Logistic Regression:

Step1:Import the dataset and segregate the the independent and dependent variables

Step 2: Then apply logistic regression to our dataset.

Formulas : F(x) = 1/

is the slope

is the y-intercept

X is independent variable.

Step3 : Predicting the values.

Knn algorithm:

Step1 : First fix value of K that means number of neighbours.

Step2:Then calculate the distance between the points by Euclidean distance.

Example : A(1,2) and B(3,4)

Distance =

Step3:Then calculate the Euclidean distance and group them to nearest neighbours.

Step4:Then calculate the number of k-neighbours in each Category.

Step5:Them assign the new data points to category of k neighbours which has maximum numbers of points.

Decision Tree

Step 1:Inlize a dataset and define the attributes

Step 2 :Calculate the entropy and the information gain of each and every attribute that present in the dataset

Formula of Entropy : E(T,X) =

Pi = probability of an event of that attribute

Information Gain = Entropy(T) – Entropy(T,X)

Step 3 : Select a attribute from the dataset which had high information gain or low entropy as the root node of tree.

Step 4 :Split the Dataset as selected attribute to obtain the subset of the data.

Step 5 : Repeat the above steps until the every attribute is selected.

Random Forest :

Step1 : First select the k random attributes from the dataset.

Step2: Construct a decision tree for each selected attribute from dataset.

Step3: Based on result produced by each decision tree. Based on the result voting will takes places.

Step4:Considering the maximum voting will be selected for the predicting the data .

Gradient Boosting :

Step 1 : Initialize the dataset and model with the constant value

And apply the loss function

Step2: Calculate the Gradient and build a new decision tree based on the gradient

Step3: Then the model is ready to predict the values.

XGBoost :

Step1 : First initialize the dataset and a single leaf tree

Step2 :And the we need to calculate the average target attributes for prediction and residuals using the loss function .

Step3 :Calculating the similarity score

Similarity score = Gradient \*(Gradient2/Hessian + )

Gradient2 is squared sum of resuduals

is the regularization hyperparameter

Step 3 : Calculate the information gain .

Information Gain = left Similarity + Right Similarity – Similarity of roots

Step 4:Creating a tree of required length with selective removing unwanted branches and regulating with regulation hyperparameter.

Step5: Then predict the residuals with help of decision tree and calculate the new set of the residuals

New residuals = old residuals + Predicted Residuals

Step 6:Repeat the process for all tress