1. **What is the difference between normalization and standardization? When to use normalization & standardization?**

**Normalization is used when the data doesn't have Gaussian distribution whereas Standardization is used on data having Gaussian distribution**. Normalization scales in a range of [0,1] or [-1,1]. Standardization is not bounded by range. Normalization is highly affected by outliers.

**Normalization** is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling.



**Standardization** is another scaling technique where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.



Normalization vs. standardization is an eternal question among machine learning newcomers. Let me elaborate on the answer in this section.

* Normalization is good to use when you know that the distribution of your data does not follow a Gaussian distribution. This can be useful in algorithms that do not assume any distribution of the data like K-Nearest Neighbors and Neural Networks.
* Standardization, on the other hand, can be helpful in cases where the data follows a Gaussian distribution. However, this does not have to be necessarily true. Also, unlike normalization, standardization does not have a bounding range. So, even if you have outliers in your data, they will not be affected by standardization.

However, at the end of the day, the choice of using normalization or standardization will depend on your problem and the machine learning algorithm you are using. There is no hard and fast rule to tell you when to normalize or standardize your data.

You can always start by fitting your model to raw, normalized and standardized data and compare the performance for best results.

**Note:** It is a good practice to fit the scaler on the training data and then use it to transform the testing data. This would avoid any data leakage during the model testing process. Also, the scaling of target values is generally not required.

Link[: Click here](https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/)

* **Below are the few ways we can do feature scaling.**
* Min Max Scaler
* Standard Scaler
* Max Abs Scaler
* Robust Scaler
* Quantile Transformer Scaler
* Power Transformer Scaler
* Unit Vector Scalar

Link: [Click here](https://towardsdatascience.com/all-about-feature-scaling-bcc0ad75cb35)

1. **What is the meaning of Overfitting in Machine learning?**

Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the models ability to generalize.

Overfitting can be seen in machine learning when a statistical model describes random error or noise instead of the underlying relationship. Overfitting is usually observed when a model is excessively complex. It happens because of having too many parameters concerning the number of training data types.

**There are multiple ways of avoiding overfitting, such as:**

* Regularization. It involves a cost term for the features involved with the objective function
* Making a simple model. With lesser variables and parameters, the variance can be reduced
* Cross-validation methods like k-folds can also be used
* If some model parameters are likely to cause overfitting, techniques for regularization like LASSO can be used that penalize these parameters

1. **Why does overfitting occur?**

The possibility of overfitting occurs when the criteria used for training the model is not as per the criteria used to judge the efficiency of a model.

1. **What is the method to avoid overfitting?**

Overfitting occurs when we have a small dataset, and a model is trying to learn from it. By using a large amount of data, overfitting can be avoided. But if we have a small database and are forced to build a model based on that, then we can use a technique known as cross-validation. In this method, a model is usually given a dataset of a known data on which the training data set is run and a dataset of unknown data against which the model is tested.

1. **How does Machine Learning differ from Deep Learning?**

Machine learning is all about algorithms which are used to parse data, learn from that data, and then apply whatever they have learned to make informed decisions.

Deep learning is a part of machine learning, which is inspired by the structure of the human brain and is particularly useful in feature detection.

1. **How is KNN different from k-means?**

KNN or K nearest neighbors is a supervised algorithm which is used for classification purposes. In KNN, a test sample is given as the class of the majority of its nearest neighbors. On the other hand, K-means is an unsupervised algorithm which is mainly used for clustering. In k-means clustering, it needs a set of unlabeled points and a threshold only. The algorithm further takes unlabeled data and learns how to cluster it into groups by computing the mean of the distance between different unlabeled points.

1. **What are the five popular algorithms we use in Machine Learning?**

Decision Tree

Probabilistic Networks

Neural Network

Support Vector Machines

Nearest Neighbor

1. **What is model selection in Machine Learning?**

The process of choosing models among diverse mathematical models, which are used to define the same data is known as Model Selection. Model learning is applied to the fields of statistics, data mining, and machine learning.

1. **What are the common ways to handle missing data in a dataset?**

Missing data is one of the standard factors while working with data and handling. It is considered as one of the greatest challenges faced by the data analysts.

There are many ways one can impute the missing values. Some of the common methods to handle missing data in datasets can be defined as deleting the

rows, replacing with mean/median/mode, predicting the missing values, assigning a unique category, using algorithms that support missing values, etc.

1. **Describe Precision and Recall?**

Precision can be said as a positive predictive value. It is the fraction of relevant instances among the received instances.

On the other side, recall is the fraction of relevant instances that have been retrieved over the total amount or relevant instances. The recall is also known as

sensitivity.

1. **Explain True Positive, True Negative, False Positive, and False Negative in Confusion**

**True Positive**

When a model correctly predicts the positive class, it is said to be a true positive.

**True Negative**

When a model correctly predicts the negative class, it is said to be a true negative.

**False Positive**

When a model incorrectly predicts the positive class, it is said to be a false positive. It is also known as 'Type I' error.

**False Negative**

When a model incorrectly predicts the negative class, it is said to be a false negative. It is also known as 'Type II' error.

1. **What, according to you, is more important between model accuracy and model performance?**

Model accuracy is a subset of model performance. The accuracy of the model is directly proportional to the performance of the model. Thus, better the performance of the model, more accurate the predictions.

1. **What do you understand by the F1 score?**

The F1 score represents the measurement of a model's performance. It is referred to as a weighted average of the precision and recall of a model. The results tending to 1 are considered as the best, and those tending to 0 are the

Worst.

1. **How is a decision tree pruned?**

Pruning is said to occur in decision trees when the branches which may consist of weak predictive power are removed to reduce the complexity of the model and increase the predictive accuracy of a decision tree model.

Pruning can occur bottom-up and top-down, with approaches such as reduced error pruning and cost complexity

pruning.

1. **What do you understand by Underfitting?**

Underfitting is an issue when we have a low error in both the training set and the testing set. Few algorithms work better for interpretations but fail for better predictions.

1. **When does regularization become necessary in Machine Learning?**

Regularization is necessary whenever the model begins to overfit/ underfit. It is a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many variables to zero and reduce cost terms. It helps to reduce model complexity so that the model can become better at predicting (generalizing).

1. **What is Regularization? What kind of problems does regularization solve?**

A regularization is a form of regression, which constrains/ regularizes or shrinks the coefficient estimates towards zero. In other words, it discourages learning a more complex or flexible model to avoid the risk of overfitting. It reduces the variance of the model, without a substantial increase in its bias.

1. **Do you think that treating a categorical variable as a continuous variable would result in a better predictive model?**

For a better predictive model, the categorical variable can be considered as a continuous variable only when the variable is ordinal in nature.

1. **How is machine learning used in day-to-day life?**

Most of the people are already using machine learning in their everyday life. Assume that you are engaging with the internet, you are actually expressing your preferences, likes, dislikes through your searches. All these things are picked up by cookies coming on your computer, from this, the behavior of a user is evaluated.

1. **Describe dimension reduction in machine learning.**

Dimension reduction is the process which is used to reduce the number of random variables under consideration. Dimension reduction can be divided into feature selection and extraction.

1. **What is the trade-off between bias and variance?**

Both bias and variance are errors. Bias is an error due to erroneous or overly simplistic assumptions in the learning algorithm. It can lead to the model under-fitting the data, making it hard to have high predictive accuracy and generalize the knowledge from the training set to the test set. Variance is an error due to too much complexity in the learning algorithm. It leads to the algorithm being highly sensitive to high degrees of variation in the training data, which can lead the model to overfit the data.

1. **What do you understand about the Decision Tree in Machine Learning?**

Decision Trees can be defined as the Supervised Machine Learning, where the data is continuously split according to a certain parameter. It builds classification or regression models similar to a tree structure, with datasets broken up into smaller subsets while developing the decision tree. The tree can be defined by two entities, namely decision nodes, and leaves.

1. **What do you mean by Genetic Programming?**

Genetic Programming (GP) is almost similar to an Evolutionary Algorithm, a subset of machine learning. Genetic programming software systems implement an algorithm that uses random mutation, a fitness function, crossover, and multiple generations of evolution to resolve a user-defined task.

1. **Why do we need to convert categorical variables into factors? Which functions are used to perform the conversion?**

Most Machine learning algorithms require numbers as input. That is why we convert categorical values into factors to get numerical values. We also don't have to deal with dummy variables. The functions factor() and as.factor() are used to convert variables into factors

1. **How Can You Choose a Classifier Based on a Training Set Data Size?**

When the training set is small, a model that has a right bias and low variance seems to work better because they are less likely to overfit.

For example, Naive Bayes works best when the training set is large. Models with low bias and high variance tend to perform better as they work fine with complex relationships.

1. **Explain the Confusion Matrix with Respect to Machine Learning Algorithms.**

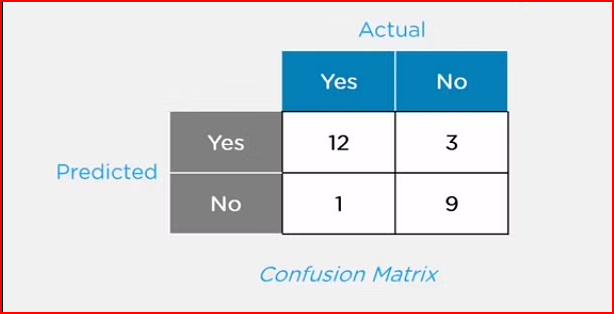
A confusion matrix (or error matrix) is a specific table that is used to measure the performance of an algorithm. It is mostly used in supervised learning; in unsupervised learning, it’s called the matching matrix.

The confusion matrix has two parameters:

* Actual
* Predicted

It also has identical sets of features in both of these dimensions

Consider a confusion matrix (Binary matrix ) shown below:



Here,

For actual values:

Total Yes = 12+1 = 13

Total No = 3+9 = 12

Similarly, for predicted values:

Total Yes = 12+3 = 15

Total No = 1+9 = 10

For a model to be accurate, the values across the diagonals should be high. The total sum of all the values in the matrix equals the total observations in the test data set.

For the above matrix, total observations = 12+3+1+9 = 25

Now, accuracy = sum of the values across the diagonal/total dataset

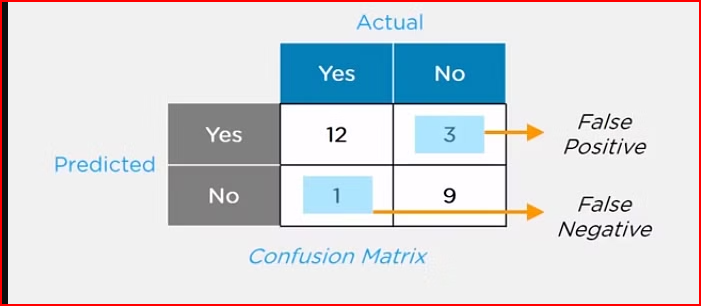
= (12+9) / 25

= 21 / 25

= 84%

### **What Is a False Positive and False Negative and How Are They Significant?**

* False positives are those cases that wrongly get classified as True but are False.
* False negatives are those cases that wrongly get classified as False but are True.
* In the term ‘False Positive,’ the word ‘Positive’ refers to the ‘Yes’ row of the predicted value in the confusion matrix. The complete term indicates that the system has predicted it as a positive, but the actual value is negative.



So, looking at the confusion matrix, we get:

False-positive = 3

True positive = 12

Similarly, in the term ‘False Negative,’ the word ‘Negative’ refers to the ‘No’ row of the predicted value in the confusion matrix. And the complete term indicates that the system has predicted it as negative, but the actual value is positive.

So, looking at the confusion matrix, we get:

False Negative = 1

True Negative = 9

1. **How is Flipkart Able to Recommend Other Things to Buy? How Does the Recommendation Engine Work?**

Once a user buys something from Flipkart, it stores that purchase data for future reference and finds products that are most likely also to be bought, it is possible because of the Association algorithm, which can identify patterns in a given dataset.

1. **Briefly Explain Logistic Regression.**

Logistic regression is a classification algorithm used to predict a binary outcome for a given set of independent variables.

The output of logistic regression is either a 0 or 1 with a threshold value of generally 0.5. Any value above 0.5 is considered as 1, and any point below 0.5 is considered as 0.

1. **What Are Some Methods of Reducing Dimensionality?**

You can reduce dimensionality by combining features with feature engineering, removing collinear features, or using algorithmic dimensionality reduction.

1. **Explain Correlation and Covariance?**

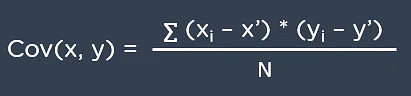
**Correlation:** Correlation tells us how strongly two random variables are related to each other. It takes values between -1 to +1.

Formula to calculate Correlation:



**Covariance:** Covariance tells us the direction of the linear relationship between two random variables. It can take any value between - ∞ and + ∞.

Formula to calculate Covariance:



1. **What are Loss Function and Cost Functions? Explain the key difference between them?**

When calculating loss we consider only a single data point, then we use the term loss function.

Whereas, when calculating the sum of error for multiple data then we use the cost function. There is no major difference.

In other words, the loss function is to capture the difference between the actual and predicted values for a single record whereas cost functions aggregate the difference for the entire training dataset.

The Most commonly used loss functions are Mean-squared error and Hinge loss.

Mean-Squared Error(MSE): In simple words, we can say how our model predicted values against the actual values.

MSE = √(predicted value - actual value)2

Hinge loss: It is used to train the machine learning classifier, which is

L(y) = max(0,1- yy)

Where y = -1 or 1 indicates two classes and y represents the output form of the classifier. The most common cost function represents the total cost as the sum of the fixed costs and the variable costs in the equation y = mx + b

1. **How do you make sure which Machine Learning Algorithm to use?**

It completely depends on the dataset we have. If the data is discrete we use SVM. If the dataset is continuous we use linear regression.

So there is no specific way that lets us know which ML algorithm to use, it all depends on the exploratory data analysis (EDA).

EDA is like “interviewing” the dataset; As part of our interview we do the following:

* Classify our variables as continuous, categorical, and so forth.
* Summarize our variables using descriptive statistics.
* Visualize our variables using charts.

Based on the above observations select one best-fit algorithm for a particular dataset

1. **How do you check the Normality of a dataset?**

Visually, we can use plots. A few of the normality checks are as follows:

* Shapiro-Wilk Test
* Anderson-Darling Test
* Martinez-Iglewicz Test
* Kolmogorov-Smirnov Test
* D’Agostino Skewness Test

1. **What is P-value?**

P-values are used to make a decision about a hypothesis test. P-value is the minimum significant level at which you can reject the null hypothesis. The lower the p-value, the more likely you reject the null hypothesis

1. **How can we relate standard deviation and variance?**

Standard deviation refers to the spread of your data from the mean. Variance is the average degree to which each point differs from the mean i.e. the average of all data points. We can relate Standard deviation and Variance because it is the square root of Variance.

1. **Is a high variance in data good or bad?**

Higher variance directly means that the data spread is big and the feature has a variety of data. Usually, high variance in a feature is seen as not so good quality.

1. **If your dataset is suffering from high variance, how would you handle it?**

For datasets with high variance, we could use the bagging algorithm to handle it. Bagging algorithm splits the data into subgroups with sampling replicated from random data. After the data is split, random data is used to create rules using a training algorithm. Then we use polling technique to combine all the predicted outcomes of the model.

1. **What is a Box-Cox transformation?**

Box-Cox transformation is a power transform which transforms non-normal dependent variables into normal variables as normality is the most common assumption made while using many statistical techniques. It has a lambda parameter which when set to 0 implies that this transform is equivalent to log-transform. It is used for variance stabilization and also to normalize the distribution.

1. **What is the difference between regularization and normalization?**

**Normalisation** adjusts the data; . If your data is on very different scales (especially low to high), you would want to normalize the data. Alter each column to have compatible basic statistics. This can be helpful to make sure there is no loss of accuracy. One of the goals of model training is to identify the signal and ignore the noise. If the model is given free rein to minimize error, there is a possibility of suffering from overfitting.

**Regularization** adjusts the prediction function. Regularization imposes some control on this by providing simpler fitting functions over complex ones.

1. **Explain how a ROC curve works.**

ROC curve is a graphical representation of the contrast between true positive rates and the false positive rate at various thresholds. It’s often used as a proxy for the trade-off between the sensitivity of the model (true positives) vs the fall-out or the probability it will trigger a false alarm (false positives).

1. **Explain the difference between L1 and L2 regularization.**

L2 regularization tends to spread error among all the terms, while L1 is more binary/sparse, with many variables either being assigned a 1 or 0 in weighting. L1 corresponds to setting a Laplacean prior on the terms, while L2 corresponds to a Gaussian prior.

1. **How would you handle an imbalanced dataset?**

An imbalanced dataset is when you have, for example, a classification test and 90% of the data is in one class. That leads to problems: an accuracy of 90% can be skewed if you have no predictive power on the other category of data!

Way to handle a Imbalanced Dataset

* Collect more data to even the imbalances in the dataset.
* Resample the dataset to correct for imbalances.
* Try a different algorithm altogether on your dataset.

1. **What is the difference between Gini Impurity and Entropy in a Decision Tree?**

* Gini Impurity and Entropy are the metrics used for deciding how to split a Decision Tree.
* Gini measurement is the probability of a random sample being classified correctly if you randomly pick a label according to the distribution in the branch.
* Entropy is a measurement to calculate the lack of information. You calculate the Information Gain (difference in entropies) by making a split. This measure helps to reduce the uncertainty about the output label.

1. **How to handle Multicollinearity?**

Multicollinearity refers to the condition when two or more independent features are correlated to each other. The change in one of the collinear features may affect the other related features. Multicollinearity in the dataset may be caused due to poor designing of experiments while collecting the data or maybe introduced while creating new features.

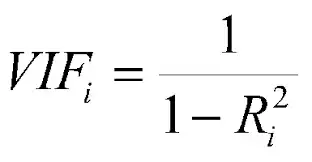
1. **Correlation Matrix:**

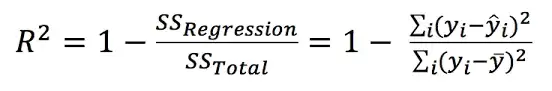
The correlation matrix can only measure collinearity between two features and is not able to detect multicollinearity in the dataset.

1. **Variance Inflation Factors (VIF):**

VIF is another technique that can be used to detect the multicollinearity of the features.

VIF measures the multicollinearity of the feature by computing the R-squared metric.





R-squared metrics measure how well the data points fit a line or curve. It ranges between 0 and 1, where values close to 1 reflect a good model.

The VIF numerical real values range between 1 to infinity, where a value of 1 reflects no multicollinearity exists for the given feature. As the VIF value increases 1 upwards, there exists multicollinearity between the features. Typically, a standard thumb rule says:

VIF=1: No multicollinearity

VIF between 1 to 5: Moderate multicollinearity

VIF > 5: Highly multicollinear

1. **Lasso Regression:**

Lasso regression is a linear regression technique with L1 prior as a regularization. The idea is to reduce the multicollinearity by regularization by reducing the coefficients of the features that are multicollinear.

By increasing the alpha value for the L1 regularizer, we introduce some small bias in the estimator that breaks the correlation and reduces the variance.

1. **Principal Component Analysis (PCA):**

PCA is a dimensionality reduction technique that uses matrix factorization under the hood to compute the eigenvalues and eigenvectors. PCA projects the given dataset into a new dimensional space based on eigenvectors.

PCA can be used to handle multicollinearity in the dataset by taking the top eigenvectors that preserve the maximum variance. The number of dimensions can be decided by observing the variance preserved for each eigenvector.

1. **Multicollinearity for Categorical Feature:**

pd.get\_dummies(), OneHotEncoder() are functions to vectorize the categorical features in the one-hot encoded format. When we one-hot encode any categorical feature, it introduces multicollinearity in the dataset.

The basic idea is to remove one feature level from the one-hot encoded vectorized data to handle the condition of multicollinearity. pd.get\_dummies() function has a parameter drop\_first that can be set to True, to remove the first feature value from vectorizing. Similarly, the scikit-learn implementation of a one-hot encoder offers a drop parameter that can remove one feature.

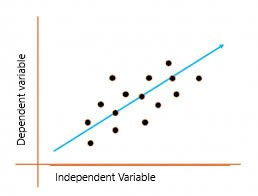
1. **More Data:**

Adding more data to the existing dataset can break the pattern of multicollinearity. This technique is often useful to remove the problem of multicollinearity.

**Linear Regression**

**Simple Linear Regression:**

Linear regression shows the linear relationship between the independent(predictor) variable i.e. X-axis and the dependent(output) variable i.e. Y-axis, called linear regression. If there is a single input variable X(independent variable), such linear regression is called **simple linear regression.**



The above graph presents the linear relationship between the output(y) variable and predictor(X) variables. The blue line is referred to as the best fit straight line. Based on the given data points, we attempt to plot a line that fits the points the best.

To calculate best-fit line linear regression uses a traditional slope-intercept form which is given below,

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.

But how does linear regression find out which is the best fit line?

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.

**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

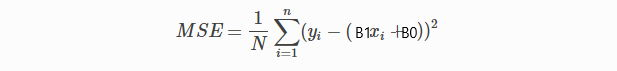
Note: best fit line is obtained by minimizing the Residual Sum of Squares(RSS).

**Cost Function for Linear Regression**

The cost function helps to work 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 y predicted and yi.

We calculate MSE using simple linear equation y=mx+b:

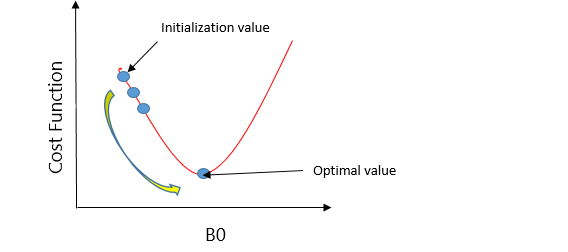


Using the MSE function, we’ll update the values of B0 and B1 such that the MSE value settles at the minima. These parameters can be determined using the gradient descent method such that the value for the cost function is minimum.

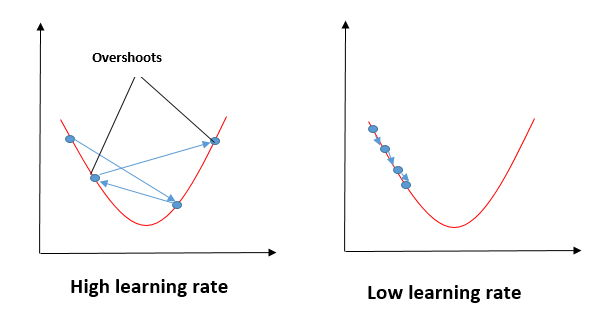
**Gradient Descent for Linear Regression**

Gradient Descent is one of the optimization algorithms that optimize the cost function(objective 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.

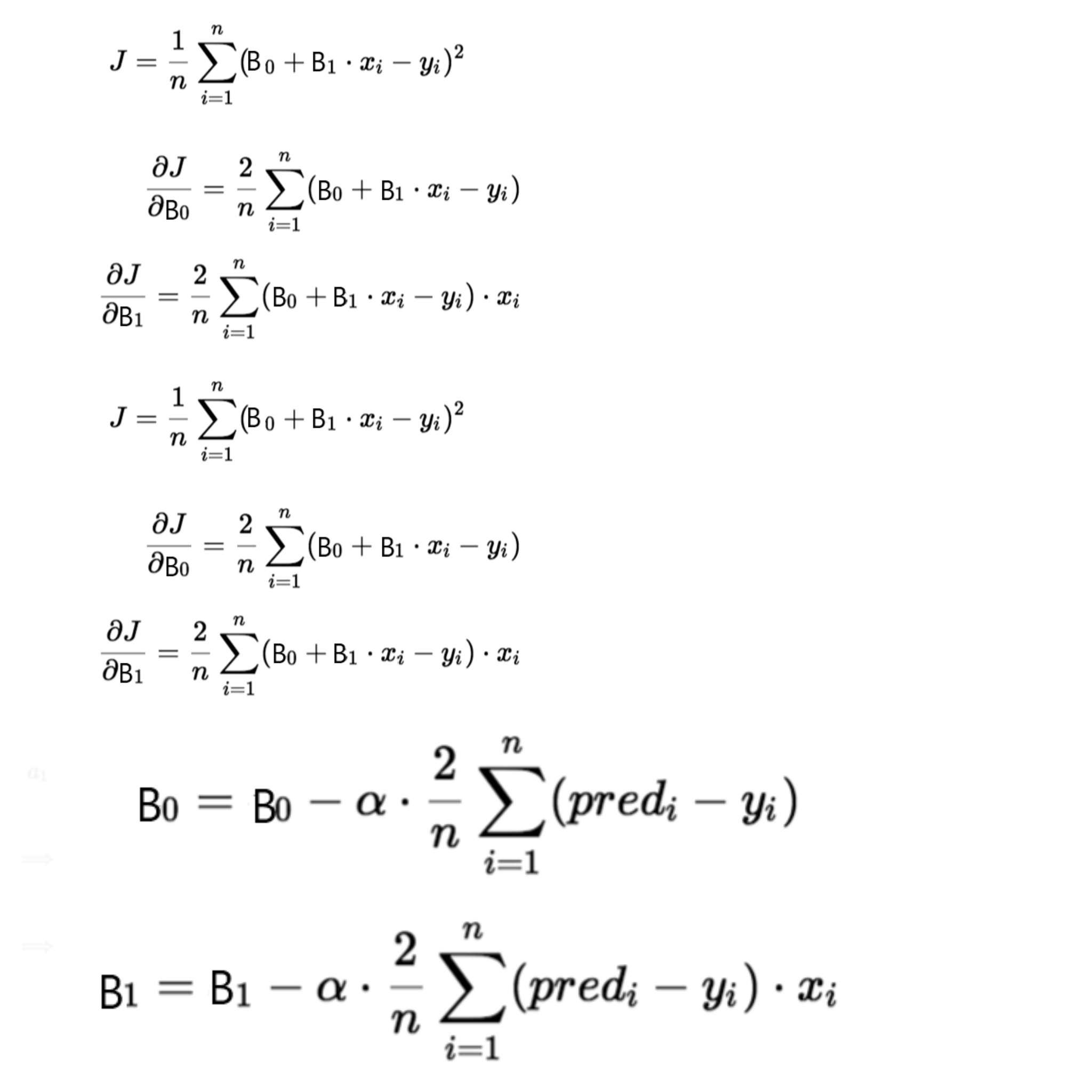
A regression model optimizes the gradient descent algorithm to update the coefficients of the line by reducing the cost function by randomly selecting coefficient values and then iteratively updating the values to reach the minimum cost function.



Let’s take an example to understand this. 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.

****

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 shows the updating of values)

**Evaluation Metrics for Linear Regression**

**The most used metrics are,**

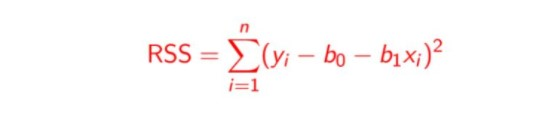
* Coefficient of Determination or R-Squared (R2)
* Root Mean Squared Error (RSME) and Residual Standard Error (RSE)

R-Squared is a number that explains the amount of variation that is explained/captured by the developed model. It always ranges between 0 & 1 . Overall, the higher the value of R-squared, the better the model fits the data.

Mathematically it can be represented as,

**R2 = 1 – ( RSS/TSS )**

**Residual sum of Squares (RSS)** is defined as the sum of squares of the residual for each data point in the plot/data. It is the measure of the difference between the expected and the actual observed output.

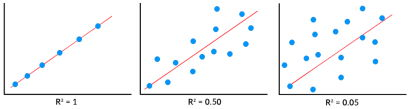
****

**Total Sum of Squares (TSS)** is defined as the sum of errors of the data points from the mean of the response variable. Mathematically TSS is,

Total Sum of Squares

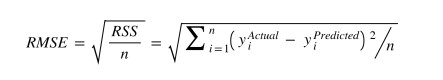
where y hat is the mean of the sample data points.

The significance of R-squared is shown by the following figures,

****

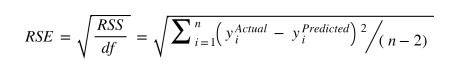
**Root Mean Squared Error:**

Root Mean Squared Error The Root Mean Squared Error is the square root of the variance of the residuals. It specifies the absolute fit of the model to the data i.e. how close the observed data points are to the predicted values. Mathematically it can be represented as,



To make this estimate unbiased, one has to divide the sum of the squared residuals by the **degrees of freedom** rather than the total number of data points in the model. This term is then called the **Residual Standard Error(RSE).**

Mathematically it can be represented as,



R-squared is a better measure than RSME. Because the value of Root Mean Squared Error depends on the units of the variables (i.e. it is not a normalized measure), it can change with the change in the unit of the variables.

**Note:**

* When a model has low bias and higher variance it ends up memorizing the data and causing overfitting.
  + There are several ways to prevent overfitting, which are stated below:
    - Cross-validation
    - If the training data is too small to train, add more relevant and clean data.
    - If the training data is too large, do some feature selection and remove unnecessary features.
    - Regularization
* When a model has high bias and low variance it ends up not generalizing the data and causing underfitting. It is unable to find the hidden underlying patterns from the data. This usually leads to low training accuracy and very low test accuracy.
  + The ways to prevent underfitting are stated below,
    - Increase the model complexity
    - Increase the number of features in the training data
    - Remove noise from the data.

[Blog to learn more in details [ Linear Regression ]](https://www.analyticsvidhya.com/blog/2021/10/everything-you-need-to-know-about-linear-regression/)

**Happy Learning!**

**Logistic Regression**

Logistic Regression is a Machine Learning method that is used to solve classification issues. It is a predictive analytic technique that is based on the probability idea. The classification algorithm Logistic Regression is used to predict the likelihood of a categorical dependent variable. The dependent variable in logistic regression is a binary variable with data coded as 1 (yes, True, normal, success, etc.) or 0 (no, False, abnormal, failure, etc.).

The goal of Logistic Regression is to discover a link between characteristics and the likelihood of a specific outcome. For example, when predicting whether a student passes or fails an exam based on the number of hours spent studying, the response variable has two values: pass and fail.

A Logistic Regression model is similar to a Linear Regression model, except that the Logistic Regression utilizes a more sophisticated cost function, which is known as the “Sigmoid function” or “logistic function” instead of a linear function.

Many people may have a question, whether Logistic Regression is a classification or regression category. The logistic regression hypothesis suggests that the cost function be limited to a value between 0 and 1. As a result, linear functions fail to describe it since it might have a value larger than 1 or less than 0, which is impossible according to the logistic regression hypothesis.

It is used when the data is linearly separable and the outcome is binary or dichotomous in nature.

That means Logistic regression is usually used for Binary classification problems.

**Binary Classification** refers to predicting the output variable that is discrete in **two** classes.

A few examples of Binary classification are Yes/No, Pass/Fail, Win/Lose, Cancerous/Non-cancerous, etc.

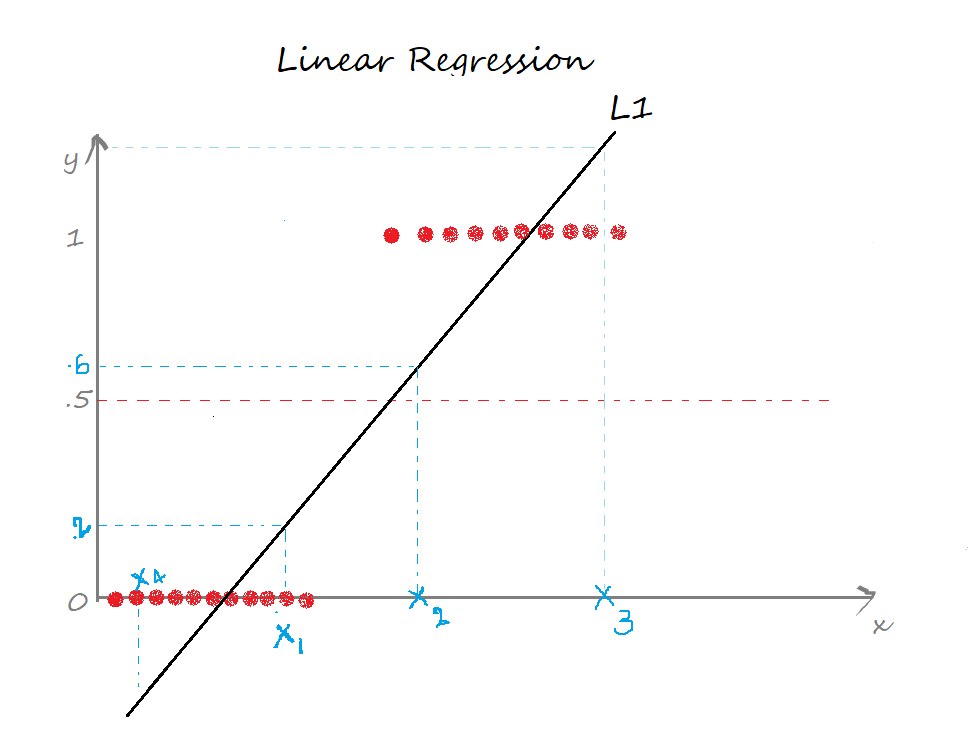
**Types of Logistic Regression**

* **Simple Logistic Regression:** a single independent is used to predict the output
* **Multiple logistic regression:** multiple independent variables are used to predict the output

[Simple Linear Regression](https://www.analyticsvidhya.com/blog/2021/05/learn-simple-linear-regression-slr/)

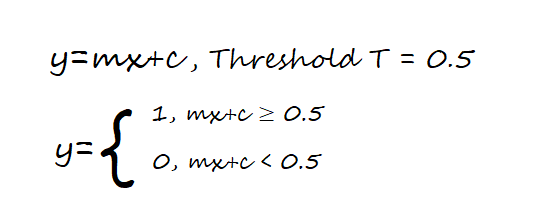
**Use Linear Regression for classification**

Now, let us try if we can use linear regression to solve a binary class classification problem. Assume we have a dataset that is linearly separable and has the output that is discrete in two classes (0, 1).

****

In Linear regression, we draw a straight line(the best fit line) L1 such that the sum of distances of all the data points to the line is minimal. The equation of the line L1 is y=mx+c, where m is the slope and c is the y-intercept.

We define a threshold T = 0.5, above which the output belongs to class 1 and class 0 otherwise.

****

Case 1: the predicted value for x1 is ≈0.2 which is less than the threshold, so x1 belongs to class 0.

Case 2: the predicted value for the point x2 is ≈0.6 which is greater than the threshold, so x2 belongs to class 1.

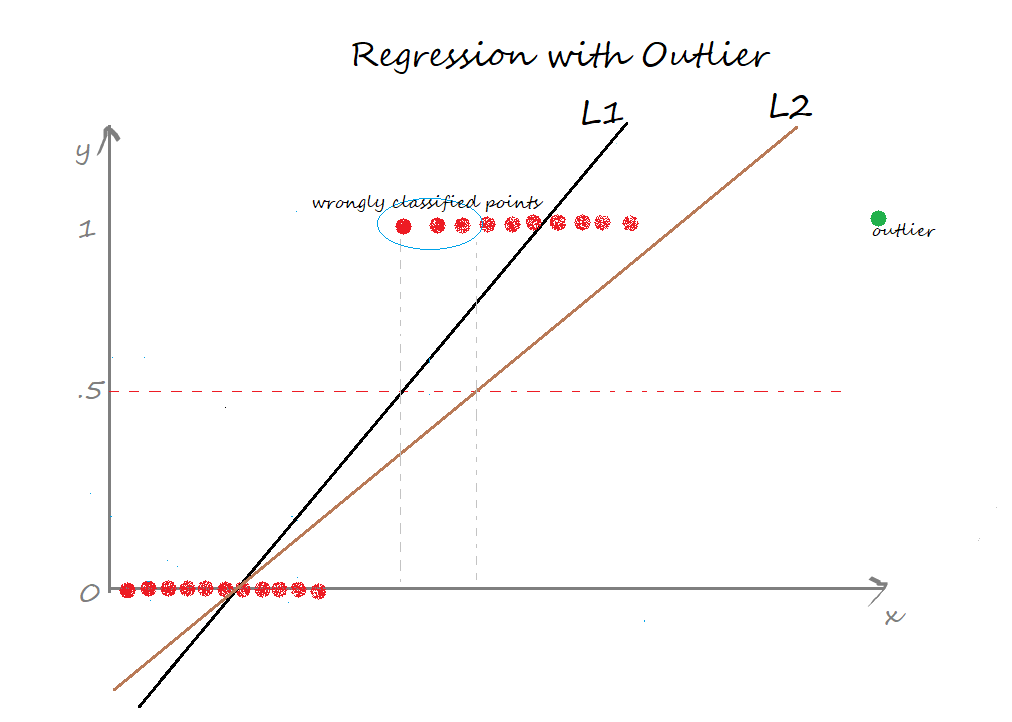
So far so good, yeah!

Case 3: the predicted value for the point x3 is beyond 1.

Case 4: the predicted value for the point x4 is below 0.

The predicted values for the points x3, x4 exceed the range (0,1) which doesn’t make sense because the probability values always lie between 0 and 1. And our output can have only two values either 0 or 1. Hence, this is a problem with the linear regression model.

Now, introduce an outlier and see what happens. The regression line gets deviated to keep the distance of all the data points to the line to be minimal.

****

L2 is the new best-fit line after the addition of an outlier. Seems good till now. But the problem is, if we closely observe, some of the data points are wrongly classified. Certainly, it increases the error term 🙁 This again is a problem with the linear regression model.

The two limitations of using a linear regression model for classification problems are:

* the predicted value may exceed the range (0,1)
* error rate increases if the data has outliers

There definitely is a need for Logistic regression here.

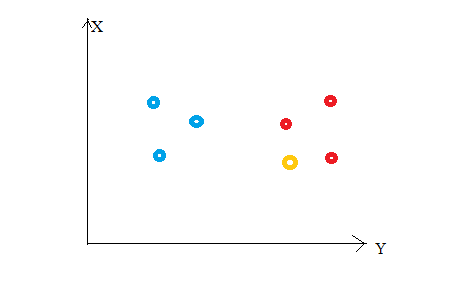
**How does Logistic Regression Work?**

# **K Nearest Neighbors**

# The K Nearest Neighbor algorithm falls under the Supervised Learning category and is used for classification (most commonly) and regression. It is a versatile algorithm also used for imputing missing values and resampling datasets. As the name (K Nearest Neighbor) suggests it considers K Nearest Neighbors (Data points) to predict the class or continuous value for the new Datapoint.

**How does KNN Work?**

**Principle**: Consider the following figure. Let us say we have plotted data points from our training set on a two-dimensional feature space. As shown, we have a total of 6 data points (3 red and 3 blue). Red data points belong to ‘class1’ and blue data points belong to ‘class2’. And yellow data point in a feature space represents the new point for which a class is to be predicted. Obviously, we say it belongs to ‘class1’ (red points) Why? Because its nearest neighbors belong to that class!



Yes, this is the principle behind K Nearest Neighbors. Here, nearest neighbors are those data points that have minimum distance in feature space from our new data point and K is the number of such data points we consider in our implementation of the algorithm. Therefore, distance metric and K value are two important considerations while using the KNN algorithm.

Euclidean distance is the most popular distance metric. You can also use

Hamming distance

Manhattan distance

Minkowski distance as per your need.

For predicting class/ continuous value for a new data point, it considers all the data points in the training dataset. Finds new data point’s ‘K’ Nearest Neighbors (Data points) from feature space and their class labels or continuous values.

**For classification:** A class label assigned to the majority of K Nearest Neighbors from the training dataset is considered as a predicted class for the new data point.

**For regression:** Mean or median of continuous values assigned to K Nearest Neighbors from training dataset is a predicted continuous value for our new data point

**How to choose the value for K?**

We don’t have a particular method for determining the correct value of K. Here, we’ll try to test the model’s accuracy for different K values. The value of K that delivers the best accuracy for both training and testing data is selected.

**Note**: It is recommended to always select an odd value of K.

When the value of K is set to even, a situation may arise in which the elements from both groups are equal. In the diagram below, elements from both groups are equal in the internal “Red” circle (k == 4).

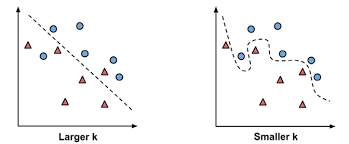
In this condition, the model would be unable to do the correct classification for you. Here the model will randomly assign any of the two classes to this new unknown data.

Choosing an odd value for K is preferred because such a state of equality between the two classes would never occur here. Due to the fact that one of the two groups would still be in the majority, the value of K is selected as odd.

**The impact of selecting a smaller or larger K value on the model**

Larger K value: The case of underfitting occurs when the value of k is increased. In this case, the model would be unable to correctly learn on the training data.

Smaller k value: The condition of overfitting occurs when the value of k is smaller. The model will capture all of the training data, including noise. The model will perform poorly for the test data in this scenario.

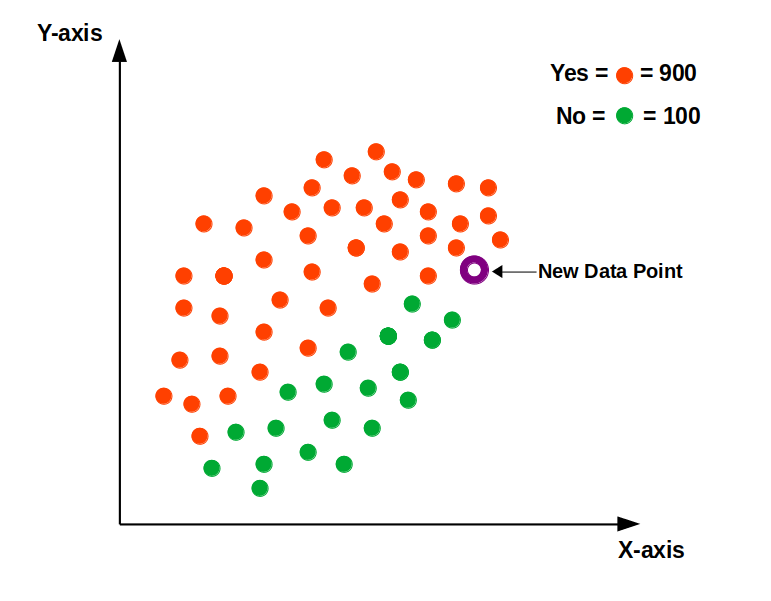
****

**Impact of Imbalanced dataset and Outliers on KNN**

**Imbalanced Dataset:**

When dealing with an imbalanced data set, the model will become biased. Consider the example shown in the diagram below, where the “Yes” class is more prominent.

As a consequence, the bulk of the closest neighbours to this new point will be from the dominant class. Because of this, we must balance our data set using either an “Upscaling” or “Downscaling” strategy.

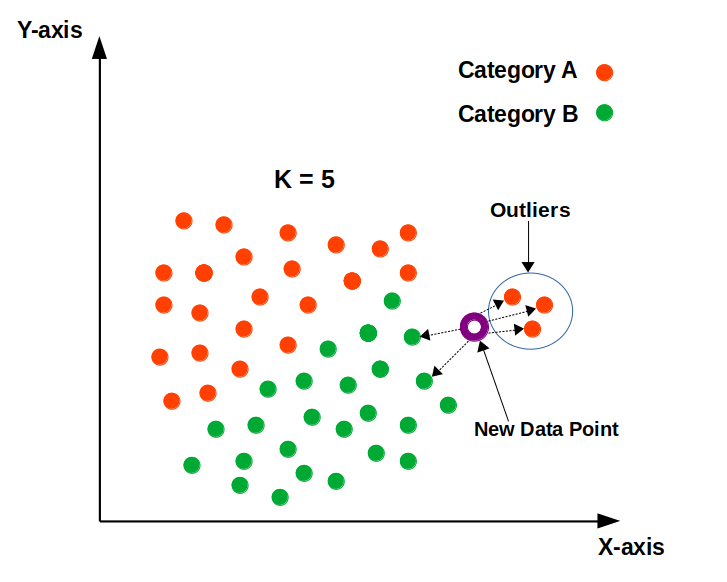


**Outliers**

Outliers are the points that differ significantly from the rest of the data points.

The outliers will impact the classification/prediction of the model. The appropriate class for the new data point, according to the following diagram, should be “Category B” in green.

The model, however, would be unable to have the appropriate classification due to the existence of outliers. As a result, removing outliers before using KNN is recommended.

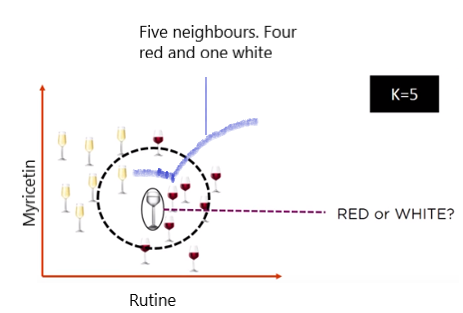
****

**How does KNN work for ‘Classification’ and ‘Regression’ problem statements?**

**Classification:**

When the problem statement is of ‘classification’ type, KNN tends to use the concept of “Majority Voting”. Within the given range of K values, the class with the most votes is chosen.

Consider the following diagram, in which a circle is drawn within the radius of the five closest neighbours. Four of the five neighbours in this neighbourhood voted for ‘RED,’ while one voted for ‘WHITE.’ It will be classified as a ‘RED’ wine based on the majority votes.



Example: Several parties compete in an election in a democratic country like India. Parties compete for voter support during election campaigns. The public votes for the candidate with whom they feel more connected.

**Regression:**

KNN employs a mean/average method for predicting the value of new data. Based on the value of K, it would consider all of the nearest neighbours.

The algorithm attempts to calculate the mean for all the nearest neighbours’ values until it has identified all the nearest neighbours within a certain range of the K value.

Consider the diagram below, where the value of k is set to 3. It will now calculate the mean (52) based on the values of these neighbours (50, 55, and 51) and allocate this value to the unknown data.

# **Decision Boundary for Knn?**