Universidade Federal do Maranhão - UFMA
Centro de Ciências Exatas e Tecnologia - CCET
Coordenação do Curso de Engenharia da Computação - CCEC
Fundamentos de Redes Neurais

TÓPICOS EM ENGENHARIA DA COMPUTAÇÃO II - FUNDAMENTOS DE REDES NEURAIS – REGRESSÃO LINEAR MULTIVARIADA

Prof. Dr. Thales Levi Azevedo Valente
Discente: Matheus Costa Alves

São Luís - MA

2025

Sumário

1 INTRODUÇÃO

A análise de dados tem se consolidado como uma ferramenta essencial no processo de tomada de decisão em diversas áreas do conhecimento. No contexto da engenharia e da ciência de dados, a modelagem preditiva é um dos pilares mais relevantes, permitindo antecipar comportamentos e estimar valores futuros com base em observações históricas. Dentre os diversos modelos disponíveis, a regressão linear multivariada se destaca por sua simplicidade, interpretabilidade e eficiência computacional. Esse modelo busca estabelecer uma relação linear entre uma variável dependente contínua e múltiplas variáveis independentes, sendo amplamente aplicado em problemas reais como precificação de imóveis, estimativas de consumo energético e análise de desempenho industrial.

Apesar de sua estrutura simples, a eficácia da regressão linear multivariada depende fortemente de aspectos como a normalização das variáveis de entrada e o método escolhido para estimação dos parâmetros do modelo. Técnicas como o Gradiente Descendente (GD) e a Equação Normal (NE) apresentam diferentes características em termos de complexidade computacional, sensibilidade à escala dos dados e velocidade de convergência. Dessa forma, compreender os efeitos dessas abordagens no desempenho do modelo é essencial para uma aplicação mais robusta em contextos reais.

Este trabalho tem como objetivo investigar o impacto da normalização das variáveis e comparar o desempenho das técnicas de otimização GD e NE na estimação dos coeficientes θ de uma regressão linear multivariada. Os experimentos foram conduzidos com base em dados reais, utilizando-se implementações em Python e visualizações gráficas para apoiar a análise dos resultados. A seguir, apresenta-se a fundamentação teórica necessária para compreensão dos conceitos aplicados ao longo deste estudo.

2 FUNDAMENTAÇÃO TEÓRICA

A regressão linear é um dos modelos estatísticos mais antigos e fundamentais da aprendizagem de máquina supervisionada (James et al., 2013). Seu objetivo principal é modelar a relação entre uma variável dependente y e uma ou mais variáveis independentes x, por meio de uma equação linear (James et al., 2013). No caso mais simples, denominado regressão linear simples, essa relação é descrita pela seguinte fórmula:

$$\hat{y} = \beta_0 + \beta_1 x \tag{1}$$

Nesta equação, $\hat{y}$ representa o valor predito pelo modelo, x é a variável explicativa, β_0 é o intercepto (ou termo independente), e β_1 é o coeficiente angular da reta, responsável por quantificar o efeito de x sobre y. O modelo pode ser generalizado para múltiplas variáveis independentes, formando o que se chama de regressão linear múltipla (James et al., 2013). A equação é então expressa como:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n \quad (2)$$

Ou, de maneira compacta e vetorial:

$$\hat{y} = x^{\top} \quad (3)$$

Onde:

- $x \in R$ é o vetor de atributos, geralmente incluindo o termo constante (intercepto),
- $\beta \in R$ é o vetor de coeficientes do modelo,
- $x^{\top}\beta$ representa o produto escalar entre os vetores.

A suposição central da regressão linear é que existe uma relação aproximadamente linear entre as variáveis independentes e a variável dependente. Ou seja, espera-se que a variação em y possa ser explicada como uma combinação linear dos x_i (Weisberg, 2014). Essa simplicidade matemática, somada à sua eficiência computacional e interpretabilidade, torna a regressão linear um modelo amplamente utilizado, especialmente como ponto de partida na análise de dados.

2.1 Função de Custo

Para que o modelo de regressão linear seja efetivamente útil na predição de valores, é necessário determinar os coeficientes β que minimizam o erro entre os valores preditos $\hat{y}$ e os valores reais y. Esse processo é orientado por uma função de custo, que quantifica a discrepância entre o modelo e os dados observados.

A função de custo mais comum utilizada na regressão linear é o Erro Quadrático Médio (Mean Squared Error – MSE), dada por:

$$J(\beta) = \frac{1}{n}\sum_{i=1}^{n}(\hat{y_i} - y_i)^2 \quad (4)$$

Onde:

- n representa o número de amostras,
- $\hat{y_i}$ é a predição do modelo para a i-ésima amostra,

- y_i é o valor real correspondente,
- x_i é o vetor de características da i-ésima amostra,
- β é o vetor de parâmetros do modelo.

Essa função impõe penalidade quadrática aos erros cometidos pelo modelo, tornando a minimização mais sensível a outliers. No entanto, essa característica também contribui para uma superfície de custo suave e convexa, que garante a existência de um mínimo global.

Minimizar o valor de $J(\beta)$ significa encontrar os valores ótimos dos coeficientes que melhor ajustam a linha de regressão aos dados observados. Para isso, diferentes técnicas podem ser utilizadas, como métodos analíticos baseados em álgebra linear (via equações normais) ou métodos iterativos como o gradiente descendente, que será explorado na próxima seção.

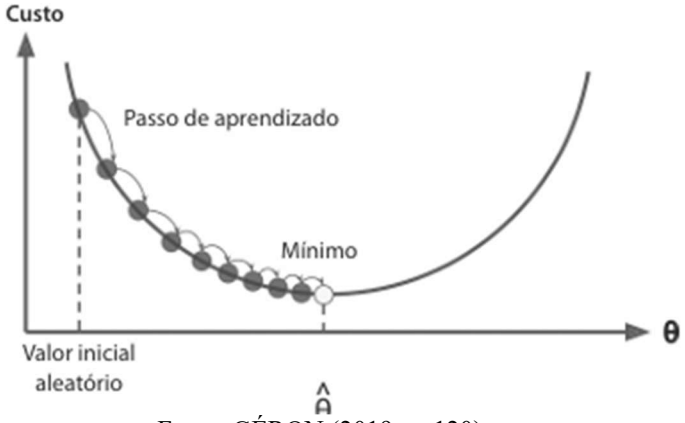
A escolha da função de custo adequada é essencial, pois influencia diretamente o comportamento do modelo durante o treinamento e sua capacidade de generalização. No contexto da regressão linear, o MSE é preferido por sua simplicidade matemática e propriedades analíticas bem definidas.

2.2 Gradiente Descendente

O gradiente descendente é um algoritmo de otimização amplamente utilizado no treinamento de modelos de regressão linear. Sua função é encontrar os valores ótimos dos parâmetros β que minimizam a função de custo do modelo, geralmente o Erro Quadrático Médio (MSE). Em termos simples, trata-se de um processo iterativo que ajusta os coeficientes do modelo com base na inclinação da função de custo — o gradiente — na tentativa de descer em direção ao ponto mais baixo da curva, ou seja, o mínimo global (GÉRON, 2019).

Géron (2019) ilustra o funcionamento do gradiente descendente com a metáfora de estar perdido em uma montanha coberta por neblina: sem enxergar o vale, você sente o terreno sob seus pés e caminha sempre na direção da descida mais íngreme. De maneira análoga, o algoritmo ajusta os parâmetros em direção à menor inclinação local da função de custo. O processo continua até que o gradiente seja praticamente nulo — indicando que um mínimo foi alcançado.

Figura 01 – Gradiente Descendente



Fonte: GÉRON (2019, p. 120).

A atualização dos parâmetros é feita por meio da seguinte fórmula:

$$\boldsymbol{\beta} \coloneqq \boldsymbol{\beta} - \eta \cdot \nabla_{\boldsymbol{\beta}} J(\boldsymbol{\beta}) \tag{5}$$

Em que:

- η é a taxa de aprendizado (*learning rate*), um hiperparâmetro que determina o tamanho dos passos,
- $\nabla_{\beta} J(\beta)$ é o vetor gradiente da função de custo com respeito aos parâmetros.

No caso da regressão linear com MSE, a derivada da função de custo é dada por:

$$\nabla_{\boldsymbol{\beta}} J(\boldsymbol{\beta}) = \frac{1}{n} \boldsymbol{X}^{\top} (\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}) \tag{6}$$

Portanto, a equação de atualização se torna:

$$\boldsymbol{\beta} \coloneqq \boldsymbol{\beta} - \eta \cdot \frac{1}{n} \boldsymbol{X}^{\top} (\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}) \tag{7}$$

Essa abordagem, conhecida como gradiente descendente em lote (batch), utiliza todo o conjunto de dados para calcular o gradiente a cada iteração. Embora isso assegure uma trajetória suave até o mínimo, o método pode ser lento quando aplicado a grandes conjuntos de dados (GÉRON, 2019).

A escolha da taxa de aprendizado (*learning rate*) é um fator crítico no treinamento de modelos de aprendizado de máquina, especialmente em algoritmos baseados em gradiente descendente. Se a taxa de aprendizado for muito pequena, o algoritmo pode convergir de forma extremamente lenta, exigindo um número excessivo de iterações para alcançar um mínimo da função de custo, o que aumenta o tempo computacional e o custo de recursos. Por outro lado, se a taxa de aprendizado for muito alta, os passos dados na direção do gradiente podem ser grandes demais, fazendo com que o algoritmo oscile em torno do mínimo ou até mesmo divirja, ultrapassando soluções ótimas e impedindo a convergência (GÉRON, 2019).

Além disso, Géron destaca que o escalonamento das características é fundamental para o bom desempenho do algoritmo. Características em escalas muito diferentes podem distorcer a superfície da função de custo, transformando a "tigela convexa" em uma "tigela alongada", o que dificulta a convergência.

Por fim, uma vantagem importante é que, como a função de custo do MSE é convexa no caso da regressão linear, o gradiente descendente tem garantia de convergir para o mínimo global, desde que a taxa de aprendizado seja bem escolhida e o número de iterações seja suficiente.

2.3 Equação Normal

A equação normal representa uma abordagem analítica direta para a estimativa dos coeficientes em modelos de regressão linear, sendo derivada da minimização da soma dos quadrados dos resíduos. Quando expressamos o modelo linear em forma matricial como $y = X\beta + \varepsilon$, onde X é a matriz de projeto e β o vetor de parâmetros, o objetivo passa a ser encontrar o vetor β que minimiza $|y - X\beta|^2$. A solução para esse problema é dada pela equação normal:
A solução para esse problema é dada pela equação normal:

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\top}\boldsymbol{X})^{-1}\boldsymbol{X}^{\top}\boldsymbol{y} \tag{8}$$

Esse método assume que $X^{\top}X$ é uma matriz inversível, o que implica ausência de multicolinearidade perfeita entre as variáveis explicativas. A vantagem principal da equação normal está na sua eficiência computacional quando o número de atributos p é relativamente pequeno, já que não envolve iteração — ao contrário do gradiente descendente, que é um método iterativo. No entanto, o custo computacional da inversão de matriz cresce com $\mathcal{O}(p^3)$, o que pode se tornar impraticável em contextos com alta dimensionalidade (GÉRON, 2019).

Além disso, a equação normal está intimamente relacionada à estimação por máxima verossimilhança no contexto da regressão linear com erros normalmente distribuídos, pois ambas produzem as mesmas estimativas pontuais para os coeficientes β. Como destacado por Géron (2019), a equação normal oferece uma solução fechada que é particularmente útil como baseline para avaliar o desempenho de métodos mais complexos, como regressão regularizada ou modelos baseados em redes neurais.

2.4 Normalização

A normalização é uma etapa fundamental no pré-processamento de dados em algoritmos de aprendizado supervisionado, especialmente em modelos que dependem de otimização iterativa como o Gradiente Descendente. Em termos gerais, ela transforma as variáveis preditoras para operarem em escalas semelhantes, evitando que atributos com magnitudes muito distintas

influenciem desproporcionalmente o comportamento do modelo (GÉRON, 2019). Neste estudo, são aplicadas duas abordagens distintas: a normalização Min-Max e a padronização Z-score, também conhecida como normalização estatística. Ambas serão descritas a seguir com maior detalhamento matemático.

2.4.1 Normalização Min-Max

A normalização Min-Max é uma técnica que consiste em reescalar os valores de uma variável para um intervalo predefinido, geralmente entre 0 e 1. A fórmula utilizada para essa transformação é:

$$x_{\text{norm}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \tag{8}$$

onde:

- x representa o valor original da variável;
- $x_{min}\ e\ x_{max}$respectivamente, os valores mínimo e máximo da variável na amostra;
- x_{norm} é o valor normalizado.

Essa técnica garante que todas as variáveis estejam na mesma escala, o que facilita a convergência dos algoritmos iterativos e permite que cada feature contribua proporcionalmente para o cálculo do gradiente. No entanto, a Min-Max é sensível a valores extremos (outliers), pois esses afetam diretamente os parâmetros $x_{min}\ e\ x_{max}$.

2.4.2 Normalização Z-score (padronização)

A padronização, também conhecida como normalização Z-score, transforma os dados de forma que passem a ter média zero e desvio padrão unitário. A equação utilizada é:

$$x_{\text{std}} = \frac{x - \mu}{\sigma} \tag{9}$$

em que:

- μ é a média da variável;
- σ é o desvio padrão da variável;
- x_{std} representa o valor padronizado.

Essa abordagem é especialmente útil quando as distribuições dos dados são aproximadamente normais, pois preserva a forma da distribuição original e reduz a influência de outliers moderados. Diferente da Min-Max, a Z-score é mais robusta a valores extremos e costuma ser preferida quando há variáveis com distribuições distintas ou dispersões muito diferentes.

Segundo Weisberg (2005), a padronização também facilita a interpretação dos coeficientes em modelos lineares, permitindo comparações entre variáveis com unidades

distintas. Além disso, ela melhora a estabilidade numérica em métodos que envolvem álgebra matricial, como a Equação Normal.

3 METODOLOGIA

Este trabalho segue uma abordagem experimental aplicada, com foco na implementação de regressão linear multivariada para predição de valores imobiliários. A pesquisa é de natureza quantitativa, estruturada em torno da análise de desempenho de diferentes técnicas de normalização e otimização.

O conjunto de dados utilizado foi ex1data2.txt, contendo variáveis referentes ao tamanho do imóvel, número de quartos e preço. Os dados foram organizados em variáveis independentes e dependentes, e submetidos a três tratamentos distintos: sem normalização, normalização Min-Max e padronização Z-score.

Foram implementadas duas abordagens de estimação dos parâmetros: Gradiente Descendente e Equação Normal. O Gradiente Descendente foi executado com diferentes taxas de aprendizado e quantidades de iterações, a depender da configuração experimental. A Equação Normal foi utilizada como base de comparação, por se tratar de uma solução analítica direta.

A experimentação envolveu testes com variados valores de iterações no Gradiente Descendente, com diferentes combinações de taxa de aprendizado, permitindo observar a sensibilidade do modelo à normalização aplicada. Para cada cenário, foram avaliados os valores de custo, a convergência dos parâmetros e as predições obtidas para uma entrada de teste.

As análises foram complementadas com visualizações gráficas, incluindo curvas de custo, superfícies da função $J(\theta)$, mapas de contorno com as trajetórias do gradiente e o plano de regressão ajustado. A implementação foi feita em Python, utilizando as bibliotecas NumPy e Matplotlib, com organização modular do código e documentação adequada para garantir reprodutibilidade.

Por fim, foram conduzidos testes de robustez com diferentes inicializações e comparações entre os métodos de normalização, a fim de avaliar a estabilidade dos resultados e a sensibilidade do modelo às variações dos hiperparâmetros.

4 RESULTADOS E DISCUSSÃO

Neste capítulo, apresentam-se e discutem-se os resultados obtidos a partir dos experimentos descritos na metodologia. São exibidos os gráficos correspondentes às análises realizadas, seguidos de interpretações que destacam as principais observações. A seção contempla a comparação entre os métodos de estimação por Equação Normal e Gradiente Descendente, bem como a influência das diferentes técnicas de normalização — min-max e z-score — no desempenho e na convergência do modelo.

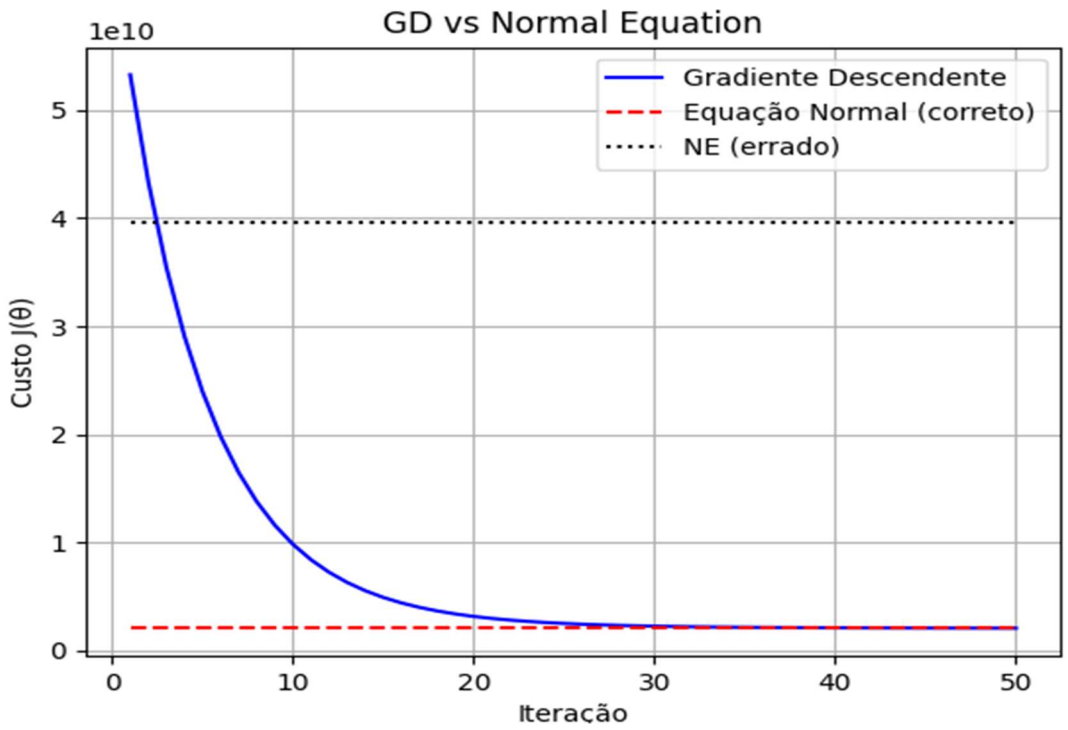
4.1 Comparação entre Gradiente Descendente e Equação Normal com Normalização Z-score.

A Figura 1 ilustra a curva de convergência do custo J(θ) ao longo de 50 iterações do Gradiente Descendente (GD) com um learning rate de 0.1, em comparação com o custo associado à Equação Normal (EN). Observa-se que o GD parte de um valor de custo extremamente elevado $\sim 6.3 \times 10^{10}$ e converge gradualmente ao longo das iterações, aproximando-se do valor ótimo após cerca de 250 iterações. Em contraste, a linha associada à EN (vermelha tracejada) permanece constante ao longo de todo o gráfico, indicando que a solução obtida por essa abordagem já parte diretamente do ponto de mínimo do custo, sem a necessidade de iterações.

Esse comportamento da Equação Normal ocorre porque ela resolve o problema de regressão linear de forma analítica, por meio da solução fechada apresentada na Equação (8), o que evita o processo iterativo de otimização presente no GD. Isso resulta em um custo imediatamente mínimo — representado por uma linha horizontal no gráfico — desde o início da comparação. A linha preta adicional representa uma tentativa incorreta de aplicar a EN sem normalização adequada, levando a um custo maior e subótimo.

Os valores de θ estimados refletem os efeitos da normalização z-score no GD, resultando em [338658.2492493 103322.82942954 -474.74249522], contra [89597.91, 139.21, −8738.02] pela EN. Apesar das diferenças numéricas entre os coeficientes, ambas as abordagens apresentaram previsões semelhantes para a entrada [1650, 3]: $292679.07 com GD e $293.081,46 com EN. Isso evidencia que os modelos estão equivalentes em desempenho preditivo, ainda que operem sobre escalas diferentes.

Figura 02 - Convergência do custo J(θ) no Gradiente Descendente em comparação com a Equação Normal

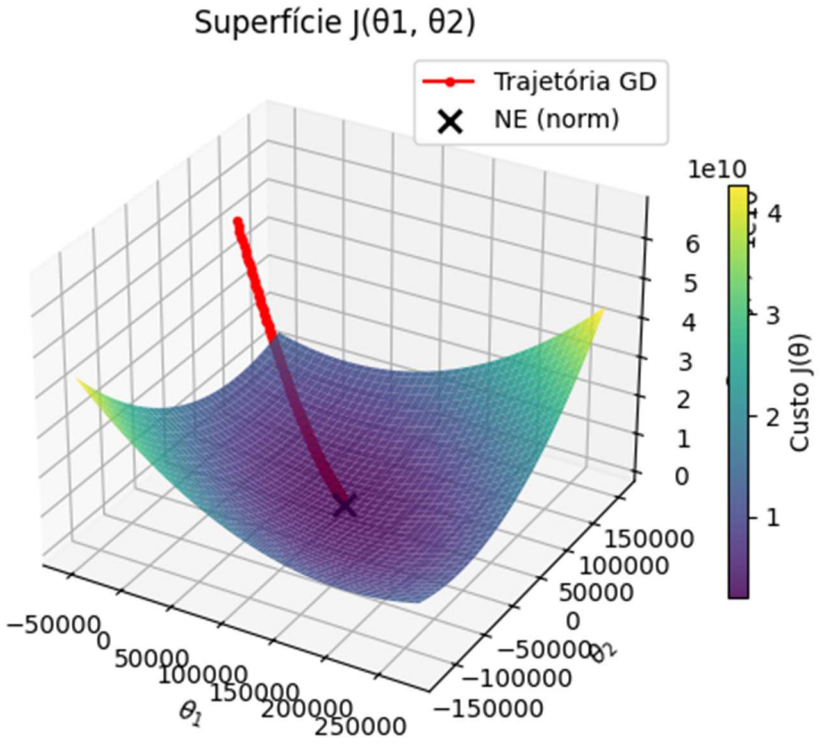


Fonte: Autor

Além da curva de convergência do custo, as Figuras 2 e 3 reforçam visualmente o comportamento do Gradiente Descendente durante o processo de otimização. A Figura 2 apresenta a superfície da função de custo $J(\theta_1, \theta_2)$ com destaque para a trajetória do Gradiente Descendente (em vermelho) em direção ao ponto ótimo (marcado com "X"), que corresponde à solução obtida pela Equação Normal após normalização. Observa-se que o algoritmo parte de um ponto elevado na superfície e percorre o vale até alcançar a região de mínimo global.

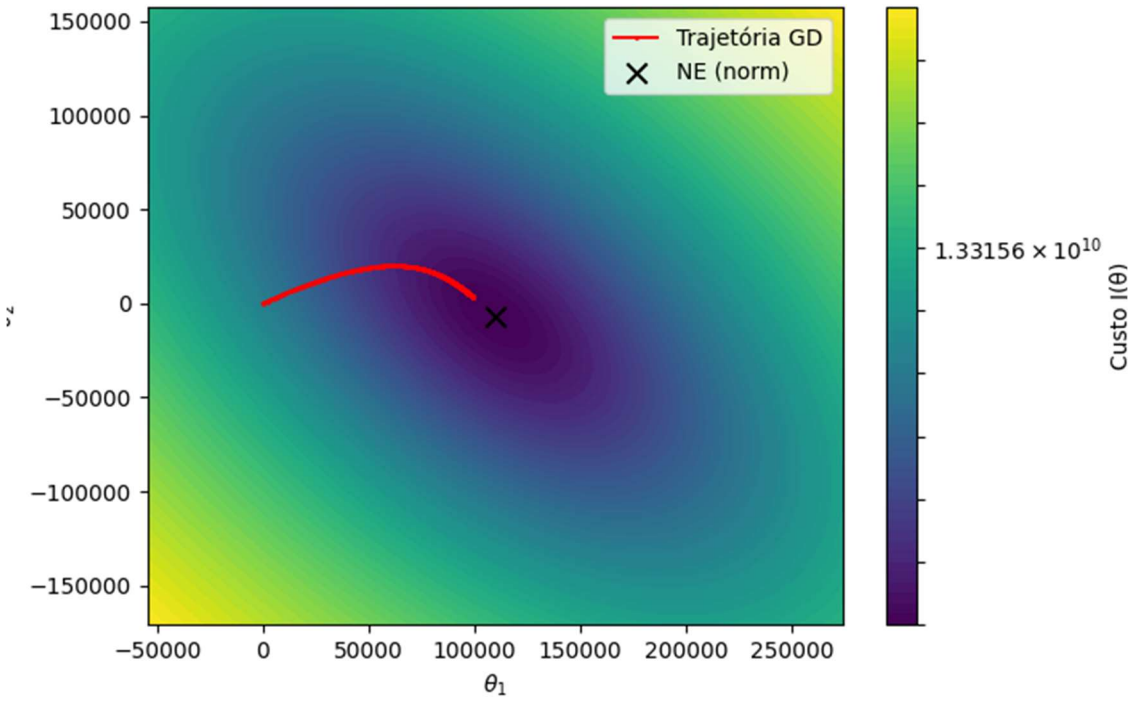
De forma complementar, a Figura 3 mostra o contorno bidimensional da mesma função de custo, com as curvas de nível representando regiões de igual valor de J(θ). A trajetória do Gradiente Descendente novamente se aproxima de forma progressiva do ponto ótimo, evidenciando a estabilidade e a eficiência da convergência quando a normalização é corretamente aplicada

Figura 03 – Superfície da função de custo J(θ1,θ2) com a trajetória do Gradiente Descendente e o ponto ótimo estimado pela Equação Normal



Fonte: Autor

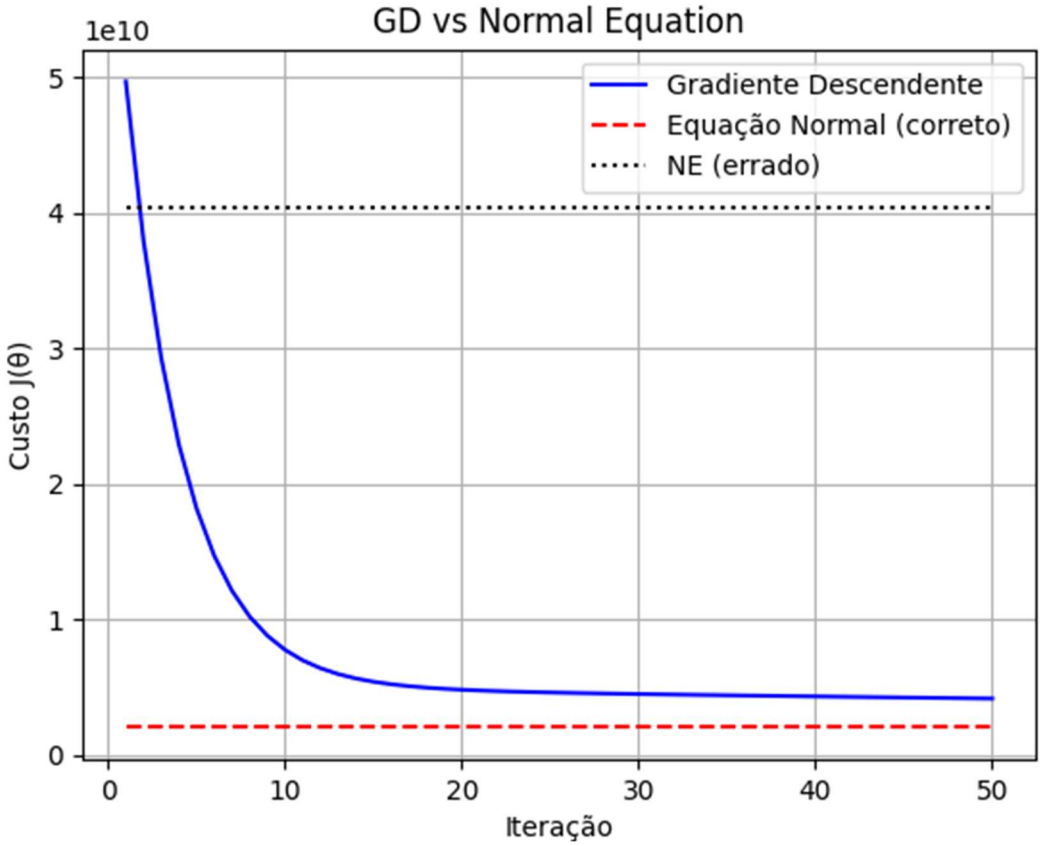
Figura 4 - Curvas de nível da função de custo J(θ1,θ2) com a trajetória do Gradiente Descendente e o ponto ótimo da Equação Normal



4.2 Comparação entre Gradiente Descendente e Equação Normal com Normalização Min-Max.

Ao implementar a normalização Min-Max com os mesmos valores de *learning rate* e número de interações usados na implementação do Z-score, verificamos que o Gradiente Descendente não converge para o ponto ótimo encontrado pela Equação Normal, como ocorria anteriormente com o Z-score implementado. A Figura 5 exemplifica esses resultados ao mostrar a convergência do custo.

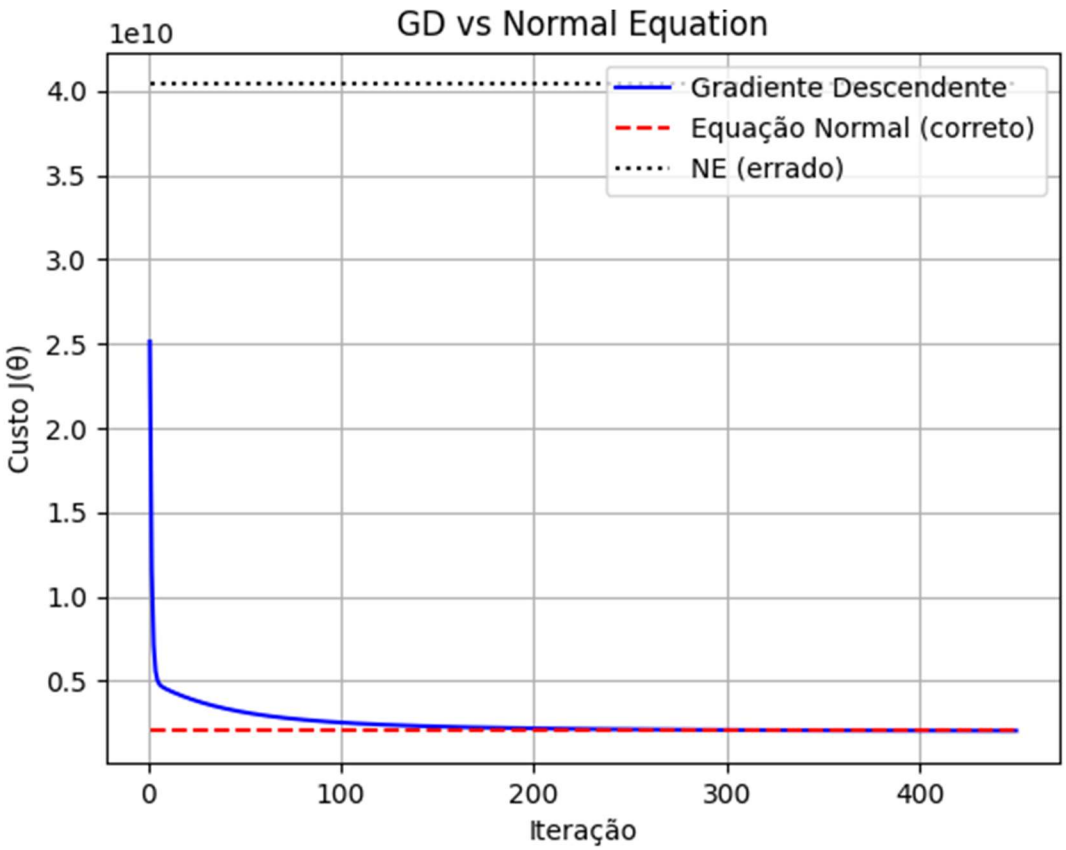
Figura 05 - Convergência do custo J(θ) no Gradiente Descendente com Min-Max em comparação com a Equação Normal



Fonte: Autor

Podemos observar que, em comparação com os resultados apresentados na Figura 2, o Gradiente Descendente, desta vez, não converge. Isso indica que os valores de hiperparâmetros utilizados não são adequados quando se aplica a normalização Min-Max a esses dados. Diante disso, foram testadas novas configurações, sendo identificada uma taxa de aprendizado de $\alpha = 0.3$ com 450 iterações como combinação eficaz, a qual permitiu a convergência do gradiente, conforme ilustra a Figura 6 a seguir.

Figura 06 - Convergência do custo J(θ) no Gradiente Descendente convergido com Min-Max em comparação com a Equação Normal

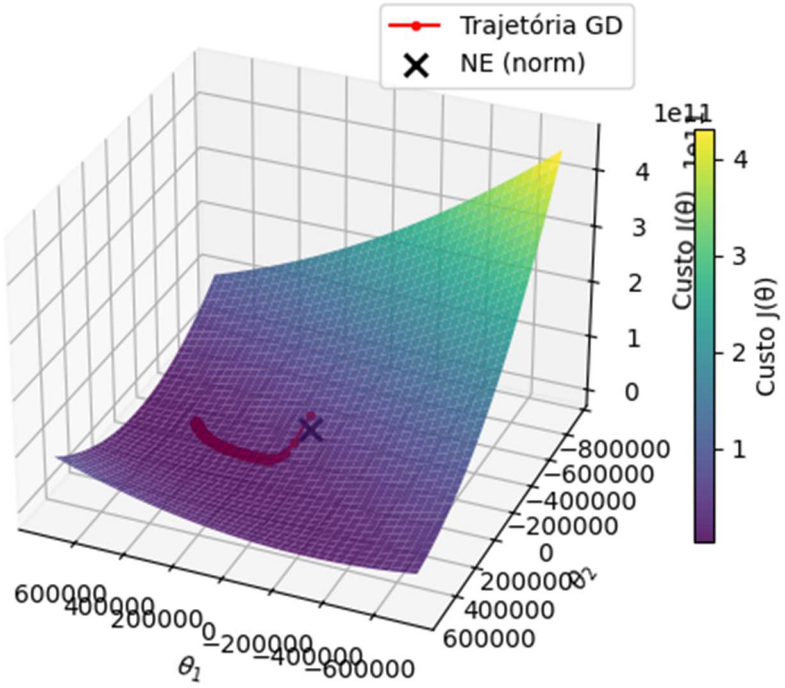


Fonte: Autor

Para essa configuração, foram encontrados os seguintes valores para θ: $[189807{,}839;\ 483637{,}316;\ -5089{,}189]$, com uma previsão para a entrada $[1650, 3]$ de $\$\ 293.700{,}80$ — valor bastante próximo daquele estimado pela Equação Normal. Assim como observado na implementação com Z-score, essa proximidade evidencia que o Gradiente Descendente, quando corretamente ajustado, é capaz de alcançar resultados compatíveis com a solução analítica, mesmo utilizando a normalização Min-Max. Para complementar esses resultados, o gráfico de cortorno é apresentado a seguir.

Nas figura, observa-se que, apesar da geometria mais alongada da superfície de custo provocada pela normalização Min-Max, o Gradiente Descendente consegue atingir a região do mínimo global após ajustes adequados de hiperparâmetros. A trajetória apresentada confirma que o algoritmo parte de uma região elevada da função de custo e, mesmo enfrentando curvaturas mais acentuadas em algumas direções, converge para o ponto ótimo identificado pela Equação Normal. Esse resultado reforça a importância do tuning da taxa de aprendizado quando se utilizam técnicas de normalização que não centralizam os dados.

Figura 07 – Superfície da função de custo J(θ1,θ2) com a trajetória do Gradiente Descendente com Min-Max e o ponto ótimo estimado pela Equação Normal



Fonte: Autor

Comparando visualmente as superfícies de custo geradas com as diferentes técnicas de normalização, nota-se que a aplicação do Z-score resulta em uma curvatura mais simétrica e convexa da função J(θ), facilitando a descida do gradiente. Já a superfície correspondente à normalização Min-Max apresenta uma forma mais achatada e alongada, com contornos menos circulares, o que exige um ajuste mais preciso dos hiperparâmetros para que o Gradiente Descendente consiga convergir. Essa diferença geométrica está diretamente relacionada à centralização dos dados promovida pela padronização Z-score, ausente na técnica Min-Max.

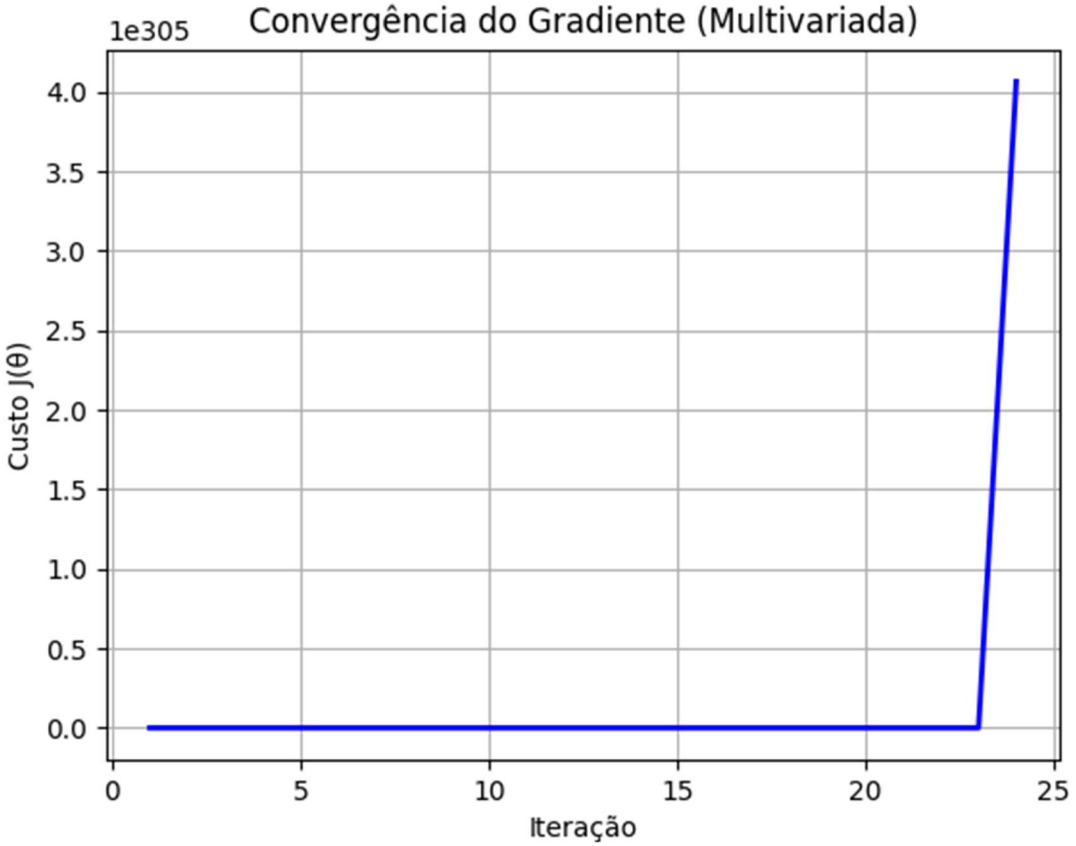
4.3 Comparação entre Gradiente Descendente e Equação Sem Normalização.

Ao tentar utilizar os mesmos valores de hiperparâmetros adotados nos experimentos com Min-Max (α = 0,3 e 450 iterações) em dados não normalizados, observou-se que o Gradiente Descendente não converge para o ponto mínimo, mas sim apresenta um comportamento de divergência. Esse resultado pode ser explicado com base na literatura, pois, segundo Goodfellow, Bengio e Courville (2016), a ausência de normalização nos dados pode ocasionar explosões ou desaparecimento do gradiente, especialmente quando os atributos possuem escalas muito diferentes entre si.

Isso ocorre porque os gradientes, ao serem multiplicados por entradas de grande magnitude, geram atualizações de pesos excessivamente amplas — o que pode causar erros de overflow e instabilidade no treinamento. A figura a seguir ilustra esse fenômeno, apresentando a explosão da

curva de custo ao longo das iterações, o que reforça a importância da normalização no pré-processamento de dados para métodos iterativos.

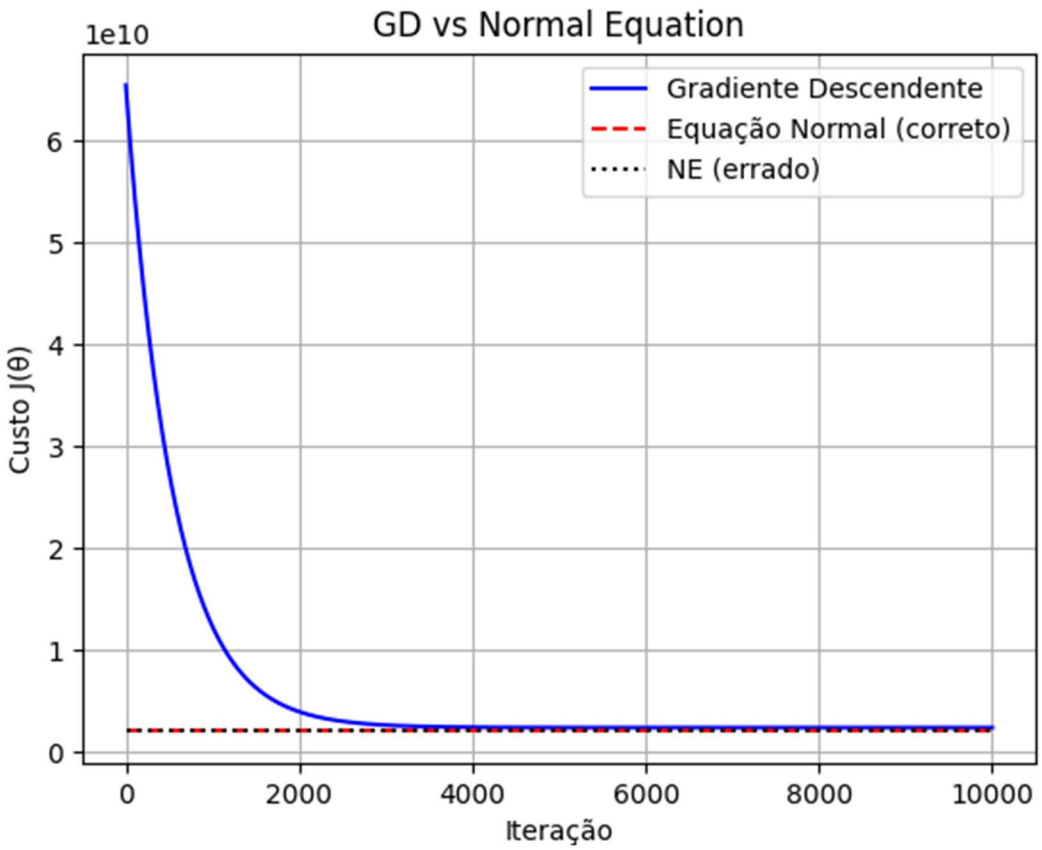
Figura 08 - Explosão do Custo no Gradiente Descendente sem Normalização



Fonte: Autor

Na Figura 08, observa-se que o custo salta para um valor extremamente elevado, caracterizando uma explosão de gradiente, causada pela inadequação da taxa de aprendizado e do número de iterações ao se trabalhar com dados não normalizados. Para mitigar esse problema, foram testadas novas configurações, sendo encontrados os valores de $\alpha = 2 \times 10^{-10}$ e 10.000 iterações como uma combinação que apresentou resultados relativamente satisfatórios. Destaca-se que a taxa de aprendizado é extremamente baixa, justamente para evitar atualizações abruptas dos pesos que poderiam comprometer a estabilidade do treinamento. A Figura 09 apresenta a curva de convergência obtida com esses novos hiperparâmetros, ilustrando a melhora no comportamento do Gradiente Descendente em um cenário sem normalização.

Figura 09 - Convergência do custo J(θ) no Gradiente Descendente convergido com Min-Max em comparação com a Equação Normal



Fonte: Autor

Observa-se que o custo se aproxima do valor mínimo encontrado pela Equação Normal, porém sem alcançá-lo completamente. Houve um ponto em que reduções adicionais na taxa de aprendizado **(α)** já não produziam efeitos significativos, uma vez que os passos tornaram-se extremamente lentos. Nessa configuração, obteve-se um valor predito para a entrada [1650, 3] de R$ 272.856,18 — relativamente próximo da previsão realizada pela Equação Normal (R$ 293.081,46), mas ainda inferior ao desempenho observado nos experimentos com normalização. Como discutido anteriormente, tanto a padronização Z-score quanto a normalização Min-Max, quando acompanhadas de hiperparâmetros bem ajustados, proporcionaram resultados substancialmente mais próximos da solução analítica, evidenciando a importância do pré-processamento dos dados para garantir a eficiência do Gradiente Descendente..

4.3 Discussão Geral dos Resultados.

Após a análise de todos os experimentos realizados, conclui-se que os testes com normalização Z-score foram os mais satisfatórios entre os três cenários avaliados: Z-score, Min-Max e ausência de normalização. Essa superioridade está relacionada à menor quantidade de iterações necessárias para alcançar a convergência e à maior estabilidade do processo de otimização.

O bom desempenho do Z-score deve-se ao fato de que essa técnica centraliza os dados, resultando em uma superfície de custo mais simétrica e convexa, conforme ilustrado na Figura 03. Essa geometria favorece a trajetória do Gradiente Descendente, permitindo uma descida mais

eficiente até o ponto de mínimo. Por outro lado, a normalização Min-Max, ao não centralizar os dados, gera uma superfície mais alongada e inclinada, o que dificultou a convergência e exigiu o uso de taxas de aprendizado maiores e um número maior de iterações.

Ao não aplicar normalização, obteve-se o pior desempenho entre todos os experimentos realizados. A necessidade de utilizar uma taxa de aprendizado extremamente baixa dificultou a convergência do Gradiente Descendente, exigindo o maior número de iterações (10.000) para alcançar um resultado aceitável. Diante disso, conclui-se que a padronização Z-score foi a técnica de normalização mais eficiente para os dados analisados, seguida pela normalização Min-Max. Já a ausência de normalização mostrou-se a abordagem menos eficaz e, portanto, não é recomendada em contextos semelhantes que envolvam algoritmos iterativos de otimização.

Ao não aplicar normalização, obteve-se o pior desempenho entre todos os experimentos realizados. A necessidade de utilizar uma taxa de aprendizado extremamente baixa dificultou a convergência do Gradiente Descendente, exigindo o maior número de iterações (10.000) para alcançar um resultado aceitável. Diante disso, conclui-se que a padronização Z-score foi a técnica de normalização mais eficiente para os dados analisados, seguida pela normalização Min-Max. Já a ausência de normalização mostrou-se a abordagem menos eficaz e, portanto, não é recomendada em contextos semelhantes que envolvam algoritmos iterativos de otimização.

Referências

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. **An introduction to statistical learning: with applications in R**. Nova York: Springer, 2013.

WEISBERG, S. **Applied linear regression**. 4. ed. Nova York: Wiley, 2014.

GÉRON, Aurélien. Mãos à obra: aprendizado de máquina com Scikit-Learn, Keras e TensorFlow: conceitos, ferramentas e técnicas para construir sistemas inteligentes. 2. ed. Rio de Janeiro: Alta Books, 2019.