

Gradiente Descendente

João Pedro Paiva
Marcos Paulo Sousa Santos
Fernanda Rafaela dos Santos

June 2023

1 Gradiente Descendente

O gradiente descendente é um algoritmo de otimização cujo objetivo é encontrar o mínimo global de uma função de custo, ajustando os parâmetros de um modelo na direção em que a função está decrescendo mais rapidamente. O algoritmo atualiza os valores dos parâmetros em passos determinados pelo tamanho do learning rate (taxa de aprendizado) e pela magnitude do gradiente em cada ponto.

O processo continua até que a função de custo atinja um mínimo global ou um critério de parada seja atingido.

$$x_{t+1} = x_t - \gamma \nabla f(x_t)$$

O algoritmo deste trabalho tem o objetivo de, por meio do gradiente descendente, encontrar a parábola que melhor se aproxima dos pontos presentes no conjunto de treinamento a seguir:

$$[(1, 4), (-1, 5), (2, 7), (-2, 8), (3, 12), (-3, 13), (4, 19), (-4, 20)]$$

O gradiente $\nabla f(xt)$ é um vetor composto pelas derivadas parciais da função objetivo em relação a cada dimensão de xt . Essas derivadas indicam a direção e magnitude do maior crescimento da função em cada dimensão. Subtraindo $\gamma \nabla f(xt)$ de xt , o algoritmo dá um passo em direção ao mínimo local, reduzindo gradualmente o valor da função.

Esse processo é repetido até que uma condição desejada seja alcançada, como um número máximo de iterações ou quando o erro for próximo o suficiente de zero.

A seguir estão as derivadas mencionadas, obtidas à mão:

$$du = dx$$

$$E(w_1, w_2, b) = \sum_{i=1}^n (y_i - \hat{y})^2 \quad \hat{y} = w_1 x^2 + w_2 x + b$$

$$\frac{\partial E(w_1, w_2, b)}{\partial w_1} = \sum_{i=1}^n 2(y_i - \hat{y}) \cdot (y_i - \hat{y})_{w_1}$$

$$= \sum_{i=1}^n 2(y_i - \hat{y}) \cdot (x^2) \rightarrow -2 \sum_{i=1}^n (y_i - \hat{y}) x_i^2$$

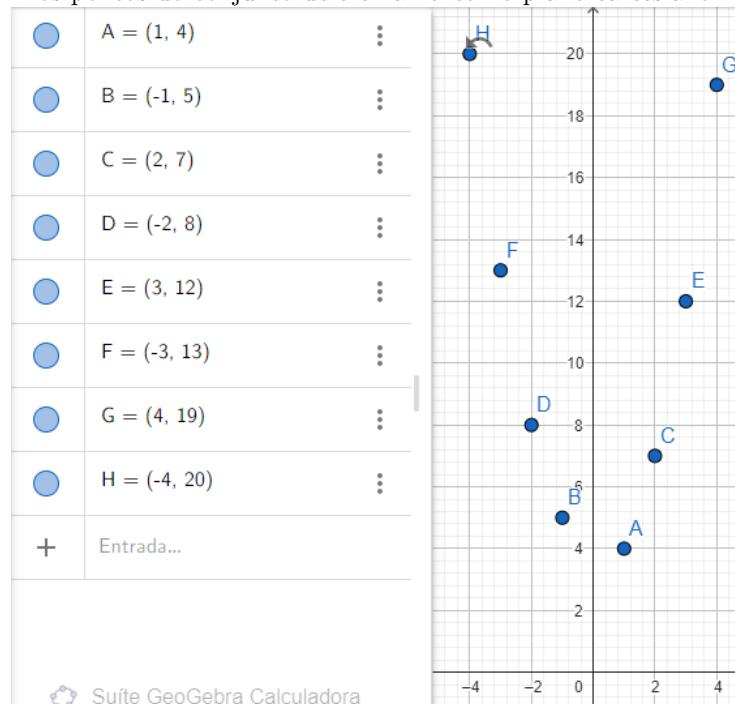
$$\frac{\partial E(w_1, w_2, b)}{\partial w_2} = \sum_{i=1}^n 2(y_i - \hat{y}) \cdot (y_i - \hat{y})_{w_2}$$

$$= \sum_{i=1}^n 2(y_i - \hat{y}) \cdot (x) \rightarrow -2 \sum_{i=1}^n (y_i - \hat{y}) x_i$$

$$\frac{\partial E(w_1, w_2, b)}{\partial b} = \sum_{i=1}^n 2(y_i - \hat{y}) \cdot (y_i - \hat{y})_b$$

$$= \sum_{i=1}^n -2(y_i - \hat{y}) \rightarrow -2 \sum_{i=1}^n (y_i - \hat{y})$$

E os pontos do conjunto de treinamento no plano cartesiano:



Rodando o algoritmo com os parâmetros (valores iniciais = (2,2,2), learning rate = 10^{-3} , número máximo de vezes que vai rodar (epochs)= 10^5 , x, y, gradiente, tolerância = 10^{-6}), obtemos:

$[1.014492660398363, -0.16666666666666655, 3.3296857288670823]$

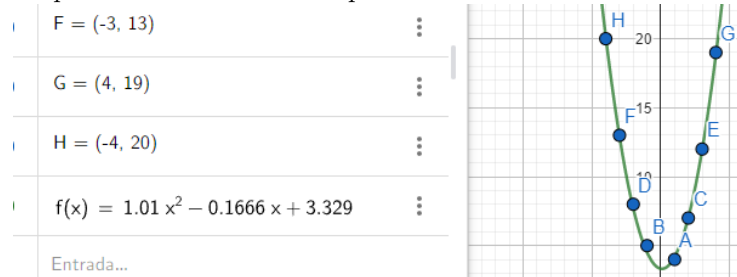
$w1 = 1.014492660398363$

$w2 = -0.16666666666666655$

$b = 3.3296857288670823$

Então, a parábola é $\hat{y} = 1.01x^2 - 0.1666x + 3.329$

E aplicando esses valores no plano cartesiano:



2 Github

Link do repositório: <https://github.com/namespaiva/calc-gradiente>