1. ***Linear Regression Models***

Models that work better with numerical variables. It is basically based on the sum of attribute values, except that weights are applied to each attribute before adding them together. The weights are commonly defined by a method called least squares.

Regression analysis is used for a huge variety of tasks and it is the most widely used method. Some of common uses are:

* Examining populations and individuals by their measured characteristics in all fields of study.
* Identifying patterns that can be used to forecast future behavior given know criteria, such as election results.

*Can be evaluated by*:

* Confusion Matrix.
* Accuracy, Recall, Precision and F1 scores.

*Pros*:

* Simple and efficient.
* Low variance.
* It provides probability score for observations.

*Cons*:

* It does not handle large number of categorical features/variables well.
* It requires transformation of non-linear features.
  1. **Logistic *Regression:***

It is a variance of linear regression model focused in a classification algorithm, that is used where the response variable is categorical. The idea of Logistic Regression is to find a relationship between features and probability of particular outcome.

Logistic regression attempts to produce accurate probability estimates by maximizing the probability of the training data.

*Types of logistic regression are*:

* Binomial Logistic Regression: it has only two target values for the target variable.
* Multinomial Logistic Regression: it has more than two target values for the target variable.
  1. **Perceptron**

It is a variance of linear regression model which has a different approach to learn a hyperplane that separates the instances pertaining to the different classes. If the data can be separated using hyperplane, it is said to be linearly separable. It turns out that if the data is linearly separable, there is a very simple algorithm for finding a separating hyperplane.

1. ***Decision Trees***

Decision tree learners are powerful classifiers that utilize a tree structure to model the relationships among the features and potential outcomes. The tree name is due to the fact that it mirrors the way a literal tree begins at a wide trunk and splits into narrower and narrower branches as it is followed upward [1].

Some of its potential uses include:

* Credit scoring models in which the criteria that cause an applicant to be reject need to be clearly documented and free from bias.
* Marketing studies of customer behavior, such as satisfaction or churn, which will be shared with management or advertising agencies.
* Diagnosis of medical conditions based on laboratory measurements, symptoms, or rate of disease progression.

*Pros:*

* Reduce bias and variance.

*Cons*:

* It can overfit.

*Can be evaluated by*:

* Confusion Matrix.
* Accuracy, Recall, Precision and F1 scores.
  1. ***ExtraTreesClassifier:***

It is an ensemble learning method fundamentally based on decision trees. It randomizes certain decisions and subsets of data to minimize over-learning from the data and overfitting.

It builds multiple trees and splits nodes using random subsets of features, but it does not bootstrap observations (meaning it samples without replacement), and nodes are split on random splits, not best splits.

In Extra Trees, randomness doesn’t come from bootstrapping of data, but rather comes from the random splits of all observations.

As in random forests, a random subset of candidate features is used, but instead of looking for the most discriminative thresholds, thresholds are drawn at random for each candidate feature and the best of these randomly-generated thresholds is picked as the splitting rule. This usually allows to reduce the variance of the model a bit more, at the expense of a slightly greater increase in bias.

* 1. ***Gradient Tree Boosting:***

Ensemble learning method based on decision tree is a generalization to arbitrary differentiable loss functions. It supports both binary and multi-class classification.

Boosting is an ensemble technique in which the predictors are not made independently, but sequentially.

* 1. ***AdaBoost*:**

The core principle of AdaBoost is to fit a sequence of weak learners (i.e., models that are only slightly better than random guessing, such as small decision trees) on repeatedly modified versions of the data. The predictions from all of them are then combined through a weighted majority vote (or sum) to produce the final prediction.

* 1. ***Random Forest***

1. **Naïve Bayes**

Naïve Bayes is a model of linear classifiers that are known for being simple yet very efficient. The probabilistic model of naïve Bayes classifiers is based on Bayes' theorem, and the adjective naïve comes from the assumption is often violated, but naïve Bayes [2].

Gaussian Naive Bayes

1. ***Support Vector Machines***

SVM is a supervised machine learning model that can be used for classification or regression problems. It has two methods: Support Vector Classifier, which is used for classification and Support Vector Regressor, which is used for regression. The main idea of this model is to find the hyperplane that separates the higher dimensional data into classes

*Pros*:

* Effective in high dimensional spaces.
* Still effective in cases where number of dimensions is greater than the number of samples.
* Uses a subset of training points in the decision function (called support vector), so it is also memory efficient.

Cons:

* If the number of features is much greater than the number of samples, avoid over-fitting in choosing kernel functions and regularization term is crucial.
  1. **SVC**

Support Vectors Classifier tries to find the best hyperplane to separate the different classes by maximizing the distance between sample points and the hyperplane. It uses the kernel, gamma, c and degree parameters to tune the model.

* 1. **LinearSVC**

1. **Stochastic Gradient Descent**

Stochastic gradient descent (SGD) is a simple yet very efficient approach to fit linear classifiers models like Support Vector Machines and Logistic Regression. It is a good option when the number of samples (and number of features) is very large.

Pros:

* Efficiency
* Ease of implementation (many opportunities for code tuning).

Cons:

* It requires a number of hyperparameters such as the regularization parameter and the number of iterations.
* It is sensitive to feature scaling.

1. **K-nearest Neighbors Algorithm**

kNN classifier is to classify unlabeled observations by assigning them to the class of the most similar labelled examples [6]. Features of the observations are collected for both training and test dataset. KNN is the most straightforward model to understand and apply**.**