1) Preprocessing:

First deal with nulls ,I replace null cells with mean of columns if columns are numerical

and with mode of columns if columns are strings ,drop nulls not preferred because data is already small.

Second deal with string columns (features)by get dummies(one hot encoder) and label by replace function or label encoder.

Third check if there is duplicate row and remove.

Fourth check outliers and index of rows that have outliers and drop them to reduce skewness of features.

Fifth split dataset to features and labels

Last step in preprocessing feature scaling on features is important to make gradient descent more faster and make all columns have same range of values.

---

2) Feature selection

It is an important step in order to drop any column that has low correlation between it and label (less than 0.1) , reduce number of features is an important step to increase accuracy.

---

3) Train and Test model

1. Split dataset into data in order to train model and small portion of data (between 20 % and 30%) to test model.
2. Random state to controls shuffling applied to data before applying the split.

---

4) Used algorithms

| Logistic Regression | SVM | Decision Tree | KNN | Random Forrest | XGbosst |
|---|---|---|---|---|---|

# 1)Logistic Regression

Logistic Regression is used for predicting categorical data

Sigmoid function is a function of linear Regression equation (w*x +b) , predict probability of Heart disease is exist ,if probability less than 0.5 then heart disease is not exist and if more than 0.5 (threshold) then heart disease is exist but label is 0(not exist) or 1(exist).

Hyperparameters

1) Solver: used to optimize data ex. lbfgs(best value in our model) deals with small dataset to make output more accurate.
2) C: regularization hyperparameter is used to decrease overfitting range(between 0.5 and 1 to avoid overfitting) , the most suitable value in our model is 1(1 is the best value in our model)

---

# 2)SVM

Support vector machine is a type of classification algorithm that used to classifies data according to the features.

It separates classes by linear kernal ,polynomial ,non linear according to data to predict what class the predicted value belong to ex. we have two classes (have heart disease , does not have heart disease) it separates between two classes and determine whether the predicted value belong to class 1 or class 2 and put the point in class it belongs to.

It should be with large margins and equidistance between support vectors from two classes

Hyperparameters

1) Kernal: it depends on data on scatter plot ,values: linear, poly, rbf(value in our model is poly to separate between classes)
2) C:Regularization parameter default is 10 increasing it reduces error but may leads to overfitting(value in our model is (value in our model is 3)

---

## 3)Decision Tree

Is used in classification is select features with the highest information gain to be splitted (root node) ,if this split have entropy more than zero (impurity) either in left branch or right branch, do another split till reach max depth or zero entropy.

Hyperparameters

1. Max depth:it should not be very large value in order not to increase complexity of algorithm and not to increase error (value in our model is 4)

2.Max _feature =consider number of feature (not all features) to calculate information gain of it in order to choose one of them to be splitted .(value in our model 3)

3.criterion=measure impurity of the split of branch ,value=entropy, gini. (value in our model is entropy)

---

## 4)knn

it is an classification algorithm ,it predict by calculate distance between point and its nearest neighbors by ecludian or matahan .

hyperparameters

1)nearest neighbors=number of nearest neighbor.(best value in our model=4)

2)metric=value(minkowski) with p=1 using Manhattan to calculate distance(this our best value for our model) rather than p=2(eucludian).

---

## 5)Random forest:

Is used in classification is select features with the highest information gain to be splitted (root node) ,if this split have entropy more than zero (impurity) either in left branch or right branch, do another split till reach max depth or zero entropy. It is better than decision because it consists of many trees by sample with replacement from original training set to build more trees.

Hyperparameters:

1.Max depth:it should not be very large value in order not to increase complexity of algorithm and not to increase error (value in our model is 4)

2.Max _feature =either log2 or sqrt , consider number of feature (not all features) to calculate information gain of it in order to choose one of them to be splitted .(value in our model is sqrt)

3.criterion=measure impurity of the split of branch ,value=entropy, gini.(value in our model is entropy)

4.number of trees= value in our model(100)

## 6)Xgboost:

- XGBoost stands for Extreme Gradient Boosting.
- It is a tree boosting algorithm that can be used for both classification and regression tasks.
- XGBoost is a popular choice for machine learning competitions because it is very efficient and can achieve state-of-the-art results.
- XGBoost has many hyperparameters that can be tuned to improve the performance of the model.
- Some of the most important hyperparameters include the learning rate, the number of trees, the maximum depth of each tree, and the regularization parameters.
- The optimal values for these parameters will vary depending on the dataset and the task at hand. It is important to experiment with different values to find the best results.

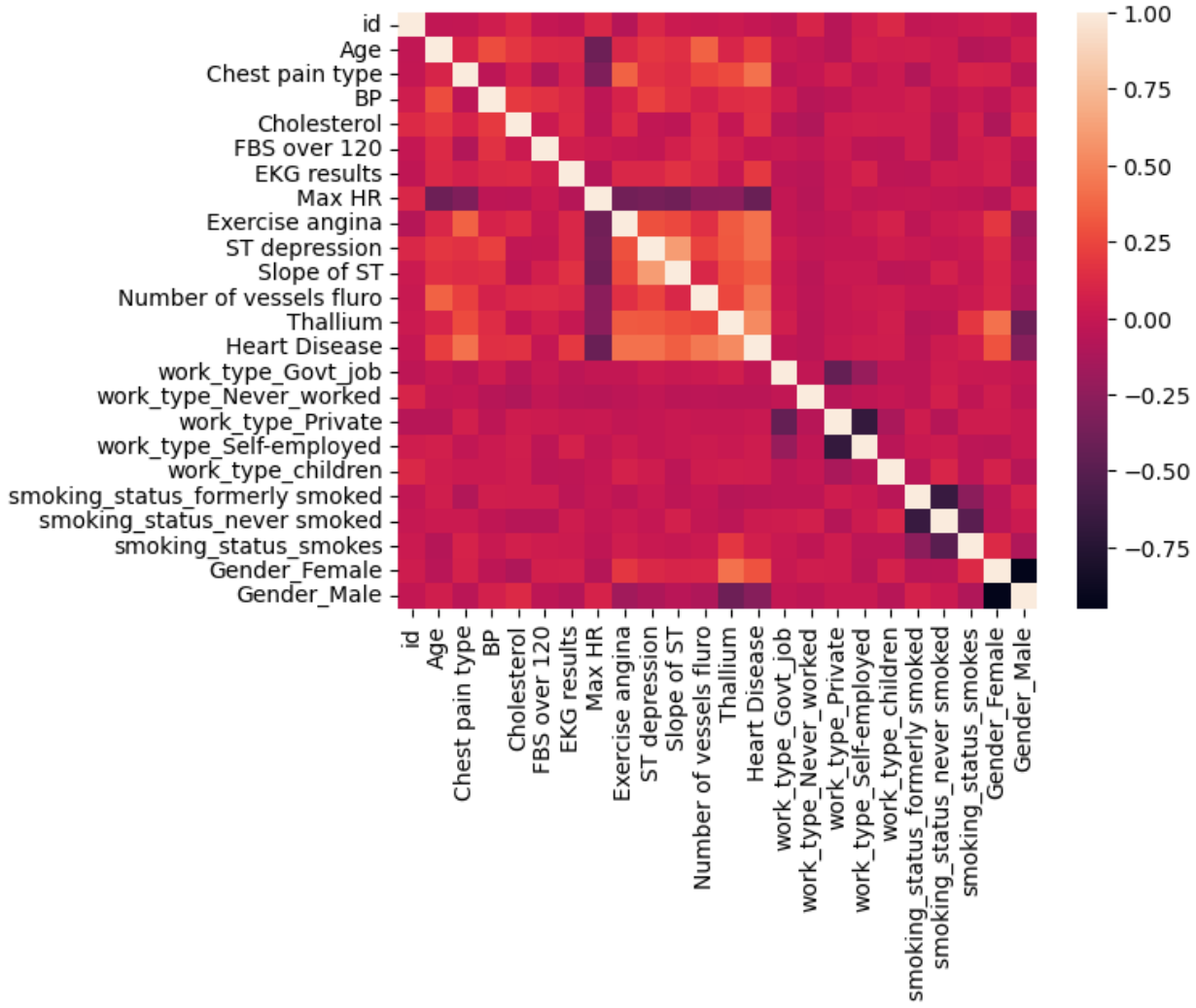
## Final conclusion

| Logistic | Svm | Decision tree | Knn | Random forest | Xg boost |
|---|---|---|---|---|---|
| Test acc=0.849 | Test acc=0.864 | Test acc=0.83 | Test acc =0.849 | Test acc=0.886 | Test acc=0.862 |
| Train acc=0.84 | Train=0.878 | Train acc=0.849 | Train acc=0.849 | Train acc=0.849 | Train acc=0.848 |

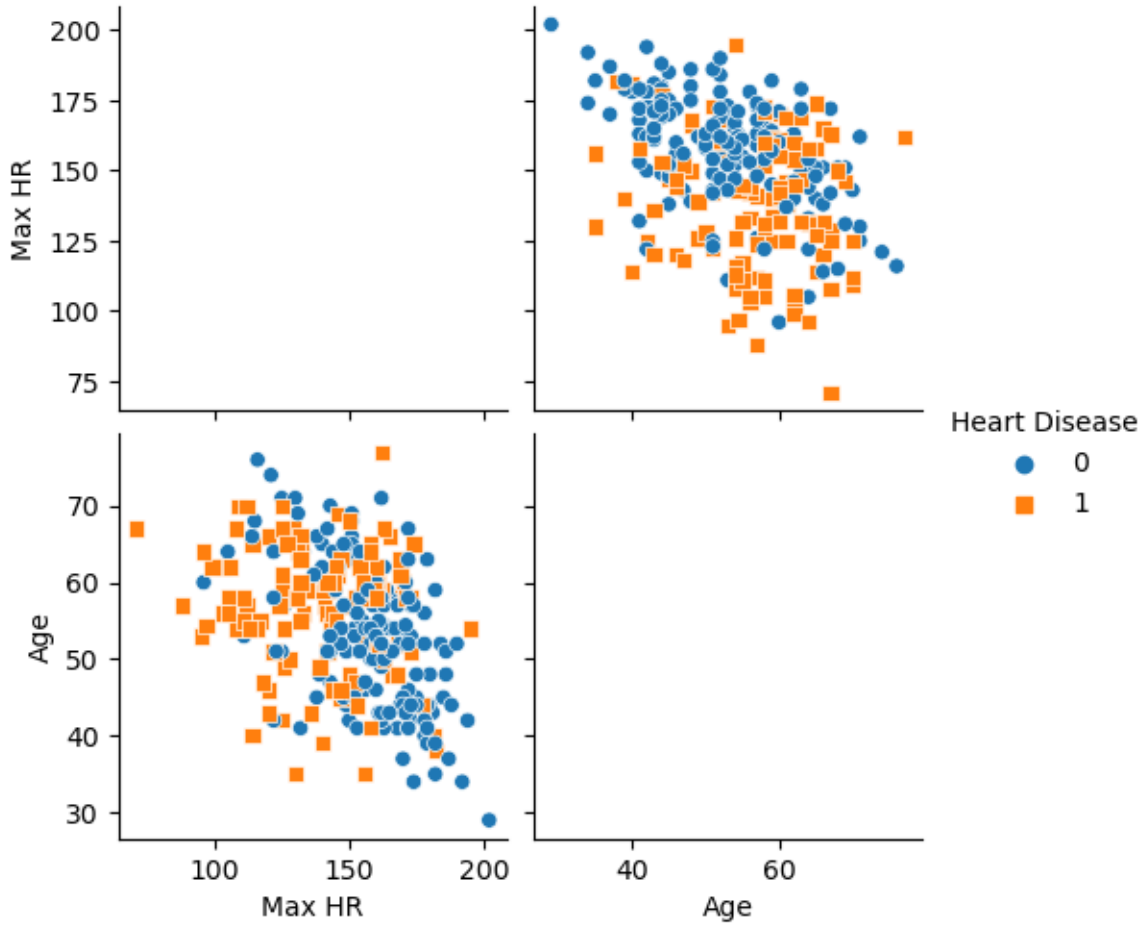
## Finally

This accuracy in each model is the best we reach ,sometimes we have very high test acc but train acc less than high by more than 4% which is not good so we should balance between them to avoid underfitting or overfitting.

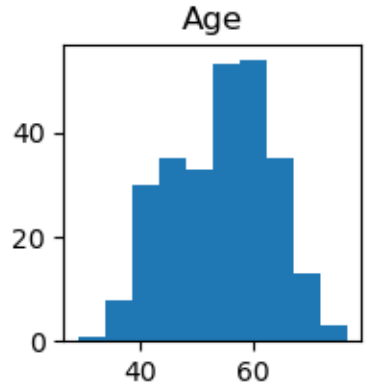
## Correlation between features and label

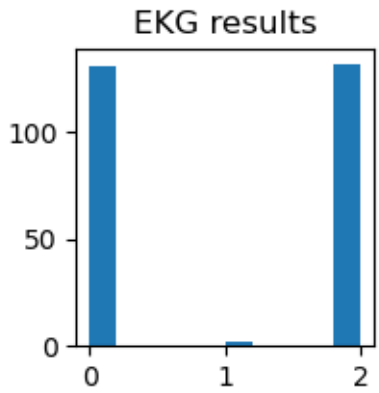


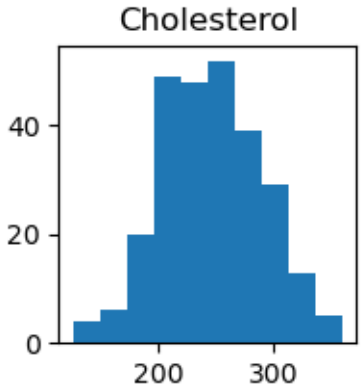

## SCATTER plot between any to columns to know type of kernal that separate two classes




Histogram to test skewness of columns




EKG results
100
50
0
0
1
2


Cholesterol
40
20
0
200
300