**Lab 4 Report**

**Vehicle fleet management**

**Task 1 “A”: Implement k-NN classifier:**

It is also called lazy algorithm and nonparametric algorithm used for both classification and regression algorithms. KNN classification is used to classify the data into different categories like(true or false,1 or 0,yes or no).KNN -regression, the output is numerical value.

The machine learning algorithm is divided into two categories into different categories:

1)Fit(Training the data to the algorithm)

2)Predict(testing the algorithm based on unseen data or unused data)

In KNN-algorithm there is nothing called training .It just save the training data in the memory during training. During the testing state the KNN algorithm will find it’s nearest neighbours and gives the result based on it’s nearest neighbours.

In KNN-classification algorithm the result of the test data or sample will be classified based on k training sample nearest to the test sample. The commonly used distance metrics is Euclidean distance which is used to find the nearest neighbours.

In evaluation result I used confusion matrix it gives num[[Tp,Tn][Fp,Fn]]

Tp,Tn: True positive , True negative

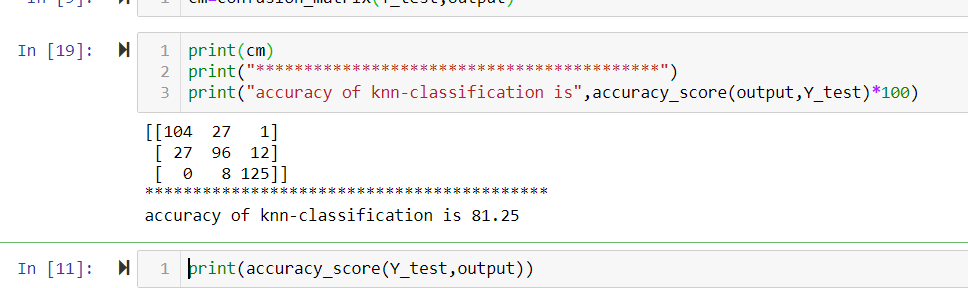
Fp,Fn: False positive , False negative

B)The evaluation result of KNN algorithm is:

The number of correct prediction is 325 out of 400

Accuracy percentage is 81.25%

Confusion matrix of KNN-classification is:



**Task 1 “B”: Investigate different classification algorithms**

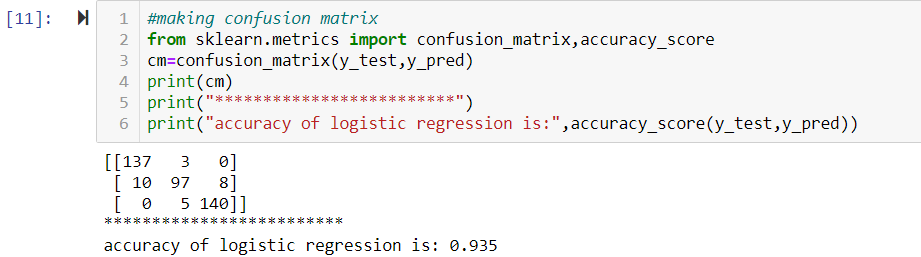
**1)Logistics Regression:**

It takes the line equation y=c+m1x+m2x+m3x…….

Result:

The number of correct prediction is 325 out of 400

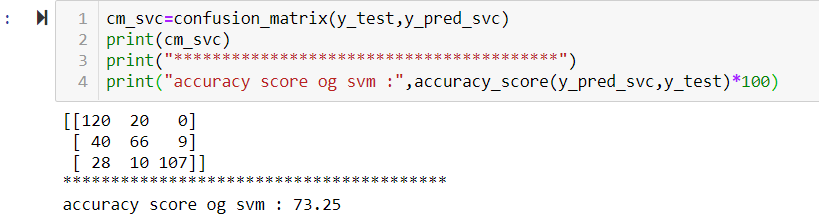
Accuracy of logistics regression is 93.5%



**2)Support vector machines(SVM) classifier:**

The number of correct prediction is 293 out of 400

Accuracy of logistics regression is 73.25%



### 3) Gaussian naive bays:

The number of correct prediction is 374 out of 400

Accuracy of logistics regression is 95.25%

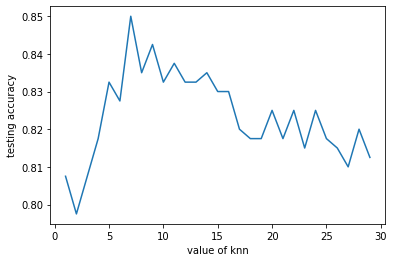
### 

**Task 1 “C”: Parameter tuning:**

**Euclidean distance:**

In parameter tuning the x-axis is k-neighbours(different values of k) an Y-axis testing accuracy from the below graph we can say that at k=7 we have the highest accuracy.

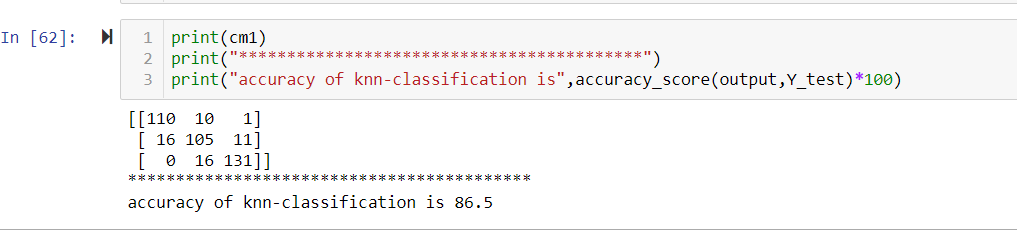
The distance measurement method is Euclidean distance.



The result at k=7 when the distance measure is Euclidean distance is:

The total number of correct predictions is 346 out of 400

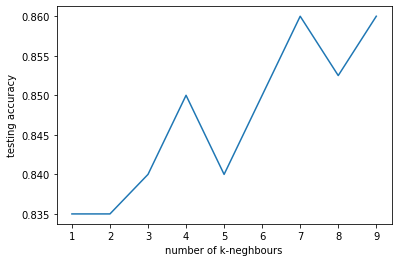
The accuracy is 86.5%



In parameter tuning the x-axis is k-neighbours(different values of k) an Y-axis testing accuracy from the below graph we can say that at k=4,7 we have the highest accuracy.

**Manhattan distance:**

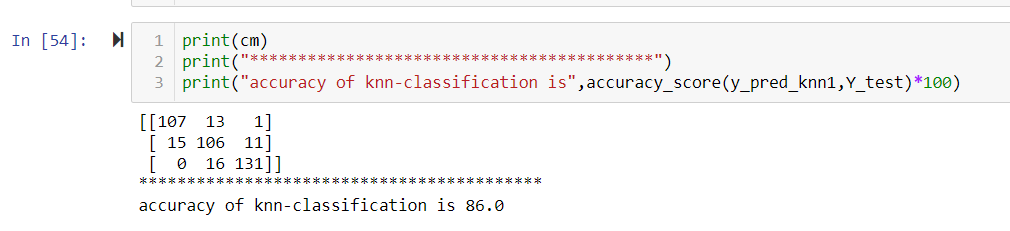
The distance measurement method is Manhattan distance



The result at k=7 when the distance measure is Manhattan distance is:

The total number of correct predictions is 344 out of 400

The accuracy is 86%



**Task 1 “D”:**

The test case accuracy of

knn for n-neighbours=7 is 83.5%

Logistic Regression accuracy: 92.25%

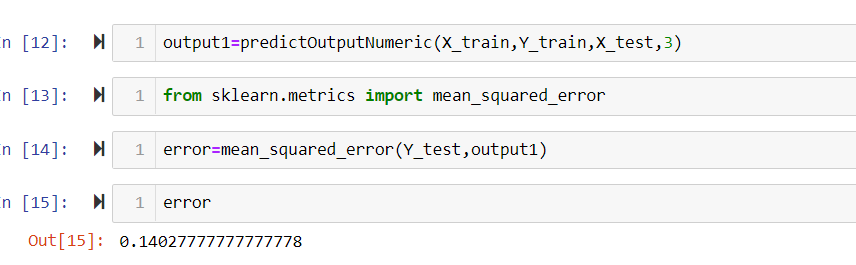
**Task 1 “E”: Implement k-NN regression:**

The KNN regression algorithm is same as KNN-classification algorithm .In knn-classification algorithm the output belongs to some class but in KNN-regression algorithm the output is value. The value assigned to the test sample is calculated is by taking the average of the K-nearest neighbours.

The accuracy is calculated by finding the error(it is calculated by subtracting the original value to the predicted value at that test sample )

If the error should be low the algorithm is best algorithm.

The mean squared error we get when we use number of neighbours is 3.The error we got is 0.1402



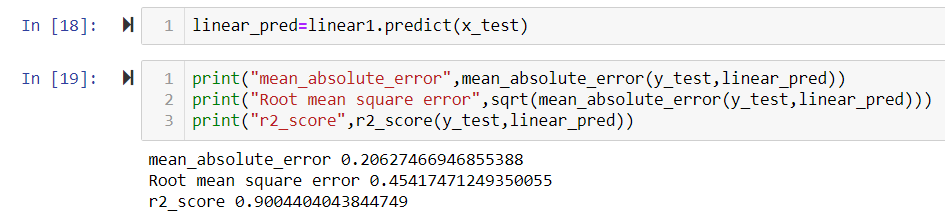
Task 1 “F”: Investigate different regression algorithms :

**1)Linear Regression:**

It takes the line equation y=c+m1x+m2x+m3x…….

**Error:**

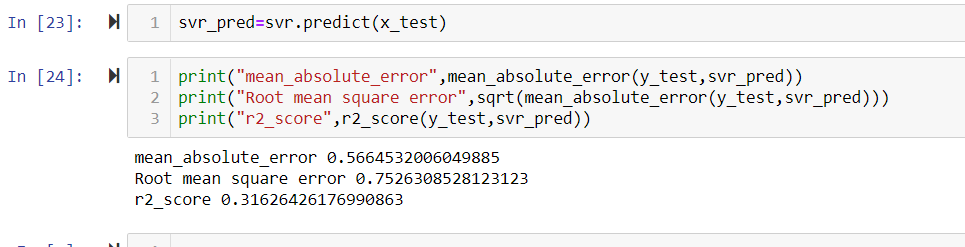
Root mean square error 0.45417471249350055



**2)SVR:**

**Error:**

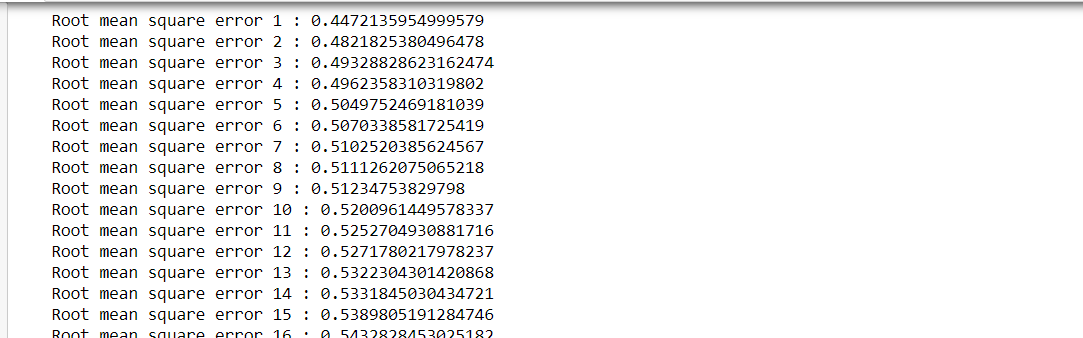
Root mean square error 0.7526308528123123



**Task 1 “G”: Parameter tuning :**

When n-neighbours is 1 we getting low root mean square value.

Root mean square error 1 : 0.4472135954999579



**Task 1 “H”:**

When we are using Linear -regression we are getting best fit line when I compared with other algorithms like Gaussian or svr..etc

**Task 2: Poker Project**

**Task 2 ”A”:**

Few numeric attributes are introduced in the poker project They are:

1)Hand category: Player 1 & Player 2 hand categories varies from 0 to 9

2)Hand rank weights from 0 to 14

3)Coins of player 1 & player 2

4)Number of raises made by player 1 & player 2

5)Average amount of money raised by player 1 & player2

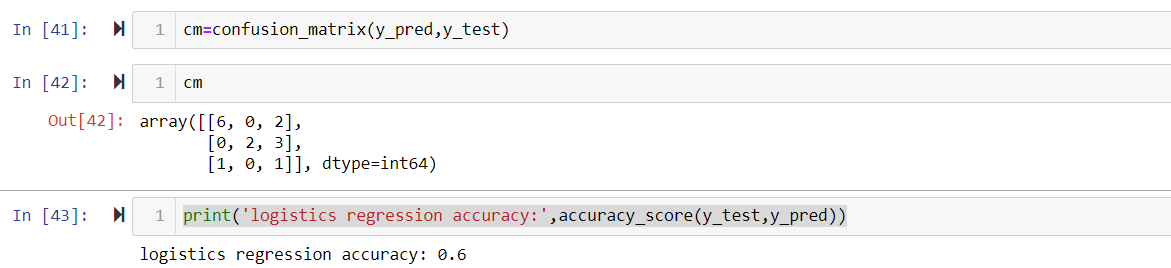
**Task 2 “B”:**

3 different learning algorithms are

**1)Logistic regression:**

The total correct predections are 9 out of 15

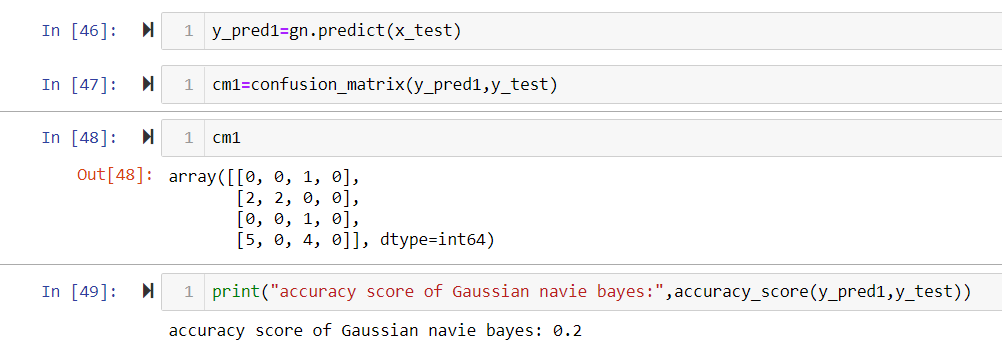
The accuracy is 60%



### 2) Gaussian navie\_bayes[¶](http://localhost:8888/notebooks/lab4/task_2.ipynb" \l "Gaussian-navie_bayes)

The total correct predections are 3 out of 15

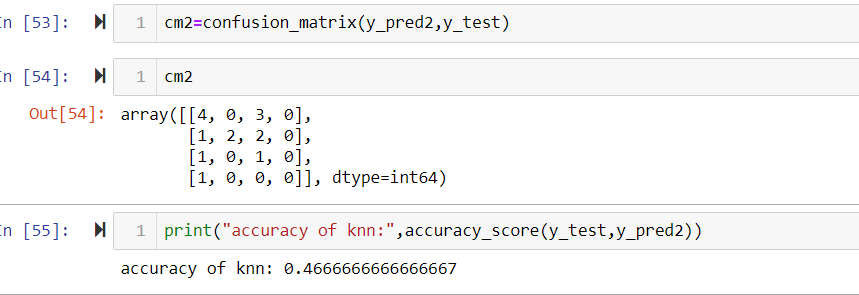
The accuracy is 20%



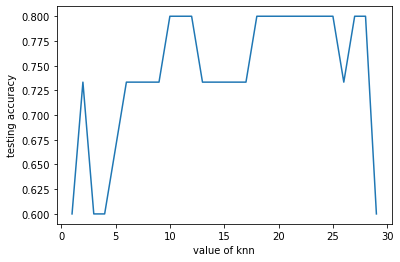
**3)KNN:**

The total correct predections are 7 out of 15

The accuracy is 46.67%



**Task 2 “C”:**

****

From the above figure on x-axis(number of neighbours) ranges from 0 to 30 and on y-axis(test accuracy).

From the above figure we can say that when number of neighbours is 7 we get accuracy of 78%

