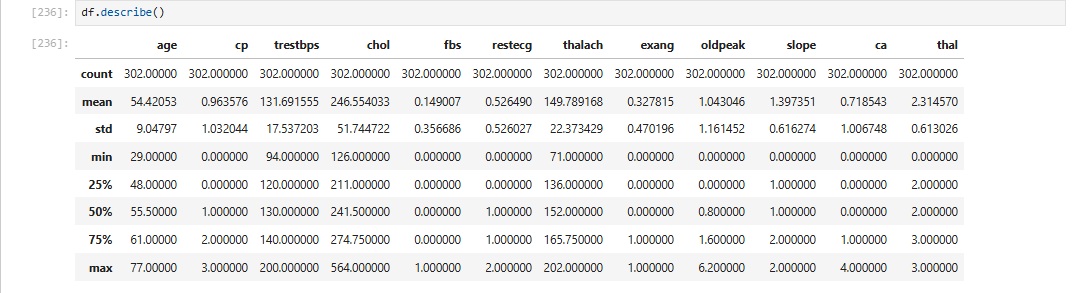
# Dataset Insights

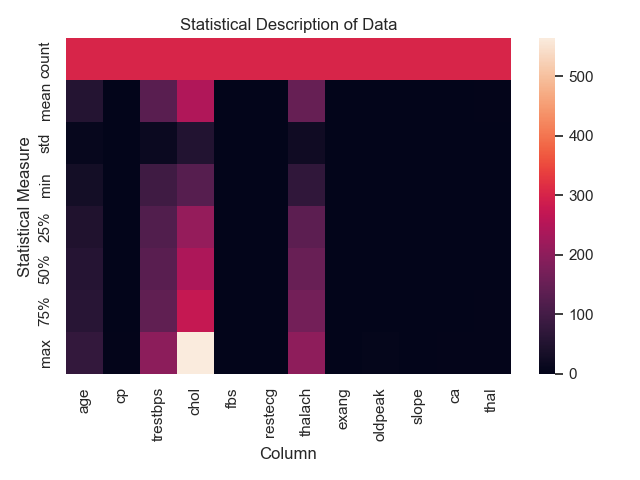
**Dataset Source: https://www.kaggle.com/datasets/sumaiyatasmeem/heart-disease-classification-dataset**

* Dataset is of Heart Disease Classification.
* It has 303 observations out of which 2 are identical (duplicated). First observation has been kept in the data while last one has been discarded.
* There are three columns that have null values which have been imputed with respective mean value.

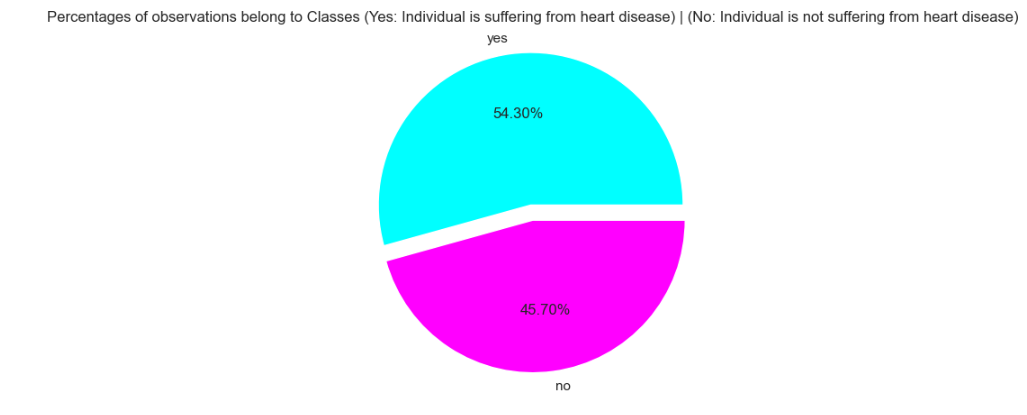
|  |  |
| --- | --- |
| **Column Name** | **Number of Null Values** |
| Trestbps | 4 |
| chol | 1 |
| Thalach | 5 |

* There are 206 entries of male and 96 entries of female in the dataset.
* Some basic statistics of the dataset have been given below:





* There are 164 patients suffering from heart disease while 138 patients who are not suffering from heart disease. There is a class imbalance problem.



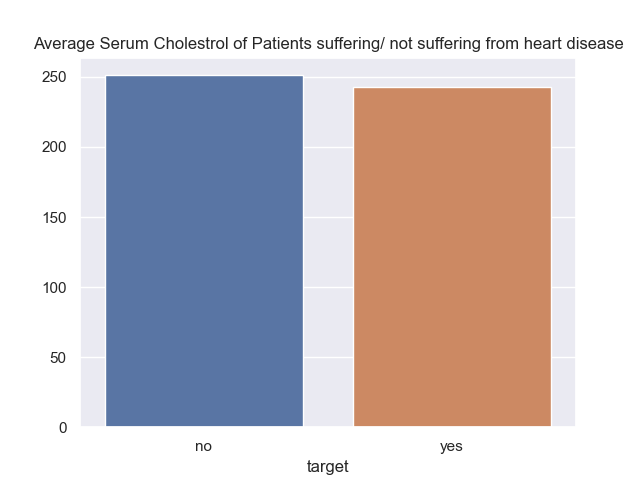
## Gender Distribution for the Patients suffering from heart disease

* Among the people 164 suffering from heart disease, 92 are male, 72 are female

## genderdist

## Average Serum Cholestrol of Patients suffering/ not suffering from heart disease

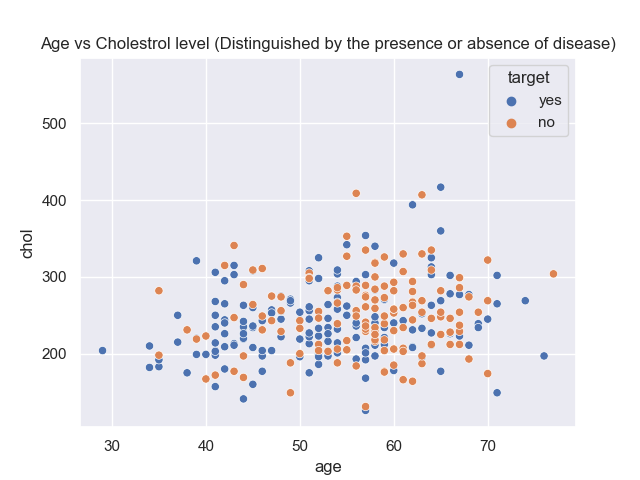
* Mean of Serum Cholestrol of Patients having heart disease: 242.640244
* Mean of Serum Cholestrol of Patients not having heart disease: 251.205202



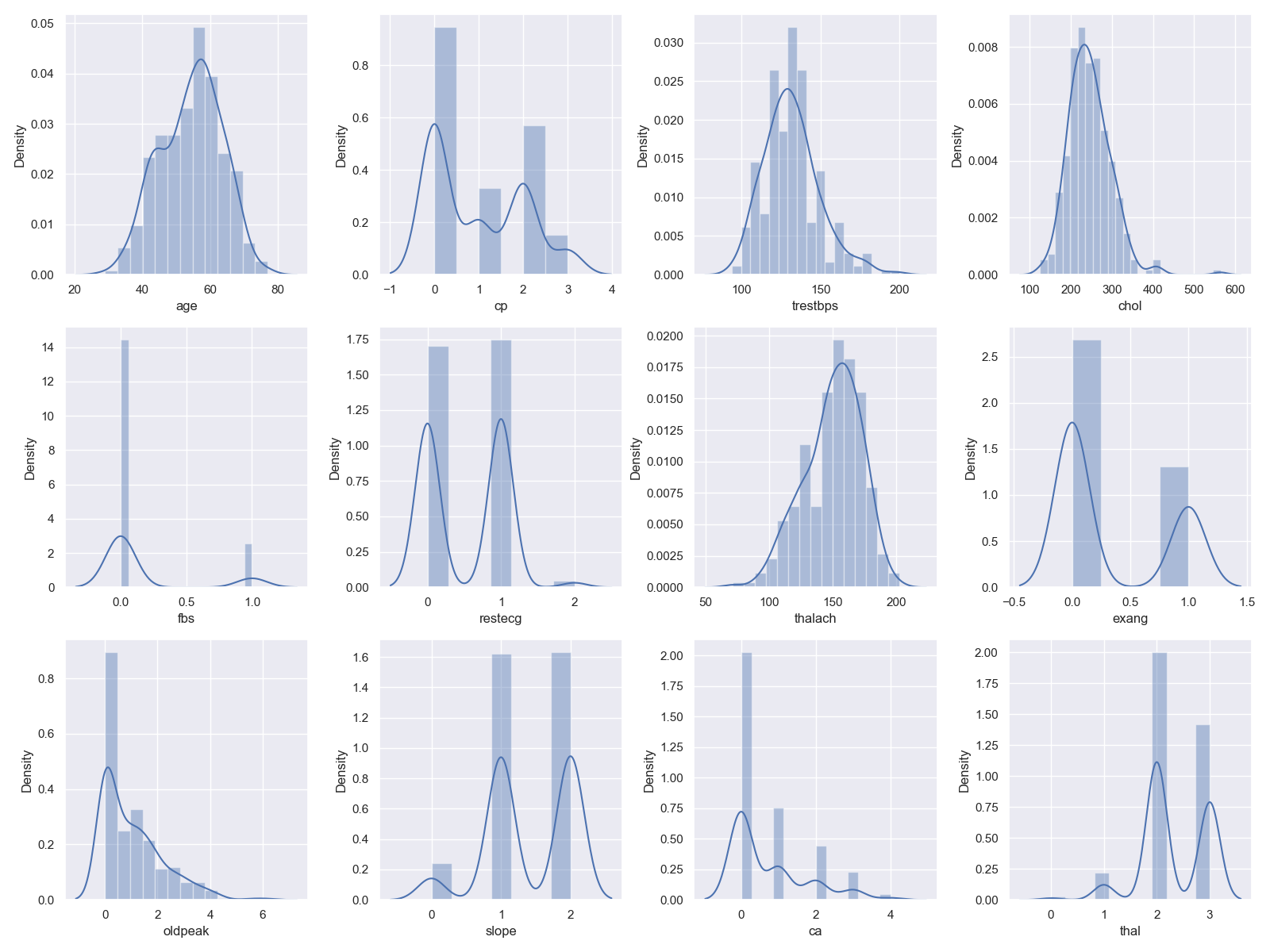
## Age Range for patient suffering from heart disease[¶](http://localhost:8889/notebooks/Desktop/BootcampAI_WP_ICODE/Assignment2_Hassan Aftab.ipynb" \l "Age-Range-for-patient-suffering-from-heart-disease" \t "http://localhost:8889/notebooks/Desktop/BootcampAI_WP_ICODE/_self)

* Minimum Age of a patient suffering from heart disease: 29 years
* Maximum Age of a patient suffering from heart disease: 76 years
* Average Age of a patient suffering from heart disease: 52.58536585365854 years

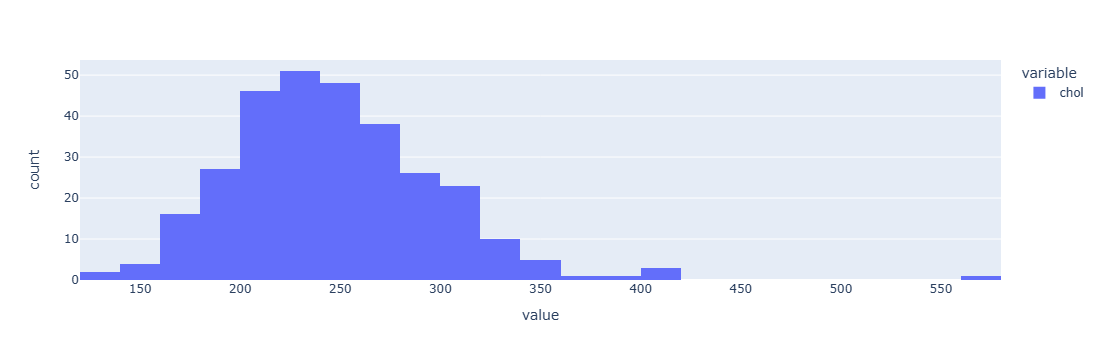
**Age vs Cholestrol level (Distinguished by the presence or absence of disease)**



## Distributions (probability density function and histogram) plots of respective features (non object data type)



## Most People's chol level lies in the range of 220-239 (51 People): Interactive visualization



**Feature Engineering**

* After doing the train test split, the column “sex” has been encoded using one hot encoding by removing the first column after being encoded to avoid multicolinearity.
* The targets y\_train and y\_test have been encoded too by Label Encoder.
* Standard scaler has been used to scale the input features.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1-score |
| KNN Classifier | 0.918033 | 0.922484 | 0.918033 | 0.918033 |
| Random Forest Classifier | 0.868852 | 0.870862 | 0.868852 | 0.868923 |
| Decision Tree Classifier | 0.770492 | 0.776666 | 0.770492 | 0.770245 |

* The results of models on the testing data have been given in Table below:

Here we can see that KNN classifier achieved the highest accuracy score.

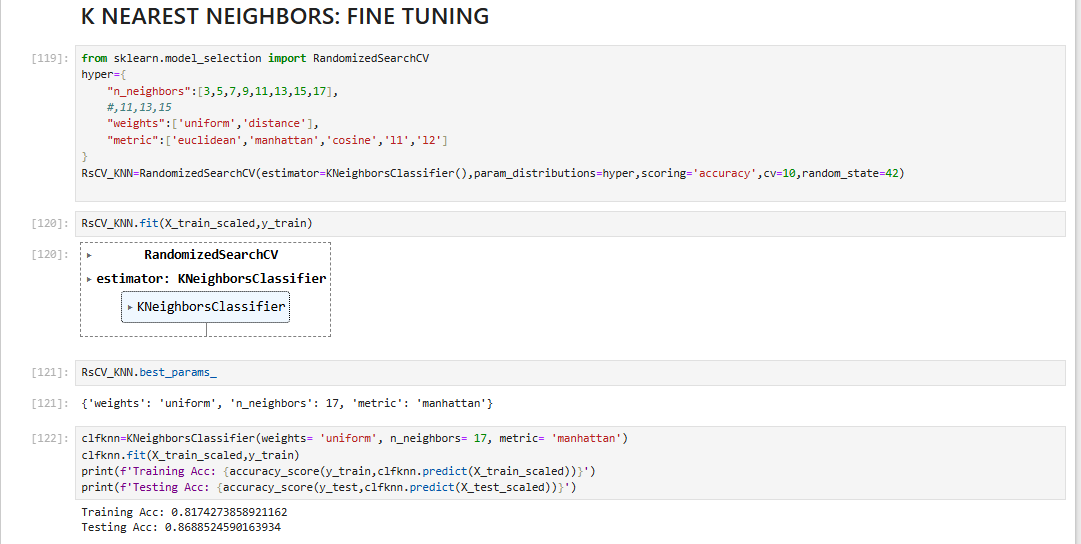
**Overfitting Scenario:**

Now we look towards the overfitting scenario:

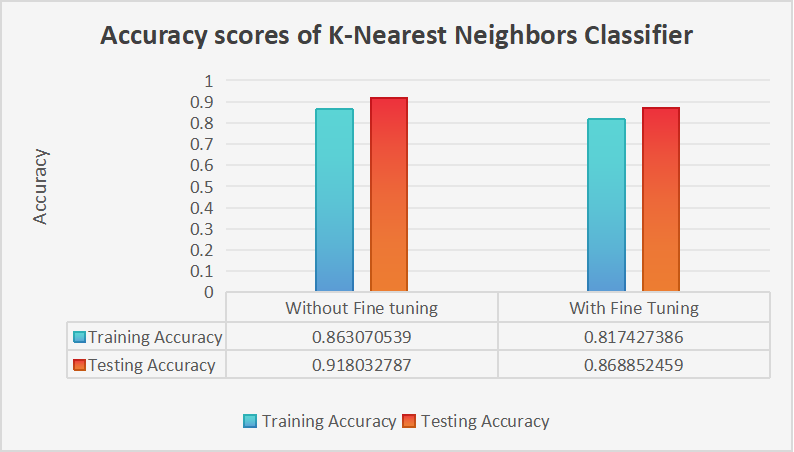
* KNN Classifier Training Accuracy: 0.8630705394190872
* KNN Classifier Testing Accuracy: 0.9180327868852459
* Decision Tree Classifier Training Accuracy: 1.0
* Decision Tree Classifier Testing Accuracy: 0.7704918032786885
* Random Forest Classifier Training Accuracy: 1.0
* Random Forest Classifier Testing Accuracy: 0.8688524590163934

It is apparent from above that results of Decision Tree Classifier and Random Forest Classifier are showing over-fitting.

**Fine Tuning using Random Search CV:**



In the case of KNN, With mentioned dictionary of parameters, you can say search space: after fine tuning, Both the training as well as testing accuracy decreased but again overfitting doesn’t seem to take place.

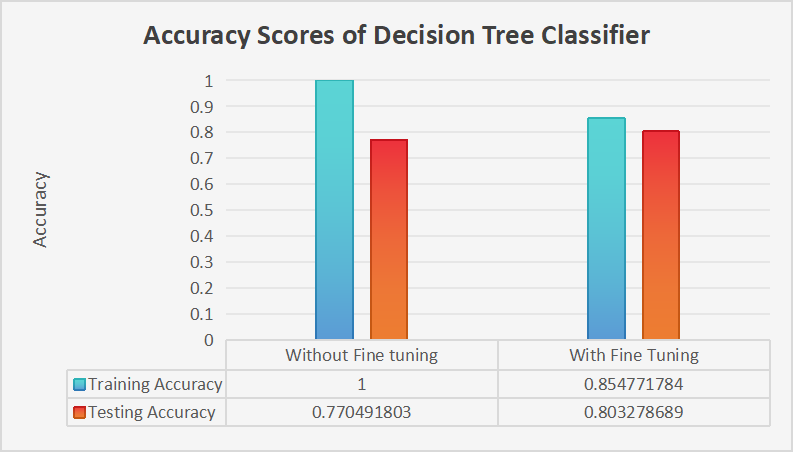


In case of KNN Classifier, we can see that even before fine tuning overfitting wasn't taking place and after fine tuning the accuracy on both the train as well as test data decreased but overfitting didn't take place.

**Decision Tree Classifier**

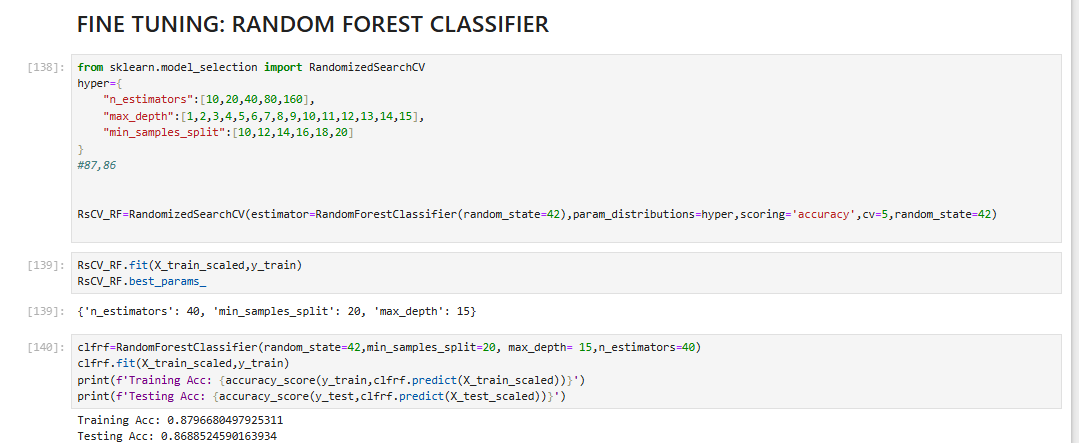


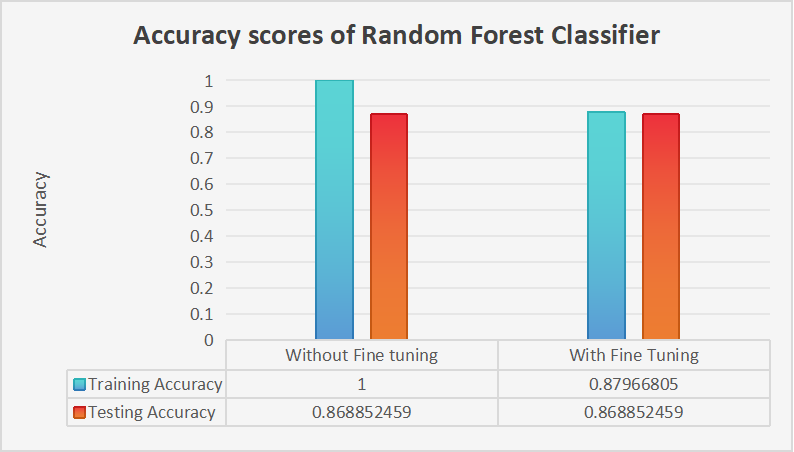
In the case of Decision Tree Classifier, after fine tuning the training accuracy saw a dip and the testing accuracy saw a rise. A big drop in the overfitting took place.



Overfitting upon fine tuning reduced to a very good extent as the gap between the accuracy score on training and test data reduced. Not only that the accuracy score on training data reduced but also the accuracy on test data increased.

**Random Forest Classifier**



In the case of random forest classifier, earlier accuracy scores were 1.0 and 0.8688 for training and testing data respectively. But now you can see here that overfitting is quite reduced as there is a very small gap between the training and accuracy scores after fine tuning of model.  
  


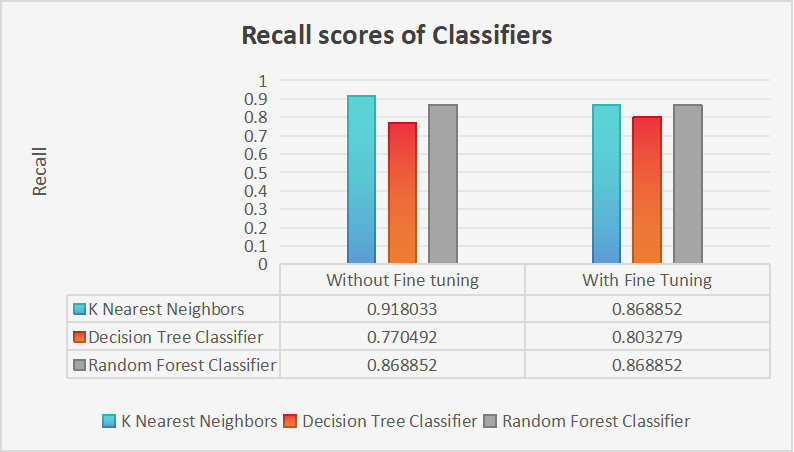
In the case of random forest classifier, earlier accuracy scores were 1.0 and 0.8688 for training and testing data respectively. But now you can see here that overfitting is quite reduced as there is a very small gap between the training and accuracy scores after fine tuning of model.

**MOST ACCURATE MODEL:**

* KNN Classifier when trained (without fine tuning) gave the best accuracy on the test data (0.9180327868852459). So, it can be concluded that KNN generalized well on the test data even without fine tuning.
* KNN is the only algorithm that is giving the testing accuracy greater than the training accuracy in either for the cases: with or without fine tuning.

**BEST MODEL BASED ON PROBLEM:**

Since, this is a disease classification problem, False Negatives mistakes are more dangerous. So, we should prefer a high recall model.



K-Nearest Neighors Classifier without fine tuning strategy gave the highest recall score on the test data.

KNN Classifier (without fine tuning) also has the highest AUC score as shown below:

