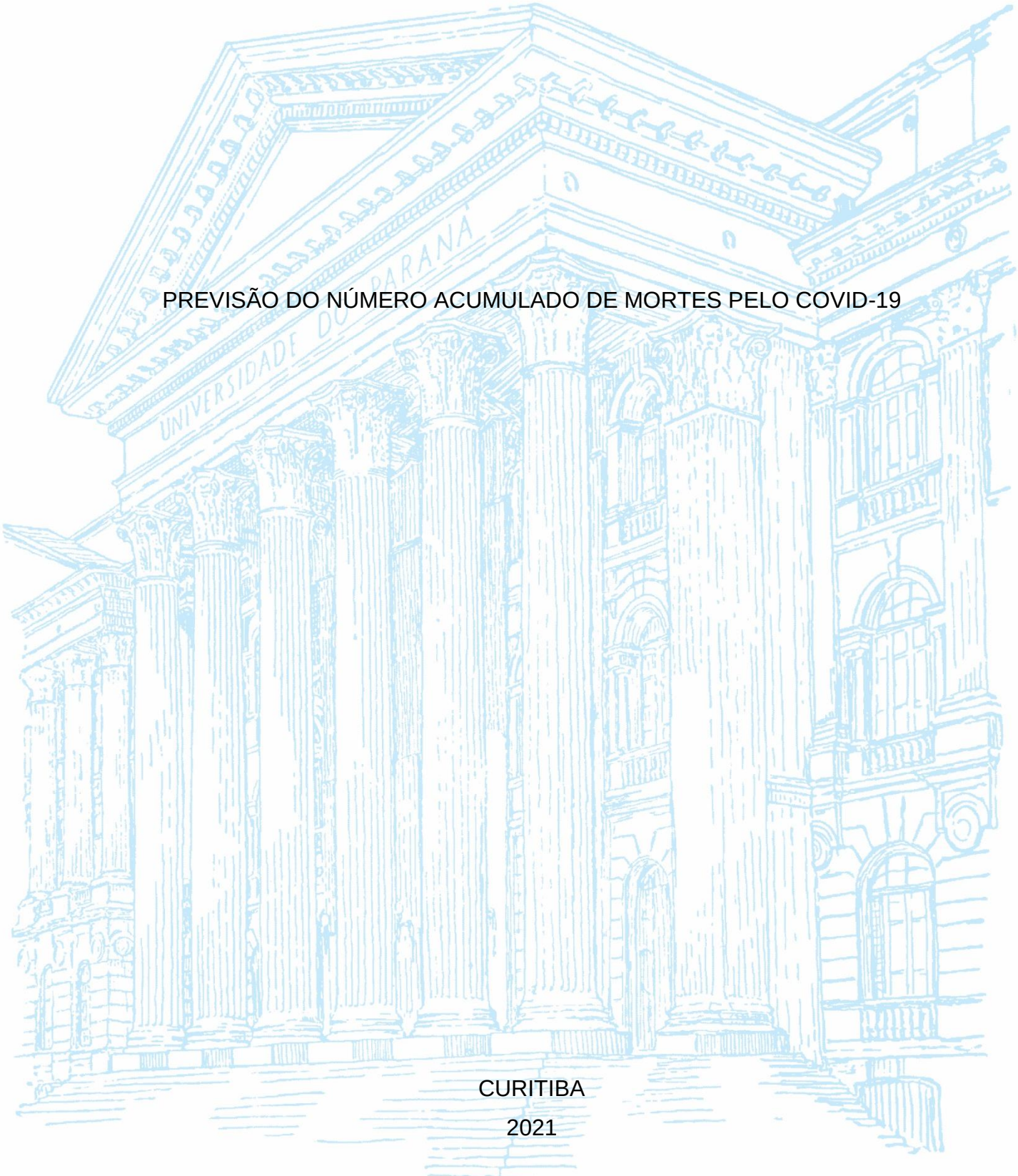
UNIVERSIDADE FEDERAL DO PARANÁ

FABRICIO YUDI SAKAI



PREVISÃO DO NÚMERO ACUMULADO DE MORTES PELO COVID-19

CURITIBA

2021

FABRICIO YUDI SAKAI

PREVISÃO DO NÚMERO ACUMULADO DE MORTES PELO COVID-19

TCC apresentada ao curso de Engenharia de Produção, Setor de Tecnologia, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Bacharel em Engenharia de Produção.

Orientador(a): Prof(a). Dr(a). Mariana Kleina

CURITIBA

2021

RESUMO

O presente estudo tem como objetivo adequar modelos de estatística e *machine learning* com intuito de prever a quantidade total acumulada de mortes pelo COVID-19. Para isso, foi utilizado uma metodologia comum em Ciências de Dados que conta com quatro etapas principais: a definição e formulação do problema, coleta de dados, modelagem e implementação. Os dados utilizados no trabalho foram retirados do site *Our World In Data*, que são os mesmos utilizados em universidades de Harvard, MIT e Oxford. E a principal ferramenta empregada foi a linguagem de programação Python, que conta com bibliotecas extensivas de análises de dados. Com isso, três modelos foram treinados e adequados aos dados: regressão linear, árvore de decisão e rede neural artificial. Ao fim desta adequação, os resultados de previsão foram comparados entre si usando métricas como MAE, MSE e RMSE; e, dentre eles, a árvore de decisão foi o modelo que teve melhores resultados de previsão.

Palavras-chaves: COVID-19. Regressão Linear Múltipla. Árvore de Decisão. Rede Neural Artificial.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Doenças infecciosas estão emergindo em uma velocidade sem precedente (BALKHAIR, 2020), e – dentre as doenças causadas por agentes patogênicos – os coronavírus ganharam atenção, já que eles foram responsáveis por três epidemias neste século (HUI et al., 2020). Estes vírus se referem a um grupo que infectam animais (KUMAR, 2020) e apresentam peplômeros – estruturas encontradas na superfície de certos vírus – que remetem à imagem de uma coroa (LIMA, 2020), por isso eles ganharam o nome de coronavírus (*corona,* em latim, que significa coroa). O primeiro caso registrado de coronavírus em humanos aconteceu em 1965, quando Tyrell e Bynoe isolaram uma das variedades do vírus do trato respiratório de um paciente que se queixava de gripe comum (JAHANGIR, 2020). Desde então, sete variedades de coronavírus em humanos foram identificadas, porém não é conhecido desde quando este grupo de vírus efetivamente existe, já que eles geralmente eram associados com doenças leves (YANG et al., 2020).

O coronavírus descoberto em dezembro de 2019 em Wuhan, China Central, foi denominado SARS-CoV-2 e é causador da doença COVID-19. Este vírus foi imediatamente identificado pela sua rápida habilidade de se espalhar e ter uma maior propensão de causar letalidades em pessoas mais velhas (ALANAGREH et al., 2020), isso fez com que – no dia 30 de janeiro de 2020 – a OMS declarasse a epidemia como uma emergência internacional (LANA et al., 2020). No Brasil, o primeiro caso foi oficialmente confirmado em 25 de fevereiro de 2020, e – três meses depois – existia mais de meio milhão de casos, espalhados ao longo de 75% dos municípios do país, com quase 30 mil fatalidades (SOUZA et al., 2020). E, nos dados até 7 de fevereiro de 2021, foram estimados globalmente mais de 100 milhões de casos de COVID-19 e mais de 2,3 milhões de mortes declaradas (WORLDOMETER, 2021).

O trabalho atual, que é centrado neste contexto, tem como objetivo encontrar o modelo que tenha maior precisão para prever o número de mortes causadas pelo COVID-19 por país. Para isso, é utilizado como metodologia base o CRISP-DM (seção 3.1), que divide o escopo de trabalho nas seguintes etapas: entendimento do fenômeno, entendimento e preparação dos dados, modelagem, avaliação e implementação.

1.1 OBJETIVOS

A seguir são detalhados os objetivos gerais e específicos deste trabalho de conclusão de curso.

1.1.1 Objetivo geral

O objetivo geral deste trabalho é identificar, dentre os três métodos (regressão linear múltipla, árvore de decisão e rede neural artificial), o mais adequado para prever, com precisão, o número acumulado de mortes por COVID-19 por país num período de curto-prazo. Neste estudo, foi adotado um período de tempo de uma semana.

1.1.2 Objetivos específicos

Com base no objetivo geral, os objetivos específicos são:

a) Estudar alguns métodos de *machine learning* e o método de regressão linear múltipla;
b) Avaliar quais variáveis tem maior impacto na previsão de mortes causadas pela pandemia do COVID-19;
c) Aplicar os métodos estudados à uma base real;
d) Comparar resultados dos diferentes modelos utilizando estas variáveis, e avaliar qual apresenta maior aderência ao fenômeno estudado.

1.2 JUSTIFICATIVA

A motivação deste trabalho se deu por ser um assunto relativamente novo e emergente (até a data de início deste estudo, faz menos de um ano que o primeiro caso de COVID-19 foi confirmado no Brasil). Além disso, não existem modelos estatísticos na literatura com objetivo de prever, entender e contextualizar o número total de mortes por país causadas por esta doença, uma vez que os artigos referentes a modelos estatísticos envolvendo a pandemia do COVID-19 geralmente ficam dentro

do escopo de efeitos de estratégias de intervenção (BOŠKOSKI et al., 2020; DING; GAO, 2020; EIKENBERRY et al., 2020; OVERTON et al., 2020).

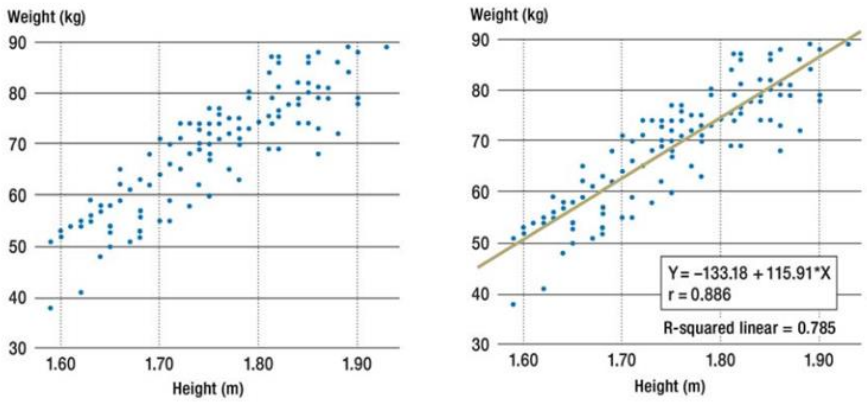
2 REVISÃO DE LITERATURA

As teorias dos modelos de Regressão Linear Múltipla, Árvore de Decisão e Rede Neural Artificial são mostradas nas seções 2.1, 2.2 e 2.3, respectivamente; e as métricas de avaliação dos modelos são apresentados na seção 2.4.

2.1 REGRESSÃO LINEAR MÚLTIPLA

A regressão linear múltipla, segundo James et al. (2013), parte do pressuposto de que existe uma relação aproximadamente linear entre as variáveis independentes e a variável dependente. E, assim, ela procura aproximar esta relação à uma reta ou um plano. A FIGURA 1a mostra um exemplo de uma relação linear entre peso e altura de uma população de pessoas, onde esta relação é aproximada pela equação linear de uma reta na FIGURA 1b.

FIGURA 1 – EXEMPLO DE UMA REGRESSÃO LINEAR



a b

Fonte: Schneider et al. (2010)

Apesar de simples, modelos lineares são bastante utilizados na prática por possibilitarem descrições estatísticas simplificadas de fenômenos complexos

(MONTGOMERY; PECK; VINING, 2013). Por isso, as aplicações de regressões lineares estão em praticamente todos os campos de estudo, como engenharias, ciências físicas, economia, ciências sociais, etc (NETER et al., 1996).

2.1.1 Formulação matemática

Uma regressão linear múltipla tem como resultado a equação de uma reta ou plano multi-dimensional (MONTGOMERY; PECK; VINING, 2013), por isto sua expressão matemática generalizada para uma regressão com k variáveis independentes, é dada pela Equação 1, onde y é a variável dependente (ou variável resposta), x são as variáveis independentes (ou variáveis preditoras), e β são os coeficientes da equação.

$$y(x) \approx \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k \quad (1)$$

É comum equações de regressão linear múltipla serem convencionadas a partir de notação vetorial como mostrado na Equação 2, onde $\beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_k \end{bmatrix}$ e $x = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_k \end{bmatrix}$.

$$y(x) \approx \beta^T x, \quad (2)$$

2.1.2 Método dos Mínimos Quadrados

O Método dos Mínimos Quadrados (MMQ) é o método mais utilizado para estimar os coeficientes de uma regressão linear.

2.1.3 Premissas subjacentes

Existem cinco premissas básicas, que devem ser atendidas para se ter garantia que um modelo de regressão linear múltipla tem resultados confiáveis utilizando o Método dos Mínimos Quadrados, que são:

a) Premissa de linearidade, que expressa que as variáveis preditoras e resposta devem ter uma relação linear entre si. Pois caso o modelo linear

seja treinado em um conjunto de dados não-lineares, sua extrapolação vai ter resultados imprecisos e errôneos (ALBERT, 2016). Esta premissa geralmente é verificada por meio da visualização das variáveis dependentes e independente em um gráfico de dispersão (BANSAL, 2014).

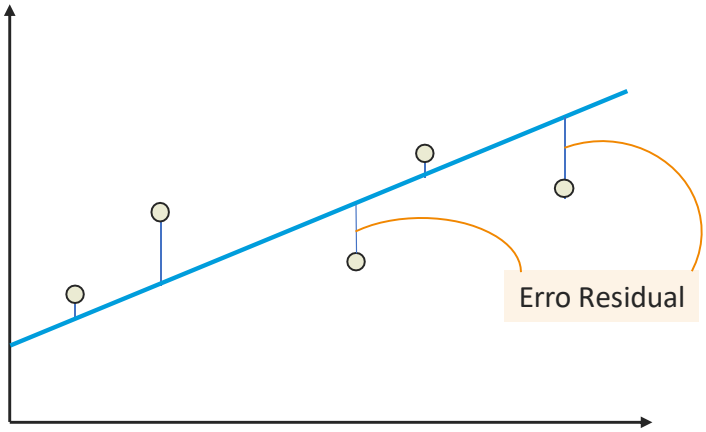
b) Premissa de independência ou de não auto-correlação, que enuncia que os residuais devem ser independentes entre si. Em outras palavras, quando o valor de y(x) é independente de y(x+1). Esta premissa é averiguada com o teste de Durbin-Watson, que deve produzir valores entre 0 e 4 (STATICSSOLUTIONS, 2010).
c) Premissa de normalidade de erros, que diz que os erros devem ter uma distribuição normal (BANSAL, 2014). Este pressuposto é validado com a visualização de um histograma com os erros residuais, que deve apresentar uma distribuição normal.
d) A premissa de multicolinearidade, que demonstra que as variáveis preditoras não devem ter uma correlação alta entre si e isso pode ser testado por meio de ferramentas ou métricas, como matriz de correlação, nível de tolerância (no contexto de colinearidade) e fator de inflação de variância (MONTGOMERY; PECK; VINING, 2012).
e) Premissa de homocedasticidade, que declara que os erros devem ser homocedásticos (em outras palavras, eles devem seguir uma distribuição regular). Em um cenário oposto, caso os erros sejam heterocedásticos, será difícil confiar nos erros padrões das estimativas do MMQ, e os intervalos de confiança serão ou muito estreitos ou muito largos (ALBERT, 2016). Este pressuposto pode ser validado utilizando-se o teste de Breusch-Pagan (STATOLOGY, 2020), no qual a hipótese nula é de que homocedasticidade está presente; e com valores superiores a 0,05, considera-se que a hipótese é rejeitada e de que não existe homocedasticidade.

2.1.4 Estimação dos coeficientes

A ideia básica do Método dos Mínimos Quadrados é minimizar o quadrado dos erros residuais (JAMES et al., 2013), erros estes que são a diferença entre o valor da variável resposta em um ponto e a estimativa por uma equação de regressão neste mesmo ponto. Na FIGURA 2, é mostrado o exemplo de erros residuais em uma

regressão linear simples, no qual os erros equivalem à distância euclidiana entre os pontos e a reta.

FIGURA 2 – EXEMPLO DE ERRO RESIDUAL EM UMA REGRESSÃO LINEAR SIMPLES



Fonte: O autor (2021)

Considerando que ε é o erro residual (Equação 3), o Método dos Mínimos Quadrados otimiza os coeficientes de uma equação linear, de modo a minimizar o quadrado dos erros residuais dado por ε^2 (Equação 4).

$$\varepsilon = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j x_{ij} \right) \tag{3}$$

$$S(\beta_0, \beta_1, \cdots, \beta_k) = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j x_{ij} \right)^2 \tag{4}$$

A minimização da função dos Mínimos Quadrados é feita por meio de derivadas parciais em relação à cada uma das estimativas dos parâmetros $\hat{\beta}_j$ igualando-as a zero (encontrando o mínimo), como mostrado na Equação 5.

$$\left. \frac{\partial S}{\partial \beta_j} \right|_{\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_k} = -2 \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \sum_{j=1}^{k} \hat{\beta}_j x_{ij} \right) x_{ij} = 0, \qquad j = 1,2, \cdots, k \tag{5}$$

Assim, chegam-se nas chamadas equações normais dos mínimos quadrados (Equações 6), nas quais a solução para as equações são os coeficientes estimados $\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_k$ da regressão linear.

$$\begin{aligned} n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1} + \cdots + \hat{\beta}_k \sum_{i=1}^{n} x_{ik} &= \sum_{i=1}^{n} y_i \\ \hat{\beta}_0 \sum_{i=1}^{n} x_{i1} + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1}^2 + \cdots + \hat{\beta}_k \sum_{i=1}^{n} x_{i1}x_{ik} &= \sum_{i=1}^{n} x_{i1}y_i \\ &\vdots \\ \hat{\beta}_0 \sum_{i=1}^{n} x_{ik} + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1}x_{ik} + \cdots + \hat{\beta}_k \sum_{i=1}^{n} x_{ik}^2 &= \sum_{i=1}^{n} x_{ik}y_i \end{aligned} \tag{6}$$

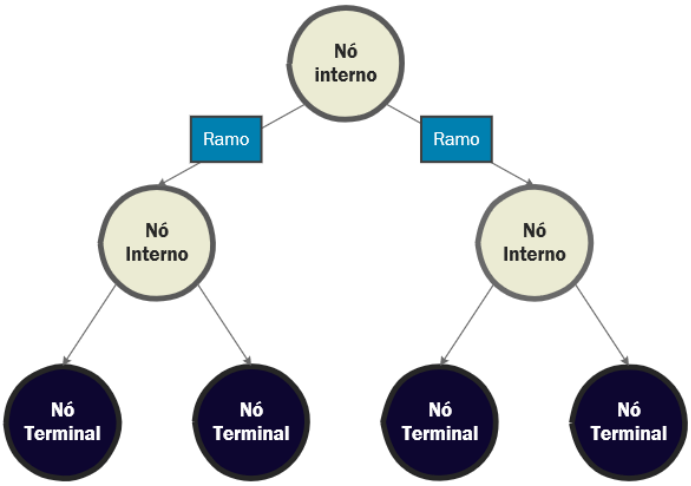
2.2 ÁRVORE DE DECISÃO

Segundo Peng et al. (2009), árvore de decisão é uma estrutura hierárquica, na qual cada nó interno representa uma escolha entre um número de alternativas, enquanto um nó terminal constitui uma decisão (a estrutura de uma árvore é detalhada na seção 2.2.1). Árvores de decisão como modelos estatísticos são amplamente utilizadas como algoritmos de machine learning pela sua efetividade e fácil interpretação (MENG et al., 2016). E, apesar de árvores de decisão serem comumente associadas com algoritmos de classificação, elas também podem ser usadas com objetivos de regressão (PATEL; SINGH, 2015), como vai ser utilizada neste trabalho.

2.2.1 Estrutura de uma árvore de decisão

Na FIGURA 3, são mostrados os principais elementos de uma árvore de decisão: nós internos (também chamados de nós não-terminais, nós de decisão ou folhas internas) são aqueles que contém algum “nó filho”; nós terminais (ou folhas terminais) são os nós finais, ou seja, aqueles que não apresentam nenhum elemento diretamente ligados a eles em um nível inferior; e ramos são estruturas que ligam nós.

FIGURA 3 – ESQUEMA DE UMA ÁRVORE DE DECISÃO



Fonte: Adaptado de Chauhan (2020)

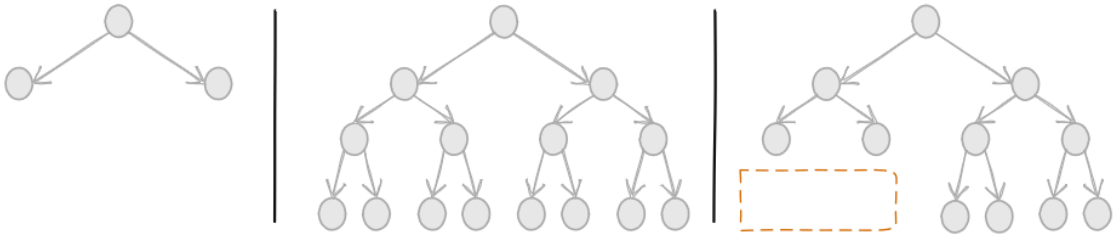
2.2.2 Algoritmo CART

Existem diversos algoritmos utilizados para construção de árvores de decisão, como o ID3 (XIAOHU; LELE; NIANFENG, 2012), C4.5 (HSSINA et al., 2014), CHAID (MILANOVIĆ; STAMENKOVIĆ, 2017) e o CART (LOH, 2011). A principal diferença entre eles está no tipo de árvore de regressão que eles constroem (regressão e/ou classificação) e no critério de escolha da melhor divisão em cada nó.

O algoritmo utilizado neste trabalho é conhecido como CART (Classification and Regression Trees ou, em português, Árvores de Classificação e Regressão), formulado por Breiman et al. (1984), e – como próprio nome diz – este algoritmo consegue modelar tanto árvores de regressão quanto de classificação (WU et al., 2008).

Segundo Timofeev (2004), o modelo CART segue duas etapas para construção do modelo: a primeira delas consiste em divisões sucessivas do espaço preditor que vai servir como base para criação da estrutura hierárquica da árvore (FIGURA 4a e FIGURA 4b); e, em seguida, é realizado um processo chamado de “poda”, na qual é feita uma compressão da árvore de decisão causando uma melhora na precisão do modelo com a diminuição de overfitting (FIGURA 4c).

FIGURA 4 – CONSTRUÇÃO DE UMA ÁRVORE DE DECISÃO CART



a b c

Fonte: O autor (2021)

2.2.3 Estratificação do espaço preditor

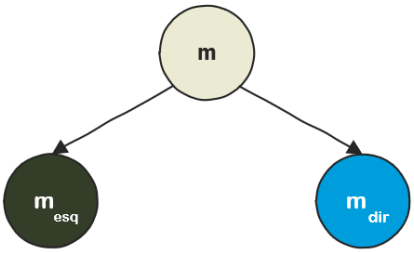
A estratificação, ou divisão, das regiões preditoras pelo algoritmo CART é o primeiro passo para construção de uma árvore de decisão. E este processo é feito de maneira binária, em outras palavras, um nó é sempre dividido em dois nós (WU et al., 2008). A forma como o algoritmo particiona e escolhe a melhor região para cada nó é detalhado a seguir. Porém, antes, considera-se:

- $x_i \in R^n \ (i = 1, .., l)$ como sendo os vetores das variáveis preditoras;
- $y \in R^l$, o vetor da variável resposta;
- Q_m, os dados de um nó m;
- N_m, a quantidade de dados do nó m;
- $\theta = (j, t_m)$, o candidato à partição que consiste na característica j e no valor t_m que é responsável por dividir Q_m em dois subconjuntos (Q_m^{esq} e Q_m^{dir}).

Neste algoritmo, cada candidato à partição θ divide a região preditora da iteração em dois subconjuntos Q_m^{esq} e Q_m^{dir} (FIGURA 5), sendo que:

- $Q_m^{esq}(\theta) = \{(x, y) | x_j <= t_m\}$
- $Q_m^{dir}(\theta) = Q_m \backslash Q_m^{esq}(\theta)$

FIGURA 5 – DIVISÃO BINÁRIA DE UM NÓ



Fonte: O autor (2021)

A qualidade de um candidato θ é caracterizado pela função G (Equação 7), e que está diretamente relacionada com a função de impureza H. Como a árvore deste trabalho é uma de regressão, a função de impureza utilizada é o Erro Quadrático Médio (Equação 8) – também conhecido pela sigla em inglês *MSE* ou *Mean Squared Error* –, que é uma métrica comum em árvores deste tipo. E apesar de não ser o escopo deste trabalho, vale notar que árvores de classificação utilizam métricas próprias para tratar variáveis categóricas como a Impureza de Gini e o Entropia (TIMOFEEV, 2004).

$$G(Q_m, \theta) = \frac{N_m^{esq}}{N_m} H(Q_m^{esq}(\theta)) + \frac{N_m^{dir}}{N_m} H(Q_m^{dir}(\theta)) \tag{7}$$

$$H(Q_m) = \frac{1}{N_m} \sum_{y \in Q_m} (y - \bar{y}_m)^2 \tag{8}$$

E, dentre todos os possíveis candidatos em um determinado nó, a escolha do melhor θ se dá pela minimização da função G (Equação 9).

$$\theta^* = argmin_\theta G(Q_m, \theta) \tag{9}$$

E uma particularidade do CART é que – a não ser que esteja explícito – uma árvore vai “aprofundar” em níveis até que não seja mais possível dividir a região preditora (TIMOFEEV, 2004). Isso difere de outros algoritmos que estabelecem critérios de paradas para o crescimento de uma árvore, como um número limite de

níveis, o não aumento da precisão do modelo com adição de novos nós, etc (MILANOVIĆ; STAMENKOVIĆ, 2017).

2.2.3.1 Procedimento de poda

O processo de poda em uma árvore de decisão é bastante comum, uma vez que este modelo estatístico tem propensão de criar estruturas muito complexas e causar o fenômeno conhecido por *overfitting* ou "sobreajuste" (PATIL; WADHAI; GOKHALE, 2010), que acontece quando o modelo está muito bem adequado aos dados utilizados para treinamento, porém ele não tem uma boa performance quando é generalizado para outros dados que não estavam contidos neste conjunto inicial. Alguns exemplos de métodos de poda são a poda do erro reduzido, poda do erro pessimista, e poda do valor crítico (ESPOSITO; MALERBA; SEMERARO, 1997). O método utilizado para poda neste trabalho é o que compõe a metodologia CART, e é denominado poda por custo de complexidade (ou poda por complexidade-custo).

O algoritmo de poda por custo de complexidade tem duas etapas: criação de um conjunto de sub-árvores $\{T_0, T_1, \cdots, T_L\}$, no qual T_0 é a árvore original, e os demais elementos são sub-árvores compostas pela remoção de um ou mais nós da T_0 (sendo T_L a sub-árvore com apenas o "nó raiz"); e seleção da sub-árvore que minimize a função da Equação 10 (JAMES et al., 2013).

A Equação 10, que é usada como critério de seleção da sub-árvore mais apta, é dividida em duas partes. Na primeira parte $\sum_{m=1}^{|T|} \sum_{x_i \in R_m} (y_i - y_{R_m})^2$, é calculada a soma dos erros residuais de cada sub-árvore. E, então, é adicionada o fator $\alpha|T|$, no qual α é chamado de parâmetro de complexidade (e que pode ser entendido como um custo pela adição de nós) e $|T|$ é a quantidade de nós terminais em uma determinada sub-árvore. Caso não existisse esse fator de penalização, o algoritmo sempre iria optar pela árvore original T_0 que sempre vai ser a maior árvore do conjunto (PATIL; WADHAI; GOKHALE, 2010).

$$\sum_{m=1}^{|T|} \sum_{x_i \in R_m} (y_i - y_{R_m})^2 + \alpha|T| \tag{10}$$

2.3 REDE NEURAL ARTIFICIAL

2.3.1 Definição

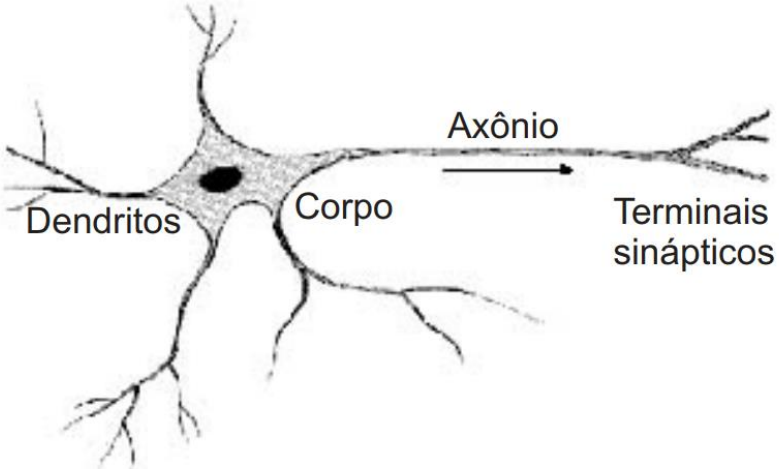
Redes neurais são conjuntos interconectados de elementos, chamados de nós, nos quais sua funcionalidade é baseada no funcionamento de um neurônio animal; e sua habilidade de processamento é armazenada em forças entre estas unidades chamadas de pesos, que são obtidos por meio de dados de treinamento (KRÖSE & SMAGT, 1996). E o treinamento pode ocorrer de maneira supervisionada ou não supervisionada (THOMAS, 2017): no treinamento supervisionado, a rede neural tem os dados e seus resultados, e a rede neural tenta modelar a função que determina estes resultados, um exemplo é o uso de redes neurais para detectar *spam* no trabalho de Silva et al. (2012); enquanto no treinamento não-supervisionado, a rede neural vai tentar entender os dados por "conta própria", como no reconhecimento de expressões faciais no estudo de Teves e Franco (2001).

Segundo Furtado (2019, p. 1), "a capacidade de resolver um determinado problema encontra-se na sua arquitetura, ou seja, no número e modo pelo qual os elementos processadores estão interconectados, nos pesos destas conexões e no número de camadas."

2.3.2 Funcionamento

O cérebro é composto por bilhões de neurônios, e cada neurônio é composto por três partes: dendritos, corpo e axônios (FERNEDA, 2006). O primeiro é responsável por captar estímulos de células vizinhas, e enviar para o corpo do neurônio, onde estes estímulos vão ser processados; quando é atingido um certo limite de estímulos, os neurônios vão enviar seus próprios estímulos para outras células por meio dos axônios, e as células vizinhas vão captá-los por meio de sinapses (SKLEARN, 2016). Um exemplo de um neurônio simples é esquematizado na FIGURA 6.

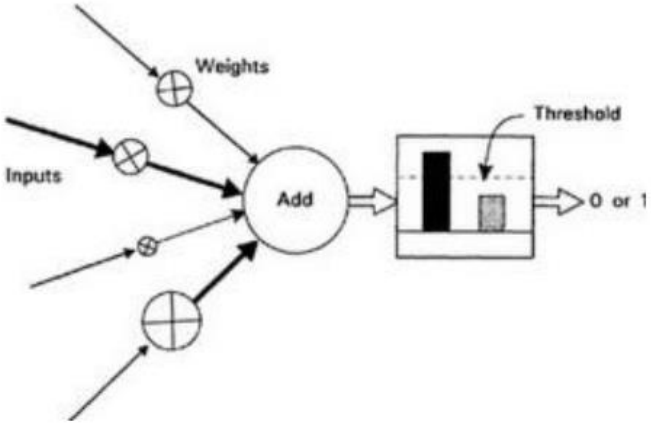
FIGURA 6 - ESTRUTURA DE UM NEURÔNIO



Fonte: Ferneda (2006)

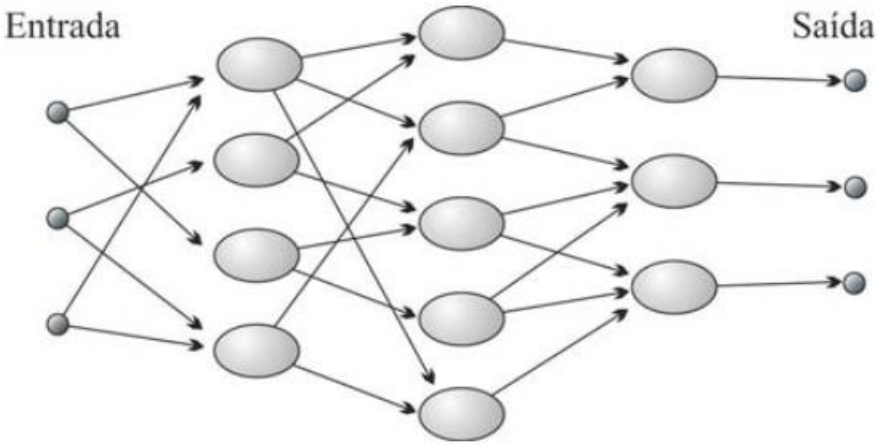
Segundo Kröse e Smagt (1996), neurônios, em uma rede neural artificial, são chamados de nós e sinapses são conhecidos por pesos; e cada *input* é multiplicado por este peso antes de chegar no equivalente ao corpo do neurônio, onde ocorrerá o processamento destes estímulos ou sinais. Dentro dele, estes sinais ponderados são somados por uma função, que pode ser uma função de soma aritmética simples, e o resultado dela é usada como dado de entrada pela função de ativação, que vai compará-la a um certo limite (chamado de *treshold* em inglês) e, então, fazer a decisão de qual estímulo enviar. Se este resultado for superior a esse limite, o nó envia como sinal um valor “alto” (convencionalmente, envia-se o número 1); caso contrário, envia como resultado o número 0 (zero). Um exemplo esquematizado é mostrado na FIGURA 7, e na FIGURA 8, é exemplificada a estrutura de uma rede neural artificial simples.

FIGURA 7 - ESTRUTURA DE UM NÓ DE UMA REDE NEURAL ARTIFICIAL



Fonte: Kröse e Smagt (1996)

FIGURA 8 - ESTRUTURA DE UMA REDE NEURAL ARTIFICIAL SIMPLES



Fonte: Ferneda (2006)

2.3.3 Formulação matemática

A seguinte formulação é de um tipo de rede neural chamada de perceptron multi-camadas (*multi-layer perceptron*, em inglês, e habitualmente abreviado como MLP), que é o algoritmo a ser utilizado neste estudo, e esta formulação trata de uma MLP com uma camada oculta (SKLEARN, 2016).

As variáveis independentes são $(x_{11}, x_{12}, .., x_{1m}), \ldots, (x_{n1}, x_{n2}, .., x_{nm})$ e as variáveis dependentes ou respostas são $y_1, .., y_k$ no modelo supervisionado, sendo que $x_{ij} \in \mathbb{R}^n$ e $y \in \mathbb{R}$. E, com uma camada oculta, a MLP aprende a função (11 pela

Equação 11, na qual $W_1 \in \mathbb{R}^m$ e $W_2, b_1, b_2 \in \mathbb{R}$, sendo que W_1 e W_2 são os pesos da camada de entrada e da camada oculta; e b_1 e b_2 são o *bias* de cada camada.

$$f(x) = W_2 g(W_1^T x + b_1) + b_2 \tag{11}$$

E, para regressão, utiliza-se a equação de perda mostrada na Equação 12; sendo que $\hat{y}$ é a previsão do algoritmo e $\frac{\alpha}{2}\|W\|_2^2$ é um termo de penalização para que o modelo não tenha *overfitting*. O modo de funcionamento da rede neural é que, começando com pesos aleatórios para as camadas, a MLP vai computar a perda de cada iteração, e então ela propaga o valor da camada de saída para as camadas anteriores atualizando o peso de cada parâmetro com o valor mais recente até que um critério de parada seja satisfeito (neste caso, até que atinja um número predefinido de iterações).

$$Perda(\hat{y}, y, W) = \frac{1}{2}\|\hat{y} - y\|_2^2 + \frac{\alpha}{2}\|W\|_2^2 \tag{12}$$

2.4 AVALIAÇÃO DOS MODELOS

Uma das etapas mais importantes de projetos de *Data Science* é a fase de avaliação, na qual os modelos treinados são avaliados utilizando dados de teste (BOTCHKAREV, 2018), ou seja, dados que não estavam contidos na fase de treinamento dos modelos. Esta etapa é importante para verificar se os modelos selecionados conseguem ser generalizados, e se é possível confiar nestas previsões e/ou estimativas (MUTUVI, 2019).

A forma como os algoritmos são avaliados é feita com base em métricas e, como é de se esperar, elas variam com base em modelos de classificação e regressão. Métricas utilizadas para classificação geralmente são acurácia, *F-scores,* taxa de erros, precisão média (HOSSIN; SULAIMAN, 2015). Enquanto as métricas para algoritmos de regressão são geralmente embasadas em ‘pontuações’ de erros, as principais são mostradas no QUADRO 1 (onde n é o número de observações ou dados, e_j é o erro no dado j, ou seja, a diferença do valor real e o valor estimado).

QUADRO 1 – PRINCIPAIS MÉTRICAS DE MODELOS REGRESSIVOS

Métrica	**Equação Matemática**
MAE: *Mean Absolute Error* (Média do Erro Absoluto)	$MAE = \frac{1}{n}\sum_{j=1}^{n}\left\|e_j\right\|$
MSE: *Mean Squared Error* (Média do Erro Quadrático)	$MSE = \frac{1}{n}\sum_{j=1}^{n} e_j^2$
RMSE: *Root Mean Squared Error* (Raiz do Erro Quadrático)	$RMSE = \sqrt{\frac{\sum_{j=1}^{n} e_j^2}{n}}$

FONTE: Botchkarev (2018)

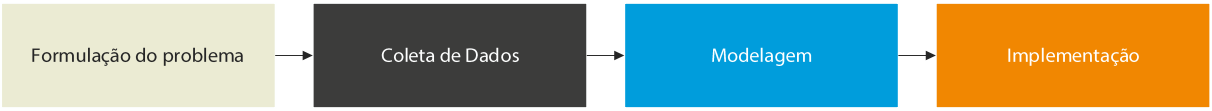
3 METODOLOGIA

O atual trabalho é caracterizado como uma pesquisa exploratória, ou seja, uma pesquisa com objetivo principal de adquirir maior familiaridade com o tema estudado (SELLTIZ, 1987). Como consequência, por meio do aumento de conhecimento sobre este determinado assunto, esse modelo de pesquisa facilita uma formulação mais precisa de problemas, hipóteses e pesquisas deste campo de estudo (MAXWELL, 2011).

Considerando que este trabalho envolve adequação de modelos de *machine learning*, foi natural a escolha de uma metodologia de *Data Science* (Ciência de Dados) como base para este estudo.

Existem diversas metodologias propostas para abordar projetos de *Data Science* e que – de acordo com Furoughi e Luksch (2018) – partem de alguns processos básicos, que são: definição e formulação do problema, coleta de dados, modelagem e implementação (FIGURA 9). Alguns exemplos de metodologias apresentadas na literatura são CRISP-DM (NETER et al., 1996); KDD (MONTGOMERY; PECK; VINING, 2013) e *Data Science Workflow Framework* (NETER et al., 1996).

FIGURA 9 – PROCESSOS BÁSICOS DE UMA METODOLOGIA DE *DATA SCIENCE*

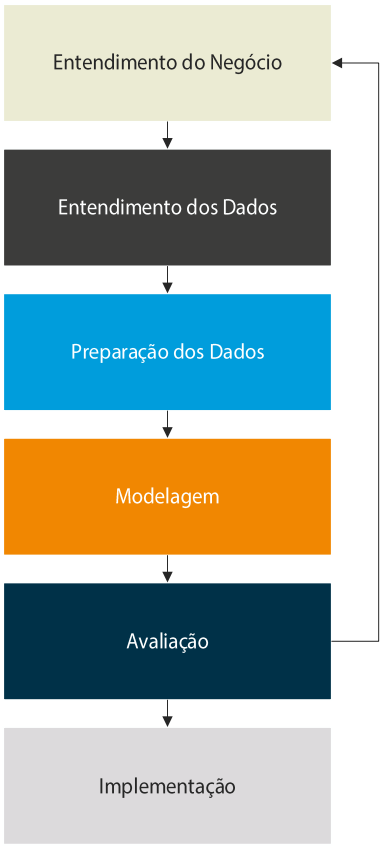


Fonte: O autor (2021)

3.1 METODOLOGIA CRISP-DM

A metodologia base escolhida para este trabalho é a CRISP-DM (*Cross Industry Standard Process for Data Mining* ou, em português, Processo Padrão da Indústria Cruzada para Mineração de Dados), que foi elaborada pelas empresas SPSS e Teradata em 1996 (MONTGOMERY; PECK; VINING, 2013), e é a metodologia mais usada para projetos de Ciência e Mineração de Dados (NETER et al., 1996).

FIGURA 10 – METODOLOGIA CRISP-DM



Fonte: O autor (2021)

Na FIGURA 10, é mostrado um fluxograma com as etapas da metodologia CRISP-DM, e que são explicadas a seguir:

a) Entendimento do negócio: o objetivo nesta etapa é entender o problema e os objetivos do projeto, e transformar esse entendimento inicial em uma formulação preliminar de um problema de Ciência de Dados;
b) Entendimento dos dados: esta etapa começa com a coleta de dados, e tarefas iniciais como familiarização com os bancos de dados e identificação de hipóteses introdutórias para o projeto;
c) Preparação dos dados: ao longo deste estágio, é feita toda transformação dos dados brutos para o formato que vai ser trabalhado até o final do projeto,
d) Modelagem: durante esta fase, são selecionadas e aplicadas modelagens estatísticas e de *machine learning*, e os parâmetros dos modelos são identificados e calibrados;

e) Avaliação: utilizando os modelos que tiveram melhor performance na etapa anterior, são feitos uma série de testes para se assegurar que eles conseguem ser generalizados para dados que não estavam contidos nos dados de treinamento e que eles vão conseguir atender os objetivos gerais do projeto. Além disso, uma vez que esta etapa é finalizada, é comum alimentar o entendimento do negócio, seja em um documento formal ou como troca de conhecimento entre times, com os *insights* e conclusões tiradas ao longo do projeto;
f) Implementação: na última etapa da metodologia CRISP, é elaborada e executada uma estratégia para implementação do projeto na empresa, e – a partir dali – ele vai ser colocado em produção.

3.2 MATERIAIS E MÉTODOS

Este trabalho foi feito utilizando a linguagem de programação Python, uma das linguagens mais populares atualmente (PYPL, 2021). E que é bastante utilizada para projetos de *Data Science*, uma vez que, além de ser uma linguagem de fácil interpretação, contém bibliotecas especializadas para aplicações de *Machine Learning* (BEKLEMYSHEVA, 2019). As principais bibliotecas utilizadas foram:

a) *Pandas* para tratamento, limpeza e manipulação de dados;
b) *Statsmodels* para adequação do modelo de regressão linear múltipla;
c) *Sklearn* para treinamento do modelo de árvore de decisão, rede neural artificial, e avaliação dos modelos;
d) *Matplotlib* para visualização dos dados.

Os dados utilizados foram retirados do site *Our World in Data*, nos quais os dados disponibilizados são utilizados para ensino em diversas universidades, como Harvard, Stanford, Cambridge, MIT e Oxford, e são reconhecidos por jornais como *The New York Times*, *BBC*, *The Washington Post* e *The Wall Street Journal* (OUR WORLD IN DATA, 2021a). Em geral, os dados do *Our World in Data* são referentes à temáticas globais como pobreza, crescimento econômico, emissões de gases de efeito estufa e doenças. Neste trabalho, foram utilizados os dados de número de mortes diárias, por país, causadas pelo COVID-19; e que são atualizados todos os

dias. O conjunto de dados contém 59 colunas, como país, número de mortes, número de casos, número de testes, taxa de pobreza, IDH do país, número de pessoas com idade superior a 65 anos, etc. O período utilizado foi do dia 1 de janeiro de 2020 até o dia 2 de fevereiro de 2021. As variáveis utilizadas foram escolhidas com base no critério de correlação com a variável dependente (número de mortes pelo COVID-19), e depois foi feito um novo filtro para excluir aquelas variáveis que eram correlacionadas entre si mesmas; foi utilizado um critério de que as variáveis deveriam ter até 0,3 de correlação entre si mesmas.

E antes que aconteça o treinamento e adequação dos modelos, é comum dividir todos os dados em treinamento e teste com base em uma proporção de 80/20 (em outras palavras, 80% dos dados destinados para treinamento, e 20% para testes dos modelos), porém foi escolhido dividir em 70/30 em uma tentativa de evitar *overfitting* e procurar ter um modelo mais generalizado possível, já que o trabalho atual tem 190 dados (que é referente ao número total de países no planeta) e que é um número relativamente pequeno.

Para adequar o modelo de regressão linear múltipla, foi utilizado a função *ols* da biblioteca *statsmodels*, que otimiza o modelo pelo Método dos Mínimos Quadrados. E para o modelo de árvore de decisão, foi utilizada a função *DecisionTreeRegressor* da biblioteca de árvores do *sklearn*. Em relação aos parâmetros de treinamento deste último modelo, o critério de qualidade das divisões foi com base no MSE (assim como explicado na seção 2.2.3.1), não foi definida profundidade máxima (ou seja, a ideia é que a árvore cresça até que não seja mais possível fazer divisões, e que é um dos pressupostos do algoritmo CART), e o número mínimo para que aconteça uma divisão é de 2 amostras (se chegar em um nível da árvore em que tenha apenas uma amostra, chega-se ao final daquele ramo específico).

No final da adequação da regressão linear múltipla, treinamento da árvore de decisão e rede neural artificial, é feito um comparativo dos dois modelos utilizando-se como critérios avaliativos MAE, MSE e RMSE (seção 2.4).

4 APRESENTAÇÃO DOS RESULTADOS

4.1 VARIÁVEIS UTILIZADAS

As variáveis independentes escolhidas foram (por país):

a) Número de casos;
b) Densidade populacional;
c) E população total.

E as correlações entre si das variáveis é mostrado no QUADRO 2.

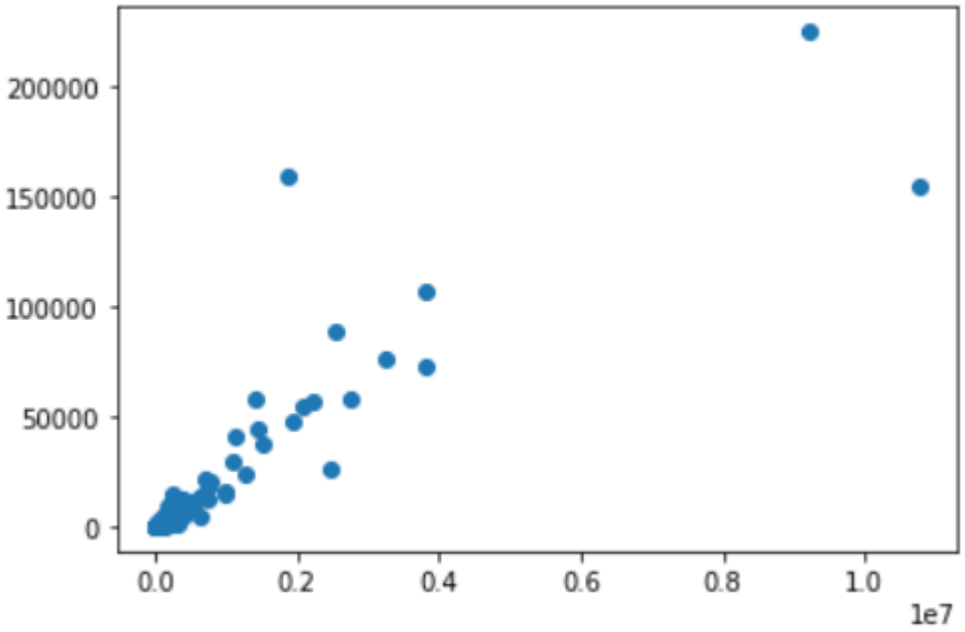
4.2 REGRESSÃO LINEAR MÚLTIPLA

Para validação da regressão linear múltipla, foram feitos os testes das premissas mencionadas na seção 2.1.2.1.

4.2.1 Premissa de linearidade

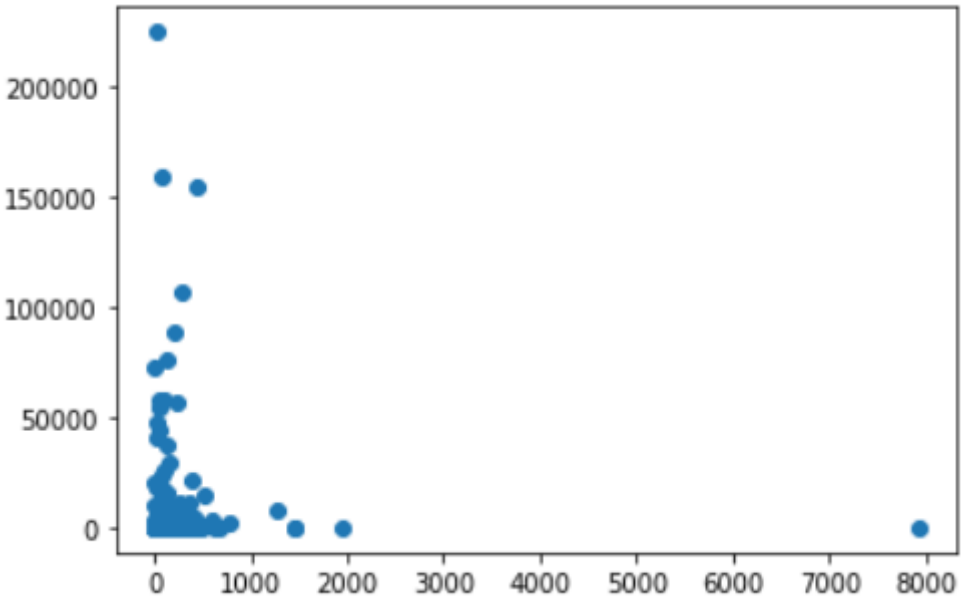
Seguem os gráficos de dispersão mostrando a relação entre cada variável dependente e a variável independente. Nos GRÁFICOS 1, 2 e 3, são mostrados, respectivamente, os gráficos de dispersão das variáveis total de casos, densidade populacional e população no eixo X com a variável dependente (neste caso, número de mortes pelo COVID-19) no eixo Y.

GRÁFICO 1 - DISPERSÃO DO TOTAL DE CASOS E NÚMERO DE MORTES



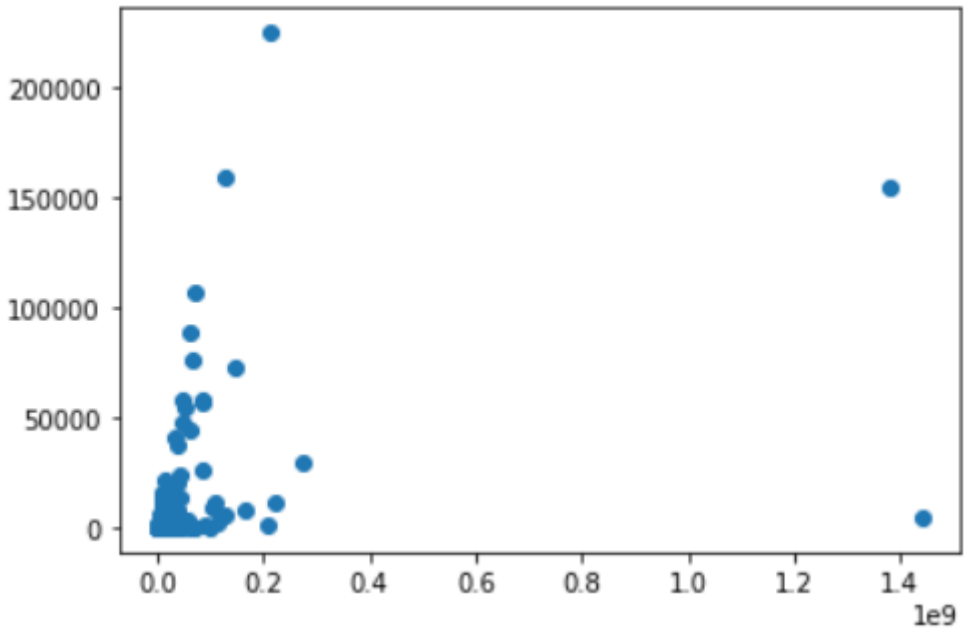
Fonte: O autor (2021)

GRÁFICO 2 - DISPERSÃO DA DENSIDADE POPULACIONAL E NÚMERO DE MORTES



Fonte: O autor (2021)

GRÁFICO 3 - DISPERSÃO DA POPULAÇÃO E NÚMERO DE MORTES



Fonte: O autor (2021)

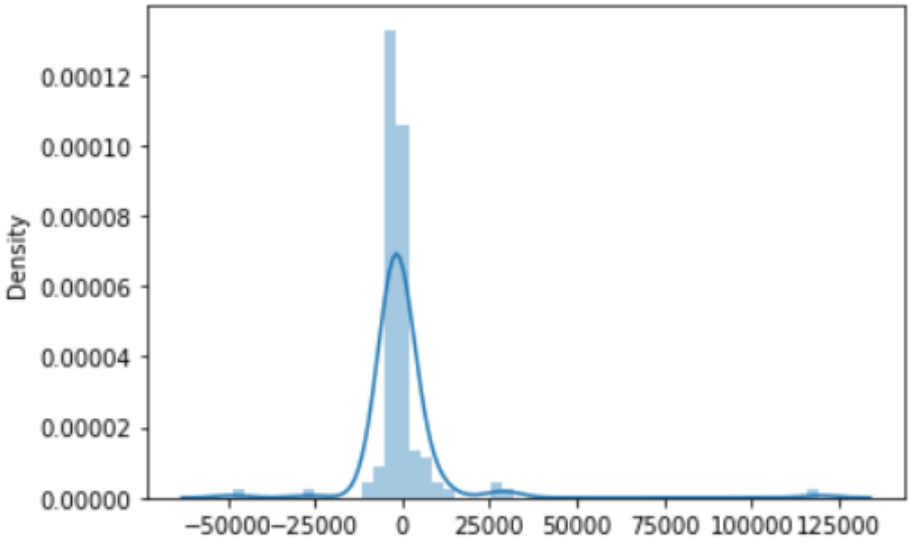
4.2.2 Premissa de não auto-correlação

O teste de Durbin-Watson produziu valor de 2,13, indicando que não existe auto-correlação na amostra estudada.

4.2.3 Premissa de normalidade dos erros

Os erros residuais, quando plotados em um histograma, tendem à uma distribuição normal validando a premissa de normalidade como mostrado no Gráfico 4. Além disso, a soma dos erros residuais é igual a zero.

GRÁFICO 4 – HISTOGRAMA E GRÁFICO DE DENSIDADE DOS ERROS RESIDUAIS



Fonte: O autor (2021)

4.2.4 Premissa de multicolinearidade

Para testar a premissa de multicolinearidade, foi utilizada uma matriz de correlação das variáveis independentes (QUADRO 2), e como nenhuma apresentou entre si correlações superiores a 0,3, foi considerado que a premissa de multicoliearidade foi atendida.

QUADRO 2 – MATRIZ DE CORRELAÇÃO DAS VARIÁVEIS INDEPENDENTES

Variáveis Independentes	Número de casos	Densidade Populacional	População Total
Número de casos	1	0,01585	0,221397
Densidade Populacional	0,01585	1	0,00868
População Total	0,221397	0,00868	1

Fonte: O autor (2021)

4.2.5 Premissa de homocedasticidade

Para checar a premissa de homocedasticidade, foi utilizado o teste de Breusch-Pagan. No entanto, o teste resultou no valor 0,02; em outras palavras, rejeita-se a hipótese de que homocedasticidade está presente. Assim, esta premissa *não* é atendida.

4.2.6 Equação resultante

A equação da regressão linear múltipla resultante é mostrada na Equação 13; sendo que x_1, x_2 e x_3 são as variáveis do total de casos, densidade populacional e população dos países, respectivamente.

$$y(x) = 1976 + \ 0{,}02199.x_1 + \ 0{,}23922.x_2 + 0{,}002279.x_3 \qquad (13)$$

4.3 ÁRVORE DE DECISÃO

Para treinar a árvore de regressão, o critério utilizado para mensurar a qualidade dos possíveis candidatos – depois de cada divisão do espaço preditor – foi a função do erro quadrático médio (como detalhado anteriormente na seção 2.2.2.1, Equação 8); a escolha da melhor divisão se deu pelo critério de “melhor” pontuação (a partir do erro quadrático médio) em vez de ser uma escolha aleatória; a profundidade máxima definida da árvore foi de 1000; o menor número de amostras

para que o algoritmo dividisse o espaço preditor foi definido como 2. Estas informações estão resumidas no QUADRO 3. Ao terminar de treinar o modelo, a árvore de decisão final teve 16 níveis (profundidade de 16), e o número de nós foi de 109. Vale notar que a otimização do número de nós e profundidade da árvore de decisão foi feito pelo próprio algoritmo do *sklearn*.

QUADRO 3 – PRINCIPAIS PARÂMETROS DA ÁRVORE DE DECISÃO

Parâmetro	**Valor**
Critério de qualidade	Erro quadrático médio
Escolha da melhor divisão	“Melhor”
Profundidade máxima	1000
Menor número de amostras para que tenha uma iteração	2

FONTE: O Autor (2021)

4.4 REDE NEURAL

Ao treinar o modelo, o critério de parada definido foi que o algoritmo parasse quando atingisse um número predefinido de iterações (neste caso, de 500); porém o algoritmo poderia terminar também se não houvesse melhoria significativa na função de perda. O alfa, termo de penalização para que o algoritmo tenha uma menor probabilidade de *overfitting*, foi definido com 0,0001 (valor padrão recomendado pelo *sklearn*, biblioteca utilizada para treinar o modelo). A tolerância e o número de iterações sem melhora na otimização foram definidos com 0,0001 e 10, respectivamente; em outras palavras, após 10 iterações sem melhora de pelo menos 0,0001 na pontuação de otimização, o algoritmo vai parar de rodar. Estes parâmetros estão resumidos no QUADRO 4. Ao final, o algoritmo teve 49 iterações e obteve o melhor resultado com 3 camadas na rede neural (1 camada de entrada, 1 camada oculta e 1 camada de saída), sendo que a camada oculta teve 100 neurônios (ou nós). Assim como caso da árvore de decisão, o próprio algoritmo da biblioteca *sklearn* foi responsável por otimizar o número final de camadas e nós.

QUADRO 4 – PRINCIPAIS PARÂMETROS DA REDE NEURAL

Parâmetro	**Valor**
Número máximo de iterações	500
Alfa	0,0001
Tolerância	0,0001
Número de iterações sem melhoria na otimização	10

FONTE: O Autor (2021)

4.5 AVALIAÇÃO DOS RESULTADOS

Com base nas métricas de avaliação de resultados explicadas na seção 2.5, chegou-se no resultados mostrados na TABELA 1. Vale notar que estes resultados são para o conjunto de teste que representa 30% das amostras do conjunto de dados total.

TABELA 1 – RESULTADOS DOS MODELOS PARA O CONJUNTO DE TESTE

Modelo	**MAE**	**MSE**	**RMSE**
Regressão linear múltipla	3.228	29.004.156	5.385
Árvore de decisão	1.872	14.848.095	3.853
Rede neural	2.261	27.791.773	5.271

FONTE: O autor (2021)

Em todas as métricas de erros avaliadas, a árvore de decisão obteve os melhores resultados entre os três modelos, seguida da rede neural e da regressão linear múltipla (lembrando que o critério de homocedasticidade não foi atendido).

5 CONSIDERAÇÕES FINAIS

Neste estudo, foram utilizados três modelos (regressão linear múltipla, árvore de decisão e rede neural artificial) e que foram escolhidos de forma arbitrária. Uma possibilidade é avaliar outros modelos de *machine learning*, em particular, *random forest* (floresta aleatória), que combina uma série de árvores de decisões para treinar o modelo, e uma vez que árvore de decisão foi o modelo que melhor se adequou ao objetivo deste estudo, *random forest* pode trazer resultados similares ou melhores.

Uma outra observação é que a regressão linear múltipla não atendeu a todas as premissas (o critério de homocedasticidade não foi validado), porém foi escolhido manter o modelo com estas variáveis (em vez de substituí-las) para ter um mesmo parâmetro de comparação com os outros modelos, que já tinham apresentado bons resultados.

E um último ponto que vale notar é que o estudo foi feito em fevereiro de 2021, e até então 1,2 milhão de pessoas tinham sido vacinadas. Porém, até os dados de 2 de agosto de 2021, mais de 1,15 bilhão de pessoas tinham sido vacinadas (OUR WORLD IN DATA, 2021), o que representa quase 15% da população mundial. Pensando nisso, o próximo passo seria colocar a variável de pessoas vacinadas em futuros estudos, e avaliar se uma árvore de decisão ainda seria o melhor algoritmo (entre os estudados) que melhor se ajusta ao conjunto de dados.

REFERÊNCIAS

ALANAGREH, L.; ALZOUGHOOL, F.; ATOUM, M. The human coronavirus disease covid-19: Its origin, characteristics, and insights into potential drugs and its mechanisms. **Pathogens**, v. 9, n. 5, 2020.

ALBERT. **Key Assumptions of OLS: Econometrics Review**. Disponível em: <https://www.albert.io/blog/key-assumptions-of-ols-econometrics-review/>. Acesso em: 24 fev. 2021.

ALPAYDIN, E. **Introduction to Machine Learning (Adaptive Computation and Machine Learning series)**. 2.ed. The MIT Press, 2009. 584p.

ANIRUDH, A. Mathematical modeling and the transmission dynamics in predicting the Covid-19 - What next in combating the pandemic. **Infectious Disease Modelling**, v. 5, p. 366–374, 2020.

BALKHAIR, A. A. Covid-19 pandemic: A new chapter in the history of infectious diseases. **Oman Medical Journal**, v. 35, n. 2, p. 2–3, 2020.

BANSAL, G. **What are the four assumptions of linear regression?**. Disponível em: <https://blog.uwgb.edu/bansalg/statistics-data-analytics/linear-regression/what-are-the-four-assumptions-of-linear-regression/>. Acesso em: 24 fev. 2021.

BEKLEMYSHEVA, A. **Why Use Python for AI and Machine Learning?**. Disponível em: <https://steelkiwi.com/blog/python-for-ai-and-machine-learning/>.

BOŠKOSKI, I. et al. COVID-19 pandemic and personal protective equipment shortage: protective efficacy comparing masks and scientific methods for respirator reuse. **Gastrointestinal Endoscopy**, v. 92, n. 3, p. 519–523, 2020.

BOŠNJAK, Z.; GRLJEVIĆ, O.; BOŠNJAK, S. CRISP-DM as a framework for discovering knowledge in small and medium sized enterprises' data. **Proceedings - 2009 5th International Symposium on Applied Computational Intelligence and Informatics, SACI 2009**, n. May 2014, p. 509–514, 2009.

BOTCHKAREV, A. Evaluating Performance of Regression Machine Learning Models Using Multiple Error Metrics in Azure Machine Learning Studio. **SSRN Electronic Journal**, n. March, 2018.

BREIMBREIMAN, L.; FRIEDMAN, J. H.; OLSHEN, R. A.; STONE, C. J. CLASSIFICATION AND REGRESSION TREES. CLASSIFICATION AND

REGRESSION TREES, P. 1–358, 2017.AN, L. et al. Classification and regression trees. **Classification and Regression Trees**, p. 1–358, 2017.

CABALION, S. et al. Middle East respiratory syndrome coronavirus and human-camel relationships in Qatar. **Medicine Anthropology Theory**, v. 5, n. 3, 2018.

CHEN, Y.; GUO, D. Molecular mechanisms of coronavirus RNA capping and methylation. **Virologica Sinica**, v. 31, n. 1, p. 3–11, 2016.

COTTEN, M. et al. Spread, circulation, and evolution of the Middle East respiratory syndrome coronavirus. **mBio**, v. 5, n. 1, 2014.

DATA, O. W. in. **Coronavirus (COVID-19) Vaccinations**. Disponível em: <https://ourworldindata.org/covid-vaccinations?country=CHN>.

DE OLIVEIRA LIMA, C. M. A. Information about the new coronavirus disease (COVID-19). **Radiologia Brasileira**, v. 53, n. 2, p. v–vi, 2020.

DE SOUZA, W. M. et al. Epidemiological and clinical characteristics of the COVID-19 epidemic in Brazil. **Nature Human Behaviour**, v. 4, n. 8, p. 856–865, 2020.

DING, Y.; GAO, L. An evaluation of COVID-19 in Italy: A data-driven modeling analysis. **Infectious Disease Modelling**, v. 5, p. 495–501, 2020.

EIKENBERRY, S. E. et al. To mask or not to mask: Modeling the potential for face mask use by the general public to curtail the COVID-19 pandemic. **Infectious Disease Modelling**, v. 5, p. 293–308, 2020.

ESPOSITO, F.; MALERBA, D.; SEMERARO, G. A comparative analysis of methods for pruning decision trees. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 19, n. 5, p. 476–491, 1997.

FAHRMEIR, L.; KNEIB, T.; LANG, S. **Regression: Models, Methods and Applications**. 2013. 714p. (Journal of Materials Processing Technology).

FERNEDA, E. Redes neurais e sua aplicação em sistemas de recuperação de informação. **Ciência da Informação**, v. 35, n. 1, p. 25–30, 2006.

FOKIN, D.; HAGROT, J. Constructing decision trees for user behavior prediction in the online consumer market. 2016.

FOROUGHI, F.; LUKSCH, P. Data science methodology for cybersecurity projects. **arXiv**, n. February, 2018.

FROST, J. **7 Classical Assumptions of Ordinary Least Squares (OLS) Linear Regression**. Disponível em: <https://statisticsbyjim.com/regression/ols-linear-regression-assumptions/>. Acesso em: 24 fev. 2021.

FURTADO, M. I. V. **Redes Neurais Artificiais: Uma Abordagem Para Sala de Aula**. 2019. (Redes Neurais Artificiais: Uma Abordagem Para Sala de Aula).

GURNEY, K. **An Introduction to Neural Networks**. 1.ed. 1997. 234p.

HOSSIN, M.; SULAIMAN, M. N. A Review on Evaluation Metrics for Data Classification Evaluations. **International Journal of Data Mining & Knowledge Management Process**, v. 5, n. 2, p. 01–11, 2015.

HSSINA, B. et al. A comparative study of decision tree ID3 and C4.5. **International Journal of Advanced Computer Science and Applications**, v. 4, n. 2, 2014.

HUI, D. S. et al. The continuing 2019-nCoV epidemic threat of novel coronaviruses to global health — The latest 2019 novel coronavirus outbreak in Wuhan, China. **International Journal of Infectious Diseases**, v. 91, n. January, p. 264–266, 2020.

JAMES, G. et al. **An Introduction to Statistical Learning: with Applications in R**. 7th.ed. Springer, 2013.

KOIRALA, A. et al. Vaccines for COVID-19: The current state of play. **Paediatric Respiratory Reviews**, v. 35, n. July, p. 43–49, 2020.

KRÖSE, B.; SMAGT, P. van der. Introduction to neural networks. 1996.

KUMAR, D. Corona Virus: A Review of COVID-19. **Eurasian Journal of Medicine and Oncology**, n. April, 2020.

LANA, R. M. et al. The novel coronavirus (SARS-CoV-2) emergency and the role of timely and effective national health surveillance. **Cadernos de Saude Publica**, v. 36, n. 3, 2020.

LAVESSON, N. **Evaluation and Analysis of Supervised Learning Algorithms and Classifiers**. 2006. 98p. (Department of Systems and Software Engineering).

LI, L. et al. Propagation analysis and prediction of the COVID-19. **Infectious Disease Modelling**, v. 5, p. 282–292, 2020.

LOH, W. Y. Classification and regression trees. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, v. 1, n. 1, p. 14–23, 2011.

MA, J. Coronavirus (COVID-19): History, Current Knowledge and Pipeline Medications. **International Journal of Pharmaceutics & Pharmacology**, v. 4, n. 1, p. 1–9, 2020.

MAXWELL, F. de O. Metodologia científica: um manual para a realização de pesquisas em Administração. **Biblioteca da UFG – Campus Catalão**, p. 72, 2011.

MEDEIROS, M. C. et al. Short-term Covid-19 forecast for latecomers. **arXiv**, p. 1–10, 2020.

MENG, Q. et al. A communication-efficient parallel algorithm for decision tree. **Advances in Neural Information Processing Systems**, n. Nips, p. 1279–1287, 2016.

MERKLE, E. C.; SHAFFER, V. A. Binary recursive partitioning: Background, methods, and application to psychology. **British Journal of Mathematical and Statistical Psychology**, v. 64, n. 1, p. 161–181, 2011.

MILANOVIĆ, M.; STAMENKOVIĆ, M. CHAID Decision Tree: Methodological Frame and Application. **Economic Themes**, v. 54, n. 4, p. 563–586, 2017.

MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. **Introduction to Linear Regression Analysis**. 5.ed. 2012.

MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. **Introduction to Linear Regression Analysis**. 6.ed. 2013. 164p.

MOREIRA, C. Neurónio. **Revista de Ciência Elementar**, v. 1, n. 1, p. 1–3, 2013.

MUTUVI, S. **Introduction to Machine Learning Model Evaluation**. Disponível em: <https://heartbeat.fritz.ai/introduction-to-machine-learning-model-evaluation-fa859e1b2d7f>.

NETER, J. et al. **Applied Linear Regression Models**. 1996. 720p.

OUR WORLD IN DATA. **Our World in Data.** Disponível em: <https://ourworldindata.org/>. Acesso em: 17 mar. 2021.

OUR WORLD IN DATA. **Coronavirus (COVID-19) Deaths**. Disponível em: <https://ourworldindata.org/covid-deaths>. Acesso em: 2 ago. 2021.

OVERTON, C. E. et al. Using statistics and mathematical modelling to understand infectious disease outbreaks: COVID-19 as an example. **Infectious Disease Modelling**, v. 5, p. 409–441, 2020.

PAPASTEFANOPOULOS, V.; LINARDATOS, P.; KOTSIANTIS, S. COVID-19: A comparison of time series methods to forecast percentage of active cases per population. **Applied Sciences (Switzerland)**, v. 10, n. 11, p. 1–15, 2020.

PATEL, N.; SINGH, D. An Algorithm to Construct Decision Tree for Machine Learning based on Similarity Factor. **International Journal of Computer Applications**, v. 111, n. 10, p. 22–26, 2015.

PATIL, D. D.; WADHAI, V. M.; GOKHALE, J. A. Evaluation of Decision Tree Pruning Algorithms for Complexity and Classification Accuracy. **International Journal of Computer Applications**, v. 11, n. 2, p. 23–30, 2010.

PENG, W.; CHEN, J.; ZHOU, H. An Implementation of ID3 Decision Tree Learning Algorithm. **Project of Comp 9417: Machine Learning**, v. 1, p. 1–20, 2009.

PETROPOULOS, F.; MAKRIDAKIS, S. Forecasting the novel coronavirus COVID-19. **PLoS ONE**, v. 15, n. 3, p. 1–8, 2020.

PYPL. **PYPL PopularitY of Programming Language**. Disponível em: <http://pypl.github.io/PYPL.html>. Acesso em: 17 mar. 2021.

ROOSA, K. et al. Real-time forecasts of the COVID-19 epidemic in China from February 5th to February 24th, 2020. **Infectious Disease Modelling**, v. 5, p. 256–263, 2020.

SALTZ, J.; SHAMSHURIN, I.; CROWSTON, K. Comparing Data Science Project Management Methodologies via a Controlled Experiment. **Proceedings of the 50th Hawaii International Conference on System Sciences (2017)**, p. 1013–1022, 2017.

SCHNEIDER, A.; HOMMEL, G.; BLETTNER, M. Linear Regression Analysis. **Deutsches Arzteblatt**, v. 107, n. 44, p. 776–782, 2010.

SELLTIZ, W. e C. Métodos de pesquisa nas relações sociais. v. 2, 1987.

SILVA, R. M.; ALMEIDA, T. A.; YAMAKAMI, A. Artificial neural networks for content-based web spam detection. **Proceedings of the 2012 International Conference on Artificial Intelligence, ICAI 2012**, v. 1, n. January, p. 209–215, 2012.

SKLEARN. Neural network models (supervised). 2016.

STATICSSOLUTIONS. **Autocorrelation**. Disponível em: <https://www.statisticssolutions.com/autocorrelation/>. Acesso em: 25 fev. 2021.

STATISTICSSOLUTIONS. **Assumptions of Linear Regression**. Disponível em: <https://www.statisticssolutions.com/assumptions-of-linear-regression/>. Acesso em: 24 fev. 2021.

STATOLOGY. **The Breusch-Pagan Test: Definition & Example**. Disponível em: <https://www.statology.org/breusch-pagan-test/>. Acesso em: 24 fev. 2021

TEVES, A.; FRANCO, L. **A Neural Network Facial Expression Recognition System using Unsupervised Local Processing**. v. 1, n. December 2012, p. 0–1, 2001.

THOMAS, A. **A beginners introduction to neural network**. 2017.

TIMOFEEV, R. **Classification and Regression Trees (CART)**. 2004.

WANG, Y. Predict new cases of the coronavirus 19; in Michigan, U.S.A. or other countries using Crow-AMSAA method. **Infectious Disease Modelling**, v. 5, p. 459–477, 2020.

WU, X. et al. **Top 10 algorithms in data mining**. 2008. 1–37p. (Knowledge and Information Systems).

XIAOHU, W.; LELE, W.; NIANFENG, L. An Application of Decision Tree Based on ID3. **Physics Procedia**, v. 25, p. 1017–1021, 2012.

YAN, X.; SU, X. G. **Linear Regression Analysis: Theory and Computing**. 2009.

YANG, Y. et al. The deadly coronaviruses: The 2003 SARS pandemic and the 2020 novel coronavirus epidemic in China. **Journal of Autoimmunity**, v. 109, n. March, p. 102434, 2020.

ZIKMUND, WILLIAM G. BABIN, BARRY J. CARR, JON C. GRIFFIN, M. **Business research methods**. 9.ed. 2012. 696p.