de simular um armazém de médio porte, foi utilizado o *software AnyLogic*, que possibilita uma grande gama de aplicações, sendo uma delas a modelagem de ambientes reais. Uma versão gratuita do *software* foi utilizada para projetar graficamente o armazém inteligente que servirá de base para o trabalho, ele está representado na Figura 1. Um material didático disponível em [17] foi utilizado como guia no uso do *software*.

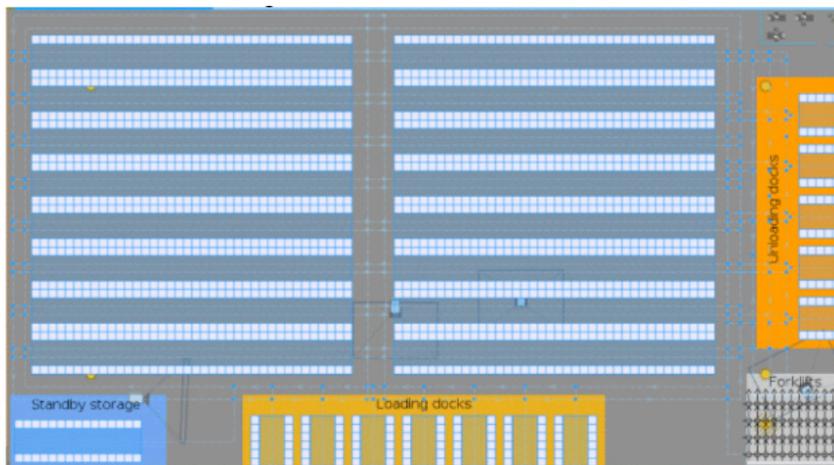


Figura 1. Modelo do armazém.

As informações do modelo desenvolvido ficam em um arquivo XML gerado pelo próprio *software*. Essas informações foram usadas para representar o modelo na forma de um grafo, nele os vértices irão representar os compartimentos das prateleiras e as arestas representarão as ligações entre esses compartimentos.

#### A. Geração do grafo

O grafo para representação do armazém foi gerado utilizando as informações do modelo contidas no XML. Para acessar o xml foi desenvolvido um algoritmo para realização do *parser*, que além de verificar se o documento obedece à sintaxe do XML, identifica se ele está de acordo com as regras definidas por um vocabulário definido pelo software gerador. O padrão para manipulação de documentos DOM (*Document Object Model*) foi utilizado. O DOM é baseado no modelo *tree-based*, ou seja, oferece uma visão de documento estruturado em árvore.

O *parser* desenvolvido, descrito pelo Algoritmo 1, carrega todo o documento para a memória e tem à disposição uma visão de todos os objetos na forma de uma árvore, que é percorrida para extração dos dados de interesse para representação do armazém.

---

**Algoritmo 1** Parser XML

---

**Input:** Raiz do XML, TAG do objetivo, Informações

**Output:** Informações do armazém

```
1: if Raiz do xml é não vazio then
2:   if Informações é não vazio then
3:     return Informações
4:   end if
5:   foreach Raiz do
6:     if Informações é vazio then
7:       if filho da Raiz = TAG do objetivo then
8:         Informações ← conteúdo de filho da Raiz
9:       end if
10:      Informações ← Pega informação xml(filho da
11:      raiz, TAG do objetivo, Informações)
12:    else
13:      Sair do loop
14:    end if
15:  end for
16: end if
return Informações
```

---

Após a realização do *Parser*, o Algoritmo 2 foi implementado para geração do grafo. As informações extraídas são: posição de início de cada prateleira do armazém, como ela é dividida e a dimensão de cada divisória da mesma. Com acesso a essas informações são estipuladas as posições de cada divisória, e um vértice é criado para cada uma delas, tendo assim uma lista de vértices. Em seguida é criada uma lista de arestas, que guarda as adjacências existentes no grafo. Por fim essas informações são armazenadas em uma estrutura do tipo dicionário, finalizando a representação do grafo com implementação em linguagem *Python* versão 3.7.

Na Figura 2 é representado o grafo gerado, este grafo contém 1380 vértices. Ao compararmos com a Figura 1, pode-se perceber que a Figura 2 está invertida, o que ocorre devido à representação do armazém no *Anylogic* representar o eixo y com seu negativo orientado para cima. O grafo criado será utilizado como base para a resolução do problema proposto.

### *B. Busca pelos caminhos mínimos*

Os algoritmos Busca de Custo Uniforme, A\* e ACS foram implementados para resolver o problema de busca proposto. Inicialmente foram implementados os 2 algoritmos clássicos na resolução de problemas de busca em grafo e na sequência o algoritmo de inteligência de enxame. Os pseudocódigos desses

---

**Algoritmo 2** Geração do grafo

---

**Input:** Informações do armazém**Output:** Grafo do armazém

- 1: **foreach** prateleiras **do**
  - 2:   crie um **vértice** para cada divisória da prateleira
  - 3:   adicione o **vértice** em Lista de vértices
  - 4: **end for**
  - 5: **foreach** Lista de Vértices **do**
  - 6:   crie uma lista de adjacentes para o **vértice**
  - 7:   adicione à lista de adjacentes os vértices adjacentes a **vértice**
  - 8:   adicione à lista de adjacentes as distâncias entre **vértice** e vértices adjacentes
  - 9:   adicione lista de adjacentes a Lista de arestas
  - 10: **end for**
  - 11: Grafo dicionário  $\leftarrow$  Cria dicionário(Lista de vértices, Lista de arestas)
  - 12: **return** Grafo dicionário
- 

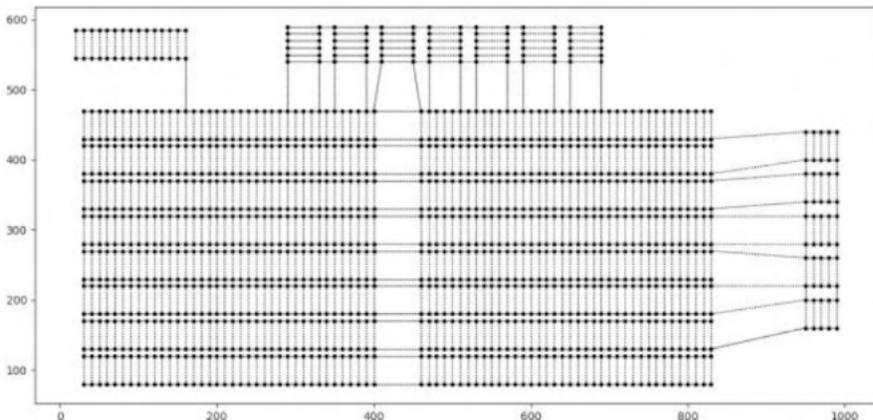


Figura 2. Ilustração do grafo gerado.

Algoritmos não são apresentados por serem bem conhecidos da literatura.

Os algoritmos, Busca de Custo Uniforme e A\* foram desenvolvidos exatamente como descritos na Seção III, sendo a heurística usada no A\* a distância euclidiana entre os vértices do grafo e o vértice objetivo. Para o ACS uma alteração foi feita, o  $\eta$  que tradicionalmente é o inverso do tamanho de um arco do grafo, foi mudado para o inverso da distância do vértice ao objetivo, a métrica usada foi a distância de Manhattan. A alteração tem o intuito de aumentar a probabilidade de a formiga ir no sentido do objetivo, fazendo também com que o método chegue à convergência com maior rapidez.

Os parâmetros utilizados no ACS foram obtidos através de uma busca em grid, onde os selecionados foram:  $\alpha = 1, 4$ ,  $\beta = 2$ ,  $\tau_0 = 0, 001$ ,  $\rho = 0, 01$ ,  $Q = 20$ ,  $q_0 = 0, 3$  e quantidade de formigas = 50. Os critérios de parada foram, convergência da população ou 500 iterações.

Os 3 algoritmos foram executados sobre o grafo para 10 instâncias diferentes do problema, ou seja, para 10 pares de origem e destino diferentes. Para as 10 instâncias foram analisadas as qualidades das soluções encontradas por cada algoritmo, bem como o tempo gasto para chegar a essas soluções. Para o ACS cada instância foi executada 10 vezes, pois ele é estocástico e diferentemente dos demais pode encontrar um resultado diferente em cada execução.

### C. Configurações da máquina

Todos os algoritmos foram implementados em linguagem *Python* utilizando o IDE (ambiente de desenvolvimento integrado) *Spyder 3.7*, sistema operacional Ubuntu 18.04 lts e processados em um computador com as seguintes configurações: Intel(R) Core(TM) i3-4005U CPU 1.7GHz, 4GB de RAM, 64-bits.

## 5 | RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados da resolução de várias instâncias do problema de busca através da aplicação dos algoritmos, Busca de Custo Uniforme, A\* e ACS, além da discussão dos resultados apresentados por cada um deles.

### A. Resultados

Foram realizados experimentos para simular um AGV com um ponto de origem e um ponto de destino para cada instância do problema, sendo que estes pontos foram gerados aleatoriamente. Os resultados obtidos consistem nas rotas encontradas em cada simulação pelos 3 métodos, bem como a qualidade desse caminho e o tempo gasto por cada um deles para resolver o problema. O algoritmo ACS foi executado 10 vezes para cada instância, enquanto os demais apenas uma vez, por estes se tratarem de métodos determinísticos.

As Tabelas I e II contêm os resultados obtidos ao executar os algoritmos Busca de Custo Uniforme e A\*. Nelas são mostrados o tempo de execução e o tamanho da solução encontrada em cada instância. A solução encontrada é ótima, pois o Busca de Custo Uniforme e o A\* com heurística adequada são ótimos, i.e., sempre encontram a solução ótima para o problema.

Tabela I  
 RESULTADOS OBTIDOS PELO BUSCA DE CUSTO UNIFORME

Instâncias	Tamanho(m)	Tempo(s)	$N^{\circ}$ de nós expandidos
1	1070	$8,995 \times 10^{-2}$	1348
2	400	$3,137 \times 10^{-2}$	480
3	740	$4,894 \times 10^{-2}$	775
4	750	$9,692 \times 10^{-2}$	1204
5	650	$7,281 \times 10^{-2}$	838
6	1335	$9,818 \times 10^{-2}$	1375
7	810	$9,931 \times 10^{-2}$	1272
8	1375	$1,033 \times 10^{-1}$	1378
9	570	$1,546 \times 10^{-1}$	1167
10	620	$4,314 \times 10^{-2}$	638

Tabela I: Resultados obtidos pelo busca de custo uniforme

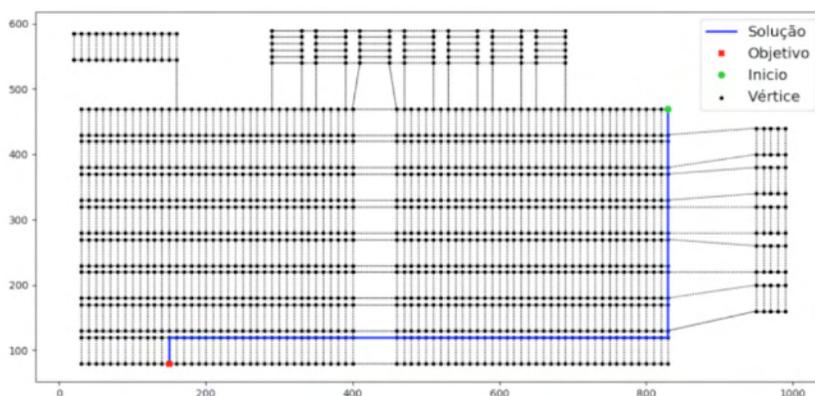


Figura 3. Resultado obtido pelo Busca de Custo Uniforme para a instância 1.

Instâncias	Tamanho(m)	Tempo(s)	$N^{\circ}$ de nós expandidos
1	1070	$8,532 \times 10^{-2}$	1061
2	400	$3,214 \times 10^{-3}$	33
3	740	$6,039 \times 10^{-3}$	58
4	750	$2,695 \times 10^{-2}$	550
5	650	$9,682 \times 10^{-3}$	164
6	1335	$1,163 \times 10^{-1}$	1251
7	810	$2,956 \times 10^{-2}$	395
8	1375	$6,451 \times 10^{-2}$	757
9	570	$2,589 \times 10^{-2}$	328
10	620	$5,114 \times 10^{-3}$	105

Tabela II: Resultados obtidos pelo a\*

As Tabelas III e IV mostram os resultados obtidos pelo algoritmo ACS. Na Tabela III é mostrado para cada instância, o tempo médio gasto em 10 execuções, o desvio padrão, o tempo gasto na execução mais demorada e o tempo gasto na execução mais rápida. A Tabela IV apresenta para cada instância, o tamanho médio das soluções em 10 execuções, o desvio padrão, o tamanho da pior e da melhor solução encontradas nessas execuções.

As Tabelas V e VI mostram para cada algoritmo os valores médios de tempo de execução e tamanho da solução, respectivamente. Os resultados apresentados nessas tabelas são relativos a 10 instâncias do problema, onde podemos notar que para os algoritmos determinísticos os valores médios de tamanho da solução são idênticos.

As figuras geradas na instância 1 podem ser observadas nas Figuras 3, 4 e 5. As demais figuras geradas foram omitidas, considerando-se as limitações de espaço no trabalho. Os resultados mostrados nessas imagens demonstram bem os resultados obtidos nas demais instâncias, onde podemos verificar que o Busca de Custo Uniforme e o A\* encontram a solução ótima e o ACS converge para uma solução sub-ótima.

### B. Discussão

Ao comparar os resultados apresentados nas Tabelas I, II, III e IV, podemos perceber que o Busca de Custo Uniforme e o A\* foram os métodos que se saíram melhor na resolução do problema, ambos encontraram o caminho mínimo e apresentaram baixo tempo de execução. O A\* foi mais eficiente no quesito tempo em 9 das 10 instâncias, e na única onde não foi a diferença foi pouco significativa, resultado esperado, pois devido a busca no A\* ser direcionada por uma informação

Instâncias	Tempo médio(s)	Desvio padrão(s)	Menor tempo(s)	Maior tempo(s)
1	$6,012 \times 10^3$	$1,184 \times 10^3$	$4,653 \times 10^3$	$7,954 \times 10^3$
2	$3,847 \times 10^2$	$2,138 \times 10^2$	$1,484 \times 10^2$	$7,843 \times 10^2$
3	$4,937 \times 10^3$	$1,533 \times 10^3$	$3,482 \times 10^3$	$8,975 \times 10^3$
4	$2,148 \times 10^3$	$6,557 \times 10^2$	$1,354 \times 10^3$	$3,568 \times 10^3$
5	$1,808 \times 10^3$	$7,693 \times 10^2$	$1,161 \times 10^3$	$3,823 \times 10^3$
6	$5,319 \times 10^3$	$8,593 \times 10^2$	$4,258 \times 10^3$	$6,514 \times 10^3$
7	$3,134 \times 10^3$	$6,709 \times 10^2$	$2,594 \times 10^3$	$4,814 \times 10^3$
8	$4,201 \times 10^3$	$1,063 \times 10^3$	$3,156 \times 10^3$	$6,631 \times 10^3$
9	$2,740 \times 10^3$	$3,388 \times 10^2$	$2,210 \times 10^3$	$3,325 \times 10^3$
10	$1,218 \times 10^3$	$3,793 \times 10^2$	$7,753 \times 10^2$	$2,091 \times 10^3$

Tabela III: Resultados de tempo de execução para o ACS.

Instâncias	Tamanho médio(m)	Desvio padrão(m)	Menor tamanho(m)	Maior tamanho(m)	Tamanho da solução ótima(m)
1	1341	46,14	1270	1410	1070
2	400	0,00	400	400	400
3	822	74,00	740	980	740
4	804	57,31	750	930	750
5	734	51,22	650	810	650
6	1773	103,71	1635	1955	1335
7	1000	76,55	890	1190	810
8	1615	77,97	1475	1735	1375
9	738	66,45	570	810	570
10	696	43,63	640	760	620

Tabela IV: Resultados dos tamanhos das rotas obtidas pelo accs.

Métodos	Tempo médio(s)
Busca de Custo Uniforme	$8,385 \times 10^{-2}$
A*	$3,725 \times 10^{-2}$
ACS	$3,190 \times 10^3$

Tabela V: Tempo médio das 10 instâncias.

Métodos	Tamanho médio(m)
Determinísticos (Busca de Custo Uniforme e A*)	832
Estocástico (ACS)	992,3

Tabela VI: Tamanho médio das soluções dos algoritmos determinísticos e estocásticos para as 10 instâncias.

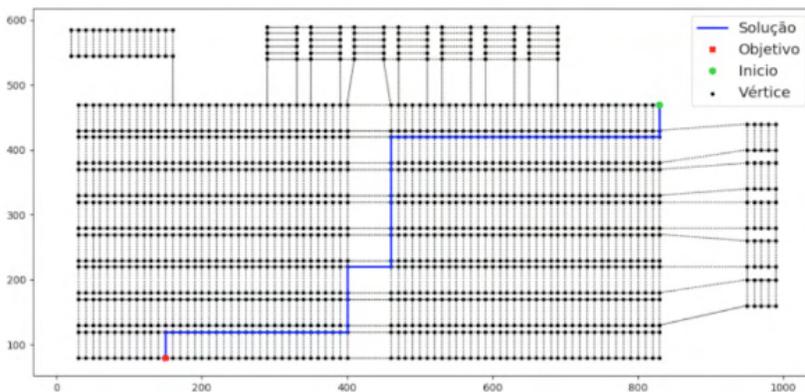


Figura 4. Resultado obtido pelo A\* para a instância 1.

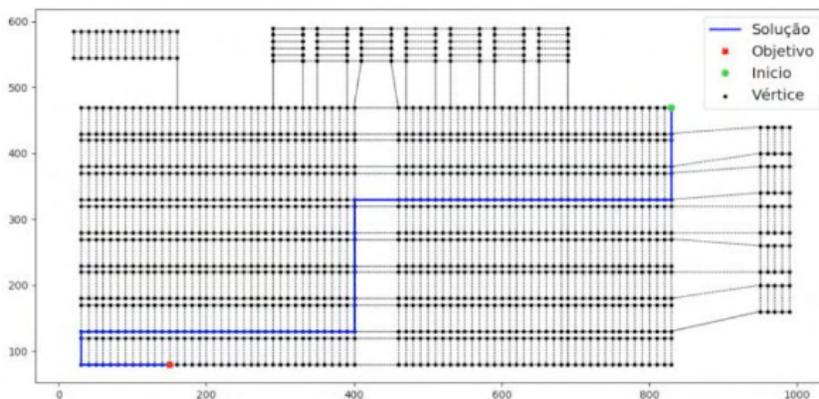


Figura 5. Resultado obtido pelo ACS na instância 1.

Heurística sua árvore de busca acaba sendo menor que a do Busca de Custo Uniforme. Podemos ver que a árvore de busca do A\* é menor ao observar nas Tabelas I e II o No de nós expandidos.

O que fez com que o Busca de Custo Uniforme fosse superior em uma das instâncias foram as características da mesma. Ao analisar a instância 6 observamos que o Busca de Custo Uniforme se saiu melhor, isso ocorre pois o vértice de origem não é rodeado por outros vértices, e como o Busca de Custo Uniforme expande os estados a sua volta num espécime de círculo, ele acabou expandindo menos estados do que se a origem fosse um vértice do grafo com mais vértices em volta. O A\* não se beneficia dessa configuração, pois a forma como explora o ambiente é diferente.

Na Tabela V podemos ver que na média o A\* foi mais eficiente que o Busca de Custo Uniforme, considerando as 10 instâncias o tempo de execução do Busca de Custo Uniforme foi mais que 2 vezes maior quando comparado ao A\*. Sendo assim foi observado que o A\* como algoritmo é mais eficiente que o Busca de Custo Uniforme.

Ao comparar os resultados dos algoritmos determinísticos das Tabelas I e II, com os resultados do algoritmo estocástico mostrado nas Tabelas III e IV, podemos perceber que o mesmo não resolveu bem o problema visto que o tempo de execução foi muito superior e os caminhos encontrados muitas vezes ficaram presos em mínimos locais. Essa má eficiência em relação ao tempo pode ser atribuída a uma implementação que não foi voltada para tal. O ACS, devido a suas características facilita a programação paralela, logo ao implementá-lo dessa forma o tempo gasto seria reduzido em muitas vezes. Além disso mais formigas e iterações se fariam viáveis podendo melhorar e muito a qualidade das soluções.

Outro aspecto importante a se ressaltar é que devido ao tamanho do problema os métodos determinísticos se mostraram suficientemente bons, fazendo com que a utilização de um estocástico não seja tão interessante, ou seja, para esse problema os

determinísticos se mostram muito superiores, porém para problemas maiores e com uma implementação mais voltada pra eficiência o ACS tem grande potencial, já que algoritmos determinísticos não se saem bem em problemas de maiores dimensões devido a terem tempo de execução exponencial.

## 6 | CONCLUSÃO

Utilizando um *software* de modelagem e simulação, o *AnyLogic*, foi possível a criação de um modelo gráfico de um armazém inteligente. Foram implementados 2 algoritmos que geram um grafo a partir do modelo gráfico, tornando possível a aplicação de algoritmos de roteirização sobre um ambiente real.

Com um estudo detalhado do problema de roteamento de veículos, foi possível a implementação de 3 algoritmos para a resolução do problema, o Busca de Custo Uniforme, o A\* e o ACS. Todos foram capazes de resolver o problema proposto, mostrando a viabilidade da aplicação desses algoritmos para problemas reais. Dois deles apresentaram melhores resultados tanto na qualidade da solução encontrada quanto no tempo de execução, o A\* e o Busca de Custo Uniforme. Ambos apresentaram tempo de execução viáveis para a resolução do problema, porém o A\* foi melhor em quase todas as instâncias, se mostrando superior aos demais métodos para o problema dado.

O ACS apresentou tempo de execução relativamente alto e ficou preso em mínimos locais em algumas situações. Porém sua implementação pode ser melhorada, trazendo mais eficiência ao método. Um aspecto importante a se ressaltar é o tamanho do problema resolvido pelos algoritmos, por ser um problema pequeno os métodos determinísticos tiveram desempenho muito bom, porém o tempo cresce exponencialmente para eles, fazendo com que para problemas muito grandes a sua utilização seja inviável computacionalmente, por isso quanto mais o problema cresce melhor o método estocástico é em relação aos determinísticos.

Como propostas de trabalhos futuros, pode-se citar uma implementação mais eficiente do ACS, fazendo um código paralelizado, assim o tempo gasto para encontrar soluções seria muito otimizado, tornando possível o acréscimo de formigas e iterações, podendo assim melhorar também a qualidade da solução. Além disso poderia-se incrementar o problema tornando-o mais desafiador, fazendo com que a utilização do ACS seja mais interessante, isso pode ser feito por exemplo, aumentando o número de objetivos.

Outra proposta de trabalho seria um sistema com vários AGVs no ambiente, trazendo uma situação mais realista, onde problemas como colisões entre os AGVs tivessem que ser tratadas.

## AGRADECIMENTOS

Os autores agradecem à UFVJM por todo o suporte prestado no desenvolvimento deste trabalho.

## REFERÊNCIAS

- [1] K. C. T. Vivaldini, "Roteamento automático de empilhadeiras robóticas em armazém inteligente," Ph.D. dissertation, Universidade de São Paulo, 2010.
- [2] M. V. C. d. S. *Ordônio et al.*, "Brasil e china no mundo 4.0: uma visão a partir da política industrial," 2020.
- [3] M. Oda, Z. A. I. Miranda, A. Itani, E. Licco, and L. A. Kulay, "Logística sustentável: contribuição a processos de gestão," *INTERFACEHS-Revista de Saúde, Meio Ambiente e Sustentabilidade*, vol. 4, no. 1, 2010.
- [4] M. Stürmer, "Sistema de controle e navegação para robôs móveis autônomos em ambientes de armazenagem," *Portuguese, Universidade do Vale do Rio dos Sinos, São Leopoldo*, 2004.
- [5] I. Beker, V. Jevtić, and D. Dobrilović, "Shortest-path algorithms as a tools for inner transportation optimization," *International Journal of Industrial Engineering and Management (IJEM)*, vol. 3, no. 1, pp. 39–45, 2012.
- [6] L. B. Branisso et al., "Sistema multiagente para controle de veículos autônomos," 2014.
- [7] N. V. Kumar and C. S. Kumar, "Development of collision free path planning algorithm for warehouse mobile robot," *Procedia computer science*, vol. 133, pp. 456–463, 2018.
- [8] N. C. Truong, T. G. Dang, and D. A. Nguyen, "Optimizing automated storage and retrieval algorithm in cold warehouse by combining dynamic routing and continuous cluster method," in *International Conference on Advanced Engineering Theory and Applications*. Springer, 2018, pp. 283–293.
- [9] E. Cardarelli, V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Cooperative cloud robotics architecture for the coordination of multiagv systems in industrial warehouses," *Mechatronics*, vol. 45, pp. 1–13, 2017.
- [10] Z. Zhang, Q. Guo, J. Chen, and P. Yuan, "Collision-free route planning for multiple agvs in an automated warehouse based on collision classification," *IEEE Access*, vol. 6, pp. 26 022–26 035, 2018.
- [11] C. Secchi, R. Olmi, F. Rocchi, and C. Fantuzzi, "A dynamic routing strategy for the traffic control of agvs in automatic warehouses," in 2015 IEEE *International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 3292–3297.
- [12] X. Li, C. Zhang, W. Yang, and M. Qi, "Multi-agvs conflict-free routing and dynamic dispatching strategies for automated warehouses," in *International Conference on Mobile and Wireless Technology*. Springer, 2018, pp. 277–286.
- [13] Y. Noh and Y. Yoon, "Tactical traffic control for multiple agv systems based on three dimensional space," in *AIP Conference Proceedings*, vol. 1790, no. 1. AIP Publishing, 2016, p. 150027.
- [14] R. L. Santos, "Uma aplicação de algoritmos de colônias de formigas em problemas de roteirização de veículos com janelas de tempo," 2006.
- [15] S. J. Russell and Norvig, *Inteligência Artificial*, 3rd ed. Rio de Janeiro: Elsevier, 2013, resolução de problemas por meio de busca.
- [16] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *biosystems*, vol. 43, no. 2, pp. 73–81, 1997.
- [17] B. Andrei, "Multimethod simulation modeling for business applications," 2017. [Online]. Available: <https://www.anylogic.com/>

## UTILIZANDO GAN E REDES NEURAS ARTIFICIAIS MLP PARA SUPORTE AO DIAGNÓSTICO PRECOCE DA DOENÇA DE ALZHEIMER: UM ESTUDO ACERCA DO POTENCIAL DA EXPANSÃO ARTIFICIAL DOS DADOS

Data de aceite: 01/03/2022

Data de submissão: 10/01/2022

### Jonathan da Silva Bandeira

Universidade de Pernambuco (UPE) – Escola  
Politécnica de Pernambuco (POLI)  
Recife – PE  
<https://orcid.org/0000-0003-1693-2091>

### Renan Costa Alencar

Universidade de Pernambuco (UPE) – Escola  
Politécnica de Pernambuco (POLI)  
Recife – PE  
<https://orcid.org/0000-0003-0028-558X>

### Mêuser Jorge Silva Valença

Universidade de Pernambuco (UPE) – Escola  
Politécnica de Pernambuco (POLI)  
Recife – PE  
<http://lattes.cnpq.br/9742028115892992>

**RESUMO:** A expectativa de vida da população dos países mais desenvolvidos cresce a cada dia e, conseqüentemente, há um aumento de diversas doenças relacionadas ao envelhecimento. No Brasil, pouco mais de 1,1 milhão de pessoas têm Doença de Alzheimer (DA). Em 2019, de acordo com a Organização Mundial da Saúde (OMS), a Doença de Alzheimer e outras demências eram a terceira principal causa de mortalidade nas Américas e na Europa. Apesar de ser uma doença degenerativa e irreversível, se diagnosticada precocemente, os tratamentos podem ser realizados a fim de retardar a progressão dos sintomas e garantir uma melhor qualidade de vida

ao paciente. A maioria dos artigos que estudam soluções de Inteligência Computacional para apoiar o diagnóstico seguem uma abordagem baseada em evidências de neuroimagem. Além disso, outra abordagem que vem ganhando destaque é a análise biomolecular. Seguindo essa abordagem, Ray *et al.*, Ravetti & Moscato e Dantas & Valença realizaram estudos com classificadores com base na área estatística ou na Inteligência Computacional para subsidiar o diagnóstico precoce da doença. O trabalho foi realizado a partir de um conjunto de dados com valores de 120 proteínas sanguíneas. Com isso, eles conseguiram classificar se o paciente poderia ou não ser diagnosticado com DA. Este trabalho visou utilizar uma abordagem tradicional com uma proposta de modelo de Rede Neural Artificial *Multilayer Perceptron* (MLP) para realizar o diagnóstico precoce de um paciente com ou sem DA e também comparar os resultados obtidos com os resultados dos trabalhos relacionados citados. Além disso, este trabalho teve como objetivo principal avaliar o potencial da utilização de dados sintéticos gerados por meio de uma *Generative Adversarial Network* (GAN) no treinamento e testes do modelo de classificação proposto. O modelo proposto de MLP apresentou como resultados taxas médias de 94% de acurácia, 100% de sensibilidade e 87% de especificidade. Esses resultados mostraram a grande capacidade do modelo, especialmente em reconhecer a classe enferma. Quanto aos dados sintéticos gerados, estes demonstraram potencial na otimização do desempenho do modelo original através da expansão artificial de dados.

**PALAVRAS-CHAVE:** Doença de Alzheimer, *Multilayer Perceptron*, Redes Neurais Artificiais, *Generative Adversarial Networks*, *Data Augmentation*.

## USING GAN AND MLP ARTIFICIAL NEURAL NETWORKS TO SUPPORT EARLY DIAGNOSIS OF ALZHEIMER'S DISEASE: A STUDY ON THE POTENTIAL OF ARTIFICIAL DATA EXPANSION

**ABSTRACT:** The life expectancy of the population in the most developed countries is growing every day and, consequently, there is an increase in various age-related diseases. In Brazil, just over 1.1 million people have Alzheimer's disease (AD). In 2019, according to the World Health Organization, Alzheimer's disease and other dementias were the third leading cause of mortality in the Americas and Europe. Despite being a degenerative and irreversible disease, if diagnosed early, treatments can be performed to slow the progression of symptoms and ensure a better quality of life for the patient. Most papers that study Computational Intelligence solutions to support diagnosis follow an approach based on neuroimaging evidence. In addition to this, another approach that has been gaining prominence is biomolecular analysis. Following this approach, Ray *et al.*, Ravetti & Moscato and Dantas & Valença performed studies with classifiers based on the statistical area or on Computational Intelligence to support the early diagnosis of the disease. The work was carried out from a data set with values of 120 blood proteins. With this, they were able to classify whether the patient could be diagnosed with AD. This work aimed to use a traditional approach with a proposed Multilayer Perceptron Artificial Neural Network model to perform the early diagnosis of a patient with or without AD and compare the results obtained with the results of the related works mentioned. In addition, this work had, as its main objective, the evaluation of potential use of synthetic data generated through a Generative Adversarial Network (GAN) in the training and testing of the proposed classification model. The proposed MLP model presented mean rates of 94% accuracy, 100% sensitivity and 87% specificity. These results showed the remarkable capacity of the model, especially in recognizing the sick class. As for the synthetic data generated, these showed potential in optimizing the performance of the original model through artificial data expansion.

**KEYWORDS:** Alzheimer's disease, Multilayer Perceptron, Artificial Neural Networks, Generative Adversarial Networks, Data Augmentation.

### 1 | INTRODUCTION

In the last decades, life expectancy has increased. Because of this, more challenges for the elderly part of the population arise (LIMA, 2018). One of the main concerns is the development of Chronic Non-Communicable Diseases (NCDs) in older adults. Among these stands out dementias such as Alzheimer's disease (AD). AD is a disease that already affects more than 47 million worldwide (APOLINÁRIO *et al.*, 2011).

Dementia declines an individual's cognitive functions, which directly affect their behavior and quality of life. Dementias can be reversible or irreversible. Irreversible dementias can also be degenerative and progressive, which get worse over time (GALLUCCI; TAMELINI; FORLENZA, 2005) . In this field, we can highlight a range of conditions from a

mild impairment, such as Mild Cognitive Impairment (MCI), to diagnoses of severe dementia conditions such as Alzheimer's Disease, discovered in 1907 by Alois Alzheimer's.

Alzheimer's Disease is a progressive neurodegenerative disorder that causes irreversible damage to an individual's brain, deteriorating their cognitive capacity and memory (the ability to recall old information or learn new information) and compromising their daily routine and behavior. AD is currently the form of degenerative dementia with the highest incidence globally, affecting mainly the elderly over 65 years of age in its late or senile manifestation. Despite this, it can also rarely affect young people at an early age.

According to the Brazilian Medical Association and the National Supplementary Health Agency, the worldwide prevalence of dementia becomes higher as age increases. In their last update, the following statistics were presented: 1.2% between 65 and 69 years old; 3.7% from 70 to 74 years old; 7.9% from 75 to 79 years old; 16.4% between 80 and 84 years old; 24.6% from 85 to 89 years old; 39.9% from 90 to 94 years old; 54.8% from 95 years onwards (APOLINÁRIO *et al.*, 2011).

Considering the Brazilian elderly population of approximately 15 million people and the incidence of dementia in Brazil, it is estimated that the scope of dementia in the country reaches a value of approximately 1.1 million individuals (CONITEC, 2017).

As stated by the American Alzheimer's Association, the situation is even more alarming in other countries, such as the USA. In 2016, estimates were of 5.4 million individuals with AD, 11% of whom were aged 65 and over 32% of those aged 85 and over (ALZHEIMER'S ASSOCIATION, 2016). Projections made this year for the figures in 2050 stated that 51% of the population over 65 years old would be affected by the disease. Also, around 7 million people over the age of 85 will be affected (ALZHEIMER'S ASSOCIATION, 2016).

Along with the World Health Organization, in the last two decades, between 2000 and 2019, Alzheimer's disease and other forms of dementia ranked the top ten diseases responsible for global mortality (WORLD HEALTH ORGANIZATION, 2020). In this ranking, AD and other dementias appeared as the third leading cause of mortality in the Americas and Europe, in the survey concluded in 2019 (WORLD HEALTH ORGANIZATION, 2020).

It is important to emphasize that, like other chronic diseases, although irreversible, AD can be controlled if diagnosed in its initial phase. This early diagnosis is crucial to delay degenerative progress and guarantee a better quality of life for the patient (APOLINÁRIO *et al.*, 2011).

At present, the set of solutions with Computational Intelligence as support for diagnosis is based on three approaches: cognitive tests, neuroimaging evidence, and biomolecular tests.

The first approach is based on exams such as the Mini-Mental State Examination and has as its main positive point the most straightforward data to understand from questionnaires (APOLINÁRIO *et al.*, 2011).

Neuroimaging is considered the current state-of-the-art of the three approaches and is based on evidence and imaging exams, such as CT scans, PET scans, and MRIs. The accuracy of the data that forms the basis used in the training of Machine Learning models is high. This high accuracy is due to the images of cerebral autopsies, in the post-mortem period of each patient, provide higher values of sensitivity and specificity. Papers that use this approach are the most common in the scientific community (SILVA *et al.*, 2019).

Definitively, biomolecular analysis is an approach that has gained notoriety in the last two years due to new evidence on the pathophysiology of the disease according to Dalmagro *et al.* (2020). The great discoveries in this area, such as a new high-precision and low-cost blood test capable of detecting the development of AD up to twenty years earlier (PALMQVIST *et al.*, 2020); and, more recently, advances in the studies of a vaccine have been capable of preventing the development of the disease (DAVTYAN *et al.*, 2019). This approach focuses on studying genetic, molecular, protein characteristics and other features present in blood plasma and are measured by laboratory tests (RAY *et al.*, 2007).

Following the last approach mentioned, Ray *et al.* (2007) developed a study using a dataset of 259 patients containing the concentration levels of 120 proteins contained in the blood plasma samples of these patients. Of these samples, 222 comprise samples of diagnoses of Alzheimer's Disease, Mild Cognitive Impairment, other dementias, and cases without dementia. This work concluded that a signature resulting from the combination of 18 proteins out of 120 allowed the early diagnosis of AD. Using the PAM technique for classification, an accuracy rate of 89% was shown. Also, 90% of correct answers were obtained in positive cases (recall), and 88% of correct answers in negative cases (specificity) in the test set of AD.

Using the same dataset cataloged by Ray *et al.*, Ravetti & Moscato (2008) used classification algorithms available in the Weka software and obtained an average accuracy rate of 93%. 96% of correctness were achieved in positive cases (recall), and 90% of correctness in negative cases (specificity) for the tests set of AD. This paper considered a better combination of 5 proteins.

In Dantas & Valença (2013), another study that used the same database, used a Reservoir Computing (RC) framework, and obtained, on average, 94.34% accuracy for the tests set of AD. It considered the same combination of 5 proteins that were used by Ravetti & Moscato.

To select a set of proteins, Ray *et al.* (2007) and Ravetti & Moscato (2008) used the already cited PAM tool to extract characteristics. The selection made by Ravetti & Moscato resulted in a subset of the selection already performed by Ray *et al.* (2007). On the other hand, Dantas & Valença (2013), used the Random Forests technique with information gain analysis (based on entropy) to find the selection, which coincided with the selection of 5 proteins by Ravetti & Moscato. This work used this same selection of 5 proteins to carry out the experiments.

The present work used an Artificial Neural Network (RNA) Multilayer Perceptron (MLP) model in the same 5-protein signature used by the two most recent works cited. The performance was compared with works in the literature and classifiers in their default settings, coming from the SciKit-Learn library written in Python. After that, a Generative Adversarial Network model was used to generate 10,000 synthetic samples for testing in two scenarios: training on real samples with testing on synthetic samples, and training on synthetic samples with testing on real samples.

## 2 | MACHINE LEARNING TECHNIQUES

### A. SciKit-Learn library algorithms

Python language was used to implement and import necessary machine learning algorithms. The following algorithms were used from the SciKit-Learn library (GRISEL *et al.*, S.I.): Random Forest, Naïve Bayes, K-Nearest Neighbors, and Support Vector Machines. For all of these, the configuration used was the standard provided by the library. The proposed classification model was a Multi-Layer Perceptron Neural Network, based on the sequential model provided by Python's Keras API (CHOLLET, S.I.).

### B. Artificial Neural Networks

Artificial Neural Networks (ANN) are computational models inspired by the biological central nervous system, particularly brain neurons, capable of machine learning and pattern recognition. Another way to define this technology is to state that an ANN is a function approximator. The primary purpose of an ANN is to simulate the connectionist behavior of a chain of neurons so it can learn from the environment and improve its performance (VALENÇA, 2009).

Computationally, the objective of an ANN is to reduce forecasting errors by minimizing a given function, called a cost function. The term cost function is used to determine the forecasting errors of the output neurons over a cycle or time. The calculation performed for a single example is called a loss function (HAYKIN, 2001).

A training procedure is carried out for the loss function. An iterative adjustment process is applied to the weights of connections (synapses) between neurons within an ANN (VALENÇA, 2009). When a neural network can generalize a solution to a specific class of problems, it is said that there was learning (VALENÇA, 2009).

The learning process of an ANN is verified by changing the weights. The initial weights, usually generated at random, are modified iteratively by a training algorithm that follows one of the following paradigms (HAYKIN, 2001):

- Supervised Learning: a training set is presented, consisting of the inputs and desired corresponding outputs.
- Reinforcement Learning: for each input presented, an indication (reinforcement)

about the adequacy of the corresponding outputs produced by the network is produced.

- **Unsupervised Learning:** the network has its weights adjusted without the use of desired input-output pairs nor indications about the adequacy of its corresponding outputs produced by the network.

To design and set an ANN, we must know some characteristics like the neuron features, the network topology, and the training rules. A well-designed and configured ANN brings in addition to learning some other benefits, such as high power of generalization, flexibility, fault tolerance, self-organization, and parallel information processing (HAYKIN, 2001).

In practice, ANN can be applied to solve problems of function approximation, classification, pattern recognition, and time-series prediction (HAYKIN, 2001).

Multilayer Perceptron is part of an ANN's class architectures known as feedforward, whose neurons are grouped in layers. The signals travel through the network in a single direction, from input to output, and neurons in the same layer are not interconnected (HAYKIN, 2001). The backpropagation algorithm is usually used to train an MLP model (VALENÇA, 2009). Fig. 1 shows the architecture of an MLP neural network.

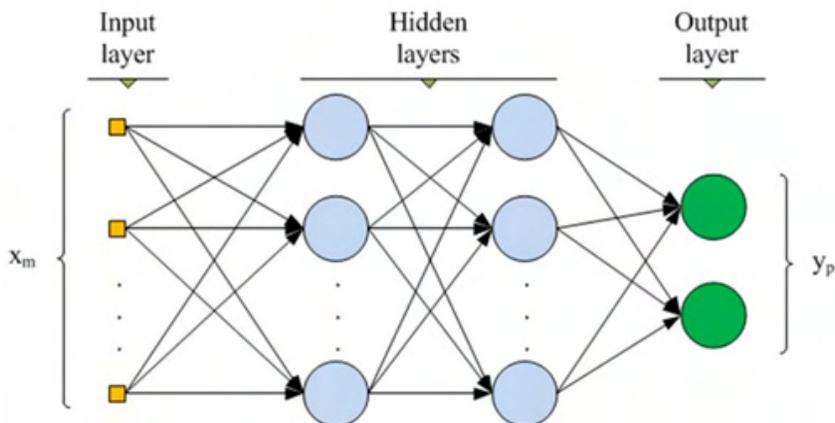


Fig. 1. Multilayer Perceptron Neural Network architecture

### C. Generative Adversarial Networks

The second proposed technique was GAN, a framework that represents a new class of machine learning models. This framework was developed by Goodfellow *et al.* (2014), which consists of two Artificial Neural Networks competing against each other in a sort of game.

In this game, one of the neural networks assumes the role of the generator while the other has the function of discriminator (GOODFELLOW *et al.*, 2014). The generator G is based on a given training set of real data, and with the addition of noise, it learns to produce

new data like the real ones and with the same statistics as the original ones (GOODFELLOW *et al.*, 2014). The new data generated is called synthetic data. The discriminator D, in turn, tries to distinguish real from synthetic instances. Fig. 2 shows the organizational structure of a GAN.

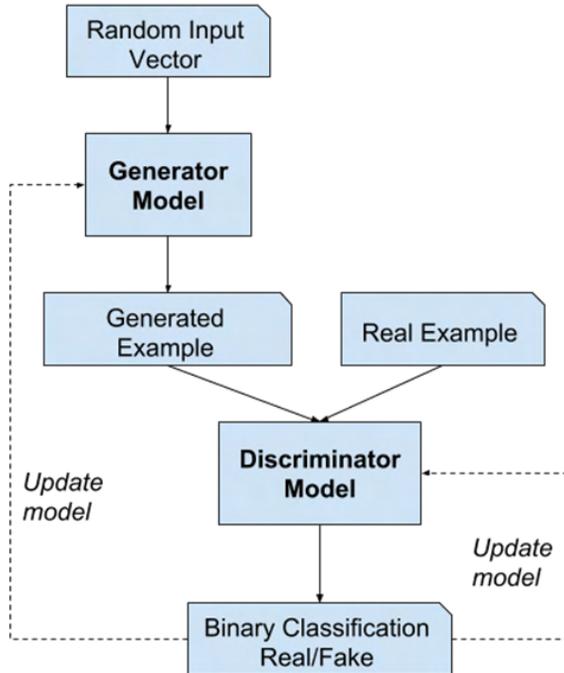


Fig. 2. GAN model architecture.

Competitive learning is possible through the loss rule defined for this process and the way neural networks are trained in the framework. The loss rule involves maximizing and minimizing two logarithmic functions, given by (1) (GOODFELLOW *et al.*, 2014).

$$\text{Min}_G \text{Max}_D \text{VGAN}(D, G) = E_{x \sim P_{data}(x)}[\log(D(x))] + E_{z \sim P_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

In  $\text{Min}_G \text{Max}_D \text{VGAN}(D, G)$ ,  $\text{Min}_G$  is the generator loss minimization given only by the second term of the equation, while  $\text{Max}_D$  is the discriminator loss maximization given by all the equation. The term  $\text{VGAN}(D, G)$  is where the maximum and minimum functions apply, which can be described as the loss vector of the GAN model, where D corresponds to the discriminating model and G is the generating model.

The discriminator D tries to maximize both functions and maintain good

performance to identify the real data as true and identify the synthetic data as false (HAYKIN, 2001). On the other hand, the generator  $G$  tries to minimize the second function, generating data increasingly like the real ones, to reduce the performance of the discriminator in both situations (real and synthetic data) (GOODFELLOW *et al.*, 2014). That way, the discriminator will not be able to specify what is real or fake.

In the first log loss function, given by  $E_{x \sim P_{data}(x)}[\log(D(x))]$ , we can highlight the discriminator applied to real data and distributions ( $x \sim P_{data}(x)$ ) in the form  $D(x)$ , where  $E$  is the loss value calculated in  $[\log(D(x))]$ .

In the calculation of the loss by the second equation, given by  $E_{z \sim P_z(z)}[\log(1 - D(G(z)))]$ , the discriminator is applied to the synthetic data and distributions generated in the form  $D(G(z))$ , where  $G(z)$  is the data generated by the generator  $G$  and ( $z \sim P_z(z)$ ) is the real data and distributions imbued with noise.

Both functions of the equation are complementary, so the second loss function ( $E$ ) is calculated by the function  $[\log(1 - D(G(z)))]$ . Thus, while the first function is  $D(a)$ , the second is  $(1 - D(a))$ . In the first function, this argument "a" is the real data  $x$ , and in the second, the generated data  $G(z)$ .

### 3 | MATERIALS AND METHODS

#### A. Dataset

The database used in the development of this work was the same developed by Ray *et al.* (2007) in their publication. It contains the values of 120 proteins found by analyzing blood samples from 222 patients diagnosed with AD, MCI, other dementias, or no dementia. The base's goal is to classify whether a patient can be diagnosed positively or negatively concerning Alzheimer's Disease (AD or NAD, respectively).

Initially, Ray *et al.* (2007) subdivided the dataset into three sets. The first one contained 83 blood samples with values of 120 proteins. The second set contained information from 92 patients diagnosed with or without AD. On the other hand, the third contained 47 samples from patients diagnosed with MCI that progressed over time to AD or other dementias or maintained the final diagnosis of MCI after a few years.

In this analysis, only the training sets and AD tests were considered. Diagnoses related to other dementias were also excluded, having been worked on instead of 92 instances of tests, only 81 instances. It contains only the cases of patients with AD or without dementias. Table I shows which of the 120 proteins were contained in the signatures defined by Ravetti & Moscato (2008) and Dantas & Valença (2013), also used in this work.

Number of proteins in signature	Proteins
5	IL-1 $\alpha$ , IL-3, EGF, TNF- $\alpha$ and G-CSF

Table I. Description of the protein signature used.

Three datasets were used in this work:

- I. A dataset with 83 samples used for training the classifiers in the first and second experiments;
- II. A dataset with 81 samples used for tests, i.e., performance evaluation of the classifiers in the first and second experiments;
- III. A synthetic dataset generated by GAN with 10,000 samples used for both training and testing in the second experiment.

The input data have been normalized, i.e., transformed to be contained in a single numerical range in these datasets. This normalization is essential, considering that the different variations of each feature could skew the training and cause a loss in the ability to generalize the models.

The input data were normalized using the MinMaxScaler function of the SciKit-Learn library. The range used as the default, between 0 and 1. The cited function uses the formula given by (2):

$$X_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

Equation  $X_{normalized}$  corresponds to the new value of a given instance of a characteristic after normalization. The value  $x_{max}$  is the highest value among all instances of a given characteristic. The value  $x_{min}$  is the smallest value among all instances of a given characteristic. Finally, the value of  $x$  corresponds to the original value of a given instance of a characteristic before it was normalized.

The labels were transformed into numerical information, assuming the values 0 (zero) for positive class and 1 (one) for negative class.

In the original dataset (RAY *et al.*, 2007), a significant number of outliers were identified within all sets. Because of the division carried out by Ray *et al.* (2007), outliers were equally distributed in both sets of training and testing, and this gives the idea that the instances are equally representative. However, using a smaller fraction of data for testing or when using a more significant number of instances in training, it is possible to notice that these outliers are primarily found in instances of positive cases. These outliers can be seen in the synthetic data generated. In practice, this means that instances of negative cases are more representative than the other ones.

## B. EXPERIMENTS

We chose to use an MLP as it is the simplest ANN model capable of solving this classification problem. As the problem is not linearly separable, elementary models are not viable.

More robust models using deeper architectures (Deep Learning) were not tested due to their high computational cost. It is necessary to verify whether the gain in this operation is worth, that is, if the cost x benefit of these models would be better than those of using an MLP.

Regarding the use of recurrent ANN models, this study has already been carried out in Dantas & Valença (2013). It has already been proved that the cost x benefit of using a recurrent model is lower than using an MLP.

Thus, in this study, we chose to use an MLP model with a good proved cost-benefit to solve this problem and to seek performance improvements through tests with artificial data expansion using GAN models for this purpose.

As already described, the classification models from the SciKit-Learn library had their default settings maintained. The MLP model, based on a sequential Keras API model, presented configurations as shown in Table II.

Parameters	Value of parameters for MLP model
Number of neurons in the input layer	5
Number of hidden layers	2
Number of neurons in the first hidden layer	120
Number of neurons in the second hidden layer	60
Number of neurons in the output layer	1
Activation function on hidden layers	ReLU
Activation function in the output layer	Sigmoid
Optimizer	Adam
Optimizer parameters	Learning rate = 0.0002; beta 1 = 0.5; beta 2 = 0.999; epsilon = 0.0000001.
Cost function	Binary cross entropy
Epochs	100
Batch size	10

Table II. MLP Model Configuration.

The GAN model was also implemented using the Keras library and had its configuration performed as shown in Tables III and IV. Table III refers to the Generator model, and Table IV is referring to the Discriminator model. The GAN model was trained

using all 164 considered AD (83 training instances and 81 test instances).

Parameters	Parameter values for layers in common for both branches	Parameter values for the categorical generation branch	Parameter values for the numerical generation branch
Number of hidden layers	2	2	2
Number of neurons in the first hidden layer	8	32	64
Number of neurons in the second hidden layer	16	64	120
Activation function on hidden layers	LeakyReLU	LeakyReLU	LeakyReLU
Number of neurons in the output layer	-	2	120
Activation function in the output layer	-	Softmax	Softmax
Latent dimension	80	-	-
Other layers	BatchNormalization (momentum = 0.8)	BatchNormalization (momentum = 0.8)	BatchNormalization (momentum = 0.8)

Table III. GAN Generator Configuration.

Parameters	Value of parameters for Discriminator model
Number of neurons in the input layer	122 (alternating between actual values and generated values)
Number of hidden layers	5
Number of neurons in the first hidden layer	128
Number of neurons in the second hidden layer	64
Number of neurons in the third hidden layer	32
Number of neurons in the fourth hidden layer	16
Number of neurons in the fifth hidden layer	8
Number of neurons in the output layer	1
Activation function on hidden layers	LeakyReLU
Activation function in the output layer	Sigmoid
Optimizer	Adam
Optimizer parameters	Learning rate = 0.0002; beta 1 = 0.5; beta 2 = 0.999; epsilon = 0.0000001.
Cost function	Binary cross entropy

Table IV. GAN Discriminator Configuration.

The first experiment involved training and testing all classifiers with only the original database sets with the selection of 5 proteins. The second experiment, on the other hand,

also used the generated synthetic database. In the end, the performances of the other classifiers were compared with the model proposed in both simulations. Finally, the result of this work is compared to the other related works cited.

For each algorithm, a seed was defined to guarantee the reproducibility of the results. For some algorithms from the SciKit-Learn library (Random Forest and SVM), the `random_state` variables of these algorithms were used. This variable was defined as 0 (zero). For the Keras MLP model, `NumPy.random.seed` and `TensorFlow.random.set` the seed were imported. Both were assigned the value 7 (seven).

Only the performances of the test set were considered to compare models. The comparison considered the accuracy metrics to ascertain the overall performance of each model: first the sensitivity, to measure the performance of each model in terms of predicting the positive class; and then, the specificity, aiming to measure the performance of each model regarding the negative class prediction.

The percentages shown in the comparative results tables of the models' performances are the average values resulting from thirty runs in different resampling of the test set.

When comparing this work with related works, the comparison methodology is similar, and the percentages also correspond to the average of the results in the test set.

## 4 | RESULTS

After the execution of the first experiments, the results presented in Table V were obtained.

<b>Techniques</b>	<b>Accuracy</b>	<b>Recall</b>	<b>Specificity</b>
<b>K-NN</b>	90%	100%	79%
<b>Naïve Bayes</b>	81%	98%	64%
<b>SVM</b>	88%	100%	74%
<b>Random Forest</b>	81%	98%	64%
<b>MLP (proposed model)</b>	94%	100%	87%

Table V. Comparison between results of the proposed model and the other classifiers.

The MLP Neural network presented the most balanced results, considering all metrics, in the first experiment. Finally, Table VI shows that the proposed model results were similar to the results of all related works cited, emphasizing the sensitivity rate of the model.

<b>Works</b>	<b>Accuracy</b>	<b>Recall</b>	<b>Specificity</b>
<b>Ray <i>et al.</i> (2007)</b>	89%	90%	88%
<b>Ravetti &amp; Moscato (2008)</b>	93%	96%	90%
<b>Dantas &amp; Valença (2013)</b>	94.34%	-	-
<b>This work</b>	94%	100%	87%

Table VI. Comparison between the performances of the model of the present work with the related works.

For the second experiment, Tables VII and VIII present the results obtained considering the training performed on synthetic data with tests on synthetic data and tests on real data. Table IX shows the results from training with the real data set and the test on a synthetic dataset.

<b>Techniques</b>	<b>Accuracy</b>	<b>Recall</b>	<b>Specificity</b>
<b>K-NN</b>	87%	92%	80%
<b>Naïve Bayes</b>	80%	86%	72%
<b>SVM</b>	88%	93%	81%
<b>Random Forest</b>	88%	91%	82%
<b>MLP (proposed model)</b>	87%	91%	82%

Table VII. Performance results of trained and tested classifiers using synthetic dataset.

<b>Techniques</b>	<b>Accuracy</b>	<b>Recall</b>	<b>Specificity</b>
<b>K-NN</b>	86%	86%	87%
<b>Naïve Bayes</b>	83%	69%	97%
<b>SVM</b>	83%	71%	95%
<b>Random Forest</b>	83%	81%	85%
<b>MLP (proposed model)</b>	84%	74%	95%

Table VIII. Performance results of classifiers trained on synthetic dataset and tested on real dataset.

Techniques	Accuracy	Recall	Specificity
<b>K-NN</b>	75%	95%	48%
<b>Naïve Bayes</b>	75%	94%	48%
<b>SVM</b>	76%	95%	49%
<b>Random Forest</b>	72%	94%	40%
<b>MLP (proposed model)</b>	78%	90%	62%

TABLE IX. Performance results of classifiers trained on real dataset and tested on synthetic dataset.

In the first case, it is possible to observe that the synthetic data has a similar distribution to the real data since the training and tests in real data present a behavior similar to the training and tests in synthetic data. Here, two subsets of 5,000 samples each out of a total of 10,000 samples were used.

The second case comprises training on real data, and tests on synthetic data impact the recognition of negative cases. Thanks to how the data was partitioned in the real training set, the outliers present in the real set make the model classify instances of healthy patients as positive AD cases.

In the third and last case of this experiment, the training was carried out with a subset of 5000 synthetic data and then test the real data set for tests. In this case, the opposite of the second case occurred. Here, the hit rate for negative cases was higher than the hit rate for positive cases. This variation was because samples from healthy patients are more representative than the other ones. Since the model was trained in a more significant number of samples, with greater access to samples in this class, this made it recognize more instances of this class to the detriment of the positive class instances.

## 5 | CONCLUSIONS

Alzheimer's disease is currently one of the three most significant causes of mortality globally due to its complications. This pathology is becoming more and more present in society as the population's longevity increases. The disease is a progressive neurodegenerative disorder that is irreversible. Although there is still no cure, existing treatments can provide higher quality and life expectancy for patients diagnosed early, whether pharmacological or behavioral.

In this way, this work aimed to verify the impacts and future potentials arising from synthetic data in improving the performance of the more traditional classifiers for this problem. We also sought to investigate the potential of the proposed model concerning other classifiers and related works.

The proposed model showed results like those presented in the related works, emphasizing its ability to predict positive cases.

Furthermore, synthetic data has shown excellent potential in optimizing the model's performance with the combination of real data and synthetic data. This potential would enable greater capacity to generalize the model and improve the training of an ANN model, which generally requires a significant amount of data.

In the more practical scope of applications, we can highlight the potential to prove that GAN models can be useful even in the expansion of highly sensitive data (such as health data), i.e., it has wide applicability during data expansion for the development of critical systems. Moreover, it is crucial to mention the possibility of developing a specialist system to aid in the early medical diagnosis of neurodegenerative diseases such as Alzheimer's disease. This specialist system would be based on models presented in this work and the approach designed by Ray *et al.* (2007).

To guarantee the reliability of the similarity of the generated data, it would be interesting, as a proposal for future work, to generate the data with a Conditional Generative Adversarial Network (CGAN) model (MIRZA & OSINDERO, 2014), as these are based on a given condition to perform the sample generation. The condition used could be the sample labels, assuring that the data generated would be more faithful to the proposed labels.

Another proposal for relevant future work would be to treat the outliers of the dataset, which could certainly add to improvements in the performance of the models.

## REFERENCES

ALZHEIMER'S ASSOCIATION. **Alzheimer's Disease Facts and Figures**. *In: Alzheimer's Dement.* 2016;12(4):459-509.

APOLINÁRIO, D.; ARAÚJO, L.M.Q.; CHAVES, M.L.F.; LOPES, L.C.; OKAMOTO, I.H.; RAMOS, A.M.; STEIN, A.T.; ANDRADA, N.C. **Doença de Alzheimer: Diagnóstico**. *In: Diretrizes Clínicas na Saúde Suplementar: Associação Médica Brasileira e Agência Nacional de Saúde Suplementar.* Janeiro, 2011.

CHOLLET, F. **The Keras API**. Available in: <https://keras.io/api/>. Last access: 2021-05-27.

COMISSÃO NACIONAL DE INCORPORAÇÃO DE TECNOLOGIAS NO SUS (CONITEC). **Protocolo clínico e diretrizes terapêuticas para Doença de Alzheimer**. *In: Relatório de Recomendação nº 285.* Julho, 2017. Available in: [http://conitec.gov.br/images/Relatorios/2017/Recomendacao/Relatorio\\_PCDDoen%C3%A7a\\_de\\_Alzheimer\\_267\\_17\\_final\\_SEC1207.pdf](http://conitec.gov.br/images/Relatorios/2017/Recomendacao/Relatorio_PCDDoen%C3%A7a_de_Alzheimer_267_17_final_SEC1207.pdf). Last access: 2020-12-12.

DALMAGRO, A.P.; CAZARIN, C.A.; ZENAIDE, F.S.; DALMAGRO, C.A. **An update about the biochemical and molecular basis of Alzheimer's Disease**. *In: Brazilian Applied Science Review, Curitiba, V. 4, N. 1, P.118-130 Jan/Fev, 2020.*

DANTAS, L. and VALENÇA, M. **Utilizando Reservoir Computing para realizar diagnóstico precoce de Mal de Alzheimer**. *In: BRICS SBIC, 2013.*

DAVTYAN, H.; HOVAKIMYAN, A.; KIANI SHABESTARI, S.; *et al.* **Testing a MultiTEP-based combination vaccine to reduce A $\beta$  and Tau Pathology in Tau22/5xfad**. *In: Bigenic Mice. Alz Res Therapy, 11, 107 (2019).*

GALLUCCI, J.; TAMELINI, M.G.; FORLENZA, O.V. **Diagnóstico diferencial das demências**. In: Revista Psiquiátrica Clínica Vol.32, Número 3, São Paulo, Maio/Junho, 2005.

GOODFELLOW, I.J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. **Generative Adversarial Nets**. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, Volume 2, PP. 2672–2680, 2014.

GRISEL, O.; LEMAITRE, G.; DU BOISBERRANGER, J. and MARMO, C. **The Scikit-learn Consortium**, at Inria Foundation. Available in: <https://scikit-learn.fondation-inria.fr/home/>. Last access: 2021-05-27.

HAYKIN, S. **Redes Neurais: Princípios e Práticas**. 2ª Ed. São Paulo: Bookman, 2001.

LIMA, V.M.H. **Fatores Sociodemográficos do envelhecimento e as universidades da terceira idade**. In: VI Congresso Nacional em Educação (CONEDU), 2018.

MIRZA, M.; OSINDERO, S. **Conditional Generative Adversarial Nets**. In: NIPS Deep Learning Workshop, 2014.

PALMQVIST, S.; JANELIDZE, S.; QUIROZ, Y.T.; *et al.* **Discriminative accuracy of Plasma Phospho-Tau217 for Alzheimer disease VS other Neurodegenerative disorders**. In: JAMA. 2020. 324(8):772–781.

RAVETTI, M.G. and MOSCATO, P. **Identification of a 5-protein biomarker molecular signature for predicting Alzheimer’s Disease**. In: PLOS ONE 3: E3111, 2008.

RAY, S.; BRITSCHGI, M.; HERBERT, C.; TAKEDA-UCHIMURA, Y.; *et al.* **Classification and Prediction of Clinical Alzheimer’s Diagnosis based on plasma signaling proteins**. In: NatMed 13: 1359–1362, 2007.

SILVA, I.R.R.; SILVA, G.S.L.; SOUZA, R.G.; SANTOS, W.P.; FAGUNDES, R.A.A. **Model based on Deep Feature Extraction for Diagnosis of Alzheimer’s Disease**. In: Neural Networks International Joint Conference on Neural Networks (IJCNN). Budapest, Hungary. IEEE, 14-19 July, 2019.

VALENÇA, M. **Fundamentos das Redes Neurais: exemplos em Java**. 2ª Ed. Recife: Livro Rápido, 2009.

WORLD HEALTH ORGANIZATION (WHO). **WHO reveals leading causes of death and disability worldwide: 2000-2019**. In: World Health DA Organization, Switzerland, 2020-12-09. Available in: <https://www.who.int/news/item/09-12-2020-who-reveals-leading-causes-of-death-and-disability-worldwide-2000-2019>. Last access: 2020-12-20.

## UTILIZAÇÃO DE UM PERCEPTRON MÚLTIPLAS CAMADAS NA APROXIMAÇÃO DE FUNÇÕES CONTÍNUAS

*Data de aceite: 01/03/2022*

**Dhiego Loiola de Araújo**

Instituto Federal de Brasília - Campus  
Taguatinga

Brasília - Distrito Federal

<http://lattes.cnpq.br/2464571396548516>

**RESUMO:** O Teorema da Aproximação Universal de Kolmogorov garante que é possível aproximar uma função contínua multidimensional via funções reais de uma única variável mas não exemplifica como construir esta aproximação. Neste trabalho, utilizamos o Teorema de Cybenko para implementar uma rede neural artificial multicamadas que realize a aproximação de funções contínuas definidas num intervalo fechado. A rede construída possui três camadas distintas: a camada de entrada, uma oculta constituída de neurônios com ativação sigmoide e uma camada de saída responsável pela combinação linear das saídas da camada anterior. Como a rede implementada é uma versão do Perceptron Múltiplas Camadas, o erro cometido na aproximação é computado através do critério dos Mínimos Quadrados. Assim, a atualização dos pesos sinápticos é realizada com o algoritmo de retropropagação do erro clássico, sem termos de momento ou resiliência.

**PALAVRAS-CHAVE:** Perceptron, Funções Contínuas, Redes Neurais Artificiais.

### A MULTILAYER PERCEPTRON TO APPROXIMATE CONTINUOUS FUNCTIONS

**ABSTRACT:** Kolmogorov's Universal

Approximation Theorem guarantees the possibility to approximate a continuous multivariable function via real valued functions of a single variable but do not give a way to build this approximation. In this work, we use Cybenko's theorem to implement an Artificial Neural Network that approximate continuous functions defined on a closed interval. This network contains three distinct layers: an input layer, a hidden layer containing neurons with sigmoid activation function and an output layer that gives a linear combination of the outputs from previous neurons. As this network is a version of the classical Multilayer Perceptron, the approximation error is computed via Least Squares criterion. Therefore, the weights update is performed with the classic backpropagation algorithm without momentum or resilient terms.

**KEYWORDS:** Perceptrons, Continuous Functions, Artificial Neural Network.

## 1 | INTRODUÇÃO

Os Perceptrons de Múltiplas Camadas (PMC) são bastante utilizados em problemas de classificação. Uma outra aplicação desta heurística é na aproximação de gráficos de funções contínuas. Como temos os pontos do domínio e sua classificação através da equação, é possível implementar uma rede que utilize os dados de entrada e saída para aproximar o gráfico desta função.

## 2 | CONSTRUÇÃO DO ALGORITMO

A ferramenta teórica que torna possível

esta aproximação é uma versão do Teorema da Aproximação Universal de Kolmogorov. O Teorema descrito por Cybenko (Cybenko, 1989) diz que é possível realizar a aproximação por funções sigmóides num domínio compacto de  $\mathbb{R}^n$ .

Teorema 2.1 (Cybenko, 1989)

Seja uma função sigmoide contínua e  $I_n \subseteq \mathbb{R}^n$  um conjunto compacto e limitado. Então, as somas finitas da forma

$$G(x) = y = \sum_{i=1}^N \alpha_i g(w_i^T x + \theta_i)$$

onde  $\alpha_i, w_i, \theta_i \in \mathbb{R}$ , são densas em  $C(I_n)$ . Em outras palavras, dada qualquer  $f \in C(I_n)$ , existe  $\varepsilon > 0$  tal que

$$|y - f(x)| < \varepsilon, \forall x \in I_n.$$

Segundo Da Silva (Da Silva, et al., 2010), uma rede PMC consegue aproximar uma função contínua se as funções de ativação forem contínuas e limitadas. Vamos considerar uma rede com uma única camada escondida contendo  $H$  neurônios, as entradas  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  e a camada de saída contendo apenas um neurônio responsável pela combinação linear das saídas dos neurônios anteriores. Com isto, o resultado fornecido pela rede no ponto  $\mathbf{x}$  será denotado por

$$N(\mathbf{x}) = \sum_{i=1}^H \alpha_i g(Z_i) \tag{1}$$

onde  $g(Z_i)$  representa a função de ativação do neurônio  $i$  da camada escondida e o parâmetro  $\alpha$  representa o peso sináptico que liga este neurônio à camada de saída. O índice  $H$  indica a quantidade de neurônios na camada anterior à saída. Ou seja, o resultado final da rede será uma combinação linear das funções de ativação na camada escondida calculadas nos elementos  $Z_i$  que são determinados por

$$Z_i = \sum_{j=1}^n w_{ij} x_j + l_i,$$

onde  $l_i$  é o limiar do neurônio  $i$  e os parâmetros  $w_{ij}$  são os pesos sinápticos que ligam as coordenadas  $x_j$  da entrada  $\mathbf{x}$  aos neurônios da camada escondida.

## 2.1 Atualização dos Pesos Sinápticos

A implementação desta rede difere um pouco do PMC clássico pois, na última camada, a função de ativação apenas realiza a combinação linear das saídas dos neurônios das camadas anteriores. Vamos considerar que os valores desejados da rede são os valores da função calculada na entrada  $\mathbf{d} = f(\mathbf{x})$  e que a saída da rede para um

determinado \(\mathbf{x}\) é  $N(\mathbf{x}) = y$ .

Para um conjunto  $\{\mathbf{x}^{(k)}\}_{k=1}^m$  com  $m$  pontos, vamos construir a função de Erro Médio Quadrático

$$E = \frac{1}{m} \sum_{k=1}^m \frac{1}{2} (\mathbf{d}^{(k)} - y^{(k)})^2 \quad (2)$$

e, para cada parâmetro da rede  $w_i$  sua atualização se dará via Algoritmo de *backpropagation* que utiliza o Método do Gradiente Descendente

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}. \quad (3)$$

Na Camada de saída, os parâmetros da rede são representados por  $\alpha$ . Assim, para um ponto  $\mathbf{x}^{(k)}$ , vamos calcular a derivada da equação (2) e, utilizando a expressão da rede neural (1), temos

$$\frac{\partial E}{\partial \alpha} = -(\mathbf{d}^{(k)} - y^{(k)})g(Z).$$

É comum, na literatura, escrever que

Para facilitar o entendimento, vamos escrever a derivada da equação do erro em relação aos pesos da camada oculta de forma individual, ou seja, vamos escrever como atualizar um único elemento. Nesta camada, os pesos sinápticos do neurônio  $i$  que se ligam às coordenadas da entrada  $\mathbf{x}^{(k)}$  são representados por  $w_{ij}$  e a equação da derivada parcial se torna

$$\frac{\partial E}{\partial w_{ij}} = -(\mathbf{d}^{(k)} - y^{(k)})\alpha_i g'(Z_i)x_j^{(k)}.$$

Como as funções de ativação  $g$  são sigmóides, suas derivadas podem ser calculadas de forma recursiva

$$g'(Z) = g(Z)(1 - g(Z)).$$

Por fim, o limiar  $\theta_i$  de cada neurônio da camada escondida nos fornecem a seguinte derivada

$$\frac{\partial E}{\partial \theta_i} = -(\mathbf{d}^{(k)} - y^{(k)})\alpha_i g'(Z_i).$$

Com a determinação de todas as derivadas parciais em relação aos parâmetros da rede, podemos escrever, a partir da equação (3), como será realizada a atualização destes parâmetros para cada ponto  $\mathbf{x}^{(k)}$  do conjunto de dados

$$\alpha_i^{(k)} = \alpha_i^{(k-1)} + \eta(\mathbf{d}^{(k)} - y^{(k)})g(Z_i),$$

$$w_{ij}^{(k)} = w_{ij}^{(k-1)} + \eta(\mathbf{d}^{(k)} - y^{(k)})\alpha_i g'(Z_i)x_j^{(k)}, \quad (4)$$

$$l_i^{(k)} = l_i^{(k-1)} + \eta(\mathbf{d}^{(k)} - y^{(k)})\alpha_i g'(Z_i).$$

Observe que o índice  $w_i^{(0)}$ , nesta notação, indica o valor de inicialização do parâmetro  $w_i$  que é atribuído aleatoriamente.

### 3 I EXPERIMENTO

O funcionamento do algoritmo segue o PMC original. Inicializamos os valores dos parâmetros de forma aleatória e, na época 1, calculamos o valor da saída da rede para todos os pontos  $\mathbf{x}^{(k)}$  e o Erro Médio Quadrático associado. Se este erro estiver dentro da precisão requerida, o algoritmo é parado e o valor de  $y$  é gerado. Caso o erro não esteja dentro da precisão, passamos para a fase de atualização dos parâmetros da rede ponto a ponto. Ou seja, no ponto  $\mathbf{x}^{(k)}$ , calculamos a atualização dos parâmetros de acordo com as equações (4). Depois de calcular em todos os pontos do conjunto de dados, passamos para a próxima época, determinamos a saída da rede e verificamos, novamente, o valor do Erro Médio Quadrático.

Para a execução do algoritmo, optamos por utilizar, na camada escondida, 10 neurônios. A taxa de aprendizagem foi  $\eta = 0.2$  e o critério de parada do algoritmo foi definido em comparando o Erro Médio Quadrático entre duas épocas consecutivas. O limite máximo para a quantidade de iterações, também chamada de quantidade de épocas, é 4000.

#### 3.1 Funções Reais de uma Única Variável

Vamos, inicialmente, utilizar o algoritmo na aproximação de funções contínuas e limitadas definidas em um conjunto compacto  $I$  em que  $I \subseteq \mathbb{R}$ . Para verificar o desempenho do algoritmo, escolhemos as seguintes funções

1. Sigmoide  $f_1(x) = \frac{1}{1+e^{-x}}$ ,
2. Linear:  $f_2(x) = 3x + 2$ ,
3. Seno:  $f_3(x) = \sin(x)$ .

Ainda no domínio real, vamos aproximar algumas das funções de *benchmark* definidas no mesmo conjunto  $I = [-1,1]$ :

4. Rastrigin:  $f_4(x) = \frac{1}{10}(10 + (x-2)^2 - 10\cos(2\pi x))$ ,

5. Michalewicz:  $f_5(x) = 100\text{sen}(x)\text{sen}^2\left(\frac{x^2}{\pi}\right)$ .

Para os conjuntos de dados, foram escolhidos 50 pontos distribuídos de forma uniforme no intervalo  $[-1,1]$  para cada uma das funções implementadas.

### 3.2 Funções Reais de Várias Variáveis

Vamos realizar a aproximação de algumas funções quadráticas definidas num quadrado limitado e fechado  $I_n \subseteq \mathbb{R}$  em que  $x = (x_1, x_2, \dots, x_n) \in I$  se  $x_i \in [-1,1] \forall i = 1, \dots, n$ .

1. Paraboloide:  $f(x, y) = x^2 + y^2$ ,

2. Hiperboloide:  $f(x, y) = x^2 - y^2$ ,

3. Metade Superior de uma Esfera:  $f(x, y) = \sqrt{4 - x^2 - y^2}$ .

Para estes conjuntos de dados, foram escolhidos 81 pontos distintos para o cálculo das funções distribuídos uniformemente no quadrado  $I = [-1,1] \times [-1,1]$ .

## 4 | RESULTADOS OBSERVADOS

Esta seção concentra os resultados obtidos após a execução do algoritmo elaborado. Como a saída da rede é, visualmente, similar ao gráfico das funções, optamos por gerar os gráficos do Erro Local que representa, em valor absoluto, a distância entre a saída da rede e o valor da função em cada ponto do conjunto de treinamento.

### 4.1 Funções Reais de uma Única Variável

Para a rede com 10 neurônios na camada escondida, ao aproximar funções reais de uma única variável, observamos os seguintes resultados com o critério de parada fixado em  $10^{-6}$ .

Função	Número de Iterações	Erro Médio de Treinamento
<b>Sigmoide</b>	40	0.00001321
<b>Linear</b>	764	0.00115531
<b>Seno</b>	37	0.00017915
<b>Rastrigin</b>	354	0.00192533
<b>Michalewicz</b>	638	0.00849207

Tabela 1: Erro Médio do Conjunto de Treinamento.

Visualmente, podemos observar os erros locais em cada uma das execuções do algoritmo.

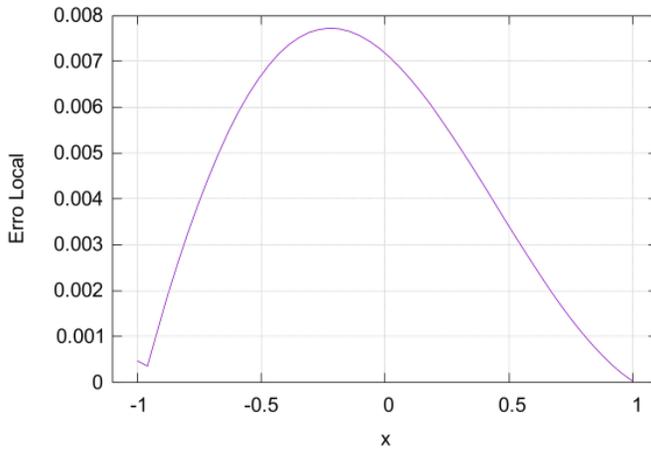


Figura 1: Erro Local Sigmoide.

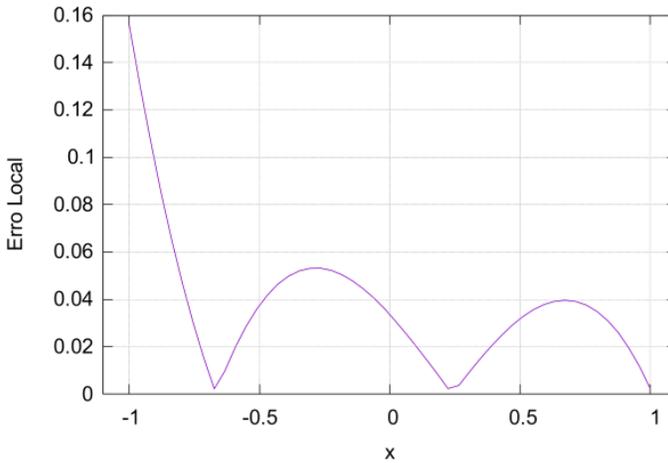


Figura 2: Erro Local Reta.

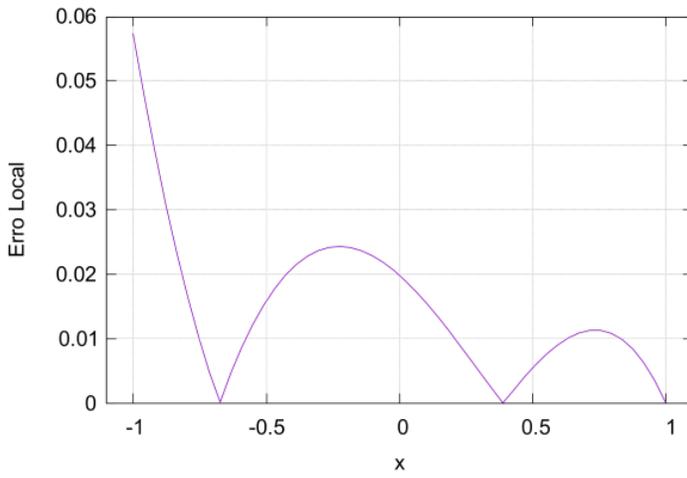


Figura 3: Erro Local Seno.

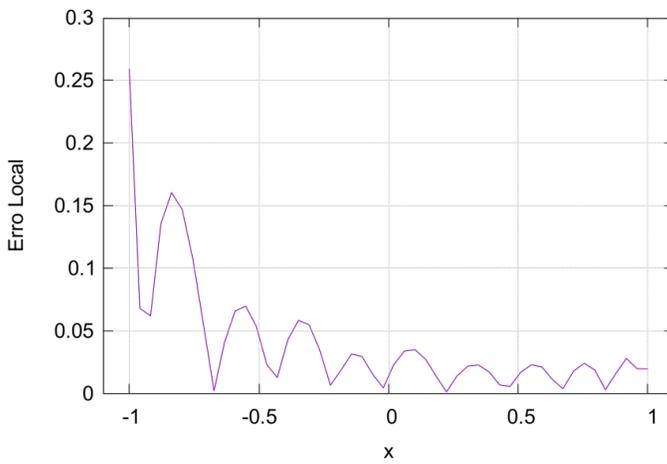


Figura 4: Erro Local Rastrigin

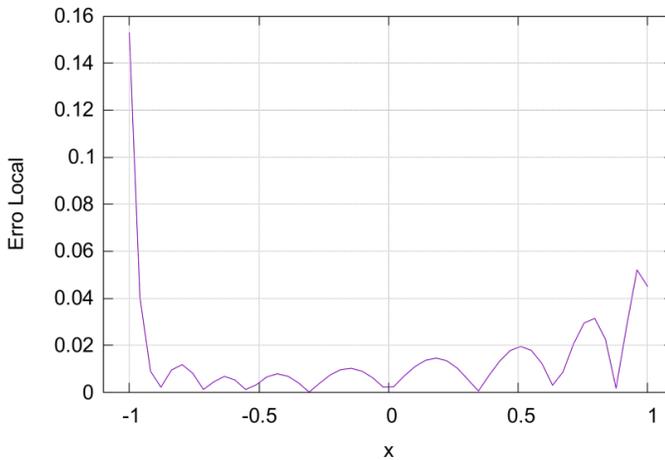


Figura 5: Erro Local Michalewicz.

Podemos observar que, ao utilizar o critério de parada em  $10^{-6}$ , o máximo do erro local ocorre no ponto inicial do intervalo quando a função a ser aproximada não é uma sigmoide e este erro local não excedeu 0.3. Caso seja necessário diminuir o valor do erro local na aproximação, deve-se alterar o critério de parada ou o valor da taxa de aprendizagem  $\eta$  de acordo com a função a ser aproximada.

## 4.2 Funções Reais de Várias Variáveis

Para estas funções, utilizamos a mesma quantidade de neurônios na camada oculta e o mesmo critério de parada. Os dados obtidos estão na tabela abaixo.

Função	Número de Iterações	Erro Médio de Treinamento
<b>Paraboloide</b>	2562	0.214552
<b>Hiperboloide</b>	1146	0.254583
<b>Esfera</b>	229	0.0212776

Tabela 2: Erro Médio do Conjunto de Treinamento.

As figuras a seguir mostram o Erro Local da aproximação para as três funções acima que foi determinado em cada um dos pontos do conjunto de treinamento.

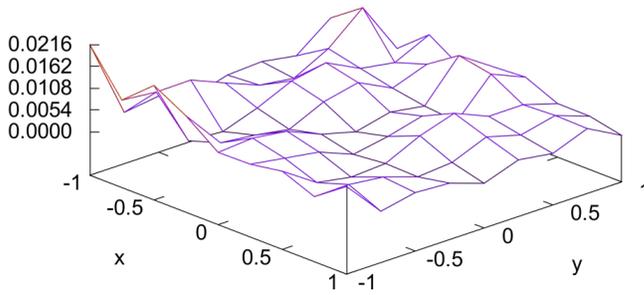


Figura 6: Erro Local Paraboloides.

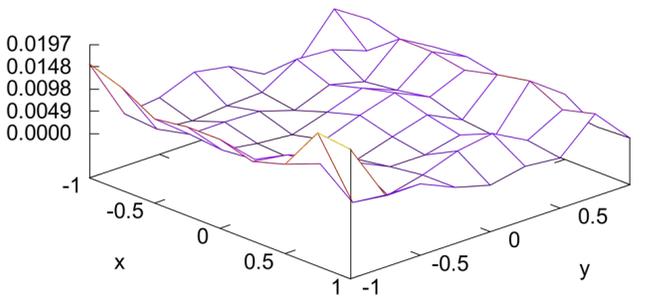


Figura 7: Erro Local Hiperboloides.

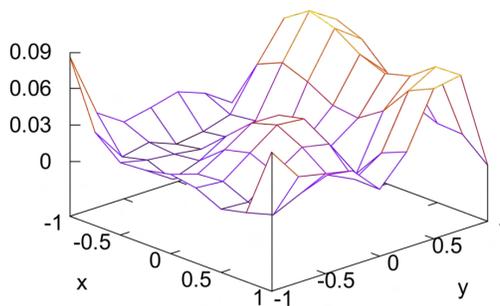


Figura 8: Erro Local Esfera.

Para as configurações da execução do algoritmo, o máximo do Erro Local ficou

abaixo de em todos os três casos.

Ao testar a capacidade de generalização da rede fora do conjunto de treinamento, realizamos novamente a aproximação da metade superior da esfera utilizando 441 pontos uniformemente distribuídos no intervalo  $[-1,1] \times [-1,1]$  e os parâmetros determinados na fase de treinamento, obtivemos o erro local descrito na imagem abaixo e seu valor máximo foi 0.0919. Observamos que o erro local mantém seu aspecto da fase de treinamento.

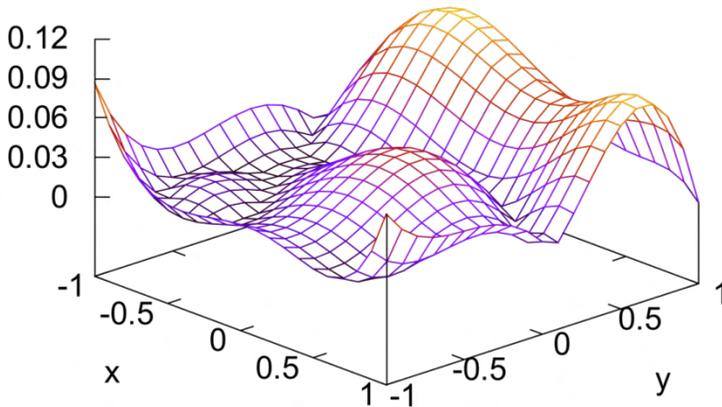


Figura 9: Erro Local em 441 pontos.

Com as funções de várias variáveis, observamos uma tendência do algoritmo em encontrar mínimos locais para a função Erro Médio Quadrático. Quando isto acontece, apesar da rede não conseguir melhorar a aproximação, o resultado final permanece, localmente, distante dos valores desejados. A imagem abaixo retrata esta situação. Para aproximar a metade superior de uma esfera, a rede utilizou 97 iterações e determinou o Erro Médio Quadrático em mas o erro local não está dentro do esperado: seu valor máximo é 0.357.

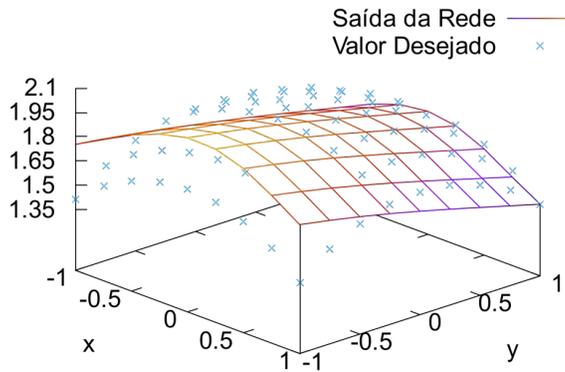


Figura 10: Mínimo Local para o Erro.

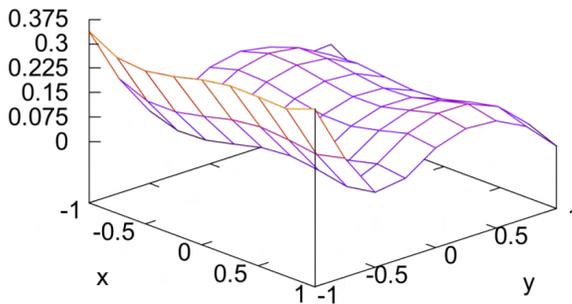


Figura 11: Erro Local na Aproximação da Esfera.

## 5 I CONCLUSÕES

A estrutura da rede implementada se assemelha ao Método de Aproximação conhecido por Mínimos Quadrados Discreto em que se escolhem funções pré-definidas para realizar uma combinação linear e determinar os melhores parâmetros para aproximar o conjunto de dados.

Com a utilização de poucos parâmetros, é possível aproximar funções contínuas dentro de uma precisão pré-determinada e generalizar esta aproximação para pontos fora do conjunto de treinamento mantendo o aspecto da saída da rede.

Há uma tendência de se obter um erro local maior nos pontos de fronteira para

aproximação de funções reais de uma única variável que não são sigmóides.

Para funções de várias variáveis, a rede fica mais suscetível a encontrar pontos de mínimo locais e o valor esperado pode ficar distante do valor desejado. Uma das possíveis soluções para este problema é a implementação de termos de momento ou resiliência. Outra alternativa pode ser a utilização de uma taxa de aprendizagem que varie ao longo das épocas.

A abordagem implementada utiliza funções de ativação sigmóides mas pode-se utilizar outros tipos de funções para a ativação dos neurônios da camada oculta. É possível alterar também a forma de atualização dos pesos sinápticos utilizando uma estratégia diferente do Método do Gradiente Descendente de modo a acelerar a convergência da rede neural.

## REFERÊNCIAS

**Cybenko, George. 1989.** Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*. 1989, pp. 303-314.

**Da Silva, Ivan Nunes, Spatti, Danilo Hernane e Flauzino, Rogério Andrade. 2010.** *Redes Neurais Artificiais para Engenharia e Ciências Aplicadas: Fundamentos Teóricos e Práticos*. São Paulo : Artliber, 2010.

## COMPUTAÇÃO EVOLUTIVA APLICADA AO MERCADO FINANCEIRO: UM SISTEMA DE RECOMENDAÇÃO ESTRATÉGICO PARA OS USUÁRIOS INICIANTES

*Data de aceite: 01/03/2022*

**Benjamin Luiz Franklin**

Universidade Estadual de Londrina – UEL.

**RESUMO:** Este trabalho aplica técnicas de computação evolutiva objetivando extrair uma estratégia de atuação para o mercado financeiro, ao abordar o fluxo de informações de ativos como um problema sistêmico. Para isso, coletou-se dados ativos financeiros para extrair-lhes os pontos favoráveis de compra e venda, para confrontá-los com o lucro gerado em suas transações, generalizando o resultado ótimo, de forma a encontrar uma solução abrangente, uma estratégia de ação. Os alertas de compra e venda, do melhor agente encontrado, são enviados às mídias sociais digitais para oferecer um sistema de recomendação a investidores iniciantes da bolsa de valores. Os resultados evidenciam a possibilidade de utilizar a computação evolutiva como um sistema de recomendações estratégicas para o mercado financeiro, disponibilizado massivamente a investidores iniciantes. As implicações dessa possibilidade, no entanto, convidam um conjunto de novos trabalhos, que possam aumentar a compreensão sobre o distanciamento entre o conhecimento estratégico, abstraído algoritmicamente de um conjunto de dados, distante da abordagem semântica, e o uso humano, cotidiano, da informação.

**PALAVRAS-CHAVE:** Mercado financeiro; Computação evolutiva; Extração de

conhecimento; Sistemas de recomendação.

### EVOLUTIONARY COMPUTING APPLIED TO THE FINANCIAL MARKET: A STRATEGIC RECOMMENDATION SYSTEM FOR NEW USERS

**ABSTRACT:** This work applies evolutionary computational techniques to extract a strategy of action in the financial market data, when approaching the flow of asset information as a systemic rather than a financial problem. For this purpose, we collected data on financial assets and sought to extract favorable purchase and sale timing, based on their continuous movement. We then compared it with the profit generated in their transactions, generalizing the optimal result through an evolutionary algorithm, in order to find an optimized solution – an action strategy. Buying and selling notifications, from the best agent found, are sent to digital social media offering a recommendation system to stock market investors. The results highlight the possibility of using evolutionary computation as a system of strategic recommendations for the financial market, which is widely available to beginner investors. We conclude that the application of non-symbolic systems to extract knowledge from the database, highlighting the financial system, presents a new range of challenges and opportunities, in which knowledge itself, as a competitive differential, gains new contours outside the semantic universe and ever more distant from ordinary human understanding.

**KEYWORDS:** Financial market; Evolutionary computing; Extraction of knowledge; Systems of recommendation.

## 1 | INTRODUÇÃO

No início do século XXI e, principalmente, após a crise de 2008, o mercado financeiro ampliou o uso da tecnologia e tem atuado sob o signo da imensa quantidade de dados minerados instantaneamente, viabilizando transações em alta frequência, inimagináveis há algumas décadas. O acesso a essa tecnologia fica restrito aos grandes atores do mercado, bancos e fundos de investimento, tornando o mercado financeiro um ambiente hostil ao pequeno investidor, pois os novos sistemas necessitam de grande estrutura técnica (BigData) e pessoal extremamente qualificado (cientistas de dados – *quants*) para viabilizar suas operações (software, modelização e manutenção de sistemas).

A participação das pessoas físicas é, no entanto, fundamental para o pleno desenvolvimento do mercado financeiro, descentralizando o poder de ação dos grandes conglomerados, tornando o mercado mais resiliente, com melhor governança e transparência (ALMEIDA, 2008). O mercado financeiro se popularizou entre as pessoas físicas, no Brasil, nos anos antes da crise de 2008, crescendo 545% em seis anos, despontando como uma alternativa de investimento à classe média. Até 2009, 600.000, ou 24,63% dos investidores eram pessoas físicas na BM&FBovespa<sup>1</sup>. As expectativas apontavam para o crescimento exponencial, que alcançaria 5 milhões de investidores até 2015.

Após as sucessivas crises políticas e econômicas, ocorridas após 2008, no entanto, os números demonstraram decréscimo, chegando a 13%, em 2013<sup>2</sup>, ensaiando uma pequena recuperação, em 2016, chegando a 17%<sup>3</sup>. Na economia norte-americana, a título de comparação, a classe média começou a participar do mercado financeiro na década de 1950. Antes da crise de 2008, 80% da população norte-americana investia no mercado de ações. Em 2016, esse número caiu para 52%<sup>4</sup>. A crise mundial afetou a credibilidade do mercado financeiro, em geral, e frustrou, em particular, as expectativas de crescimento da participação de pessoas físicas na bolsa de valores brasileira.

Esse cenário de crise somou-se ao incremento tecnológico como um diferencial competitivo na adoção de estratégias de operação no mercado, abrindo espaço para o uso de modelos preditivos dos comportamentos dos ativos financeiros e sistemas de negociação em alta frequência, que compram e vendem ações, tão rapidamente e em tão grande volume, que excedem a capacidade humana de operação no mercado (PATTERSON, 2011).

Como consequência, as formas tradicionais de investimento de longo prazo (*buy*

1 Detalhes em BM&FBovespa. Histórico de pessoas físicas. Recuperado em 8 de abril de 2017, de [http://www.bmfbovespa.com.br/pt\\_br/servicos/market-data/consultas/historico-pessoas-fisicas/](http://www.bmfbovespa.com.br/pt_br/servicos/market-data/consultas/historico-pessoas-fisicas/).

2 Detalhes em: Bolsa: participação de investidor pessoa física cai em dezembro - 03/01/2013 - InfoMoney. Recuperado em 8 de abril de 2017, de <https://economia.uol.com.br/ultimas-noticias/infomoney/2013/01/03/bolsa-participacao-de-investidor-pessoa-fisica-cai-em-dezembro.jhtm>.

3 Detalhes em: Cresce número de pessoas físicas como profissionais na Bolsa | EXAME.com - Negócios, economia, tecnologia e carreira. Exame. Recuperado em 8 de abril de 2017, de <http://exame.abril.com.br/seu-dinheiro/cresce-numero-de-pessoas-fisicas-como-profissionais-na-bolsa/>, 31 de outubro de 2016.

4 Detalhes em: GALLUP. *Just Over Half of Americans Own Stocks, Matching Record Low*. Recuperado em 8 de abril de 2017, de <http://www.gallup.com/poll/190883/half-americans-own-stocks-matching-record-low.aspx>.

*and hold*), pelas quais o investidor compra ações e não se preocupa com as variações de curto prazo, encontram-se questionadas, indicando participação de investidores focada em transações de curto (*day trade* – para operações no mesmo dia) e médio prazos (*swing trade* – para operações de até quatro semanas). Essas transações exigem dos investidores mais experiência e tempo empregado no acompanhamento e na análise das tendências do mercado, principalmente as de curto prazo, sendo estas, usualmente, atividades exercidas por profissionais das grandes corretoras e bancos<sup>5</sup>.

Os investidores não profissionais, em sua maioria, ficaram mais vulneráveis às oscilações de curto e médio prazo e não conseguem ainda estabelecer estratégias para se proteger de crises graves, em que podem perder grande volume de recursos e desistir de investir no mercado financeiro. Falta-lhes experiência, conhecimento e tempo para dedicar à análise do mercado, além de sistemas de informação – já usados amplamente pelos grandes atores<sup>6</sup>. Este trabalho justifica-se por viabilizar um sistema orientado à recomendação estratégica de operações, gratuito e focado no pequeno investidor, que poderia colaborar para a reintegração das pessoas físicas na arena do mercado financeiro, ao oferecer uma estratégia consistente de investimento e alertas, para momentos de atenção no comportamento dos ativos. O problema de pesquisa enfrentado consiste em verificar a possibilidade de extração de uma estratégia de ação para o mercado financeiro derivada do histórico de dados disponíveis, publicamente, na internet, sem a interferência de um especialista na área e que esta possa, ainda, ser amplamente difundida nas mídias sociais, de forma a beneficiar os entrantes inexperientes no mercado financeiro, como um sistema de recomendação. Nas próximas seções serão descritas a metodologia usada, destacando o caráter experimental do trabalho, além de seus conceitos fundamentais: a abordagem sistêmica do mercado financeiro e a computação evolutiva como instrumento de extração de estratégia para sua operação. Nas últimas seções do trabalho, abre-se uma constelação de sugestões para futuras pesquisas, dada as conclusões emergentes, em que o conhecimento extraído de bases de dados, por algoritmos evolutivos e conexionistas, apresentariam um desafio, ao distanciarem-se do modo semântico de representação linguística “tradicional”.

## 2 | MÉTODO

Esta pesquisa tem caráter aplicado e qualitativo, uma vez que pretende, por meio da análise quantitativa de dados, extrair deles as relações qualitativas e os pontos de compra e venda de ativos financeiros, como resultado da aplicação de algoritmos evolutivos

5 Detalhes em: “Bolsa não é para pessoas físicas”, diz especialista | EXAME.com - Negócios, economia, tecnologia e carreira. Exame. Recuperado em 9 de abril de 2017, de <http://exame.abril.com.br/seu-dinheiro/bolsa-nao-e-para-pessoas-fisicas-diz-especialista/>, 30 de dezembro de 2013.

6 Detalhes em: Acionista pessoa física, o maior abandonado. Valor Econômico. Recuperado em 9 de abril de 2017, de <http://www.valor.com.br/valor-investe/o-estrategista/2628796/acionista-pessoa-fisica-o-maior-abandonado>, 23 de abril de 2012.

em um corpus documental (a base de dados dos preços das ações, recolhidos de sites especializados), daí seu tipo exploratório, bibliográfico e documental.

O conceito de conhecimento que pretende-se extrair da série histórica de dados será resumido na relação entre os objetos em uma estrutura (GIL, 2008 p. 24), ou seja, dado o conceito de um ativo financeiro, quanto e quando serão a eles imputadas ações verbais (compra e venda) em dado momento, de forma a promover uma ação possível a um objeto.

Para isso, após a coleta de documentos que possam fornecer uma base de dados contendo o preço histórico dos ativos financeiros, diversos agentes de software serão criados, para que testem os pontos de compra e venda desses ativos, a partir de um modelo de indicadores que qualificariam os momentos de compra e venda, conforme sua relação com toda a série temporal. Cada agente, dessa forma, terá uma configuração diferente (aleatória) para produzir pontos de compra e venda de ativos, no decorrer da série histórica. Esse procedimento resultará em um número, indicando lucro ou prejuízo, como função de ajuste. O algoritmo evolutivo em questão tratará de reproduzir, promover mutação (nas configurações dos agentes) e selecionar os agentes mais bem-sucedidos, durante determinado número de gerações, fazendo emergir, possivelmente, uma estratégia global de investimento, como anuncia a computação evolutiva (MITCHELL, 2001). Diversas técnicas de inteligência computacional têm sido aplicadas, nos últimos vinte anos, no mercado financeiro, destacando Redes Neurais Artificiais, Sistemas Especialistas e Técnicas Estatísticas (TKÁČ; VERNER, 2016). As abordagens vindas da Inteligência artificial têm, no entanto, oferecido resultados mais consistentes que os métodos puramente estatísticos (HSU et al., 2016). Este trabalho aplicará, então, a computação evolutiva e sua característica de otimização global de estratégia, como método para emergir comportamentos locais de agentes, em situações específicas.

### **3 | O MERCADO FINANCEIRO VISTO COMO UM PROBLEMA SISTÊMICO**

O mercado financeiro tem sido analisado, tradicionalmente, por duas correntes distintas e antagônicas: a análise fundamentalista e a técnica, ou gráfica (BULKOWSKI, 2005). A análise fundamentalista busca entender os atributos financeiros reais das empresas, como balanços financeiros, previsões de investimento, passivos e ativos; enquanto a análise gráfica procura encontrar padrões nas séries históricas dos preços das ações, para encontrar momentos adequados de compra e venda. A fundamentalista necessita de conhecimento técnico específico, para que as análises ganhem consistência e sejam baseadas em argumentos explícitos, enquanto a análise gráfica exige experiência em encontrar alguns padrões “chave”, dispensando o conhecimento formal imprescindível na análise fundamentalista. Dessa forma, a análise gráfica é utilizada por investidores não iniciados e sem informações privilegiadas, contemplando uma ampla camada de usuários

leigos entrantes no mercado.

A análise técnica procura, então, encontrar padrões nas séries históricas de dados. A objeção feita a ela, neste trabalho, no entanto, é a de que esses padrões não emergem da experiência real da coleta de dados, de forma empírica, tratando de um consenso histórico que, não necessariamente, se repetirá no fluxo dos ativos financeiros.

A objeção dese trabalho à análise técnica consiste, basicamente, em duvidar que existam padrões fora da experiência sensível – local – de cada ativo financeiro. Não obstante, espera-se que padrões possam ser retirados das séries históricas e, nesse sentido, concorda-se com a análise gráfica, mas sem a presunção de que esses existam na tradição, mas na experiência cotidiana do mercado financeiro. Os padrões procurados, em suma, emergiriam de dados reais coletados do mercado, em vez de padrões esperados intrinsecamente, que dispensariam a experiência empírica localizada em um mercado específico.

Se, para a análise gráfica, o preço de um ativo já reflete toda a informação possível a seu respeito, é porque nela já existe uma visão sistêmica do fluxo informacional. A visão sistêmica, surgida na modernidade, procura abordar seus problemas utilizando uma metalinguagem, ou seja, transformando todas as questões abordadas sob um olhar unificado: uma questão de relacionamento e fluxo de informações entre o todo e suas partes. Essa abordagem prosperou, no século XX, com o advento da cibernética e, mais, tarde, com a teoria da complexidade (BAR-YAM, 1997). O ponto fundamental dessas abordagens é que fazem convergir diferentes disciplinas em uma forma unificada de representação computacional, em que diferentes domínios epistemológicos ganhariam um modelo de comunicação e relação ao participarem, nessa operação, de um contexto comum, capaz de operar e produzir transformações em uma estrutura unificada. O advento do sistemismo permitiu incluir uma série de disciplinas em seu paradigma unificador, dentre elas, a biologia, as ciências cognitivas e a própria economia (NEUMANN ; MORGENSTERN, 2007).

Sabe-se, com as investigações do pós-estruturalismo (POSTER, 1990), que não existe uma linguagem universal capaz de unificar a experiência humana, mas que, no entanto, para que os sistemas convirjam para o mesmo modelo de operação, estes devem ser integrados a um mesmo regime de informação, termo cunhado por Frohmann (GÓMEZ, 2012), para designar o contexto sociotécnico e jurídico: um mercado global, com os mesmos modelos de transformação, operação e comunicação, para todos os agentes envolvidos. A transformação digital, sendo um modelo de convergência sociotécnica, proveria esse novo contexto global unificador, associando ao fluxo dos ativos financeiros, toda a cadeia de relações e causalidades inscritas no mesmo “alfabeto que pensa”, usando o termo de Lanham (2007).

A partir do contexto unificado de causalidade, transformação e comunicação, possibilitado pelo regime global de informações, os atores sociais ganhariam novas ferramentas de perscrutação de seus horizontes, de produção de um sistema de inferências

e conclusões. O novo modo epistêmico unificador das operações é baseado em dados – a ciência de dados – em um modo transdisciplinar, porque é transobjetivo, unificador, mas indiferente às operações de sentido, quando, tudo o que interessa é apenas a correlação entre conceitos, objetos e eventos.

## 4 | MINERAÇÃO DE DADOS

A convergência tecnológica, ao prover um aparato comum de transformação de atributos para as entidades representadas, responde ao relacionamento mútuo entre classes e propriedades, convergindo diferentes problemas a uma estrutura comum (BRATTON, 2015). A resposta epistemológica equivalente, vinda da deriva do pensamento sistêmico, linguístico, cibernético e complexo, que agora unifica e converge as operações do mundo contemporâneo, aparece nas formas de análise, seleção, promoção e herança, dos pontos de um modelo comum, nas operações sistêmicas de agregação, categorização e consolidação: operações básicas e universais, aderentes a qualquer arranjo particular de um estruturalismo, garantido, materialmente, pelo aparato computacional globalizado.

Uma vez que grande parte da aparição social seja automaticamente convertida em rastros de dados, o sistemismo passa a oferecer um arsenal de possibilidades de investigação empírica que, antes, não poderiam sequer serem imaginadas. A mineração de dados oferece um desses conjuntos instrumentais, como as operações fundamentais de agregação, seleção e categorização, necessárias para a análise de um amplo conjunto de objetos passíveis da abordagem sistêmica, em busca de padrões e descobrimento de conhecimento (HAN; KAMBER, 2011).

Se, neste trabalho, aborda-se o fluxo do mercado financeiro como um problema sistêmico, é porque entende-se que o problema pode ser enfrentado a partir da gramática de tratamento de sistemas e suas operações genéricas de extração de padrões, correlações e categorias, ou seja, aproximando-se de uma visão cibernética interdisciplinar, em que as questões envolvidas são abstraídas da sua raiz informacional.

*[...] seu caráter interdisciplinar (da Cibernética) emerge quando considera a economia não como um economista, a biologia não como um biólogo, e a máquina não como um engenheiro. Em cada caso seu tema permanece o mesmo, isto é, como os sistemas se regulam, se reproduzem, evoluem e aprendem. Seu ponto alto é de como os sistemas se organizam. (Gordon Pask apud EPSTEIN, 1973, p. 9).*

Essa gramática genérica de tratamento sistêmico se traduz nas operações de mineração de dados e sua miríade de técnicas de extração, manipulação e categorização de padrões. De forma geral, as abordagens de extração de conhecimento são reduzidas às mesmas categorias das abordagens da inteligência artificial, ou seja: a simbólica, a conexãoista e a evolutiva (BARRETO, 2001).

A simbólica parte de um tratamento semântico do conhecimento, em que os

modelos do mundo são representados de forma silogística, de modo a permitir inferências e conclusões lógicas a cerca dos modelos. Esses modelos usam ontologias, bases de conhecimento e mecanismos de raciocínio como base de sua abordagem e são utilizados quando existe farto material documentado sobre o domínio escolhido e especialistas disponíveis para a estruturação desse conhecimento. A vantagem dessa abordagem está no domínio semântico dos modelos gerados, que podem oferecer explicações lógicas sobre seus procedimentos e conclusões.

As abordagens conexionistas e evolutivas, por outro lado, não dependem tanto da disponibilidade de especialistas para a formalização semântica do domínio específico. Por outro lado, dependem da ampla disponibilidade de dados qualificados para treinamento e extração de padrões. A abordagem conexionista procura imitar o funcionamento do aprendizado humano, a partir de suas conexões neurais, em que o aprendizado ocorre pela adaptação de uma rede de neurônios artificiais, representante de uma função que liga dados de entrada com dados de saída. O aprendizado consiste em encontrar a topologia correta da rede, por meio dos pesos das relações entre os neurônios, capazes de relacionar os dados de entrada e saída. Dessa forma, com uma grande base de dados, o sistema encontraria uma série de exemplos de entrada e saída e procuraria balancear a rede de forma a extrair um padrão que possa generalizar as relações. O conhecimento, então, seria o resultado dessa relação estrutural, entre os neurônios artificiais (ZUBEN, 2003).

Destaca-se que existe uma diferença fundamental nessas duas abordagens. Se a abordagem simbólica estrutura o conhecimento como um cômputo semântico de relações lógicas e, para isso, precisa de especialistas humanos capazes de formular essas relações, a abordagem conexionista, por outro lado, precisa de vasta quantidade de dados para abstrair suas relações entre os neurônios, resultando em “conhecimento” fora do sistema semântico de representação, ou seja, ela não favorece a explicação lógico-formal dos modelos gerados (IDEM).

A computação evolutiva, por sua vez, procura aplicar metáforas do modelo evolutivo a problemas computacionais, transformando sua solução em adequação evolutiva resultante do processo de seleção, promoção e herança de indivíduos de uma mesma espécie, no decorrer de várias gerações. Cada indivíduo – agente de software – é representante de um tipo de solução possível (genes), que é contrastado com a solução desejada, gerando um valor para a função de ajuste. Os indivíduos bem-adaptados – com alto valor na função de ajuste – serão selecionados para futuras gerações, em que se reproduzirão (trocarão genes entre si, formando outras soluções possíveis), podendo, inclusive, sofrerem mutações (alterações em seu código genético, originando novas soluções inesperadas), repopulando a espécie e reiniciando o ciclo evolutivo.

A computação evolutiva depende, de forma geral, da possibilidade de contrastar um indivíduo, representante de um código genético – uma solução possível para um problema computacional – a uma função de ajuste, para que sua configuração genética

possa ser avaliada e generalizada como solução de um problema. A evolução da espécie, como um todo, dependerá de grande poder computacional, uma vez que o processo evolutivo (a configuração genética adequada) poderá demandar grande quantidade de processamento computacional. O ponto a ser destacado é que essa abordagem também favorece o poder computacional e a grande quantidade de dados empíricos, como elementos centrais na extração de conhecimento, em detrimento do modelo semântico lógico-formal, dependente de um sistema silogístico de inferências, compatível com as capacidades linguísticas humanas.

Se, no modelo simbolista, fica em destaque a forma humana “tradicional” de conhecimento semântico, implicando a flexão verbal de entidades, criando-lhes acidentes e eventos, mas preservando-lhes a identidade em uma narrativa e, conseqüentemente, um “estilo” humano de comunicação, nas abordagens conexionista e evolutiva, o conhecimento emergente das grandes bases de dados dispensa o formato humano consagrado pelos cômputos semânticos, levando o próprio conhecimento a um estatuto desconhecido e alienado ao humano.

O grande poder de operação dos nossos dias, seja na convergência epistemológica dos modelos sistêmicos, na unificação sintética computacional, ou na grande capacidade de extração de conhecimento de bases de dados com diversas estruturas, amplia poderes de síntese, correlação e inferência nunca antes imaginados, mas oferece, também, desafios à própria ordem epistemológica, ao separar, ainda mais, a operação do mundo dos afazeres humanos cotidianos.

## 5 | EXTRAÇÃO DE ESTRATÉGIA

A ênfase deste trabalho é na possibilidade que existam pontos de compra e venda ideais no histórico dos preços dos ativos financeiros na bolsa de valores, e que encontrar esses pontos – o sonho dos investidores: “comprar na baixa e vender na alta” – possa ser possível, minerando-se os dados históricos dos ativos. Objetiva-se, em suma, encontrar um padrão derivado desses pontos de compra e venda e, assim, extrair uma estratégia lucrativa de investimentos, em que a diferença entre os pontos seja positiva, ou seja, que haja lucro, conforme a figura 1.

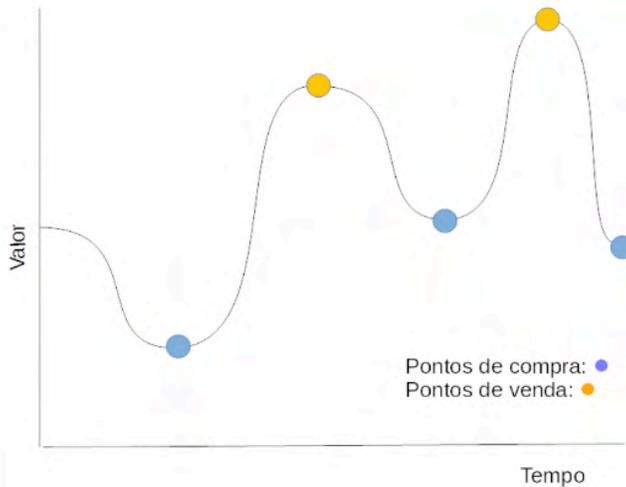


Figura 1: Pontos de compra e venda.

O padrão dessa série histórica tem a premissa de que se repetirá, enquanto a conjuntura sociotécnica permanecer e, enquanto isso ocorrer, seja possível inferir os momentos de compra e venda de ativos com o menor risco possível. Nossas premissas são de que esses padrões sejam voláteis, mas não caóticos, e indicam conjunturas que possam ser modeladas computacionalmente. O primeiro passo, então, para extrair os padrões de compra e venda, foi a construção de um banco de dados contendo os preços dos ativos financeiros a serem estudados, em que, a cada 15 minutos, diariamente, um agente de software busque nas agências de notícias financeiras (google finance e outros) seu valor atualizado e o armazene. Os ativos investigados nesse trabalho foram: PETR4 (Petróleo Brasileiro S.A.), VALE5 (VALE S.A), CSNA3 (Companhia Siderúrgica Nacional) e EMBR3 (Embraer S.A.). Esses não foram os únicos coletados, mas representam parte do acervo de vinte ativos, coletados desde 2009, perfazendo um conjunto de 1 GB de dados, que servirão como base para futuros trabalhos.

Uma vez coletados, esses ativos foram, programaticamente – utilizando a plataforma JAVA –, normalizados (valores entre 0 e 1) e agrupados em quartis móveis, contados a partir da data atual, para cada instante  $t$  e data menos quatro meses, normalizando os pontos de baixa e alta na série de dados.

Dessa forma, para cada instante  $t$ , na série histórica, tem-se um valor entre 0 e 1 equivalente (IQM4), que indica a posição relativa no quartil dado pela janela dos quatro meses anteriores, conforme a figura 2. Dessa maneira, pretende-se indicar os momentos em que um ativo se aproxima de um ponto de compra (quartil inferior,  $q_1$ ) ou de venda (quartil superior,  $q_4$ ). Se o valor do ativo ( $VA$ ), em um instante  $t$ , estiver em seu quadrante superior ( $q_4$ ) em IQM4 (seu quartil móvel normalizado em quatro meses) estará em uma

situação de possível venda. Da mesma forma, se estiver em  $IQM4-q1$ , ou seja, seu quartil inferior, estará em situação de possível compra.

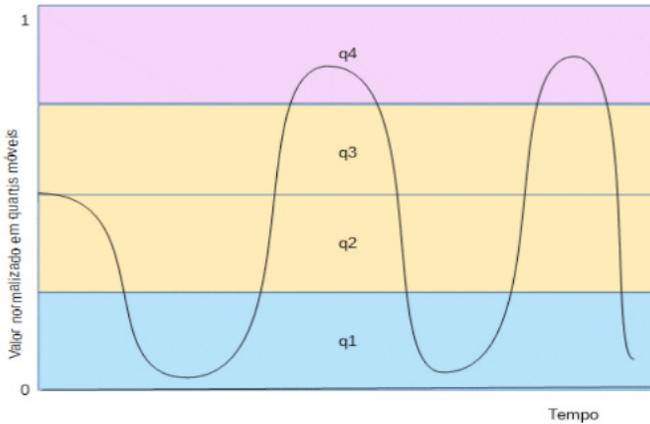


Figura 2: Valor normalizado para quartis móveis.

Estar em uma situação de compra ( $IQM4-q1$ ) ou de venda ( $IQM4-q4$ ), não é, no entanto, suficiente para detectar a realização de compra. É preciso que se confirme a situação de mudança de tendência, para uma tendência de alta, em  $IQM4-q1$ , ou queda, para  $IQM4-q4$ . Para isso, adotou-se mais um quartil de curto prazo IQD2, com um horizonte de apenas dois dias ( $d-2$ ), para detectar a mudança no curto prazo, quando a média dos valores entre  $t$  e  $t-d2$ , estará no quartil superior  $q1$ , para venda, ou  $q4$ , para compra, indicando mudança de tendência, conforme a figura 3.

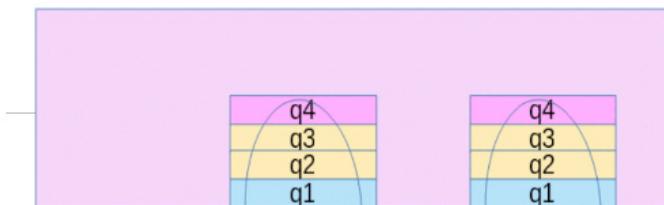


Figura 3: Mudança de tendência.

Dessa forma, tem-se dois índices:  $IQM4$  e  $IQD2$ , indicando, conjuntamente, pontos de compra e venda. A combinação desses dois indicará a pontuação de compra e venda, por exemplo, quando  $IQM4(0.20)$  e  $IQD2(0.90)$  caracterizam um ponto de compra, e  $IQM4(0.80)$  e  $IQD2(0.20)$ , um de venda. A experimentação dessas condições para a implementação de pontos de compra e venda, aplicados à série histórica do ativo, resultará em uma função de ajuste, necessária ao algoritmo evolutivo, para avaliar a configuração

genética do agente (IQM4 e IQD2). Dessa forma, a partir da população inicial de agentes, com configurações aleatórias de IQM4 e IQD2, converge-se uma solução ótima para os valores de IQM4 e IQD2, caracterizando uma solução global para determinado ativo, uma estratégia de investimento, como mostra o algoritmo da figura 4.

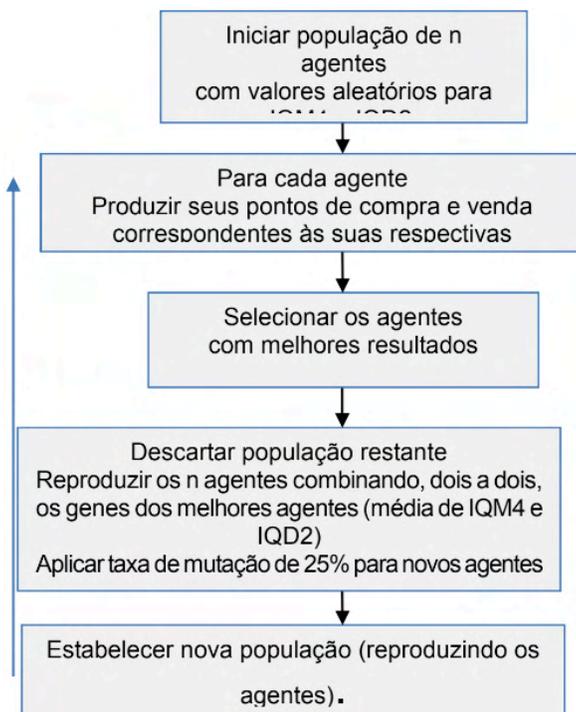


Figura 4: Algoritmo de evolução.

## 6 | RESULTADOS

O algoritmo foi aplicado a quatro ativos escolhidos, durante todo o ano de 2016, com população de 20 agentes, durante 20 gerações, obtendo os seguintes resultados, conforme a tabela 1.

Ativo	Compra (IQM4)	Compra (IQD2)	Venda (IQM4)	Venda (IQD2)	Lucro %
CSNA3	0,12	0,25	0,68	0,32	<b>48</b>
PETR4	0,17	0,81	0,57	0,84	<b>74</b>
VALE5	0,15	0,74	0,83	0,6	<b>52</b>
EMBR3	0,09	0,91	0,9	0,09	<b>26,5</b>

Tabela 1: Resultados do treinamento de algoritmo para o ano de 2016.

As configurações encontradas no treinamento foram testadas para um outro conjunto de dados, no primeiro semestre do ano de 2017, obtendo-se os seguintes resultados, conforme a tabela 2.

Ativo	Lucro %
CSNA3	47
PETR4	74
VALE5	51
EMBR3	27

Tabela 2: Resultados do teste de algoritmo para primeiro semestre do ano de 2017.

A semelhança nos resultados obtidos no treinamento e no teste do algoritmo aponta, conforme discutidos nas seções anteriores, que as mesmas tendências ainda atuam no mercado no ano de 2016 e no primeiro semestre de 2017, provavelmente indicando uma recuperação. Seria recomendável aplicar o algoritmo em um cenário diferente, com mais dados da atual tendência, para testar, com maior rigor, a capacidade de abstração e generalização do sistema, em um cenário livre da tendência negativa predominante nos anos anteriores, desde o início crise.

Para caracterizar o sistema de recomendação, integrou-se o conjunto resultante, contendo as configurações generalizadas para os ativos escolhidos, em um processo periódico de avaliação do estado atual do valor dos ativos financeiros, conforme a figura 6. Caso o valor do respectivo ativo, a cada 15 minutos, normalizado, conforme as seções anteriores, encontre-se nos valores críticos encontrados para IQM4 e IQD2, então um alerta é enviado para a rede social, no perfil do agente, para participar o usuário das condições de compra ou venda, recomendadas pelo sistema.

Para testar a integração do sistema, criou-se um perfil temporário, na mídia social Twitter®, para recomendar ações de compra e venda, em: <https://twitter.com/mymarx>.

## 7 | CONSIDERAÇÕES FINAIS

Neste trabalho, aplicou-se a computação evolutiva para extrair uma estratégia de ação para o mercado financeiro. Os resultados mostram a viabilidade dessa abordagem, apontando a lucratividade dessas operações. Dado o limitado poder computacional empregado no experimento, é recomendável, tentar alcançar resultados superiores, em futuros trabalhos, com uma estrutura tecnológica mais adequada.

É importante notar, também, que os padrões localizados são contextualizados à conjuntura social e política, devendo os modelos serem recalibrados, periodicamente, para manterem a coerência com os novos dados gerados, consistindo com as premissas iniciais do trabalho.

Ao tratar o movimento dos ativos financeiros como um problema sistêmico, conclui-se que a abordagem orientada a dados que desponta como paradigma unificador, na ciência do século XXI, apresenta um desafio epistemológico, uma vez que extrai conhecimentos cujo modelo de causalidade e relacionamentos se distancia do esquema semântico de representação, distanciando o conhecimento humano do produzido pelas máquinas. Esta e outras questões, no entanto, devem ser abordadas em futuros trabalhos, evidenciando a aptitude do tema.

## REFERÊNCIAS

- Almeida, J. (2013). In: *Gazeta mercantil/caderno A*. 21/08/2008. P. 3. Disponível em: <<http://index.gazetamercantil.com.br/arquivo/2008/08/21/101>> . Acesso em: 21 de março de 2013.
- Bahrammirzaee, A. (2010). A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. *Neural Computing and Applications*, v. 19, n. 8, p. 1165–1195. doi: 10.1007/s00521-010-0362-z, 2010.
- Bar-yam, Y. (1997). *Dynamics of complex systems*, Studies in nonlinearity. Reading, Mass: Addison-Wesley, 1997.
- Barreto, J. M. (2001). *Inteligência Artificial no limiar do século XXI*. Florianópolis.
- Bratton, B. H. (2015). *The stack: on software and sovereignty*, Software studies. Cambridge, Massachusetts: MIT Press.
- Bulkowski, T. N. (2005). *Encyclopedia of chart patterns*. 2<sup>nd</sup> ed. Hoboken, NJ: John Wiley.
- Epstein, I. (1973). *Cibernética e comunicação*. São Paulo: EDUSP.
- Feldman, R. (2007). *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge; New York: Cambridge University Press, 2007.
- Gartner (2012). *Technology Trends You Can't Afford to Ignore*. Disponível em: [http://www.gartner.com/it/content/1503500/1503515/january\\_19\\_tech\\_trends\\_you\\_cant\\_afford\\_to\\_ignore\\_rpaquet.pdf](http://www.gartner.com/it/content/1503500/1503515/january_19_tech_trends_you_cant_afford_to_ignore_rpaquet.pdf)>. Acesso em: 21 de março de 2013.
- Gil, A. C. (2008). *Métodos e técnicas de pesquisa social*. São Paulo: Atlas.
- Gómez, M. N. G. de (2012). Regime de informação: construção de um conceito. *Informação & Sociedade: Estudos*, v. 22, n. 3. Recuperado em 19 de agosto de 2017, de <http://www.ies.ufpb.br/ojs/index.php/ies/article/view/14376>, 2012.
- Han, J.; Kamber, M. (2011). *Data mining: concepts and techniques*. 3<sup>rd</sup> ed. Burlington, MA: Elsevier.
- Hsu, M. W.; Lessmann, S.; Sung, M. C.; Ma, T.; Johnson, J. E. V. (2016). Bridging the divide in financial market forecasting: machine learners vs. financial economists. *Expert Systems with Applications*, v. 61, n. Supplement C, p. 215–234. doi: 10.1016/j.eswa.2016.05.033.
- Lanham, R. A. (2007). *The economics of attention: style and substance in the age of information*. Paperback ed. Chicago: Univ. of Chicago Press.

Mitchell, M. (2001). An introduction to genetic algorithms, Complex adaptive systems. 7. print ed. Cambridge, Mass.

Neumann, J. Von; Morgenstern, O. (2007). Theory of Games and Economic Behavior. Princeton University Press.

Pang, B.; Lee, L. (2008). Subjectivity Detection and Opinion Identification. Opinion Mining and Sentiment Analysis. Now Publishers Inc.

Patterson, S. (2011). The quants: the maths geniuses who brought down Wall Street. London: Random House Business.

Poster, M. (1990). The mode of information: poststructuralism and social context. Chicago: University of Chicago Press.

Reynolds, R.; Liu, D. (2011). Multi-objective cultural algorithms. 2011 IEEE Congress of Evolutionary Computation (CEC) . p.1233–1241. doi: 10.1109/CEC.2011.5949757.

Tkáč, M.; Verner, R. (2016). Artificial neural networks in business: Two decades of research. Applied Soft Computing, v. 38, n. Supplement C, p. 788–804. doi: 10.1016/j.asoc.2015.09.040.

Witten, I. H.; Frank, E. (2005). Data Mining: Practical Machine Learning Tools and Techniques. [S.l.]: Morgan Kaufmann.

Zuben, F. (2003). Uma caricatura funcional de redes neurais artificiais. Rev. SBRN.

## ESTUDO DA REPROVAÇÃO NO CURSO DE SISTEMAS DE INFORMAÇÃO DA UNIMONTES NO PERÍODO DE 2014-1 A 2019-2

*Data de aceite: 01/03/2022*

### **Marilée Patta**

Universidade Estadual de Montes Claros -  
Unimontes, Montes Claros-MG, Brasil  
Lattes: 9384001525524653

### **Reginaldo Morais de Macedo**

Universidade Estadual de Montes Claros -  
Unimontes, Montes Claros-MG, Brasil  
Lattes: 0917411323478857

**RESUMO:** Gradativamente, percebeu-se aumento do número de alunos reprovados nas disciplinas do curso de Sistemas de Informação da Unimontes. Preocupados com a evasão e proposta de medidas para a permanência do aluno do curso, a reprovação se mostra como um complexo problema que sofre influências diversas, sendo pouco estudada. Dessa forma, considerando-se os semestres letivos de 2014-1 a 2019-2 geraram-se as médias de alunos matriculados, de aprovados e reprovados para se identificar as disciplinas e períodos letivos (1ºP a 8ºP) que mais reprovaram, assim como buscar subsídios para entender se a carga horária da disciplina agudiza os índices de reprovação. Os dados foram obtidos da Secretaria Geral da Unimontes e da Coordenação do Curso de Sistemas de Informação e, com uso da Estatística descritiva foi possível apresentar índices de alunos matriculados, aprovados e reprovados. Assim, verificou-se diminuição de alunos matriculados nos períodos centrais, do 3ºP até o 7ºP, e maior número de alunos no 1ºP

quando comparado com o 8ºP. As maiores médias percentuais de aprovações são percebidas entre o 4ºP e 6ºP e o 1ºP e o 2ºP são os que mais reprovaram. As três disciplinas de 108 horas, com maiores cargas horárias, foram as que mais reprovaram: Cálculo diferencial e integral (2ºP), Matemática Computacional e Algoritmos e estruturas de Dados I (1ºP). Porém, identificaram-se consideráveis índices de reprovação em disciplinas de 36h e 72h, tanto as mais árduas e robustas quanto aquelas consideradas mais leves. Embora os resultados forneçam um panorama das reprovações no curso, eles ainda são incipientes e indicam que os estudos deverão ser categorizados, pois há grupos de disciplinas com comportamentos semelhantes e grupo com uma ampla variabilidade.

**PALAVRAS-CHAVE:** Reprovação no Curso de Sistemas de Informação; Índices de reprovação.

**ABSTRACT:** Gradually, there was an increase in the number of students who failed the disciplines of the Information Systems course at Unimontes. Concerned with evasion and proposed measures for the permanence of the student in the course, failure is shown as a complex problem that suffers different influences, being little studied. In this way, considering the academic semesters from 2014-1 to 2019-2, the averages of students enrolled, passed and failed were generated to identify the subjects and academic periods (1st P to 8th P) that failed the most, as well as to seek subsidies to understand if the workload of the discipline exacerbates the failure rates. Data were obtained from the General Secretariat of Unimontes and from the Information Systems

Course Coordination and, with the use of descriptive statistics, it was possible to present rates of enrolled, approved and failed students. Thus, there was a decrease in the number of students enrolled in the central periods, from the 3rd to the 7th, and a greater number of students in the 1st when compared to the 8th. The highest average percentages of approvals are perceived between the 4th P and 6th P and the 1st and 2nd P are the ones that failed the most. The three subjects of 108 hours, with the highest workloads, were the ones that failed the most: Differential and Integral Calculus (2nd P), Computational Mathematics and Algorithms and Data Structures I (1st P). However, considerable failure rates were identified in subjects of 36h and 72h, both the most arduous and robust and those considered lighter. Although the results provide an overview of failures in the course, they are still incipient and indicate that the studies should be categorized, as there are groups of disciplines with similar behaviors and a group with a wide variability.

**KEYWORDS:** Failure in the Information Systems Course; Disapproval rates.

## INTRODUÇÃO

Discussões sobre currículos e taxas de reprovações em cursos na área da Computação devem se tornar rotinas nos colegiados para a busca de informações norteadoras nas Diretrizes e Referências Curriculares do MEC e em eventos da área e interdisciplinares. As Diretrizes Curriculares Nacionais para os cursos na área da Computação - DCN (MEC, 2016) reforçam que as organizações (empresas e governo), dependem dos sistemas de informação para a sua operação nas áreas funcionais (produção, marketing, recursos humanos, finanças, etc.) e que o uso destes sistemas representa potenciais ganhos de eficiência, de produtividade e competitividade para as empresas e as inserem em um cenário globalizado e competitivo. Na área, os profissionais requerem conhecimentos técnicos e organizacionais que afetam as operações e as estratégias da organização. Alinhado às Diretrizes e Referências curriculares do MEC, O Curso de Sistemas de Informação da Universidade Estadual de Montes Claros - Unimontes, implantado em 1996, procura atender demandas regionais e globais que propiciam benefícios para a sociedade. Conta com carga horária de 3672 horas, divididas em 8 períodos e um total de 57 disciplinas, com 28 vagas em cada semestre.

Com a grande demanda mundial por profissionais de tecnologia de informação e comunicação, o egresso do Curso de Sistemas de Informação passou a ser elemento estratégico para o mundo dos negócios, o que pode garantir boas vagas de emprego ao profissional da área.

O Curso de Sistemas de Informação, inserido em função social relevante, desde a sua criação, inseriu egressos no mercado de trabalho. Conta com casos de sucesso de ex-alunos que se dedicaram ao curso e se tornaram profissionais de empresas públicas e/ou privadas. Outros se inseriram em programas de pós-graduação *strictu-sensu* e tornaram-se professores de Faculdades, Institutos Federais e Universidades.

Por outro lado, em função da diversidade de demandas da sociedade por tecnologia, as vagas de trabalho na área de Computação têm se diversificado, registrando-se aumento em todo o mundo. Porém, as empresas, ao anunciarem vagas na área de tecnologia, muitas vezes, continuam sem supri-las por profissionais qualificados, com perfis que se alinham às suas demandas.

Neste contexto, apesar das promissoras perspectivas dos profissionais da área da computação, deve-se refletir sobre alguns fatores que podem levar a falta de profissionais. O número de cursos no mundo e no Brasil, é suficiente para composição de força de trabalho? Há poucas vagas disponíveis? O nível de especialização do conteúdo é denso, árduo e extenso, acarretando dificuldades no aprendizado? O ingressante tem conhecimento do que é realmente o curso? O estudante tem que trabalhar para se sustentar ou sustentar sua família? Considerando-se tais hipóteses, há de se afirmar que no curso de Sistemas de Informação na Unimontes percebe-se desmotivação de parte dos acadêmicos, sem a devida atenção aos conteúdos e dedicação aos estudos. Alguns, pela condição socioeconômica deixam o curso em prol do estágio ou emprego. Assim, um processo resultante é o elevado índice de reprovações ocorridas nas disciplinas, o que contribui diretamente para o aumento do índice de evasão.

A este respeito, e no intuito de monitorar a evolução dos acadêmicos, decidiu-se realizar investigação e análise dos dados visando entender a dinâmica de aprovação e reprovação dos acadêmicos do curso de Sistemas de Informação entre 2014 a 2019. Assim, este estudo tem como objetivo calcular e apresentar, a partir dos dados acadêmicos de 2014-1 a 2019-2, índices gerais de aprovação e reprovação, índices de reprovação em cada disciplina, além de identificar períodos (1º a 8º) e disciplinas que mais reprovam

Poucos trabalhos acadêmicos quantificam índices temporais relativos à reprovação, causando a ausência de parâmetros para guiar ações de gestores, professores e até mesmo desenvolvedores de ferramentas educacionais que visam a melhoria da qualidade de conteúdo e de ensino no curso. Por outro lado, também os alunos, sem tais índices, ficam sem parâmetros para se prepararem, de forma mais produtiva, para o processo ensino-aprendizagem que ocorre no curso.

## **MATERIAL E MÉTODOS**

Fez-se revisão teórica em livros, artigos científicos e Diretrizes e Referências curriculares do MEC. Os dados foram obtidos dos registros acadêmicos da Coordenação do Curso e da Secretaria Geral da Unimontes relativos aos períodos letivos de 2014-1, 2014-2, 2015-1, 2015-2, 2016-1, 2016-2, 2017-1, 2017-2, 2018-1, 2018-2 e 2019-2. Ressalta-se a falta de dados de 2019-1 pela ocorrência de greve na Unimontes. Obtiveram-se: estrutura curricular; ano e semestre que os acadêmicos cursaram cada disciplina; total de alunos matriculados por disciplina em cada semestre; departamento de origem das disciplinas;

identificador do aluno; frequência, nota e situação de aprovação (aprovado ou reprovado) em cada disciplina cursada pelo aluno.

Com uso da Estatística descritiva, foram apuradas as frequências e calcularam-se as médias de alunos matriculados, de aprovados e de reprovados em cada disciplina. As reprovações foram divididas em reprovação por nota e reprovação por zero ou falta (considerada desistência da disciplina).

Por fim, ranquearam as disciplinas das maiores para as menores taxas de reprovação, e apuraram-se os períodos (de 1º a 8º) que mais reprovaram no curso.

## RESULTADOS E DISCUSSÃO

O termo reprovação provém de um conceito latino e refere-se à falta de aprovação e ao não consentimento que um sujeito atribui a uma situação, uma coisa ou uma pessoa, que significa não aprovar (ou desaprovar). Reprovar é obter uma má qualificação, ou pelo menos uma qualificação que não é suficiente ou não cumpre um determinado objetivo. Por outro lado, o termo aprovar qualifica como bom, alguém ou algo. A reprovação é uma regra geral utilizada no âmbito da educação. Devido ao fraco rendimento, um acadêmico pode ser reprovado em uma disciplina, devendo cursá-la posteriormente para obter as qualificações necessárias para a sua aprovação e prosseguimento do curso.

Muitas são as causas associadas às reprovações dos alunos nos cursos superiores. Entre os fatores que podem levar o aluno a reprovação, destacam-se a dificuldade dos conteúdos, a desmotivação no estudo, a falta de compreensão da informação; a falta de tempo para dedicação aos estudos, a correção/avaliação subjetiva dos docentes, entre outros.

No curso de Sistemas de informação da Unimontes não é diferente. Na estrutura curricular constam disciplinas oriundas da Administração, Matemática, Computação, de cunho tecnológico, além de áreas suplementar e complementar. Contudo, as disciplinas voltadas para cálculos matemáticos e elaboração e uso de algoritmos carregam, em seus conteúdos, base necessária às várias disciplinas dos períodos subsequentes. Constata-se, em geral, baixo rendimento nas habilidades relacionadas a Matemática dos alunos que ingressam no curso.

Uma outra reflexão fiz respeito aos alunos que se frustram por terem uma interpretação errônea do objetivo do Curso. Para esta parcela de alunos, o curso formaria usuários de computadores, havendo visão distorcida das habilidades, objetivos e perfil do egresso. Esperam aprender a trabalhar com softwares básicos, tais como editores de texto, navegadores de Internet, dentre outros. Assim, muitos alunos desanimam, são reprovados em disciplinas e acabam evadindo. Adicionalmente, há relatos de problemas com a didática do ensino.

Porém, para construir soluções tecnológicas, o acadêmico deve desenvolver

habilidades e competências que envolvem o pensamento lógico lapidado. Para Paula *et al.* (2009), é difícil para os alunos conseguirem construir representações mentais que de fato abstraíam por completo um problema, sendo necessário buscar meios que estimulem a desenvolver tal capacidade. Hinterholz Jr. (2009) afirma que disciplinas de algoritmos costumam ter altos índices de evasão e reprovação, sendo um dos gargalos existentes nos cursos de graduação, particularmente de computação, dificultando ou até mesmo impedindo a continuidade dos alunos no curso. A necessidade de desenvolver um raciocínio lógico e descrevê-lo em linguagem específica, é determinante para este alto índice de reprovações.

Assim, apresentam-se desde a quantidade e a média de alunos matriculados, até a taxa de reprovação no intervalo.

De 2014-1 a 2021-1 computaram-se 2243 (duas mil quatrocentos e trinta e cinco) matrículas nas 57 disciplinas do curso. Dados de 2020-1 não entram no cálculo de reprovações (ver Figura 1).



Figura 1 – Alunos matriculados por semestre

Calculou-se a média de alunos matriculados nos períodos (1ºP a 8ºP), no intervalo de 2014 a 2019. A média de matriculados variou de 15 a 33 alunos e as maiores médias de matrículas se apresentam no 1ºP (33,13) e 2ºP (29,98) períodos, precedidas do 8ºP (19,04). As menores médias de alunos matriculados se concentram nos períodos intermediários, sendo no 3ºP (18,07), 4ºP (17,86), 6ºP (16,99), 7ºP (16,28) e 5ºP (15,91). Exibem-se também as médias de alunos aprovados e reprovados nos semestres letivos (ver Figura 2).

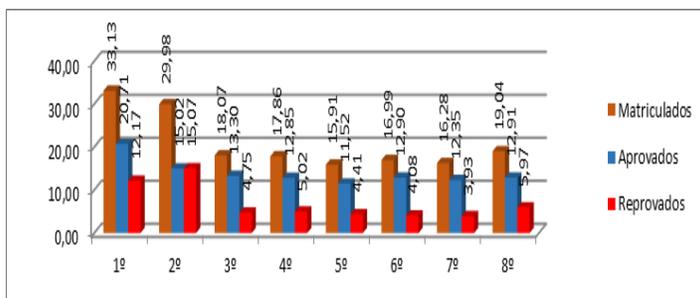


Figura 2 – Média de Alunos Matriculados, Aprovados e Reprovados no período de 2014 a 2019.

De 2014 a 2019, o índice médio de aprovação é de 73% e o de reprovação é de 27%. A reprovação foi estratificada em dois casos: 1) aluno frequente sem rendimento suficiente para aprovação (12,75%); 2) por falta ou zero, não participando de nenhuma atividade avaliativa ou não frequente na disciplina (14,4%) (ver Figura 3).

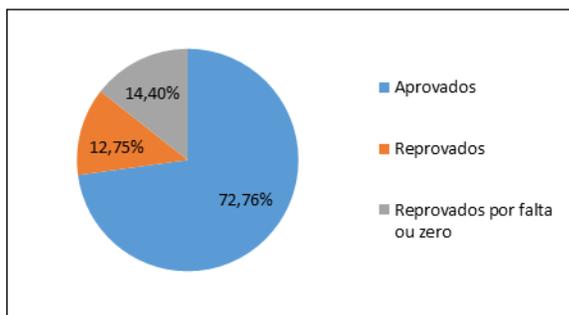


Figura 3 – Reprovados e Aprovados.

Os períodos com maiores médias de aprovação são o 4ºP (88,81%) e 6ºP (88,11%), precedidos do 7ºP (76,64%), 5ºP (76,20%), 8ºP (76,11%), 3ºP (74,62%), e ainda, o 1ºP (64,32%) e o 2ºP (53,26%), com menores médias de aprovação, sendo os que mais reprovaram. Em seguida, aparecem o 3ºP (25,35%), 8ºP (23,50%), 5ºP (23,84%) e 7ºP (23,37%) com empate técnico, o 6ºP (19,89%) e 4ºP (19,27%) com as menores médias de reprovações (ver Figura 4).

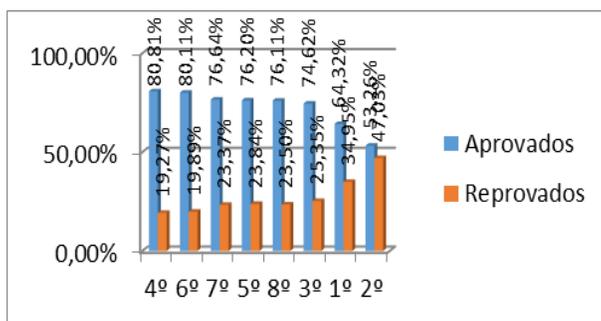


Figura 4 – Médias de aprovados e reprovados por período do curso.

Entendido como fator de evasão, fez-se um ranking em ordem decrescente de reprovações por falta ou zero. Dessa forma, os períodos com maiores índices de reprovação são o 2ºP (25,23%), 1ºP (19,7%), precedidos pelo 3ºP (13,87%), 5ºP (13,77%), 8ºP (12,33%), 7ºP (11,42%), e ainda o 4ºP (10,09%) e o 6ºP (8,81%), com menores índices (ver Figura 5).

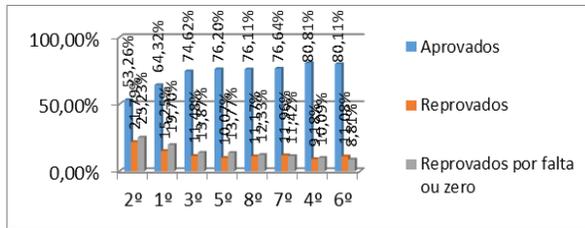


Figura 5 – Reprovação por falta ou zero.

Do primeiro ao oitavo período, os índices de reprovação nas disciplinas apresentaram grande variação, partindo de 1,94% na disciplina de Programação II, chegando a 83,29%, em Cálculo diferencial e integral. Apesar de os autores apontarem dificuldade em relação à aplicação da lógica, a disciplina de Algoritmos e estruturas de dados I (56,29%) é a sexta que mais reprova e Algoritmos e estruturas de dados II (43,88%), a décima. Antes aparecem, cálculo diferencial e integral (83,29%), comunicação de dados (74,77%), Redes de computadores (74,16%), Trabalho de conclusão de curso II (66,41%), Geometria analítica e álgebra linear (62,75%). Antes da décima estão Matemática Computacional (50,58%), Projeto de trabalho de conclusão de curso (47,64%) e Probabilidade e Estatística (44,99%) (ver Figura 6).

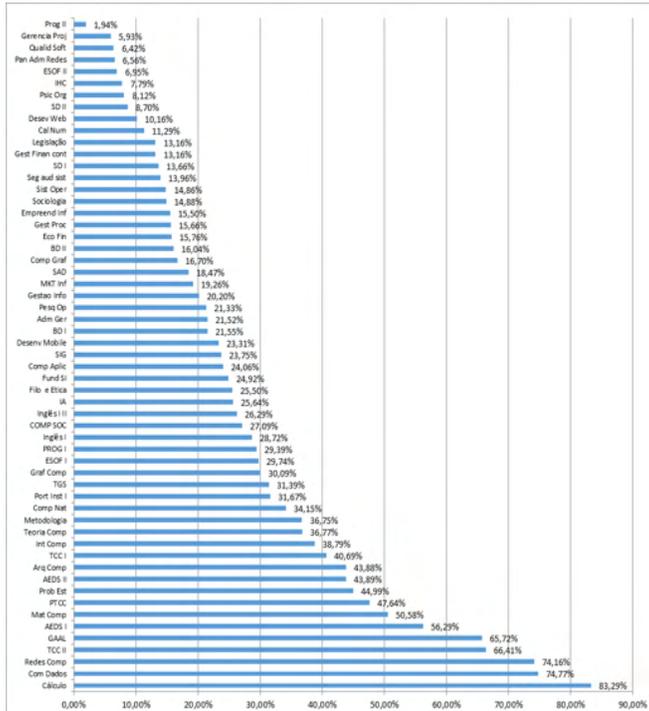


Figura 6 – Média de Reprovação nas disciplinas no período de 2014 a 2019.

Uma outra análise foi a identificação das disciplinas que mais reprovam nos períodos (1° a 8°) do curso, considerando suas cargas horárias.

No 1°P, as duas disciplinas que mais reprovaram são de 108 horas (Algoritmos e Estruturas de dados (56,29%) e Matemática Computacional (50,58%)), precedida de Introdução à Computação (38,79%) (72 h).

Interessante destacar que três disciplinas de 36 horas (Português Instrumental I (31,67%); Inglês Instrumental (28,72%); e Computador e Sociedade (27,09%)) registram maiores percentuais que outras duas de 72 horas (Fundamentos de Sistemas de Informação (24,92%); e Administração Geral (21,52%)) (ver Figura 7).

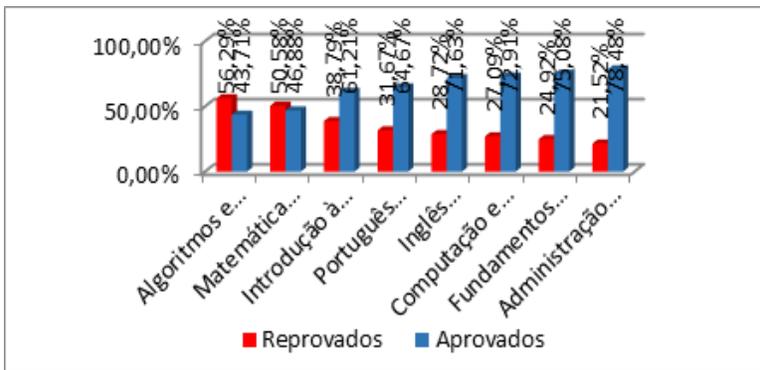


Figura 7 – Reprovação nas disciplinas do 1° período de 2014 a 2019.

No 2°P, a disciplina que mais reprovou foi Cálculo Integral e Diferencial (83,29%) de 108 horas. Em seguida, aparecem 4 disciplinas de 72 horas: Geometria Analítica e Álgebra Linear (65,72%); Algoritmos e Estruturas de Dados II (58,46%); Probabilidade e Estatística (44,99%); Arquitetura de Computadores (43,88%). Em seguida, registram-se Metodologia Científica (36,75%) (36 h) e Teoria Geral dos Sistemas (31,39%) (72 h). Por fim, aparece Inglês Instrumental II (26,29%) (36 h) (ver Figura 8).

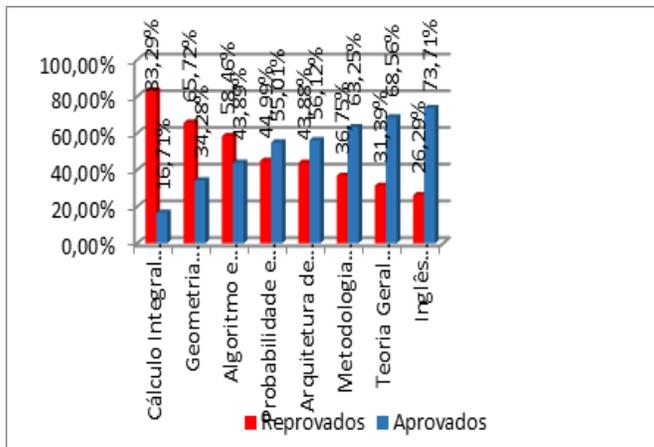


Figura 8 – Reprovação nas disciplinas do 2º período de 2014 a 2019.

No 3ºP, as médias de reprovação são bem menores que as do 2º e 1º períodos. A disciplina que mais reprovou foi Teoria da Computação (36,37%) (72 h), precedida de Grafos Computacionais (30,09%) (36 h). As demais disciplinas do período são de 72 horas: Engenharia de Software I (29,74%); Programação I (29,39%); Sistemas de Informação Gerencial (23,75%); Banco de Dados I (21,55%); Gestão da Informação e do Conhecimento (20,20%); 8) Cálculo Numérico (11,29%) (ver Figura 9).

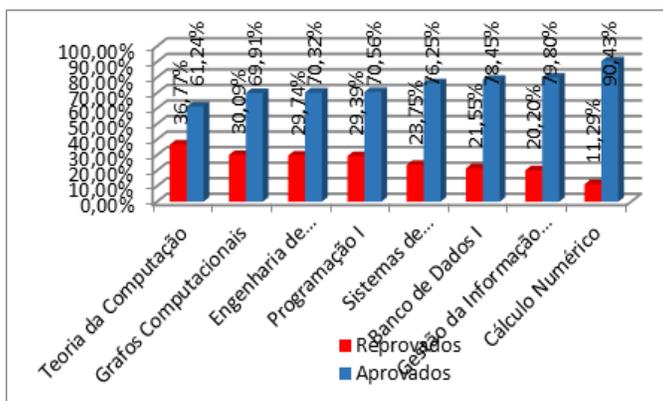


Figura 9 – Reprovação nas disciplinas do 3º período de 2014 a 2019.

No 4ºP, com alto índice, a disciplina que mais reprovou foi Redes de Computadores (74,16%) (72 h), precedida de Marketing em Informática (19,26%) (36 h). Ainda com 72 horas se encontram Banco de Dados II (16,04%), Sistemas Operacionais (14,86%), Gestão Financeira e Contábil (13,61%), Interação Humano Computador (7,79%), Engenharia de Software II (6,95%), e Programação II (1,94%) (ver Figura10).

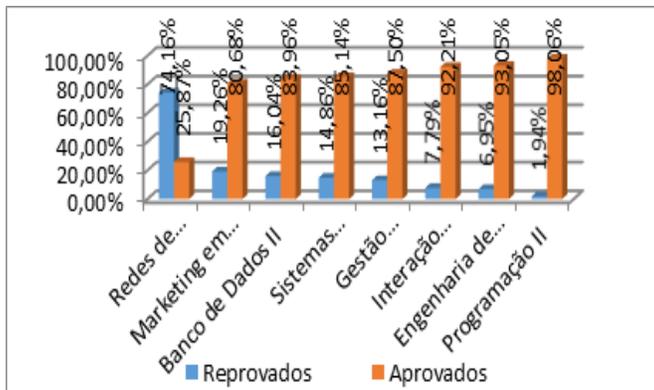


Figura 10 – Reprovação nas disciplinas do 4º período de 2014 a 2019.

No 5ºP, todas as disciplinas são de 72 horas. Destaca-se a disciplina Comunicação de Dados (74,77%) com alto percentual de reprovação. Em seguida aparecem Pesquisa Operacional (21,33%); Computação Gráfica (16,70%); Sistemas Distribuídos I (13,66%); Desenvolvimento Web (10,16%); e Qualidade de Software (6,42%) (ver Figura 11).

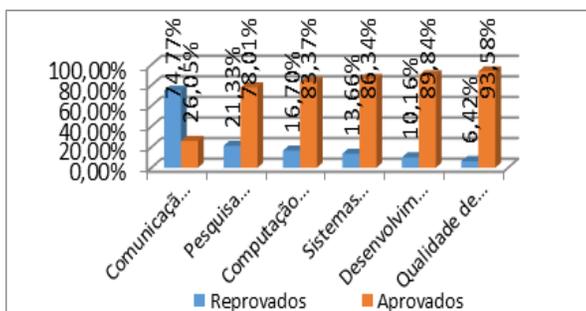


Figura 11 – Reprovação nas disciplinas do 5º período de 2014 a 2019.

No 6ºP destaca-se Projeto de Trabalho de Conclusão de Curso (47,64%) (36 h), com maior índice de reprovação. Em seguida, com 72 horas, registram-se Inteligência Artificial (25,64%), Desenvolvimento de Software para dispositivos móveis (23,31%), e Sistemas Distribuídos II (8,70%). Com menores índices, registram-se Psicologia Organizacional (8,12%) (36 h) e Gerência de Projeto de Software (5,93%) (72 h) (ver Figura 12).

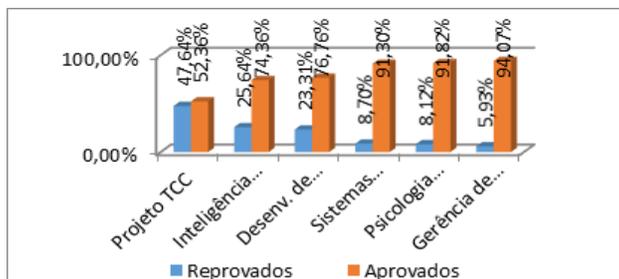


Figura 12 – Reprovação nas disciplinas do sexto período de 2014 a 2019.

No 7ºP, os maiores índices foram registrados nas disciplinas Trabalho de Conclusão de Curso I (40,69%) (36 h) e Computação Natural (34,15%) (72 h). As demais disciplinas registraram índices aproximados: Sistema de apoio à decisão (18,47%) (72 h); Economia e Finanças (15,76%) (36 h); Gestão de Processos (15,66%) e Empreendedorismo em informática (15,50%), (72 h) (ver Figura 13).

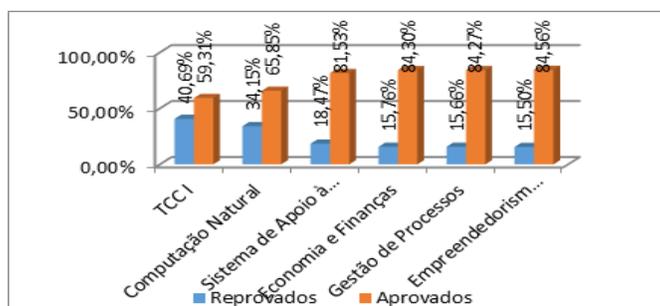


Figura 13 – Reprovação nas disciplinas do sétimo período de 2014 a 2019.

No 8ºP destaca-se a disciplina Trabalho de Conclusão de Curso II (66,41%) (36 h) com maior índice. Em seguida registram-se Filosofia e ética (25,5%), (36 h) e Computação aplicada - Informática na educação (24,06%) (72 h). Com menores índices aparecem Sociologia (14,88%) (36 h), Segurança e Auditoria de Sistemas (13,96%) (72 h), Legislação e ética Profissional (13,16%) (36 h) e Planejamento e Administração de Redes (6,56%) (72 h) (ver Figura 14).

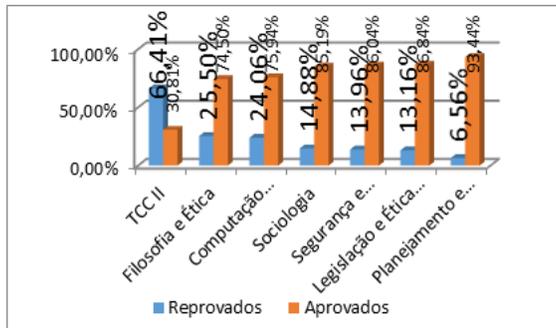


Figura 14 – Reprovação nas disciplinas do oitavo período de 2014 a 2019.

## CONCLUSÃO

A reprovação no Curso de Sistemas de Informação da Unimontes se mostra como um complexo problema que sofre influência de variáveis como fatores institucionais/acadêmicos, vocacionais, socioeconômicos, pessoais, familiares, de saúde, e demais outros.

Analisando-se a média de alunos matriculados, verifica-se sua diminuição nos períodos centrais, do 3ºP o 7ºP, com mais alunos no 1ºP e 8ºP que retratam o ingresso no curso e a formatura, respectivamente. Em função do menor número médio de alunos do 8ºP (19,04) em relação ao 1ºP (33,18), suspeita-se que nos períodos centrais está havendo trancamentos, e que aparentemente a desistência de alunos mais se configura entre o 4º e o 7º períodos.

Porém, pode-se verificar que as maiores médias percentuais de aprovações são percebidas entre o 4ºP e 6ºP e que o 1ºP e o 2ºP mais reprovaram, tanto por deficiência de nota ou de ausência do aluno na disciplina. Os índices do terceiro período são menores em relação ao segundo e ao primeiro. Para a segunda metade do curso, do quinto ao oitavo períodos, a disciplina Comunicação de Dados se destaca quanto à reprovação. Nos períodos finais, as disciplinas relacionadas ao Trabalho de Conclusão de Curso (36 h), ocuparam as primeiras posições. A este respeito, cabe considerar que grande parte dos alunos desiste das disciplinas PTCC, TCC I e TCC II mais de uma vez, havendo alta taxa de desistência, o que indica a necessidade de estudo específico sobre isso.

As três disciplinas de 108 horas foram as que mais reprovaram: Cálculo diferencial e integral (2P), Matemática Computacional e Algoritmos e estruturas de Dados I (1P). Em relação as disciplinas de 72 horas destacam-se maiores índices de reprovações na área tecnológica, da computação e de Ciências exatas. Porém, disciplinas de 36 horas de área complementar, suplementar e da computação também se destacam em índices de reprovação. A disciplina que menos reprovou foi Programação II (1,94%). Observa-se que as causas de tão baixo percentual de reprovação ainda não são conhecidas, pois fogem

fortemente ao comportamento de outras com mesma dinâmica de sala de aula, como por exemplo a disciplina de Algoritmos e estruturas de dados, com alto índice de reprovação.

Assim, as disciplinas de 108 horas se destacam, mas as outras, com 36 e 72 horas apresentam índices ainda não determinísticos, indicando que categorizações entre disciplinas deverão ser feitas na busca por padrões ou comportamentos de origem semelhante. Cabe assim destacar que os maiores índices de reprovação ocorrem tanto em disciplinas mais árduas e robustas, quanto em disciplinas de cunho teórico, consideradas mais leves pelos acadêmicos.

No intervalo estudado, pode-se perceber que não só disciplinas da área da computação como algoritmo ou programação, bem como da área das ciências exatas, como cálculo integral e diferencial tem forte representatividade quanto as taxas de reprovação no curso de sistemas de informação.

Por fim, pode-se conhecer, no período de 2014 a 2019, o índice de aprovação (73%) e de reprovação (27%) do Curso de Sistemas de Informação.

Há de se ressaltar que os resultados são ainda muito incipientes. Apresentaram-se disciplinas e períodos que mais reprovaram no intervalo estudado. Porém, sem demonstrar clara tendência quando se confronta a carga horária com reprovação, ou quando se verifica alto índice de reprovação em disciplinas de redes de computadores (4º e 5ºP) e de Trabalho de conclusão de curso (6º, 7º, 8ºP). Pode-se inferir que um estudo aprofundado deve ser realizado considerando-se as peculiaridades das disciplinas, como também os fatores que causam as reprovações, pois torna-se complexo encontrar tendências com este resultado.

O objetivo é utilizar este cenário para o planejamento de ações relacionadas à diminuição destes índices.

## AGRADECIMENTOS

À Secretaria Geral da Unimontes e à Coordenação do Curso de Sistemas de Informação pela disponibilização dos dados e pelo apoio.

## REFERÊNCIAS

Hinterholz Jr., Ornélio. **Tepequém: uma nova ferramenta para o ensino de algoritmos nos cursos superiores em computação**. In XVII-Anais do Workshop sobre Educação em Informática, 2009. 20, 21.

LAUDON, K.C., LAUDON, J.P. **Management Information Systems**. Pearson. 14 ed. Educação em Informática, 2009. 20, 21.

MEC, 2016. **Diretrizes Curriculares Nacionais para os Cursos de Graduação em Computação (DCN)**. Disponível em: <[http://portal.mec.gov.br/index.php?option=com\\_docman&view=download&alias=11205-pces136-11-pdf&category\\_slug=julho-2012-pdf&Itemid=30192](http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=11205-pces136-11-pdf&category_slug=julho-2012-pdf&Itemid=30192)> Acesso em Set. 2019.

PAULA, Leda Queiróz; PIVA JÚNIOR, Dilermando; e FREITAS, Ricardo Luis. **A leitura e a abstração do problema no processo de formação do raciocínio lógico-abstrato em alunos de computação.** Reverte-Revista de Estudos e Reflexões Tecnológicas da Faculdade de Indaiatuba, n.7, 2009. 22.

SEBESTA, R. W. **Conceitos de linguagem de programação.** Porto Alegre: Editora Bookman, 2002.p. 36.

## **SOBRE A ORGANIZADORA**

**LILIAN COELHO DE FREITAS** - Lilian received the BSc degree in Computer Engineering from Federal University of Pará (UFPA) - Brazil in march 2007. She received the MSc. degree in Electrical Engineering from UFPA in december 2009. Nowadays, she is a professor at Federal Institute of Education, Science and Technology of Pará (IFPA), in Brazil. Since 2003, she has been developing research in the wireless network area.

## ÍNDICE REMISSIVO

### A

Aprendizado de máquina 3, 1, 2, 3, 5, 6, 7, 9, 13, 18, 20, 33, 43, 44

Armazém inteligente 87, 88, 90, 94, 103, 104

### B

Bloom 51, 52, 54, 63, 65, 66, 69, 70, 71, 72, 73

Busca de custo uniforme 87, 89, 91, 92, 96, 97, 98, 99, 100, 102, 103

### C

Colônia de formigas 87, 91, 93

Computação evolutiva 4, 133, 135, 136, 139, 144

Covid-19 3, 33, 34, 35, 41, 42, 43, 45, 46, 47, 48, 49, 50, 75, 82

### D

Data augmentation 106

Doença de alzheimer 4, 105, 106, 119

### E

Experiência do usuário 3, 74, 75, 76

Extração de conhecimento 133, 138, 140

### F

Fluxo de carga linearizado 3, 21, 22, 23, 24, 25, 26, 29, 30, 31

Funções contínuas 4, 121, 124, 131

### G

Generative adversarial networks 106, 110

### I

Imagens de raio X 3, 33, 46

Índices de reprovação 147, 149, 152, 153, 158, 159

Inteligência lógico-matemática 3, 51, 52, 56, 57, 59, 72

### J

Jogos digitais de entretenimento 3, 51, 52, 53, 63, 64, 71

### M

Mercado financeiro 2, 4, 133, 134, 135, 136, 137, 138, 144

Multilayer perceptron 8, 35, 38, 105, 106, 109, 110, 121

## **O**

Otimização matemática 22, 23, 26, 28, 31

## **P**

Perceptron 4, 8, 35, 38, 105, 106, 109, 110, 121

Portais de notícias 3, 74, 76, 77, 79, 80, 82, 84, 85

## **R**

Rastreamento 3, 74, 75, 76, 77, 78

Reconhecimento de padrões 20, 33, 35, 40

redes neurais artificiais 20, 35, 49, 146

Redes neurais artificiais 4, 105, 106, 121, 132, 136

Redes neurais convolucionais 33, 34, 37, 46

Reprovação no curso de sistemas de informação 4, 147, 159

Roteirização 87, 89, 103, 104

## **S**

Sistemas de recomendação 133

Sistemas elétricos de potência 21, 22, 32

Solver knitro 22

## **T**

Teoria das Inteligências Múltiplas 51, 54, 71, 72

 [www.atenaeditora.com.br](http://www.atenaeditora.com.br)  
 [contato@atenaeditora.com.br](mailto:contato@atenaeditora.com.br)  
 @atenaeditora  
 [www.facebook.com/atenaeditora.com.br](http://www.facebook.com/atenaeditora.com.br)

*Collection:*

# APPLIED COMPUTER ENGINEERING 2

 [www.atenaeditora.com.br](http://www.atenaeditora.com.br)  
 [contato@atenaeditora.com.br](mailto:contato@atenaeditora.com.br)  
 @atenaeditora  
 [www.facebook.com/atenaeditora.com.br](http://www.facebook.com/atenaeditora.com.br)

---

*Collection:*

# APPLIED COMPUTER ENGINEERING 2