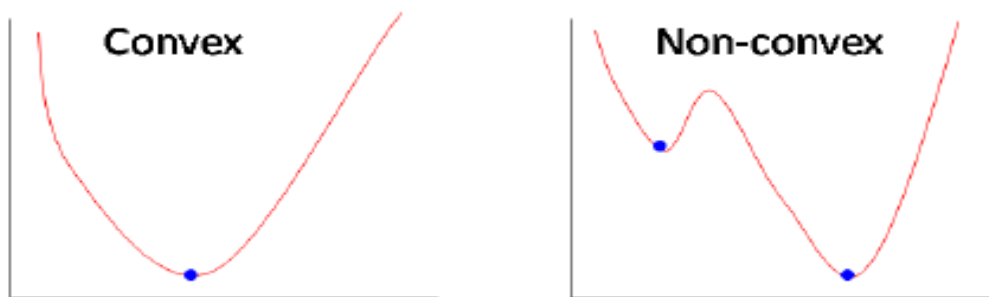


Gradient Descent Intuition

Gradient Descent is the iterative algorithm that is used by various machine learning algorithms to learn about the data especially the algorithms that applies regression in any form in their core mathematically speaking it uses some cool differential calculus to find the minima of a convex function, while if the function is non-convex, various techniques are applied to find the global minima as in this case it is very likely that the learning stops at any local minima or a saddle point, but those concepts are for later understanding as of now the idea here is to get a basic understanding of how we make machines learn about regression tasks using gradient descent.

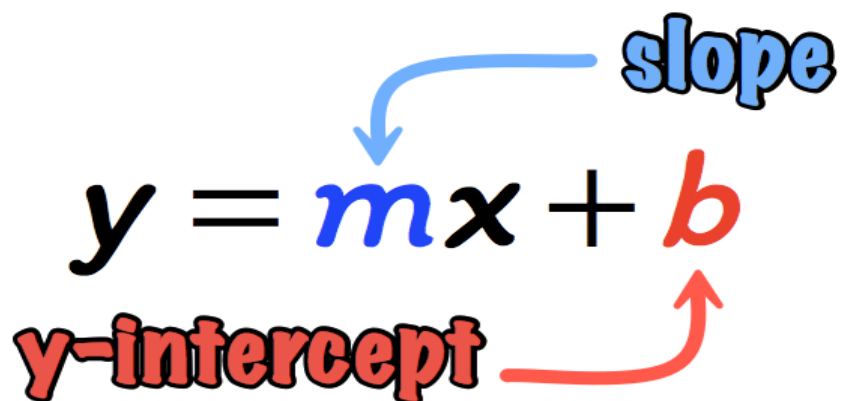


Basically, the function we take for this is called the cost function, and we try to reduce the cost function by taking partial derivatives of the cost function with respect to each of

the independent features in the function, these partial derivatives state the change in the cost function with respect to the change in the respective features after this we deduct these partial derivatives from the respective features but before that the partial derivative is multiplied with a hyperparameter called the learning rate which helps in controlling the rate of the deductions so that it is neither too high nor too low but close enough to converge at the global minima, if the deductions is too high it would miss the global minima while if the deductions is too low it won't converge at the global minima.

Let's take up linear regression problem for simplicity and understand how it works

This is the equation of line that we try to fit in a linear regression problem.



The diagram shows the linear regression equation $y = mx + b$. The variable m is colored blue, and the variable b is colored red. A blue arrow points from the word "slope" to the blue m . A red arrow points from the word "y-intercept" to the red b .

$$y = mx + b$$

Here “m” and “b” are the independent variables slope and y-intercept which is used to calculate the predicted values with the correct values of “m” and “b”.

Here the cost-function is mean-squared error

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$

Here “ y_i ” is the actual values while “ \tilde{y}_i ” is the predicted values and “ n ” is the sample size.

Now we substitute the value of “ \tilde{y}_i ” in the function and the function becomes like this:

$$f(m, b) = \frac{1}{n} \sum_{i=1}^n (y_i - (mx_i + b))^2$$

Now we take partial derivative w.r.t the independent features that is “ m ” and “ b ”, and we get:

$$\frac{\partial}{\partial \mathbf{m}} = \frac{2}{N} \sum_{i=1}^N -x_i (y_i - (mx_i + b))$$

$$\frac{\partial}{\partial \mathbf{b}} = \frac{2}{N} \sum_{i=1}^N -(y_i - (mx_i + b))$$

Now the independent features are updated as:

$$m = m - \text{learning rate} * d/dm$$

$$b = b - \text{learning rate} * d/db$$

Until the cost function converges at the global minimum, the corresponding values of “m” and “b” are the correct value that will be used in the equation of the line $y = mx + b$ for the predictions.