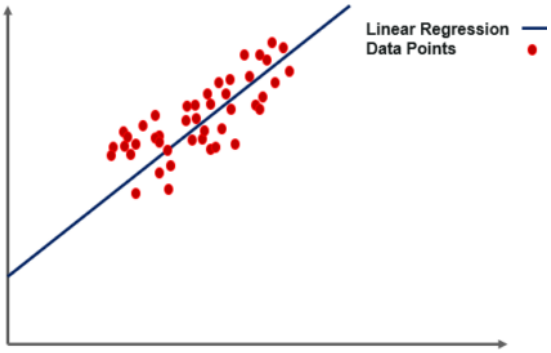
Supervised Learning – Regression - Lecture Notes

1. Regression – an overview

Linear Regression is a machine learning algorithm based on **supervised learning**. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.



Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x)). Hence, the name is Linear Regression. For example, in the figure above, X (input) is the work experience and Y (output) is the salary of a person. The regression line is the best-fit line for the model.

The formula for a simple linear regression is:

$$y = \beta_0 + \beta_1 X + \epsilon$$

- y is the predicted value of the dependent variable (y) for any given value of the independent variable (x).
- B_0 is the intercept, the predicted value of y when the x is 0.
- B_1 is the regression coefficient – how much we expect y to change as x increases.
- x is the independent variable (the variable we expect is influencing y).
- e is the error of the estimate, or how much variation there is in our estimate of the regression coefficient.

There are a range of different approaches used in machine learning to perform regression. Some of the most common regression techniques in machine learning can be grouped into the following types of regression analysis:

Simple Linear Regression

A simple straight-line equation involving slope (dy/dx) and intercept (an integer/continuous value) is utilized in simple Linear Regression. Here a simple form is: y=mx+c where y denotes the output x is the independent variable, and c is the intercept when x=0. With this equation, the algorithm trains the model of machine learning and gives the most accurate output

Multiple linear regression

When a number of independent variables more than one, the governing linear equation applicable to regression takes a different form like: y= c+m1x1+m2x2… mnxn where represents the coefficient responsible for impact of different independent variables x1, x2 etc. This machine-learning algorithm, when applied, finds the values of coefficients m1, m2, etc., and gives the best fitting line.

Logistic regression

Logistic regression is used when the dependent variable can have one of two values, such as true or false, or success or failure. Logistic regression models can be used to predict the probability of a dependent variable occurring. Generally, the output values must be binary.

Let us look at a few advantages and disadvantages of linear regression for machine learning.

Advantages	Disadvantages
Linear regression performs exceptionally well for linearly separable data	The assumption of linearity between dependent and independent variables
Easier to implement, interpret and efficient to train	It is often quite prone to noise and overfitting
It handles overfitting pretty well using dimensionally reduction techniques, regularization, and cross-validation	Linear regression is quite sensitive to outliers
One more advantage is the extrapolation beyond a specific data set	It is prone to multicollinearity

2. Linear Regression – How does it work

Linear regression uses the relationship between the data-points to draw a straight line through all them. This line can be used to predict future values.

Thus, linear regression is a supervised learning algorithm that simulates a mathematical relationship between variables and makes predictions for continuous or numeric variables such as sales, salary, age, product price, etc.

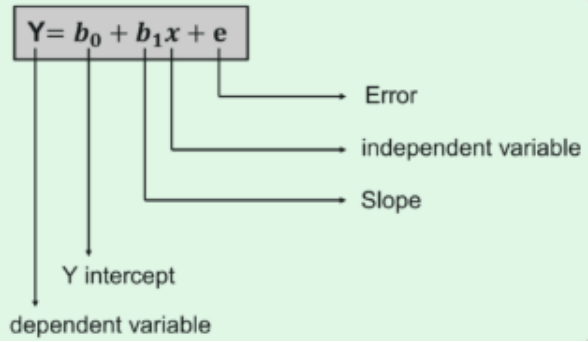
We now will consider the process of applying Linear Regression in a Machine Learning project.

3. Linear Regression Terminologies

The following terminologies are important to be familiar with before moving on to the linear regression algorithm.

Cost Function

The best fit line can be based on the linear equation given below.



The dependent variable that is to be predicted is denoted by Y.

A line that touches the y-axis is denoted by the intercept b0.

b1 is the slope of the line, x represents the independent variables that determine the prediction of Y.

The error in the resultant prediction is denoted by e.

The cost function provides the best possible values for b0 and b1 to make the best-fit line for the data points. We do it by converting this problem into a minimization problem to get the best values for b0 and b1. The error is minimized in this problem between the actual value and the predicted value.

$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

$$J = \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

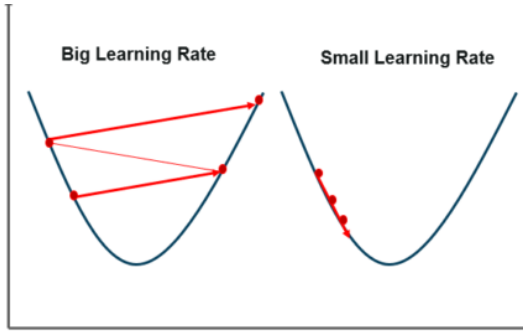
We choose the function above to minimize the error. We square the error difference and sum the error over all data points, the division between the total number of data points. Then, the produced value provides the averaged square error over all data points.

It is also known as MSE(Mean Squared Error), and we change the values of b0 and b1 so that the MSE value is settled at the minimum.

Gradient Descent

The next important terminology to understand linear regression is **gradient descent.** It is a method of updating b0 and b1 values to reduce the MSE. The idea behind this is to keep iterating the b0 and b1 values until we reduce the MSE to the minimum.

To update b0 and b1, we take gradients from the cost function. To find these gradients, we take partial derivatives with respect to b0 and b1. These partial derivatives are the gradients and are used to update the values of b0 and b1.



4. Implementing Linear Regression

The process takes place in the following steps:

1. Loading the Data
2. Exploring the Data
3. Slicing The Data
4. Train and Split Data
5. Generate The Model
6. Evaluate The accuracy

5. Linear regression in Python – Use case