**Regression Techniques**

**What is regression?**

Regression analysis is the fundamental concept of machine learning. It falls under [supervised learning](https://builtin.com/machine-learning/supervised-learning) wherein the algorithm is trained with both input features and output labels. It helps in establishing a relationship among the variables by estimating how one variable affects the other.

**Definition:** Regression in machine learning consists of mathematical methods that allow data scientists to predict a continuous outcome (y) based on the value of one or more predictor variables (x).

Regression is used to finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables. It is mainly used for **prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables**.

In Regression, we plot a graph between the variables which best fits the given datapoints, using this plot, the machine learning model can make predictions about the data. In simple words, **"Regression shows a line or curve that passes through all the datapoints on target-predictor graph in such a way that the vertical distance between the datapoints and the regression line is minimum."** The distance between datapoints and line tells whether a model has captured a strong relationship or not.

Some examples of regression can be as:

* Prediction of rain using temperature and other factors
* Determining Market trends
* Prediction of road accidents due to rash driving.

**Why we used regression analysis?**

Regression analysis helps in the prediction of a continuous variable. There are various scenarios in the real world where we need some future predictions such as weather condition, sales prediction, marketing trends, etc., for such case we need some technology which can make predictions more accurately. So, for such case we need Regression analysis which is a statistical method and used in machine learning and data science.

By performing the regression, we can confidently determine the **most important factor, the least important factor, and how each factor is affecting the other factors**.

**What is Linear Regression?**

Linear regression is a type of statistical analysis used to predict the relationship between two variables. It assumes a linear relationship between the independent variable and the dependent variable, and aims to find the best-fitting line that describes the relationship. The line is determined by minimizing the sum of the squared differences between the predicted values and the actual values.

**Definition:** Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.

If there is only one input variable (x), then such linear regression is called **simple linear regression**. And if there is more than one input variable, then such linear regression is called **multiple linear regression**.

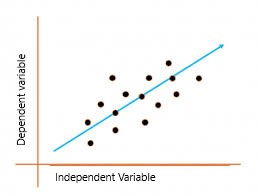
Some popular applications of linear regression are:

* Analyzing trends and sales estimates
* Salary forecasting
* Real estate prediction
* Arriving at ETAs in traffic.

**Simple Linear Regression**

Simple Linear regression is a quiet and the simplest statistical regression method used for predictive analysis in machine learning. In a simple linear regression, there is one independent variable and one dependent variable.

The model estimates the slope and intercept of the line of best fit, which represents the relationship between the variables. The slope represents the change in the dependent variable (Y axis) for each unit change in the independent variable (X axis), while the intercept represents the predicted value of the dependent variable when the independent variable is zero.



The above graph presents the linear relationship between the output(y) variable and predictor(X) variables.  The line is referred to as the*best fit* straight line.

To calculate best-fit line, linear regression uses a traditional slope-intercept form

**Yi= β0 + β1Xi**

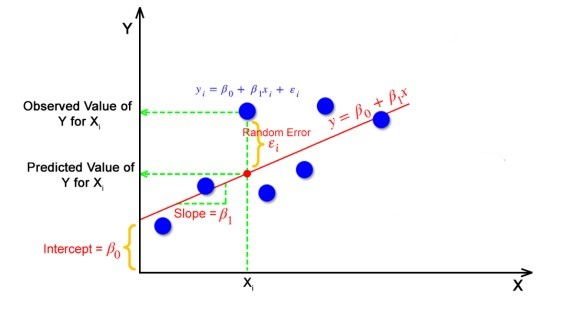
Where, **Yi** = Dependent variable,

**β0** = constant/Intercept,

**β1** = Slope/Intercept,

**Xi** = Independent variable.

This algorithm explains the linear relationship between the dependent(output) variable y and the independent(predictor) variable X using a straight-line Y= B0 + B1 X.



The goal of the linear regression algorithm is to get the **best values for B0 and B1** to find the best fit line. The best fit line is a line that has the least error which means the error between predicted values and actual values should be minimum.

**Random Error (Residuals)**

In regression, the difference between the observed value of the dependent variable(**yi**) and the predicted value(**predicted**) is called the residuals.

**εi=** **ypredicted** –   **yi**

**where ypredicted =   B0 + B1 Xi**

**Best fit line**

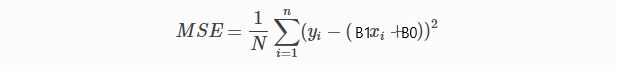
In simple terms, the best fit line is a line that fits the given scatter plot in the best way. Mathematically, the best fit line is obtained by minimizing the Residual Sum of Squares (RSS).

**Cost Function for Linear Regression**

The [cost function](https://www.analyticsvidhya.com/blog/2021/03/data-science-101-introduction-to-cost-function/) helps to find out the optimal values for B0 and B1, which provides the best fit line for the data points.

In Linear Regression, generally **Mean Squared Error (MSE)** cost function is used, which is the average of squared error that occurred between the **ypredicted** and **yi**.

MSE using simple linear equation y=mx+b:

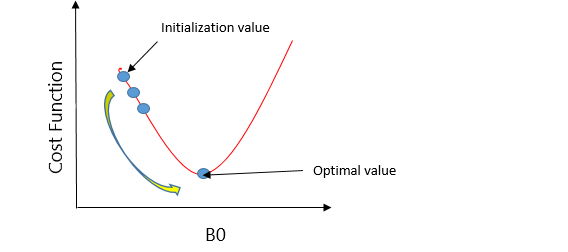


Using the MSE function, we’ll update the values of B0 and B1 such that the MSE value settles at the minima.  These parameters can be determined using the gradient descent method such that the value for the cost function is minimum.

**Gradient Descent for Linear Regression**

Gradient Descent is one of the optimization algorithms that optimize the cost function to reach the optimal minimal solution. To find the optimum solution we need to reduce the cost function (MSE) for all data points. This is done by updating the values of B0 and B1 iteratively until we get an optimal solution.

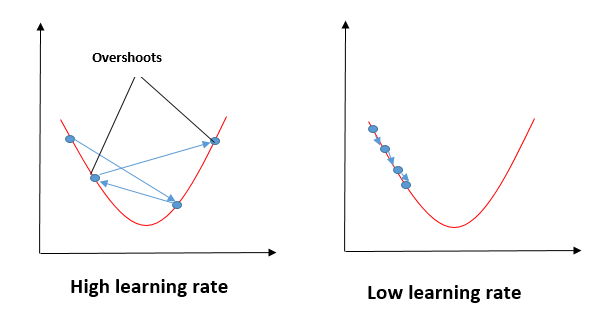
A regression model optimizes the gradient descent algorithm to update the coefficients of the line by reducing the cost function by randomly selecting coefficient values and then iteratively updating the values to reach the minimum cost function.



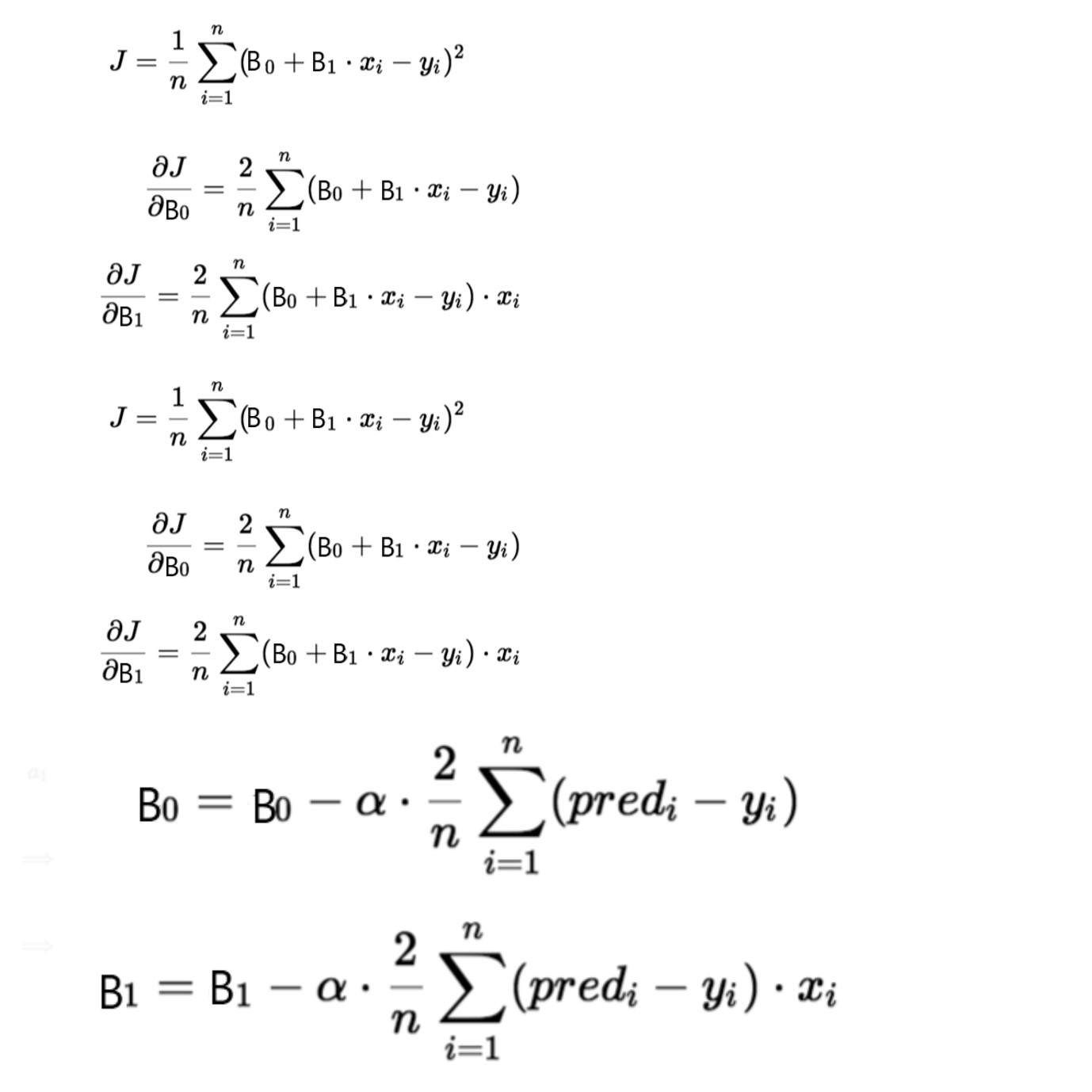
**(**Imagine a U-shaped pit. And you are standing at the uppermost point in the pit, and your motive is to reach the bottom of the pit. Suppose there is a treasure at the bottom of the pit, and you can only take a discrete number of steps to reach the bottom. If you opted to take one step at a time, you would get to the bottom of the pit in the end but, this would take a longer time. If you decide to take larger steps each time, you may achieve the bottom sooner but, there’s a probability that you could overshoot the bottom of the pit and not even near the bottom. In the gradient descent algorithm, the number of steps you’re taking can be considered as the **learning rate**, and this decides how fast the algorithm **converges** to the minima.**)**

**Learning rate** (also referred to as step size or the alpha) is the size of the steps that are taken to reach the minimum. This is typically a small value, and it is evaluated and updated based on the behaviour of the cost function. High learning rates result in larger steps but risks overshooting the minimum. Conversely, a low learning rate has small step sizes. While it has the advantage of more precision, the number of iterations compromises overall efficiency as this takes more time and computations to reach the minimum.

**The cost (or loss) function** measures the difference, or error, between actual y and predicted y at its current position. This improves the machine learning model's efficacy by providing feedback to the model so that it can adjust the parameters to minimize the error and find the local or global minimum. It continuously iterates, moving along the direction of steepest descent (or the negative gradient) until the cost function is close to or at zero. At this point, the model will stop learning. Additionally, while the terms, cost function and loss function, are considered synonymous, there is a slight difference between them. It’s worth noting that a loss function refers to the error of one training example, while a cost function calculates the average error across an entire training set.



To update B0 and B1, we take gradients from the cost function. To find these gradients, we take partial derivatives for B0 and B1.



We need to minimize the cost function J. One of the ways to achieve this is to apply the batch gradient descent algorithm. In batch gradient descent, the values are updated in each iteration. (Last two equations show the updating of values)

The partial derivates are the gradients, and they are used to update the values of B0 and B1. Alpha is the learning rate.

**Types of Gradient Descent**

There are three types of gradient descent learning algorithms: batch gradient descent, stochastic gradient descent and mini-batch gradient descent.

**Batch Gradient Descent**

**Stochastic Gradient Descent**

**Mini-batch Gradient Descent**

**Multiple Linear Regression**

Multiple linear regression is a technique to understand the relationship between a single dependent variable and multiple independent variables.

The formulation for multiple linear regression is also similar to simple linear regression with

the small change that instead of having one beta variable, you will now have betas for all the variables used. The formula is given as:

Y = B0 + B1X1 + B2X2 + … + BpXp + **ε**