OBJETOS DE APRENDIZAGEM PARA ESTUDO DO ARCABOUÇO MATEMÁTICO DE TÉCNICAS DE CLASSIFICAÇÃO DO APRENDIZADO DE MÁQUINA E SUAS MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO

Rubens Dias Rodrigues Junior e Prof. Dr. Ivan Carlos Alcântara de Oliveira

Apoio: PIVIC Mackenzie

RESUMO

Um objeto de aprendizagem (OA) é todo e qualquer recurso educacional com a finalidade de apoiar o estudo e o aprendizado dos estudantes. Eles são reutilizáveis e podem incorporar texto, gráficos, animações, áudio e vídeo. Com base no descrito, este trabalho realizou o projeto e o desenvolvimento de três OAs para orientar os interessados no tema a compreenderem os componentes matemáticos dos algoritmos KNN (K-Nearest Neighbors), Naïve Bayes e SVM (Support Vector Machines) usando Python como plataforma de codificação e páginas do Jupyter Notebook contendo textos, gráficos e vídeos para enriquecer a experiência de aprendizagem.

Palavras-chave: Objetos de Aprendizagem, Aprendizagem de Máquina, SVM, KNN, Naive Bayes

ABSTRACT

A learning object (LO) is any and all educational resources intended to support student study and learning. They are reusable and can incorporate text, graphics, animations, audio and video. Based on what was described, this work carried out the design and development of three LOs to guide those specific to the subject to understand the mathematical components of the KNN (K-Nearest Neighbours), Naïve Bayes and SVM (Support Vector Machines) algorithms using python as a platform of acquisitions and Jupyter Notebook pages containing texts, graphics and videos to enrich the learning experience.

Keywords: Learning Objects, Machine Learning, SVM, KNN, Naive Bayes

1. INTRODUÇÃO

Objetos de aprendizagem são recursos educacionais que podem ser usados para apoiar o processo de ensino e aprendizagem em diferentes contextos educacionais, como escolas, universidades, treinamentos corporativos e cursos online. Segundo Wiley (2006), os objetos de aprendizagem são “qualquer recurso digital que possa ser reutilizado para o suporte ao ensino”. Nesse sentido, há uma ampla variedade de recursos em formatos diversos que podem ser trabalhados, por exemplo, na forma de vídeo, animação, podcast, simulação ou uma combinação deles. Esses recursos, podem ser utilizados para apresentar conceitos complexos de forma visual e interativa, proporcionando uma experiência de aprendizagem envolvente.

Tendo objetos de aprendizagem como recurso é apresentada a problemática do entendimento matemático no estudo de algoritmos de aprendizagem de máquina (em inglês, *Machine Learning - ML*). A busca de informações sobre o funcionamento de modelos de aprendizagem de máquina e suas métricas nem sempre retornam informações muito didáticas e de fácil entendimento, o que emerge a oportunidade de criação de um recurso educacional para o auxílio no arcabouço matemático e de aplicação dessas técnicas nos mais diversos cenários de negócio demandados atualmente. Não obstante, a complexa literatura, materiais técnicos e cursos disponíveis que carecem de didática eficaz, contendo linguagem matemática de difícil compreensão (MASOLA, 2019, apud SELBACH, 2010, p. 40) que muitas vezes não exploram outros meios de apresentar o conhecimento como, por exemplo, recursos visuais e simulações.

Mediante o exposto este artigo tem a intenção de explorar a construção de recursos educacionais digitais fazendo uso de formatos variados para fornecer conhecimento matemático e de aplicabilidade de modelos supervisionados de ML e suas métricas para auxiliar nos estudos de profissionais de ciência de dados e estudantes interessados nesse tema.

Os algoritmos selecionados para o estudo foram o K-NN (*K-nearest neighbors*), *Naive Bayes* e SVM (*Support Vector Machines*). O repositório dos objetos selecionado foi o jupyter notebook e a linguagem de programação python.

O conteúdo deste artigo é apresentado nas seções a seguir. Na Seção 2, tem-se o referencial teórico que aborda os conceitos de objetos de aprendizagem, técnicas de ML supervisionado com ilustrações do arcabouço matemático e as métricas de desempenho, bem como alguns simuladores de algoritmos de ML. A metodologia utilizada contendo uma síntese das etapas encontra-se na Seção 3. Na Seção 4, são apresentados os resultados

obtidos com destaque para os três OAs produzidos e algumas considerações finais. Por fim, na Seção 5, são destacadas as considerações finais e propostas de trabalhos futuros.

2. REFERENCIAL TEÓRICO

2.1 INTRODUÇÃO

Em sua tese Souza (2020) constata que o ensino de Matemática foca no produto, em fórmulas prontas e afirma que: “na maioria das Instituições de Ensino, principalmente em relação ao ensino de Matemática, a ênfase maior é dada ao produto em detrimento do processo, implicando na má qualidade do primeiro” (SOUZA, 2020, p. 23).

Esta constatação não é diferente quando entramos no campo da ciência de dados. Os problemas de negócio ficam cada vez mais complexos, assim como o volume de dados vem aumentando o que impulsiona o desenvolvimento de ferramentas computacionais mais sofisticadas baseadas em Aprendizado de Máquina (FACELI, 2021, p.2), que envolvem algoritmos mais complexos e de difícil compreensão matemática.

Levando-se em conta essa problemática, assim como uma demanda maior por profissionais de ciência de dados no mercado, há uma quantidade significativa de cursos de formação nessa área sendo ofertada, objetivando ensinar a aplicação de inteligência artificial em cenários de negócio, porém sem focar no processo de aplicação dos algoritmos em si e nem no processo matemático envolvido, mas apenas oferecer um *pipeline* de execução, o que gera o surgimento de profissionais de ciência de dados executores de algoritmos, mas sem conhecimento matemático e procedural adequado do que estão fazendo.

Ao se deparar com problemas reais em uma empresa ou pesquisa perguntas como “Qual algoritmo usar?”, “Que métrica(s) de avaliação utilizar?”, “Como compreender as métricas?”, “Por onde começar?”, tornam-se muito comuns e buscar literatura é um desafio. Uma das consequências é o grande uso de fóruns digitais, que tem uma “roupagem” de recursos educacionais digitais, os quais se tornam um dos principais recursos de aprendizagem para desenvolvedores de algoritmos computacionais atualmente.

2.2 OBJETOS DE APRENDIZAGEM

Como definir um Objeto de Aprendizagem? Embora encontradas muitas definições, aqui destaca-se:

> “Qualquer entidade, digital ou não digital, que possa ser usada, reutilizada ou referenciada durante o uso de tecnologias que suportem o ensino.” (WILEY, 2006)

Com o objetivo de identificar as características que um objeto de aprendizagem deve ter Santórum et al. (2021) realizou uma revisão da literatura e constatou que, realmente, não existe uma definição consensual entre os autores pesquisados sobre objetos de aprendizagem. Nesse estudo estabeleceu-se que um objeto de aprendizagem é um recurso digital, independente, reutilizável, interoperável e com um propósito educacional. Além disso, deve ser encontrado com algoritmos de busca simples e ter uma avaliação por sua efetividade, custo e usabilidade. Essa abordagem foi adotada neste projeto de pesquisa.

Os recursos didáticos digitais tem sido um elemento de motivação para os estudantes em seu processo ensino-aprendizagem, seja pelo apelo inovador, seja pela interação que estes recursos proporcionam. Os objetos de aprendizagem surgem da necessidade de aplicação desses recursos, aproveitando seu potencial de reutilização, portabilidade, classificação e identificação.

Uma das primeiras dificuldades ao trabalhar com objetos de aprendizagem é a multiplicidade de recursos disponíveis para sua concepção. As concepções têm variado desde a especificação da natureza do objeto (digital / não digital), seu tamanho (mínimo, extenso), suas características (estrutura de três componentes / estrutura livre), sua condição aberta / fechada, entre outras (CASTRO; DURÁN, 2008).

Por se tratar de recursos digitais que complementam o aprendizado, deve-se considerar as habilidades individuais e as diferentes formas de aprender dos estudantes ao conceber um objeto de aprendizagem. Este é um dos potenciais do uso de objetos de aprendizagem, pois eles facilitam os múltiplos meios de representação e transmissão do conteúdo pedagógico: textos, imagens, áudios, vídeos, animações, jogos (OLIVEIRA, 2018). Também promove a personalização dos recursos didáticos conforme as necessidades do estudante e seu estilo de aprendizagem.

Considerando a acessibilidade, os objetos de aprendizagem podem auxiliar as tecnologias assistivas (dispositivo ou recurso que auxilia pessoas com deficiência a realizarem suas atividades de forma independente) ampliando o acesso ao conteúdo didático de forma eficaz.

Indo para a ciência Matemática e Ciência de Dados, a demanda por objetos de aprendizagem parece alavancar o desenvolvimento de ferramentas digitais que vem mudando a forma de aprender. Inclusive, com o advento da pandemia do coronavírus iniciada em 2019 e o aumento das modalidades *on-line* de ensino, cada vez mais ser autodidata se faz necessário para conseguir acompanhar os estudos. Contudo, o uso da tecnologia deve ser valorizado pela natureza do conhecimento que pode se construir e não apenas porque pode oferecer motivação ou facilitar o aprendizado (TRIANA-MUÑOZ et al,

2016). Em Matemática, por exemplo, diversos recursos são encontrados na internet para apoio ao estudante, dos quais gostaria de destacar:

- Geogebra (Geogebra, 2023): Ferramenta para aprender matemática que traz a possibilidade de simulações em geometria, álgebra, gráficos estatísticos, cálculo etc.;
- Symbolab (Sy, 2023): Ferramenta para resolução de equações algébricas, trigonométricas, cálculo etc.;
- *Derivative Calculator (*Derivative Calculator*, 2023)*: Ferramenta para cálculo de equações diferenciais com passo-a-passo;
- *Integral Calculator (*Integral Calculator, 2023*)*: Ferramenta para cálculo de integrais com passo-a-passo.

Como exemplos da utilidade desse tipo de recurso, foram encontrados alguns sítios de simulação na internet, também chamados de *playground*, nos quais é possível aprender modelos de aprendizado de máquina de maneira didática por simulação, agregando valor à aprendizagem, tais como:

- *Tensorflow Playground (*Maruseac, 2023): É possível simular os parâmetros de uma rede neural para entender seu funcionamento;
- *Time-Series Playground* (Encora, 2023*)*: Simulador de séries temporais que possibilita testar parâmetros, decompor séries, avaliar métricas e obter conhecimento matemático envolvido nos algoritmos.

2.3 MODELOS DE APRENDIZAGEM DE MÁQUINA SUPERVISIONADOS

Segundo Faceli (2021), tarefas de ML podem ser divididas em Preditivas e Descritivas. Modelos preditivos podem ser utilizados, por exemplo, para, a partir de seus sintomas, predizer o estado de saúde de um paciente. Nessas tarefas, em geral são utilizados algoritmos de ML que seguem o paradigma de aprendizado supervisionado (FACELI, 2021, p. 3).

Dentro da categoria de aprendizado de máquina temos os modelos de classificação. Tais modelos consideram em sua etapa de treinamento, rótulos conhecidos (também chamados de etiquetas ou categorias) com o objetivo de aprender padrões para fazer melhores predições em dados futuros (dados de teste ou dados em produção).

Segundo Olson (2017), os algoritmos a seguir merecem atenção:

- Naive Bayes;
- Regressão Logística;

- Árvore de Decisão;
- SVM: Máquina de Vetor de Suporte;
- *Random Forest*: Florestas Aleatórias;
- K-NN: Vizinhos mais próximos;
- XGBoosting.

Neste trabalho foram selecionados os algoritmos K-NN, Naive Bayes e SVM. Essa escolha levou em consideração as características matemáticas variadas envolvidas na resolução desses algoritmos. O KNN envolve o cálculo de distância entre pontos (geometria analítica), o Naive Bayes é baseado em probabilidade e o SVM utiliza elementos de álgebra linear.

Uma descrição desses algoritmos é realizada por Mahesh (2020), a saber:

- **K-*nearest neighbors* (K-vizinhos mais próximos):** O algoritmo K-NN é um algoritmo simples e supervisionado de aprendizado de máquina que pode ser usado para resolver problemas de classificação e regressão. É fácil de implementar e entender, mas tem a grande desvantagem de ficar significativamente lento à medida que o tamanho dos dados em uso aumenta;
- ***Naive Bayes*:** É uma técnica de classificação baseada no Teorema de *Bayes* com uma suposição de independência entre os preditores;
- **Máquina de Vetor de Suporte (SVM):** Os SVMs podem executar com eficiência uma classificação não linear usando o que é chamado de truque do *kernel*, mapeando implicitamente suas entradas em espaços de recursos de alta dimensão.

2.3.1 KNN (K-NEAREST NEIGHBORS)

O modelo K-*Nearest Neighbors* (KNN – K-vizinhos mais próximos) é um algoritmo de aprendizado de máquina utilizado para classificação e regressão. Ele é um dos métodos mais simples e intuitivos para classificar novos pontos de dados com base na proximidade com exemplos existentes.

O funcionamento do KNN é bastante simples. Ele armazena um conjunto de dados de treinamento com rótulos conhecidos e, quando recebe um novo exemplo sem rótulo, procura pelos k vizinhos mais próximos no conjunto de treinamento. A distância entre os pontos é calculada usando uma métrica, como a distância euclidiana, e os k vizinhos mais próximos são selecionados.

No caso da classificação o rótulo mais comum entre os k vizinhos é atribuído ao novo exemplo. No caso da regressão a média dos rótulos dos k vizinhos é usada como o valor previsto para o novo exemplo.

O valor de k é um parâmetro importante no KNN. Um valor menor de k torna o modelo mais sensível a ruídos e flutuações nos dados, enquanto um valor maior suaviza as fronteiras de decisão e pode perder detalhes finos.

O KNN tem algumas limitações como a necessidade de manter todo o conjunto de treinamento na memória para realizar as consultas de vizinhos mais próximos, o que pode ser computacionalmente caro para grandes conjuntos de dados. Além disso, a escolha adequada da métrica de distância e do valor de k também é importante para obter resultados precisos.

Em resumo, o KNN é um algoritmo de aprendizado de máquina simples e eficaz que classifica novos exemplos com base na proximidade com exemplos de treinamento, tornando-o adequado para problemas de classificação e regressão.

Segundo Faceli(2021), o KNN apresenta as seguintes vantagens e desvantagens:

Vantagens:

- Algoritmo de treinamento Simples;
- Simplificação de função objetivo para otimização;
- Aplicável em problemas simples e complexos;
- Algoritmo incremental para fins de treinamento.

Desvantagens:

- Maior esforço computacional na fase de treinamento;
- Para classificar uma nova observação, requer calcular a distância deste objeto em relação a todos os objetos de treinamento.

Matematicamente, o algoritmo KNN pode ser explicado com a aplicação do teorema de Pitágoras através de uma definição simples de geometria analítica, conforme complementa Faceli em sua literatura através da Eq. 1.

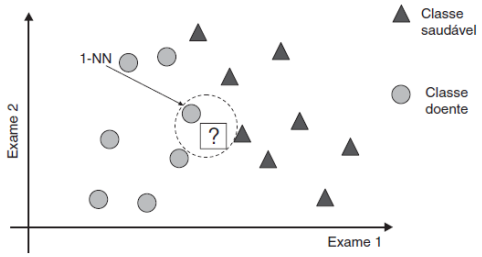
$$|PQ| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad \text{(Eq. 1)}$$

Para tornar a matemática mais conveniente, podemos generalizar a formulação e descrever a distância euclidiana, ou distância entre 2 pontos, como ilustrado na Eq. 2.

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{d} \left(x_i^l - x_j^l\right)^2} \qquad \text{(Eq. 2)}$$

A Figura 1 apresenta um exemplo ilustrativo da aplicação do KNN com o K=1, fazendo a divisão em duas classes.

Figura 1: Exemplo de 1-NN. K=1



Fonte: Faceli (2021)

2.3.2 NAIVE BAYES

O algoritmo Naive Bayes é um método de classificação probabilístico baseado no teorema de Bayes. Ele é amplamente utilizado em tarefas de classificação de texto como análise de sentimento, detecção de spam e categorização de documentos.

O algoritmo Naive Bayes assume que as características são independentes entre si, o que pode não ser verdade na prática. No entanto, mesmo com essa simplificação, o Naive Bayes pode ser muito eficaz em muitos casos.

Existem variações do algoritmo Naive Bayes, incluindo o Naive Bayes Multinomial, o Naive Bayes Bernoulli e o Naive Bayes Gaussiano. Cada variação é adequada para diferentes tipos de dados e problemas de classificação.

Segundo sua formulação matemática, o algoritmo Naive Bayes parte da premissa de uma probabilidade a priori para encontrar, então, uma probabilidade a posteriori de uma observação pertencer a uma determinada classe dentro das observações disponíveis, conforme explicitado na Eq. 3 (Mahesh, 2020).

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \qquad \text{(Eq. 3)}$$

Na aplicação da fórmula de Bayes podemos encontrar valores muito baixos de probabilidade. Para que possamos nos prevenir deste comportamento também chamado de *underflow*, a formulação é incrementada pela função *log*, conforme explicita Faceli(2021) na Eq. 4.

$$\log(P(y_i \mid \mathbf{x})) \propto \log(P(y_i)) + \sum_j \log(P(x^j \mid y_i)) \qquad \text{(Eq. 4)}$$

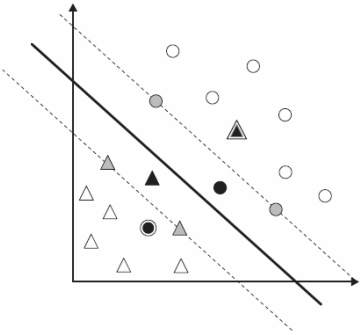
Segundo Jadhav et al (2016), as vantagens do algoritmo de Bayes é sua performance, requer pouco recurso computacional no treinamento e melhora a performance da classificação por remover variáveis irrelevantes. Contudo, para que os resultados do modelo sejam bons, o classificador Naive Bayes necessita de uma base de dados grande e também apresenta menos acuracidade para alguns tipos de *datasets*.

2.3.3 SVM (SUPPORT VECTOR MACHINES)

Support Vector Machine (SVM) é um algoritmo de aprendizado de máquina supervisionado que é frequentemente usado para classificação e regressão. Ele funciona encontrando um hiperplano que separa os dados em diferentes classes ou que melhor se ajusta aos dados em uma regressão.

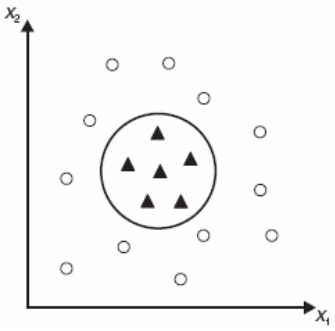
Basicamente, o algoritmo SVM cria margens de decisão em um hiperplano, sendo que tais margens são definidas pelo que chamamos vetor de suporte. Estes vetores determinarão os limites de decisão para que uma observação pertença a uma classe ou outra, segundo podemos ver nas Figuras 2 e 3, encontradas em (Faceli, 2021).

Figura 2: SVM Linear



Fonte: Faceli (2021)

Figura 3: SVM Não-Linear



Fonte: Faceli (2021)

Simplificando a matemática existente neste algoritmo, temos a formulação Eq. 5 e Eq. 6 para entender o algoritmo SVM (Winston, 2010).

$(w, x_n) + b \geq 0 \; quando \; y_n = +1$ (Eq.5)

$(w, x_n) + b < 0 \; quando \; y_n = -1$ (Eq. 6)

Traduzindo as Eq. 5 e 6, o algoritmo SVM calcula o produto escalar entre vetores e, dependendo da direção do vetor em relação ao vetor de suporte determinado pelo algoritmo, ele consegue classificar uma observação em sua classe adequada.

O SVM é um algoritmo de aprendizado de máquina popular devido à sua capacidade de lidar com dados complexos e de alta dimensionalidade. No entanto, sua sensibilidade a hiperparâmetros e sua dificuldade em lidar com grandes conjuntos de dados são algumas das desvantagens que precisam ser consideradas ao escolher o SVM como método de classificação (Zhang, 2021).

2.4 MÉTRICAS DE DESEMPENHO

A tarefa de avaliar os modelos de classificação é medir o grau em que a classificação sugerida usando o modelo correspondente à classificação real do caso (NOVAKOVI´C et al, 2017). Dependendo do método de observação existem diferentes medidas para avaliação do desempenho do modelo. A seleção das medidas mais adequadas deve ser feita dependendo das características do problema e formas de sua implementação.

Dentre as métricas que podem ser avaliadas, algumas se destacam, considerando que não é possível estabelecer a priori as melhores técnicas, já que depende do problema a ser estudado (FACELI, 2021, p. 148). As métricas a seguir, por exemplo, são calculadas a partir de uma matriz chamada “matriz de confusão”, que, segundo Düntsch (2019), possui o formato apresentado no Quadro 1.

Quadro 1: Matriz de confusão:

		Valor Verdadeiro	
		P	***N***
Valor Predito	***P***	***Verdadeiro Positivo***	***Falso Positivo***
	N	***Falso Negativo***	***Verdadeiro Negativo***

Fonte: Adaptado de Düntsch (2019)

A partir da matriz de confusão, segundo Castro (2016), as métricas acurácia, precisão, revocação e F1-Score podem ser obtidas e são facilmente entendidas matematicamente.

A acurácia (Eq. 7) mede o número de classificações corretas dividida pelo número total de classificações.

$$Acurácia = \frac{VP+VN}{VP+FP+VN+FN} \quad \text{(Eq. 7)}$$

A precisão (Eq. 8) mede a exatidão do algoritmo, ou seja, a probabilidade de um item recuperado ser relevante.

$$Precisão = \frac{VP}{VP+FP} \quad \text{(Eq. 8)}$$

A revocação (Eq. 9) mede a completude do algoritmo, ou seja, a probabilidade da recuperação de um item ser relevante.

$$Revocação = \frac{VP}{VP+FN} \quad \text{(Eq. 9)}$$

F1-Score (Eq. 10) é usada para avaliar o desempenho da classificação em um intervalo [0,1].

$$F1Score = \frac{2.Precisão.Revocação}{Precisão+Revocação} \quad \text{(Eq. 10)}$$

Problemas de classificação ainda remetem às formulações matemáticas mais complexas, trazendo mais oportunidades de criação de objetos de aprendizagem de auxílio didático, como por exemplo, uma métrica chamada *Log Loss*, sendo essa uma métrica de classificação baseada em probabilidade (BARRON, 2019).

2.5 MACHINE LEARNING PLAYGROUNDS

Atualmente é comum encontrar simuladores de algoritmos de *Machine Learning* disponíveis na internet e estes vem sendo chamados de *Machine Learnig Playgrounds*. Estes ambientes têm o propósito de simular comportamentos de algoritmos de acordo com

dados fictícios criados pelo próprio usuário no momento da sua execução. Alguns exemplos interessantes são: *Encora Time Series Playground* para simulação de séries temporais, *Tensor Flow Playground*, para simulação de redes neurais e outros exemplos mais simples como o *Machine Learning Playground* (ml-playground), que simula os algoritmos mais comuns que conhecemos, como KNN, SVM, árvore de decisão, dentre outros.

Esses aplicativos de suporte mais abstratos têm se tornado muito importantes como suporte à aprendizagem, como também para testes mais agnósticos de modelos de ML. Um exemplo é CFU *playground*, uma estrutura de código aberto para aceleração de aprendizado de máquina para desenvolvimento de aceleração de hardware para processamento de redes neurais. Esta estrutura *full-stack* dá aos usuários acesso para explorar arquiteturas experimentais e sob medida que são personalizadas e otimizadas para ML incorporado (PRAKASH et al., 2023).

De acordo com as definições de nosso estudo estes simuladores poderiam se encaixar no conceito de Objetos de Aprendizagem digitais.

3. METODOLOGIA

Esta pesquisa teve natureza aplicada e envolveu o desenvolvimento de artefatos na forma de objetos de aprendizagem, ou recursos educacionais digitais, para o estudo do arcabouço matemático dos algoritmos de ML KNN, Naive Bayes e SVM. O seguinte conjunto de atividades foi realizado no período de vigência deste projeto:

- Pesquisa bibliográfica inicial: refere-se à pesquisa de trabalhos em bases de artigos, revistas, dissertações e teses disponíveis na internet, de conteúdo privado e público, utilizando como palavras-chave os principais conceitos considerados foco e essenciais, sendo eles: objetos de aprendizagem matemática, tipos de objetos de aprendizagem digitais, *Objetos de aprendizaje*, *The current state of open educational resources*, *supervised machine learning techniques*, *roc curve, evaluation of classification models*, *matthews correlation coeficient*, *logarithmic loss machine learning*, *cosine similarity measure text classification* e acurácia precisão e recall e f1.
- Leitura de bibliografia pesquisada: envolve a leitura da bibliografia selecionada nas pesquisas realizadas na etapa anterior;
- Elaboração de mapa conceitual identificando os principais conceitos relacionados ao foco desta pesquisa;
- Seleção das técnicas de aprendizagem de máquina e suas métricas de avaliação para desenvolvimento;

- Desenvolvimento de algoritmos de aprendizagem de máquina para avaliação de métricas de erro e desempenho;
- Realização das formulações matemáticas das métricas estudadas;
- Estudo e seleção do conteúdo e formatos dos recursos digitais que compõem o objeto de aprendizagem;
- Construção e disponibilização dos objetos de aprendizagem digitais.

Primeiramente foi adotada a Pesquisa Bibliográfica a respeito de objetos de aprendizagem digitais e algoritmos de aprendizagem de máquinas e suas métricas.

Na etapa de desenvolvimento do projeto no que tange à parte matemática, foram criados algoritmos de aprendizagem de máquina utilizando linguagens de programação computacional adequada (Ex.: Python, R etc.) com o objetivo de estudar suas métricas de erro e desempenho.

A partir do estudo dos algoritmos e suas métricas, foi feita a formulação matemática das métricas estudadas visando uma didática mais inclusiva no estudo dos algoritmos de aprendizagem de máquina.

Tendo uma visão mais tangível da complexidade dos problemas matemáticos a serem compartilhados, foram definidos os recursos digitais mais adequados para os objetos de aprendizagem para uma abordagem didática dos algoritmos estudados.

Por fim, foram criados objetos de aprendizagem digitais para disponibilização dos resultados, visando o acesso de um público que busca por conhecimento matemático dos algoritmos de ML selecionados.

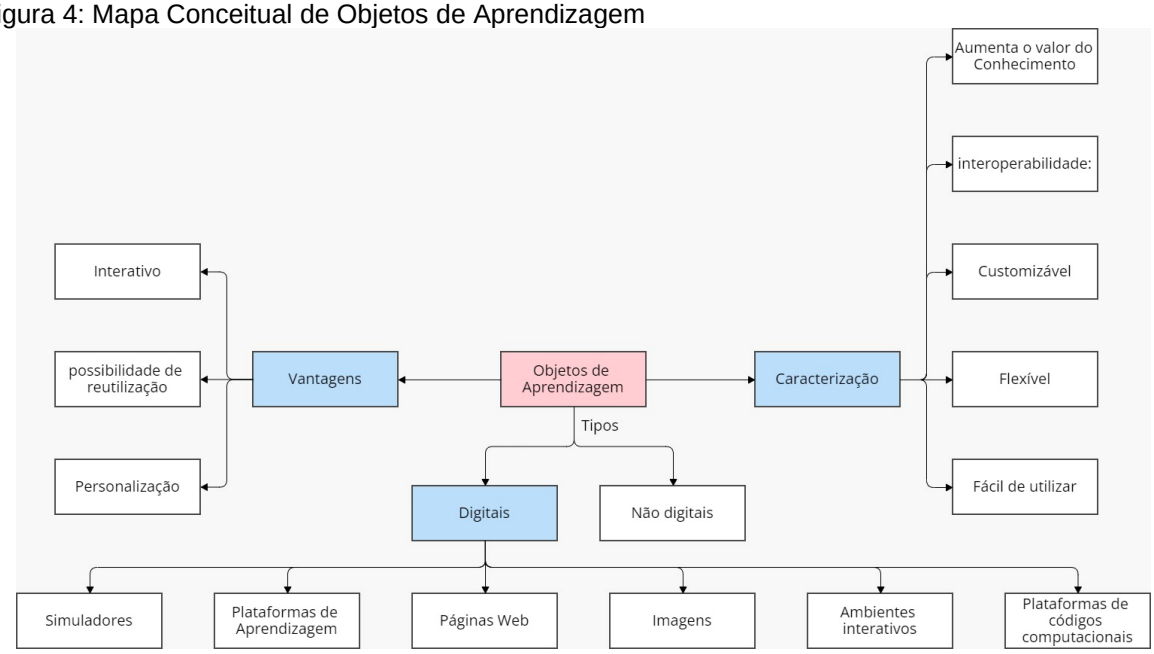
4. RESULTADOS

Nesta seção, é apresentada uma síntese dos resultados obtidos. Na Seção 4.1, são mostrados dois mapas conceituais elaborados na fase de estudos desta pesquisa. O processo de construção e a estrutura dos OAs elaborados são destacados na Seção 4.2. Na Seção 4.3, o OA relativo ao KNN é descrito. O OA sobre o Naive Bayes é apresentado na Seção 4.4. Na Seção 4.5, é destacado o OA do SVM. Por fim, na Seção 4.6, são realizadas algumas discussões.

4.1 MAPA CONCEITUAL

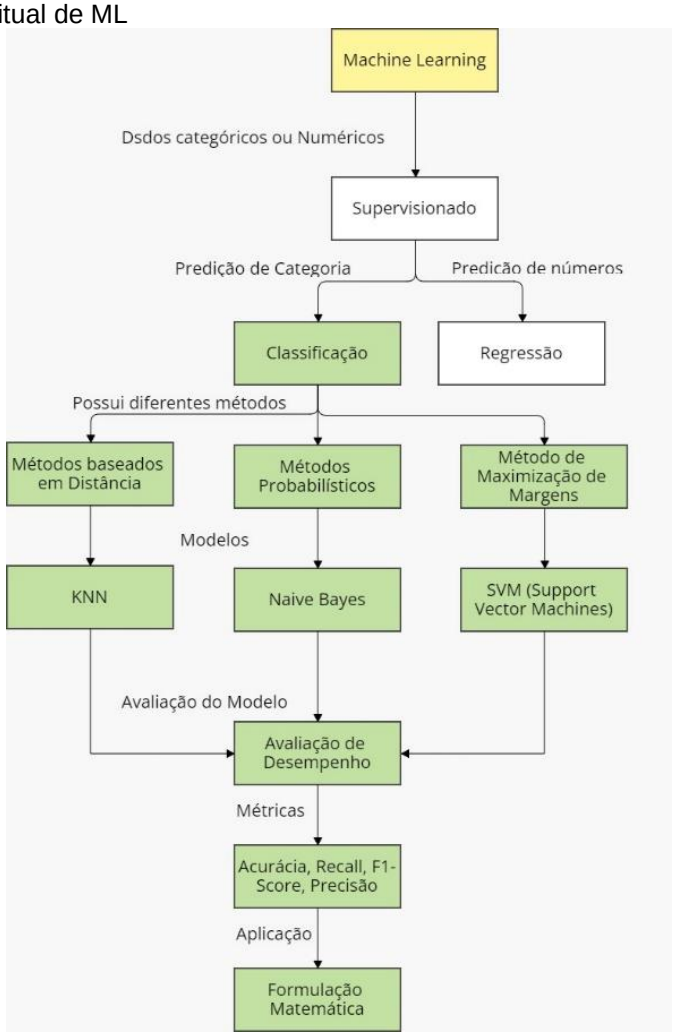
Na fase de estudos desta pesquisa foram elaborados dois mapas conceituais a respeito de Objetos de Aprendizagem e *Machine Learning*, ilustrados na Figuras 4 e 5. Esses mapas foram fundamentais para embasar os conhecimentos necessários sobre a temática deste trabalho.

Figura 4: Mapa Conceitual de Objetos de Aprendizagem



Fonte: Elaborado pelo autor (2023)

Figura 5: Mapa Conceitual de ML



Fonte: Elaborado pelo autor (2023)

4.2 OBJETOS DE APRENDIZAGEM: ESTRUTURA E PROCESSO DE CONSTRUÇÃO

Para desenvolvimento dos recursos educacionais deste projeto, foram utilizados o Jupyter notebook que é uma plataforma utilizada para desenvolvimento de soluções em Python e outras linguagens de programação que permite visualizar códigos e resultados de execução, além de incorporar códigos HTML, que aceita a adição de vídeos e imagens para explicação dos algoritmos. A versão do Python utilizada foi a 3.8.0, sendo executado no gerenciador de ambientes Anaconda. Como base de dados para teste, foi utilizada a base Iris (Scikit-learn, 2023).

Foram criados recursos educacionais digitais para cada modelo estudado: KNN, Naive Bayes e SVM. A estrutura de seções de código e formatos de recursos utilizados seguiu uma mesma sequência padrão para todos os objetos de aprendizagem, conforme descrito a seguir:

- Introdução: Texto contendo uma explanação do modelo a ser estudado;
- Vídeo introdutório: Apresentação do objeto de aprendizagem e resumo do que irá ser estudado;
- Carga da Base dados;
- Análise Exploratória de Dados: Utilização de gráficos como Box plot, coordenadas paralelas, pair plot, violino e similares;
- Vídeo explicativo sobre a leitura de cada tipo de gráfico;
- Preparação dos dados para execução do modelo de ML;
- Vídeo elucidativo a respeito do arcabouço matemático do modelo a ser criado:
    - Modelo KNN: Abordagem de geometria analítica, teorema de Pitágoras, distância Euclidiana e uso de um simulador de KNN através da plataforma ML Playground.
    - Modelo Naive Bayes: Abordagem da fórmula geral do modelo Naive Bayes com e sem uso de logaritmo e ilustrações.
    - Modelo SVM (*Support Vector Machine*): Ilustração do conceito de margens, produto escalar, direção de vetores e simulação do modelo através da *plataforma Interactive demo of Support Vector Machines (SVM)*.
- Construção do modelo de ML;
- Avaliação das métricas de desempenho;
- Vídeo esclarecedor dos resultados do modelo e métricas de desempenho;

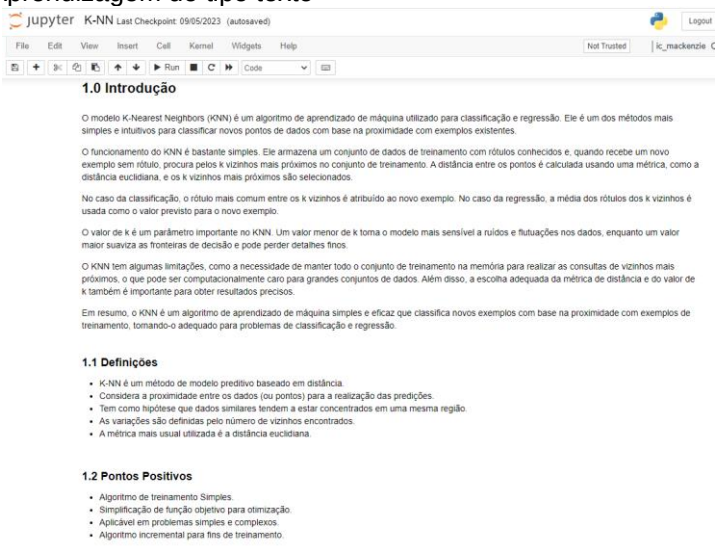
- Material complementar: Vídeos incorporados do Youtube com abordagens matemáticas dos modelos estudados nos objetos de Aprendizagem. Um dos canais mais explorados foi o StatQuest.
- Referências.

Os objetos de aprendizagem desenvolvidos encontram-se disponíveis no endereço https://github.com/rrodriguesrr/learning_objects.git. No interior da página desse link, atentar ao arquivo LEIAME.txt que contém as instruções mínimas de uso.

4.3 K-NN (K-NEAREST NEIGHBORS):

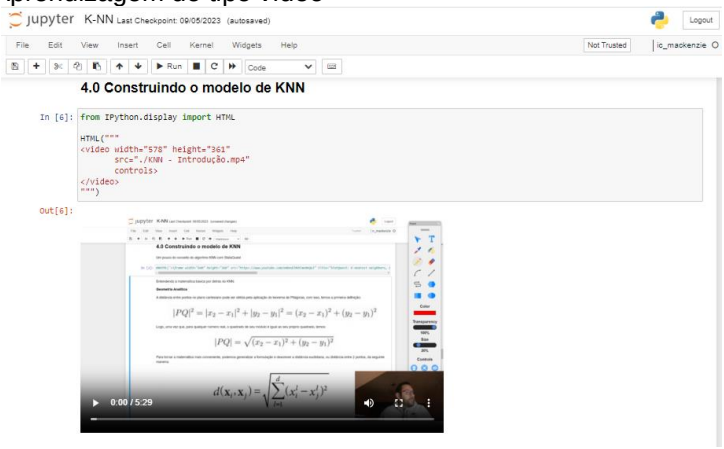
Seguindo o modelo padrão de seções do Jupyter notebook, foi criado o modelo de KNN, tendo como base de dados a base Iris (Scikit-learn, 2023). Foram feitas análises exploratórias de dados, com vídeos e textos introdutórios. O modelo utilizado foi o KNeighborsClassifier do pacote *sklearn*. Para avaliação do modelo foi utilizada a métrica de acuracidade. Ilustrações do objeto obtido são destacadas nas Figuras 6 e 7.

Figura 6: Objeto de Aprendizagem do tipo texto



Fonte: Elaborado pelo autor (2023)

Figura 7: Objeto de Aprendizagem do tipo vídeo



Fonte: Elaborado pelo autor (2023)

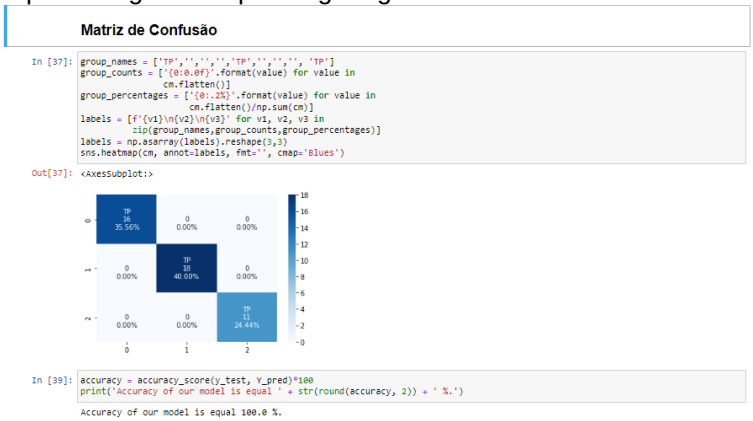
4.4 NAIVE BAYES

O recurso educacional digital criado para o *Naive Bayes* seguiu o modelo padrão de seções do Jupyter notebook, tendo a Iris (Scikit-learn, 2023) como base de dados. Também foram feitas análises exploratórias de dados, com vídeos e textos introdutórios. O modelo utilizado foi o GaussianNB do pacote *sklearn*. Para avaliação do modelo foi utilizada a métrica de acuracidade. Algumas capturas dos resultados obtidos para o objeto de aprendizagem podem ser visualizadas nas Figuras 8 e 9.

Figura 8: Objeto de Aprendizagem do vídeo incorporado do Youtube

Fonte: Elaborado pelo autor (2023)

Figura 9: Objeto de Aprendizagem do tipo imagem/gráfico.

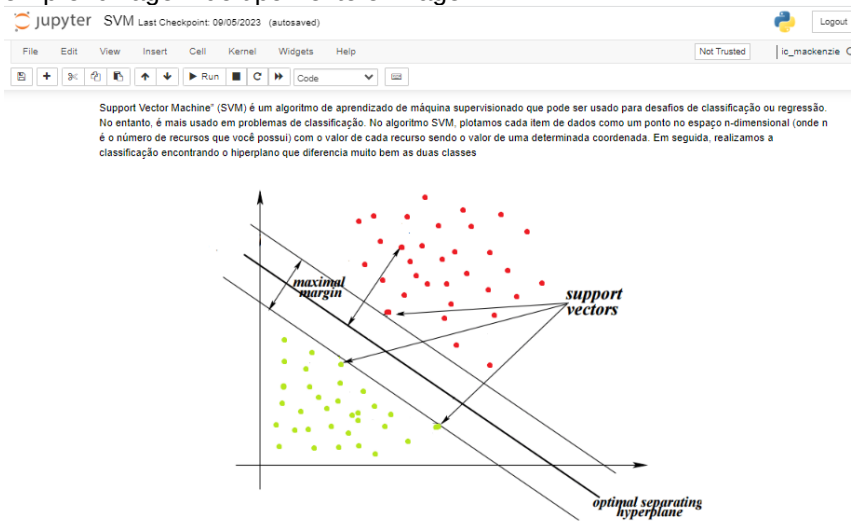


Fonte: Elaborado pelo autor (2023)

4.5 SVM (*Support Vector Machine*)

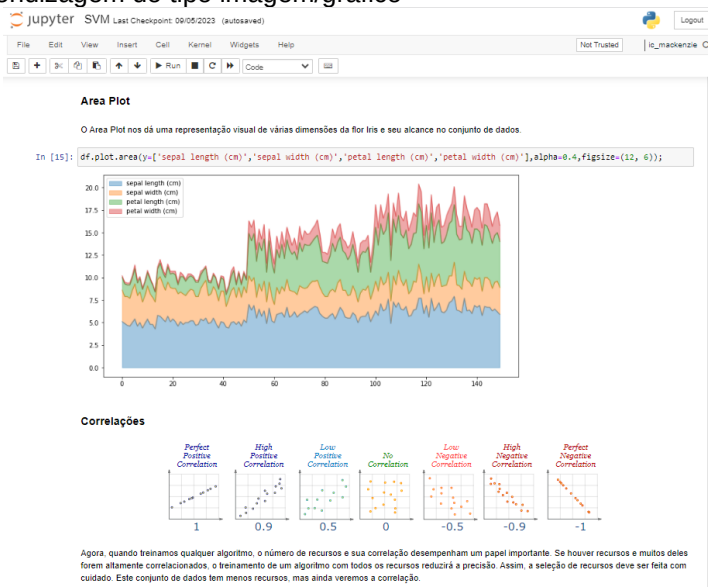
A construção do objeto de aprendizagem correspondente ao SVM seguiu a sequência padrão descrita anteriormente e o modelo padrão de seções do Jupyter notebook, tendo a Iris (Scikit-learn, 2023) como base de dados. Dessa forma, nesse recurso foram feitas análises exploratórias de dados, com vídeos e textos introdutórios. O modelo utilizado foi o LinearSVC do pacote *sklearn*. A métrica utilizada para a avaliação do modelo foi a acurácia. Exemplos de interfaces do objeto podem ser observadas nas Figuras 10 e 11.

Figura 10: Objeto de Aprendizagem do tipo Texto e Imagem



Fonte: Elaborado pelo autor (2023)

Figura 11: Objeto de Aprendizagem do tipo imagem/gráfico



Fonte: Elaborado pelo autor (2023)

4.6 DISCUSSÃO

O uso de objetos de aprendizagem digitais contendo recursos com formatos variados na exploração dos algoritmos KNN, Naive Bayes e SVM e do arcabouço matemático

relacionado, forneceu mecanismos para abordar os conceitos de maneira visual e textual incorporadas em uma página do Jupyter Notebook com execução dinâmica dos modelos.

Fazer o uso de vídeos, imagens e explicações textuais contextualizadas tem potencial para trazer diferenciais, geralmente não presentes em livros ou literaturas e tem a capacidade de auxiliar na compreensão do conteúdo estudado, especialmente na explicação da formulação matemática presente nesses algoritmos.

Tendo por base que os objetos de aprendizagem foram criados como uma página do Jupyter Notebook, Python e seus pacotes, esses OAs fornecem um *Storytelling* da sua execução.

5. CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho foram desenvolvidos três objetos de aprendizagem considerando os algoritmos de ML supervisionados KNN, Naive Bayes e SVM, com ênfase ao arcabouço matemático que norteia seu funcionamento e avaliação.

Na construção deles foi adotada uma sequência padrão do processo de geração de modelos fazendo uso da base de dados Iris que, em síntese, envolveu: uma introdução ao algoritmo, carga da base de dados, análise exploratória dos dados com gráficos e vídeo explicativo, preparação dos dados e vídeo explicativo do arcabouço matemático, construção do modelo e avaliação do modelo e vídeo explicativo da métrica utilizada.

Nos OAs também foram utilizados formatos variados de mídias, compreendendo uma combinação de texto, imagem e vídeo, que foram incorporados em uma página do Jupyter Notebook em Python, com o pacote sklearn para geração dos modelos. Dessa forma, o estudante pode seguir a sequência sugerida no OA e interagir com ele. A interação ocorre pela execução passo a passo dos códigos contidos na página em Python e os resultados obtidos.

Os recursos educacionais digitais criados são uma alternativa didática para o estudo de algoritmos de ML, uma vez que sua concepção não teve foco apenas em computação, mas também em metodologias de ensino de matemática. Dessa forma, tem potencial para auxiliar pessoas interessadas no estudo e aprendizado desse tema e profissionais ingressantes na área de ciência de dados com interesse em adquirir subsídios para realizar uma argumentação mais sólida sobre os resultados obtidos em seus modelos e a base matemática utilizada. No entanto, uma avaliação e validação desses OAs com usuários é necessária para comprovar sua real eficácia.

Este projeto foi apresentado no WTT 2023 (Workshop de Tendências Tecnológicas) para alunos e professores da Faculdade de Computação e Informática (FCI) da

Universidade Presbiteriana Mackenzie (UPM). Por meio dele, foi possível identificar melhorias que foram adicionadas aos recursos produzidos.

A partir dos resultados obtidos o próximo passo seria selecionar pessoas para realizar uma avaliação do seu conteúdo como potencial facilitador de estudo e aprendizado. Ademais, produzir novos OAs com outros algoritmos de ML e outras métricas de avaliação de desempenho.

AGRADECIMENTOS

Agradeço à Universidade Presbiteriana Mackenzie pela Oportunidade de realizar minha primeira pesquisa científica.

Ao professor Dr. Ivan Carlos Alcântara de Oliveira que me orientou nesta jornada não apenas com seu conhecimento, mas também com sua compreensão devido ao meu momento profissional, que tornou meu tempo muito restrito para realização desta pesquisa.

À professora Daniela Cunha da Faculdade de Computação e Informática que me orientou no início do processo e me direcionou ao professor orientador.

À professora Eliza Hidemi Sadaike que me direcionou em minha comunicação com a Faculdade de Computação e Informática para que fosse possível concretizar esta realização.

REFERÊNCIAS

BARRON, Jonathan T. **A general and adaptive robust loss function**. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019. p. 4331-4339.

CASTRO, Eduardo Peñalosa; DURÁN, Patricia Landa. **Objetos de aprendizaje: una propuesta de conceptualización, taxonomía y metodología**. Revista electrónica de psicologia Aplicada, v. 11, n. 3, 2008.

CASTRO, Leandro Nunes de; FERRARI, Daniel Gomes**. Introdução à mineração de dados: conceitos básicos, algoritmos e aplicações**. São Paulo: Saraiva, v. 5, 2016.

DERIVATIVE CALCULATOR. **Calculate Derivatives online**, c2023. Página inicial. Disponível em: < https://www.derivative-calculator.net/ >. Acesso em: 01 de set. de 2023.

DÜNTSCH, Ivo; GEDIGA, Günther. **Confusion matrices and rough set data analysis**. In: Journal of Physics: Conference Series. IOP Publishing, 2019. p. 012055.

ENCORA. **Time-Series Playground**, c2023. Página inicial. Disponível em: < https://encoradigital.github.io/SAB-timeseries-playground/ >. Acesso em: 01 de set. de 2023.

FACELI, Katti; LORENA, Ana C.; GAMA, João; AL, et**. Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina.** 2 ed. Rio de Janeiro: Grupo GEN, 2021.

GEOGEBRA. **GeoGebra for Teaching and Learning Math**, c2023. Página inicial. Disponível em: < https://www.geogebra.org/>. Acesso em: 01 de set. de 2023.

Greitemann Jonas. **Interactive demo of Support Vector Machines (SVM),** c2023. Página inicial. Disponível em: < https://greitemann.dev/svm-demo/>. Acesso em: 01 de set. de 2023.

INTEGRAL CALCULATOR. **Calculate Integrals online**, c2023. Página inicial. Disponível em: < https://www.integral-calculator.com/ >. Acesso em: 01 de set. de 2023.

JADHAV, Sayali D.; CHANNE, H. P. **Comparative study of K-NN, naive Bayes and decision tree classification techniques**. International Journal of Science and Research (IJSR), v. 5, n. 1, p. 1842-1845, 2016.

MAHESH, Batta. **Machine learning algorithms-a review.** International Journal of Science and Research (IJSR).[Internet], v. 9, p. 381-386, 2020.

MARUSEAC, Mihai. **Tensorflow Playground**, c2022. Página inicial. Disponível em: < https://playground.tensorflow.org/ >. Acesso em: 01 de set. de 2023.

MASOLA, Wilson; ALLEVATO, Norma. **Dificuldades de aprendizagem matemática: algumas reflexões**. Educação Matemática Debate, v. 3, n. 7, p. 52-67, 2019.

ML Playground. **Machine Learning Playground**. Página inicial. Disponível em: < https://ml-playground.com/>. Acesso em: 01 de set. de 2023.

NOVAKOVIĆ, Jasmina Dj et al. Evaluation of classification models in machine learning. **Theory and Applications of Mathematics & Computer Science**, v. 7, n. 1, p. 39–46-39–46, 2017.

OLIVEIRA, Ivan Carlos Alcântara de. **Recursos Educacionais Abertos e Objetos de Aprendizagem. 2018.** [Arquivo PDF de apoio no curso de pós-graduação da Universidade Cruzeiro do Sul].

OLSON, Randal S. et al. **PMLB: a large benchmark suite for machine learning evaluation and comparison**. BioData mining, v. 10, n. 1, p. 1-13, 2017.

PRAKASH, **Shvetank et al. Cfu playground: Full-stack open-source framework for tiny machine learning (tinyml) acceleration on fpgas**. In: **2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)**. IEEE, 2023. p. 157-167.

Scikit-learn. **The iris Dataset**, c2023. Disponível em: < https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html/>. Acesso em: 01 de set. de 2023.

SOUZA, José Ocimar Barros de. **Modelagem matemática como processo para o desenvolvimento do pensamento analítico e reflexivo**. 2020. Tese de Doutorado.

SANTÓRUM, M., CARRIÓN-TORO, M., GUACAPIÑA, M-B, ACOSTA-VARGAS, P. and JADÁN-GUERRERO, J. "**A Case Study: Developing reusable Learning Objects**," *2021 Second International Conference on Information Systems and Software Technologies **(ICI2ST)***, 2021, pp. 79-86, doi: 10.1109/ICI2ST51859.2021.00019.

STATQUEST. **An epic journey through statistics and machine learning**, c2023. Página inicial. Disponível em: < https://statquest.org/>. Acesso em: 01 de set. de 2023.

SY. **Symbolab**, c2023. Página inicial. Disponível em: < https://pt.symbolab.com/ >. Acesso em: 01 de set. de 2023.

TRIANA-MUÑOZ, Mónica María; CEBALLOS-LONDOÑO, Juan Fernando; VILLA-OCHOA, Jhony Alexander. **Una dimensión didáctica y conceptual de un instrumento para la Valoración de Objetos Virtuales de Aprendizaje**. El caso de las fracciones. Entramado, v. 12, n. 2, p. 166-186, 2016.

WILEY, David. **The current state of open educational resources**. 2006.

Winston, Patrick H., **Lecture 16: Learning: Support Vector Machines: Artificial Intelligence MIT OCW**. Fall 2010. Notas de Aula.

Zhang, Y., & Yang, Y. **A Comparative Study of Machine Learning Algorithms for Predicting the Risk of Heart Disease**. Journal of Healthcare Engineering, 2021.

Contatos:

- Rubens Dias Rodrigues Junior: rubensdias.rodrigues@mackenzista.com.br
- Prof. Dr. Ivan Carlos Alcântara de Oliveira: ivan.oliveira@mackenzie.br