# INDIVIDUAL ASSIGNMENT

## STATISTICS & MACHINE LEARNING

SUBMISSION DATE – 31ST March, 2022.

DONE BY

Kamalakannan Thayanidhi

# Table of Contents

# Introduction

Statistics is a subfield of mathematics, and this gives a focus of well defined, carefully chosen methods to predict or visualize the relationships between chosen variables. Machine Learning is a subfield of computer science where the main focus is on algorithm and coding. We are going to use & apply the knowledge of these two fields combined together for this individual assignment.

# Assignment Objective

The Main Assignment Objective is to display our understanding of different Machine Learning Algorithms based on how they work, what are their strength & weaknesses. Moreover, we have to setup a whole benchmark experiment on a given Dataset to conduct a comparison between different models and the metrics which they yield. Since the objective of the given dataset is to find best classification model. We are going to go ahead with 5 models which are much more suited for Classification Modelling Technique.

# Machine Learning Algorithms

## I) Logistic Regression –

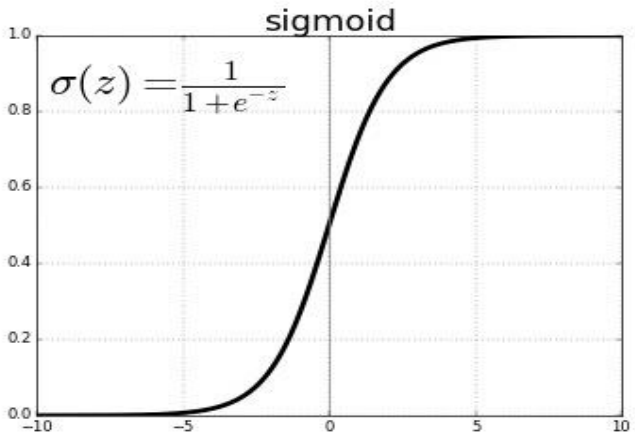
### General Idea -

The Logistic Regression is one of the **simplest** & **efficient classification** machine learning algorithms that exists. It is used when the data is linearly separable, and the **outcome** is **binary** (i.e. Binary refers to predicting the output variable that is discrete in two classes.). Even though Logistic is mainly used for binary classification it can also be extended to solve Multi-Class Classification problems.

From a **Mathematical** point of view, it is a "supervised machine learning" algorithm that can be used to estimate the probability of a certain class or event. Thus, while **estimating** the **probability** of an event occurring or it can categorize them into separate event outcomes. Since Logistic Regression works

on estimating the Probability of events/class. The main outlying function should **predict** values only between **0 and 1**.

Working / Objective Function -




Since Logistic Regression works on estimating the Probability of events/class. The main outlying function should **predict** values only between **0 and 1**. So Logistic Regression should use a function where the values fit between 0 and 1 for every prediction. In order to do this, we can use **sigmoid type function.** This yields us an S-Shaped curve.

The Basic Logistic Regression Function looks in this form. (i.e. The P(X) is the probability that observation X is in the defined class).

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

But this alone doesn't give us the categorization of the variables. We need to further use **Maximum Likelihood Estimation.** This MLE function always return a large probability when the model is close to the matching class value, and a small value when it is far away, for both classification of a Yes/No, Up/Down etc.

Advantages -

1) It is easier to Implement & Interpret.
2) It has really good Accuracy for Simple datasets.
3) It is very Efficient to train.
4) It can easily extend to multiple classes.

Dis – Advantages –

1) Non-linear problems can't be solved with logistic regression.
2) Logistic Regression requires no multicollinearity between independent variables.
3) It is tough to obtain complex relationships.

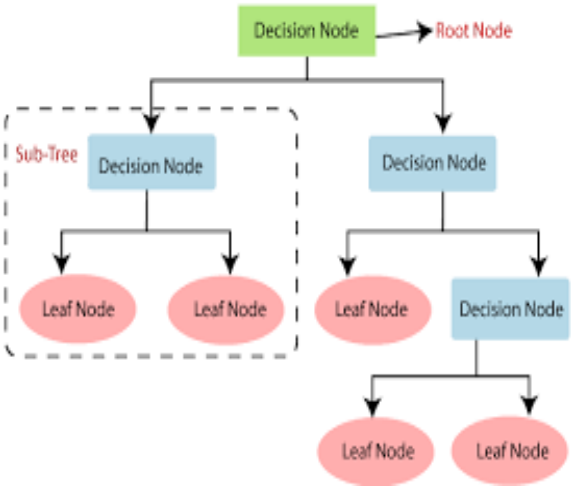
## II) Decision Tree –

General Idea -

The Decision Tree Algorithm as the name suggests works on a **Tree-Based Model**. The way the model splits the data into Different Branches, Sub-branches & Leaf Nodes is represented as a Tree. Simply put, decision trees are a collection of if-else statements. It checks if the condition is true and if it is then it proceeds to the next node attached to that decision.

Working / Objective Function -

The Decision Tree Algorithm **binarily splits** the data, but the method the decision tree uses to split into categories is what differentiates this algorithm with other algorithms. The Decision Tree uses **Entropy of the Data** to decide whether to make a split at a particular Data Point or not. Entropy basically measures the **degree of randomness** of a node.




The Main goal of a Decision Tree is to decrease the uncertainty or **degree of randomness** in the splitting of the dataset. Therefore, the Decision Tree incorporates the **reduction of uncertainty** of a given attribute, as well as **determining** whether a given attribute could be used as a **root node** or decision node.

The Way Decision Tree does this is through is either by considering the **Classification Error – Rate** or **Node Impurity**. So here we try and fit every value present in the set and create the conditions. Next for the created conditions we check which provides the **minimum** Node Impurity index or which provides the **minimum** Error – Rate (i.e.Maximum Information Gain).

Depending upon the Number **of Depths, Leaves & Features** to include the Decision Tree algorithm can **easily overfit** the Data. In order to stop this from happening we need to do an additional step called **pruning** which improves the performance of the tree by cutting the nodes or sub-nodes which have low importance.

Advantages –

1) It requires less effort for data preparation during pre-processing
2) It is very intuitive and easy to explain.
3) It does not require normalization/scaling of data.

Dis – Advantages –

1) Small change is dataset value can cause large change in structure of tree.
2) It takes longer time to train the model as it becomes complex.
3) It is usually less accurate than other similar classification algorithms.
4) It can easily overfit the Data (i.e. unless we Prune,etc.)

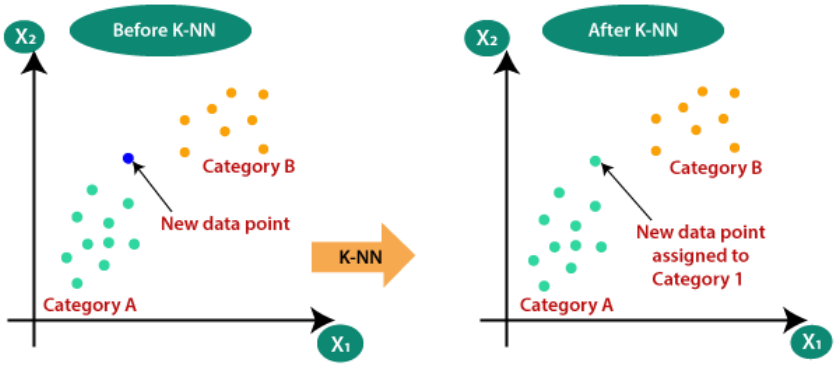
## III) K-Nearest Neighbors –

General Idea -

KNN is a supervised machine learning algorithm that can be used for classification. The KNN Algorithm observes the behaviors of the **nearest points** and **classifies** itself accordingly. This classification is a type of **lazy learning** as it does not attempt to construct a general internal model, but simply stores instances of the training data. Also, K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

Working / Objective Function -

The Algorithm needs to be fed values like **K – Value** which denotes how many **neighbors to consider**. Furthermore, we can also define how the KNN would calculates the mean/mode distances of its neighbors based on different methods.




First the algorithm needs to be given the K-Value which would determine how many neighbors to consider. Then it **calculates** the **distance** between all the Training Data Points and the newly introduced Data points. Then it **Sorts** the computed distance in **Ascending** Order between the points and choose the **First K-Distances** from this sorted list. Depending upon whether it is a classification/regression problem it takes the **mode/mean** of the classes associated with the Distances (i.e Classification – Mode, Regression – Mean).

The Important factors which determine how good our KNN Classification are the K-Value & Distance Calculation Method. If we **have very small K-Value** our **predictions** become very **unstable**. If we have **very High K-Value** give us **high error rate**.

There are mainly four ways to calculate the distance measure between the data point and its nearest neighbor**: Euclidean distance, Manhattan distance, Hamming distance,** and **Minkowski distance**. **The Most commonly used** distance calculating method is the **Euclidean distance.** From these methods since each have a different algorithm each might ultimately yield different model & results.

Advantages –

1) It is simple to implement.
2) It is robust to the noisy training data.
3) It can be more effective if the training data is large.

Dis – Advantages –

1) Always needs to determine the value of K which may be complex some time.
2) The computation cost is high because of calculating the distances between the data points for all the training samples.

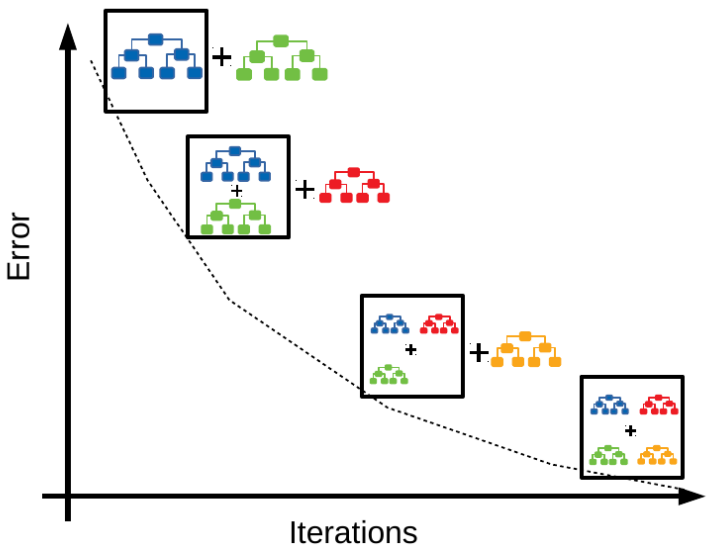
## IV) Gradient Boosting –

General Idea -

The Gradient Boosting Method builds models sequentially and each successive model tries to reduce the errors from the previous model. The Algorithm does this by minimizing the loss function by adding weak learners using gradient descent. The Loss function changes depending upon problem. (i.e. For Classification we use Log-Likelihood Function).

Working / Objective Function -

The first step of the algorithm is to initialize the model with some constant value, for initializing our model for a classification problem we have to use the log(odds) function. The minimum value of this loss function will be our first prediction. The Next step is to calculate all the pseudo residuals based on our first prediction. After completing this step, we make a decision tree with all independent variables and target variables as Residuals.




The Final step that algorithm does is it gets new predictions by adding our base model with the new tree we made on residuals. This minimizes the overall prediction error.

Now after all these steps have been completed the Gradient Boosting methods repeats the whole process from step 1-3 and does this over and over. The intuition is that the Gradient boosting method tries to minimize the prediction error by learning from the previous models one step at a time.

One of the main reasons Gradient Boosting is so powerful is because these methods deal with both aspect of Bias-Variance Tradeoff and hence why they are very effective. Furthermore, this method allows us to define a lot of parameters which include the Tree-Based Parameters, Boosting Based Parameters & Miscellaneous Parameters which improve the overall function and predictive performance of this algorithm.

Advantages –

1) It is usually the model that gives the best Prediction Accuracy.
2) It is Highly Flexible.
3) It can handle missing data.

Dis – Advantages –

1) Less interpretive in Nature.
2) Computationally expensive.
3) It can pretty easily Overfit as GBM always tries to minimize error rate.

## V) Support Vector Machines –

General Idea -

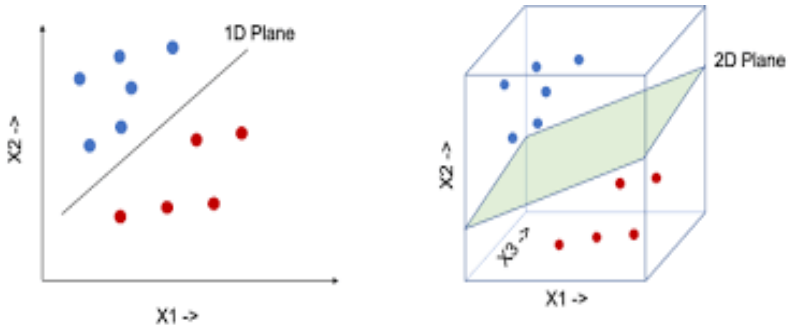
The SVM is a supervised machine learning algorithm which is mostly used for classification type problems. In Simple words to explain the SVM finds the best divide there is between data points that segregates them into distinct categories. It accomplishes by using representations of vectors.

Working / Objective Function -

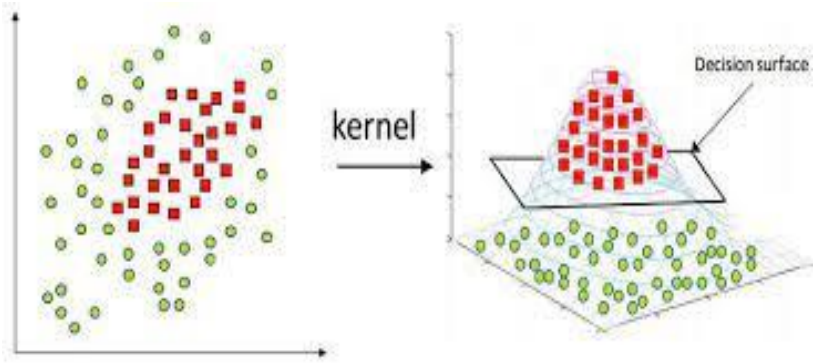
As the Name of the Algorithm suggests we use the help of vectors. Using the method of vectors, we represent data where we plot each data item as a point

in n-dimensional space (where n represent the number of features) with the value of each feature being the value of a particular coordinate.

The Next step after the Vector representation is we try to find the hyperplane that differentiates the two classes very well. We accomplish this by maximizing the distances (Margin) between nearest data point (either class) and a particular hyper-plane. We go for the hyperplane with the maximum distance because then our model will be more robust and won't miss-classify data points.




But how does SVM kernel separate two categories if they are not separable by basic linear splitting. This is where SVM Kernels come into action. The SVM kernel is a function that transforms low-dimensional input space to a higher-dimensional space, i.e. it transforms a not-separable problem into a separable problem.




As a result, it performs some complex data transformations, then determines the best way to separate the data based on what we have defined as labels.

## Advantages –

1) It is effective in high dimensional spaces.
2) It is memory efficient.
3) It works well with a clear margin of separation.

Dis – Advantages –

1) Training time is higher when involved with large datasets.
2) Does not perform well when target classes are overlapping.
3) The probability estimates are computed separately during cross validation.

## Dataset for the Benchmarks & Testing

The Dataset we are going to use for the Benchmarking of Machine Learning algorithms is a Bank Telemarketing Dataset. Using various metric, we have to determine whether a particular client would subscribe or not to the campaign telemarketing we are sending out.

This Dataset has a good mix of both Continuous Variables as well as Categorical Variables which we need to preprocess a little bit before we can go ahead and implement the models on them. After that basically we have to test out which of these models are going to produce the least AUC Test Mean Score through Cross Validation.

## Benchmark Experiment

### I) Variable Selection –

The First step of setting up our benchmark is to of course select the list of variables which are most significant. This needs to be done in order to decrease the computational time taken by the models as after our feature engineering step we have a lot of dummy variables which are created.

**Boruta** is the variable selection method we are going to use for this benchmark experiment. The Boruta is an Automated feature selection process which finds a subset of features from the dataset which are relevant to given classification task. The Core algorithm behind Boruta is Random Forest.

In essence, randomly shuffled attributes are defined to establish a baseline performance for predicting the target variable. Using a hypothesis test, we determine, whether each variable is only randomly correlated. Any variable that does not pass the hypothesis test is discarded.

## II) Cross Validation –

The Cross Validation Setup we have setup for this benchmark experiment has five iterations. This Cross-Validation method evaluates the model performance on different subset of the training data and then calculate the average prediction error rate.

The Setup works as follows –

1) Randomly split the data set into k-subsets (or k-fold) (for example 5 subsets)
2) Reserve one subset and train the model on all other subsets
3) Test the model on the reserved subset and record the prediction error
4) Repeat this process until each of the k subsets has served as the test set.
5) Compute the average of the k recorded errors. This is called the cross-validation error serving as the performance metric for the model.

## III) Evaluation Metric –

As far as AUC score goes the higher the AUC the Better a Model Performs. The Evaluation Metric we are going to use for this setup is the AUC Test Mean values that we get from the Cross-Validation of sets.

The AUC Score would help us identify how different models performs accurately for the same subset of the processed train set of data.

The Given below is the All the mentioned 5 Machine Learning Algorithms and how their AUC Score Looks like -

| Algorithm | Logistic | Decision T | KNN | GBM | SVM |
|---|---|---|---|---|---|
| Auc Test | 0.7844010 | 0.6830197 | 0.7606045 | 0.7933681 | 0.6921302 |

## Conclusion

From the previous set of AUC Test scores we can clearly see that the Best Performing Model among the Set of Algorithms is GBM closely followed by Logistic Regression Algorithm & KNN.

The Model which performed the least was Decision Tree with an Score of 0.68 AUC.

Furthermore, these algorithms performance and prediction accuracy can be further improved by tuning their respective Hyper Parameters especially in the case of GBM where it offers a lot of flexibility for Model Tuning.

## References:


1. https://machinelearningmastery.com/relationship-between-applied-statistics-and-machine-learning/
2. https://www.analyticsvidhya.com/blog/2021/09/gradient-boosting-algorithm-a-complete-guide-for-beginners/
3. https://monkeylearn.com/blog/classification-algorithms/
4. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761
5. shorturl.at/dqBN
6. https://dataaspirant.com/feature-selection-techniques-r/
7. https://www.analyticsvidhya.com/blog/2016/03/select-important-variables-boruta-package/
8. http://www.sthda.com/english/articles/38-regression-model-validation/157-cross-validation-essentials-in-r/


THANK YOU