# Gradient Descent and Least Squares in Linear Regression

Sidharth Rao

February 27, 2022

## 1 Introduction

Least Squares is a method of Linear Regression which involves finding the squared distance of a set of points from a graph, and then varying the slope and y-intercept (weight and bias) of a line in order to minimize the squared distance and find a line of best fit.

Gradient Descent is a concept in Machine Learning in which a set of weights and biases are changed in order to minimize loss. In the case of Linear Regression, the loss function in Gradient Descent incorporates the Least Squares method.

## 2 Derivations

### 2.1 Least Squares

The Least Squares formula: Let $(x_1, y_1)$, $(x_2, y_2) \ldots$, $(x_n, y_n)$ be a set of points to find a line of regression for and let $w$ and $b$ represent the weight and bias of a linear equation. Let

$$SE = \sum_i^n (wx_i + b - y_i)^2$$

denote the error of the linear equation. As $SE$ approaches 0, the accuracy of the linear equation increases. Therefore, we need to minimize $SE$.

$$\frac{\partial SE}{\partial w} = 0$$

$$2w \sum_i^n x_n^2 + 2b \sum_i^n x_n - 2 \sum_i^n x_n y_n = 0$$

$$w \sum_i^n x_n^2 = \sum_i^n x_n y_n - b \sum_i^n x_n$$

$$\frac{\partial SE}{\partial b} = 0$$

$$2bn + 2w \sum_i^n x_n - 2 \sum_i^n y_n = 0$$

$$b = \frac{\sum_i^n y_n - w \sum_i^n x_n}{n}$$

Plugging in b gets us:

$$w = \frac{n \sum_i^n x_n y_n - \sum_i^n x_n \sum_i^n y_n}{n \sum_i^n x_n^2 - (\sum_i^n x_n)^2}$$

Using the w and b above, a line of best fit can be found using least squares in 2 dimensions.

### 2.2 Gradient Descent

The Gradient Descent formula: Let $(x_1, y_1)$, $(x_2, y_2) \ldots$, $(x_n, y_n)$ be a set of points to find a line of regression for and let $w$ and $b$ represent the weights (plural) and bias of a linear equation. Let

$$MSE = \frac{1}{n} \sum_i^n (w_1(x_1)_i + w_2(x_2)_i + ... + w_m(x_m)_i + b - y_i)^2$$

denote the Mean Squared Error of the linear equation. As $MSE$ approaches 0, the accuracy of the linear equation increases. Therefore, we need to minimize $MSE$.
Rather than fully calculating the optimization, Gradient Descent is used in the field of Computer Science to get highly accurate approximations of the weights and biases with the least loss. The first step is to find the gradient of MSE.

$$\frac{\partial SE}{\partial w_m} = \sum_i^n 2x(w_1(x_1)_i + ... + w_m(x_m)_i + b - y_i) \quad \frac{\partial SE}{\partial b} = \sum_i^n (w_1(x_1)_i + ... + w_m(x_m)_i + b - y_i)$$

$\frac{\partial SE}{\partial w_m}$ applies to each weight.
Remember that the directional derivative of a function is maximized when the gradient is in the same direction as the unit vector. To find the greatest decrease, we just find the gradient in the opposite direction. So, subtract the gradient from the weights and bias to make sure that the loss function always moves towards the minimum loss.[Aca21]. At each iteration of the gradient descent process, we follow this process:

$$w_m = w_m - \alpha \frac{\partial SE}{\partial w_m} \qquad\qquad b = b - \alpha \frac{\partial SE}{\partial b}$$

where $\alpha$ is the learning rate, multiplied by the gradient to make sure that the function does not move too far beyond the minimum loss by changing weights too fast.

### 2.3 Gradient Descent in 2 Dimensions

Running my gradient descent code on one weight with an $\alpha$ of .1 and 3000 epochs (iterations) resulted in values similar to the ones below:

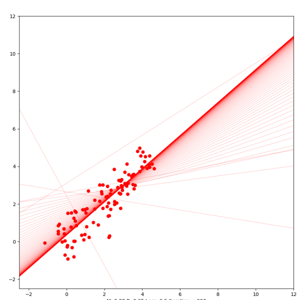


Figure 1: y=.88x+.35 (Loss = .5)

### 2.4 Gradient Descent in 3 dimensions

Running my gradient descent code on one weight with an $\alpha$ of .01 and 1500 epochs (iterations) resulted in values similar to the ones below:

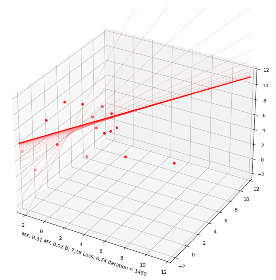

Figure 2: z=.31x+.02y+7.18 (Loss = 6.74)

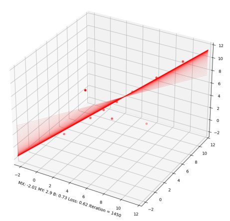

Figure 3: z=-2.01x+2.9y+.73 (Loss = .62)

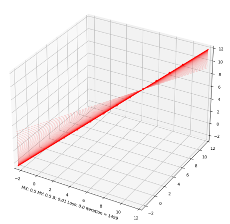

Figure 4: z=-.5x+.5y+.01 (Loss = .00 (Rounded))

### 2.5 Links

You can access my code at:
https://github.com/sidharthmrao/SVM
You can access my slideshow at:
https://docs.google.com/presentation/d/1qEJGzKYBfBNYXTz2mjPXXkYq3ONve
FTfCSqscYyvdA8/edit?usp=sharing

## References

[Aca21] Khan Academy. Gradient Descent. `https://tinyurl.com/3j3tercf`, 2021. [Online; accessed 02/23/2022. Used for basics, all mathematics and code is my own work.].

*On my honor, I have neither given nor received unauthorized aid on this assessment.*