Train Test Split:

The train-test split is a technique for evaluating the performance of a machine learning algorithm.

It can be used for classification or regression problems and can be used for any supervised learning algorithm.

The procedure involves taking the dataset and splitting it into 2 subsets.

- **Train Dataset**: Used to fit the machine learning model.
- **Test Dataset**: Used to evaluate the fit machine learning model.

The objective is to estimate the performance of the machine learning model on new data: data not used to train the model.

When to Use the Train-Test Split:

When the dataset is sufficiently large and each of the train and test datasets are suitable representations of the problem domain.

The train-test split evaluation procedure is computational efficiency

the train-test split procedure is used to verify the performance of the model quickly across vast data already present.

How to Configure the Train-Test Split

one main configuration parameter, which is the **size of the train and test sets**

You can split based on your project objectives:

- Computational cost in training the model.
- Computational cost in evaluating the model.
- Training set representativeness.
- Test set representativeness.

Commonly used splits:

- Train: 80%, Test: 20%
- Train: 67%, Test: 33%
- Train: 50%, Test: 50%

Stratified Train-Test Splits:

Some classification problems do not have a balanced number of examples for each class label.

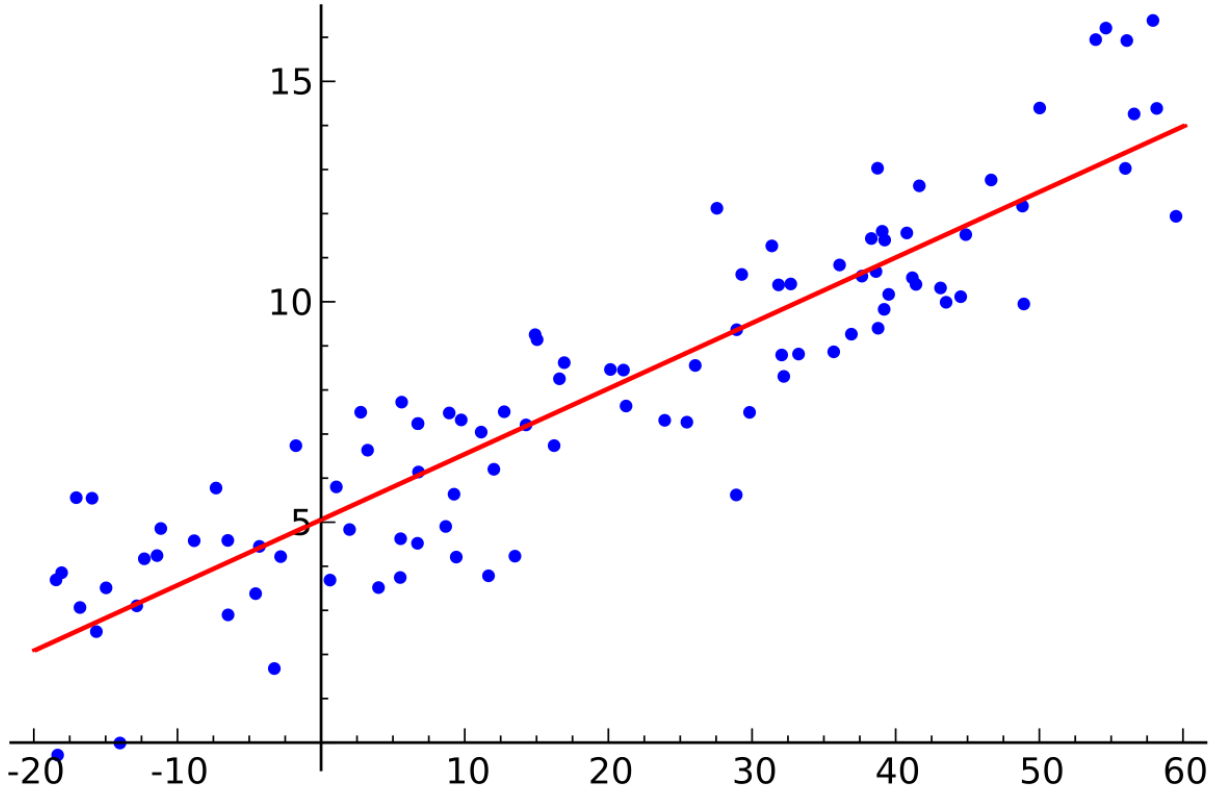
As such, it is desirable to split the dataset into train and test sets in a way that preserves the same proportions of examples in each class as observed in the original dataset.

This is called a **stratified train-test split.**

We can achieve this by setting the "***stratify***" argument to the y component of the original dataset. This will be used by the *train_test_split()* function to ensure that both the train and test sets have the proportion of examples in each class that is present in the provided "*y*" array.

Linear Regression:

Regression is a method of modeling a target value based on independent predictors. This method is mostly used for forecasting and finding out cause and effect relationships between variables. Regression techniques mostly differ based on the number of independent variables and the type of relationship between the independent and dependent variables.



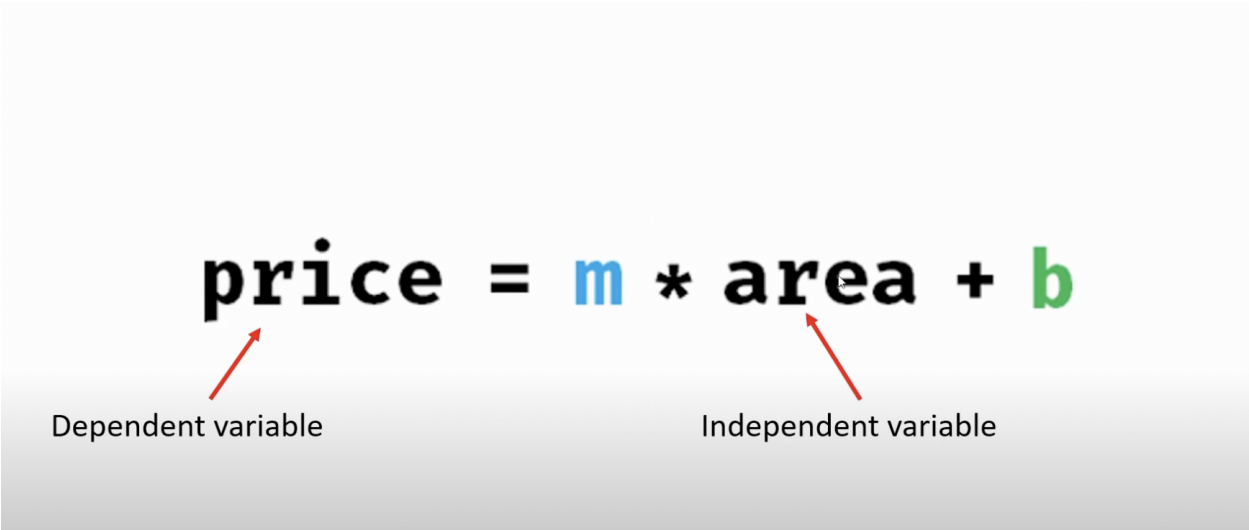
The red line in the above graph is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best. The line can be modelled based on the linear equation shown below.

A linear regression line has an equation of the form

Y = mX + c

where X is the explanatory/independent variable and Y is the dependent variable. The slope of the line is m, and c is the intercept (the value of y when x = 0).

In case of Housing price prediction:



Simple linear regression is used to estimate the relationship between two quantitative variables. You can use simple linear regression when you want to know:

How strong the relationship is between two variables (e.g. the relationship between rainfall and soil erosion).

The value of the dependent variable at a certain value of the independent variable (e.g. the amount of soil erosion at a certain level of rainfall).

Prior moving to Linear regression code/algorithm lets understand few important concepts required for Linear regression.

Correlation: explains the association among variables within the data

Variance: the degree of the spread of the data

Standard deviation: the square root of the variance

Normal distribution: a continuous probability distribution(gaussian distribution), it's sort of a bell curve in which the right side of the mean is the mirror of the left side

Cost Function

Cost function helps to find the best possible value for m and c, such that we can provide the best fit line for the data points.

Cost function measures how a machine learning model performs.

The cost function is the calculation of the error between predicted values and actual values, represented as a single real number.

The difference between the cost function and loss function is as follows:

The cost function is the average error of n-samples in the data (for the whole training data) and the loss function is the error for individual data points (for one training example).

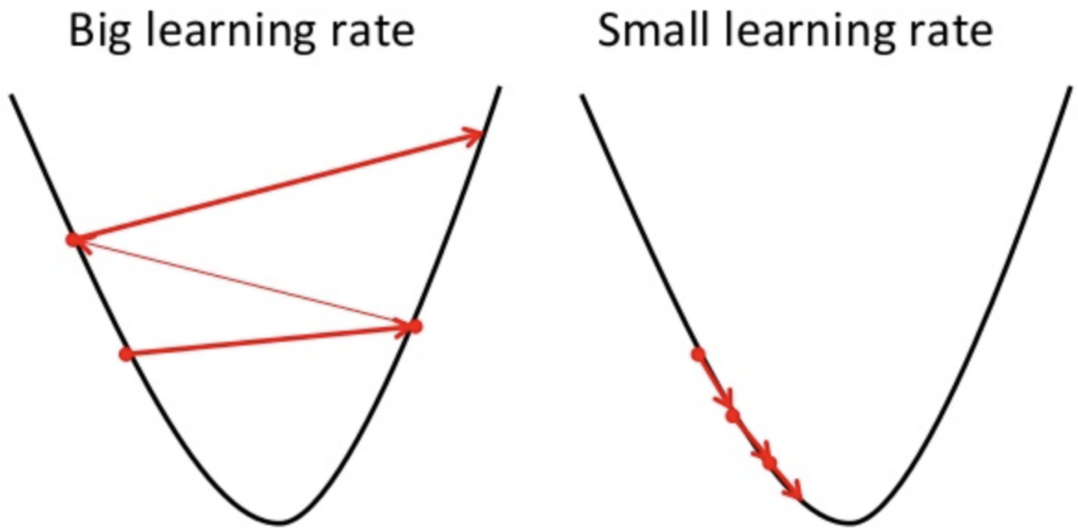
Gradient Descent

the word gradient means an increase and decrease in a property or something! whereas Descent means the act of moving downward

Gradient descent is an iterative optimization algorithm to find the minimum of a function. Here that function is our Loss Function.

Gradient descent is a method of updating m and c to reduce the cost function(MSE).

The concept is that we start with some m and c values and then reduce the cost by changing them iteratively. Gradient descent assists us in changing the values.



Usual example in Gradient descent:

Imagine a valley and a person with no sense of direction who wants to get to the bottom of the valley. He goes down the slope and takes large steps when the slope is steep and small steps when the slope is less steep. He decides his next position based on his current position and stops when he gets to the bottom of the valley which was his goal.

Why?

- **Gradient descent is by far the most popular optimization strategy used in Machine Learning and Deep Learning at the moment.**
- **It is used when training Data models, can be combined with every algorithm and is easy to understand and implement.**

Feature Assessment and Selection:

Gini score:

It measures the purity of the sample. Split happens based on the score.

Formula:

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

-

Gini score ranges from 0 to 0.5 whereas entropy ranges from 0 to 1.

Why do we need a Gini score if we have entropy?:

Gini score is computationally efficient(no need for log computation).

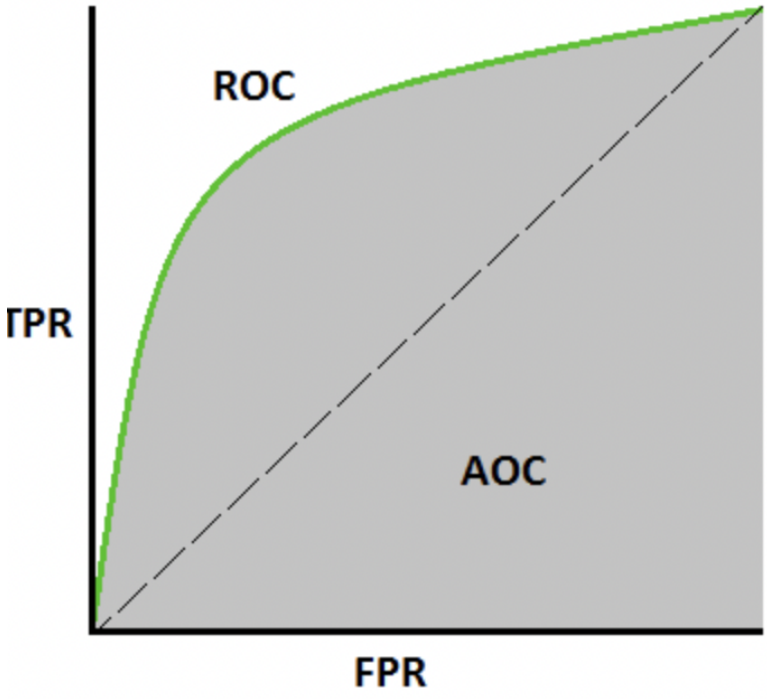
It mostly used metrics in ensemble techniques.

Understanding AUC - ROC Curve:

AUC - ROC curve is a performance measurement for the classification problems

ROC is a probability curve and AUC represents the degree or measure of separability

The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.



TPR / Recall = TP / TP+FN

Specificity = TN/TN+FP

FPR = FP / TN+FP

AUC value interpretation:

An excellent model has AUC near to the 1 which means it has a good measure of separability.

A poor model has an AUC near 0 which means it has the worst measure of separability.

when AUC is 0.5, it means the model has no class separation capacity whatsoever.

ROC is a curve of probability.

The relation between Sensitivity, Specificity, FPR, and Threshold.

Sensitivity and Specificity are inversely proportional to each other. So

when we increase Sensitivity, Specificity decreases, and vice versa.

When we decrease the threshold, we get more positive values thus it increases the sensitivity and decreasing the specificity.

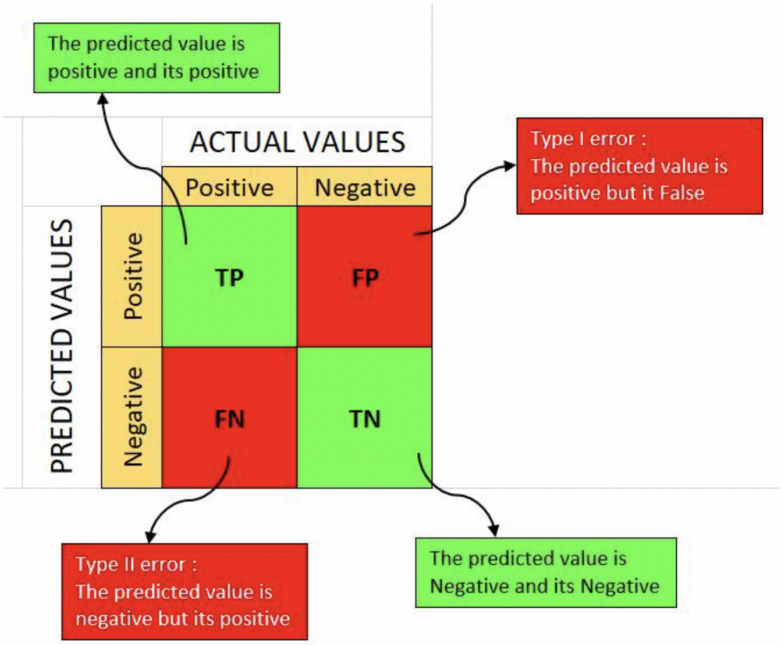
Similarly, when we increase the threshold, we get more negative values thus we get higher specificity and lower sensitivity. As we know FPR is 1 - specificity. So when we increase TPR, FPR also increases and vice versa.

The Confusion Matrix:

A confusion matrix is a summary of prediction results on a classification problem.

The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix.

The confusion matrix shows the ways in which your classification model is confused when it makes predictions.



LIft:

lift = (predicted rate / average rate)

Lift helps you get a better picture of the overall performance of your model. You can quickly spot flaws if the slope of the lift chart is not monotonic.