Classification| DA-1

2019101101
Stella Sravanthi

I. Cleaning the data

Removing all unnecessary columns for the analysis from the metadata. I've considered LAT (N), LONG (E), DEPTH (km) and all others to be removed.
Then the required dataset is later named as **useful .**
Shape of the dataset is **(52989, 4)**

	DEPTH (km)	LAT (N)	LONG (E)	MAGNITUDE
1	0.0	71	24	7.5
2	0.0	71	24	7.5
3	0.0	72.9	33.72	7.5
4	NaN	17.3	80.1	6.1397
5	80.0	26	97	6.1397
...	...	...	...	...
52985	10.0	32.8°N	78.4°E	3.2
52986	70.0	25.5°N	90.4°E	3.6
52987	22.0	23.2°N	86.5°E	4
52988	20.0	32.8°N	76.4°E	4.3
52989	10.0	20.0°N	72.8°E	3

52989 rows × 4 columns

Modifying the values of the useful columns

Columns like LAT (N) , LONG (E) have values along with symbols(°) and characters(N,S,E,W) which are removed and converted to float.

	DEPTH (km)	LAT (N)	LONG (E)	MAGNITUDE
52985	10.0	32.8	78.4	3.2
52986	70.0	25.5	90.4	3.6
52987	22.0	23.2	86.5	4
52988	20.0	32.8	76.4	4.3
52989	10.0	20.0	72.8	3

Threshold value taken is 4.5 and then values of MAGNITUDE column changed to 0 or 1 depending on that threshold.

[1. 1. 1. ... 1. 1. 1.]

```
useful['MAGNITUDE'].describe()
```

```
count    40107.000000
mean         4.533235
std          0.612819
min          2.000000
25%          4.200000
50%          4.400000
75%          4.800000
max          9.100000
Name: MAGNITUDE, dtype: float64
```

Here I've found 4.5 is the mean of the value. Therefore it is an ideally correct value to be chosen for the threshold.

Dealing with all NULL values

The data set is cleaned when all the null values are taken care.Rows containing NaN or infinity are removed and the number of rows in the dataset from 52989 to 40107.

Splitting the data

Data is divided into X and the features considered for X are Longitude,Latitude, Depth and the value to be predicted Y. Here using the information of whether the earth occurred or not (from X) the magnitude is predicted.
The **training** data and **testing** data (X and Y) in a **70:30 ratio.**

II. KNN

K Nearest Neighbours is applied on the dataset using **KNeighboursClassifier** from **sklearn** module.So neighbours are defined with `knn = KNN(n_neighbors=i)`

Firstly to train the dataset using `knn.fit(X_train,Y_train)` and then predict the result on the test set using `knn.predict(test_X)` . So we got the highest AUC score of **0.56** for K = 2,3,4 respectively 4.5 threshold.

III. Decision Tree

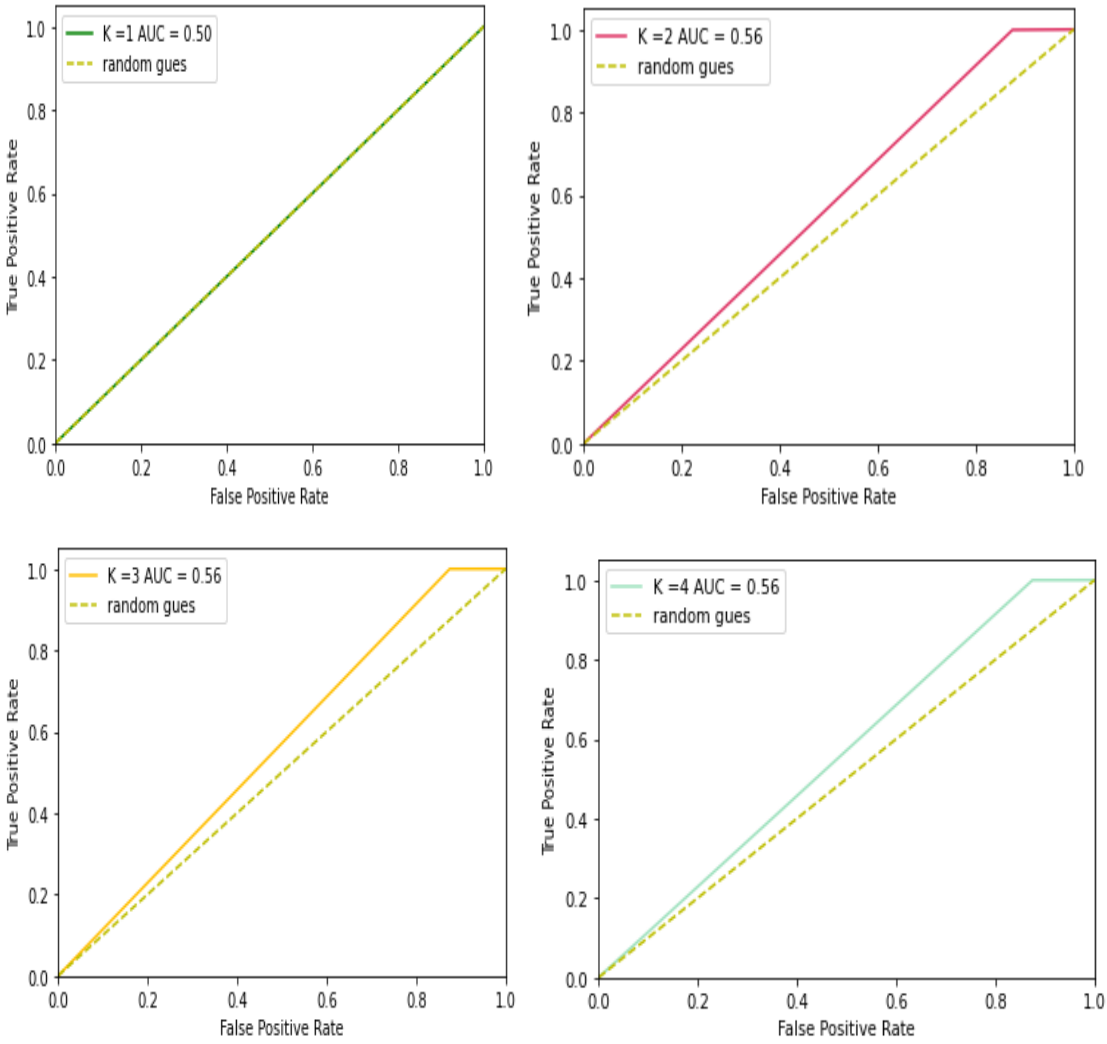
Decision Tree is applied on the dataset using **DecisionTreeRegressor** from **sklearn** module.So depth is defined with `dt = DTR(max_leaf_nodes=i, random_state=1).`

Firstly to train the dataset using `dt.fit(X_train,Y_train)` and then predict the result on the test set using `dt.predict(test_X)` . So we got the highest AUC score of **0.62** for depth = 18 to 26 respectively 4.5 threshold.

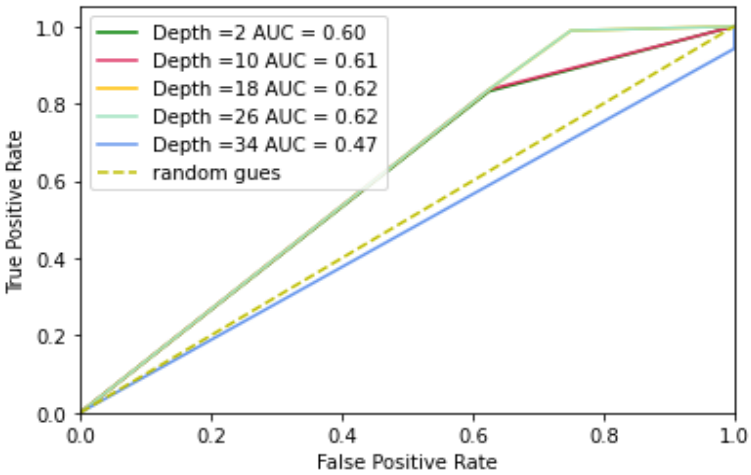
IV. TASKS

1. **Plot ROC for both these classifiers for K as parameter in KNN, preprune depth as a parameter in Decision Tree and number of estimators as parameter in ensemble learning.**

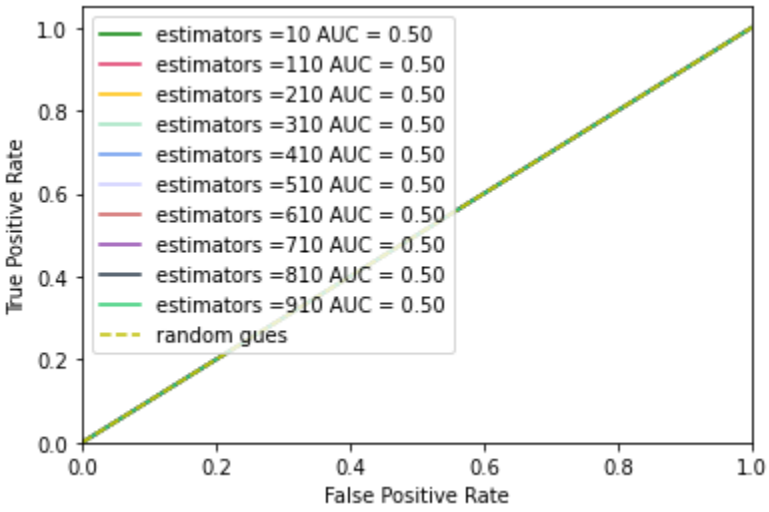
- **ROC for KNN for different values of K**



❖ **ROC for Decision Tree for different depth values**



❖ **ROC for Ensemble learning for different estimator values**



2. **Which is the better classifier for this data amongst the three? Give Reasoning.**

Decision tree seems to be a better classifier for this data and it is better than KNN model and ensemble learning. Because KNN is ideal for distance measures and here we worked with depth and coordinates which are different types of data whereas the decision tree doesn't depend on any relations.Ensemble learning maintained auc score around 0.5 so it can be suitable for dataset where underfitting is seen.But overall I've found Decision tree model is best classifier for this data.

3. **What could be the best possible values of the parameters for each classifier based on the ROC curves? Give Reasoning.**

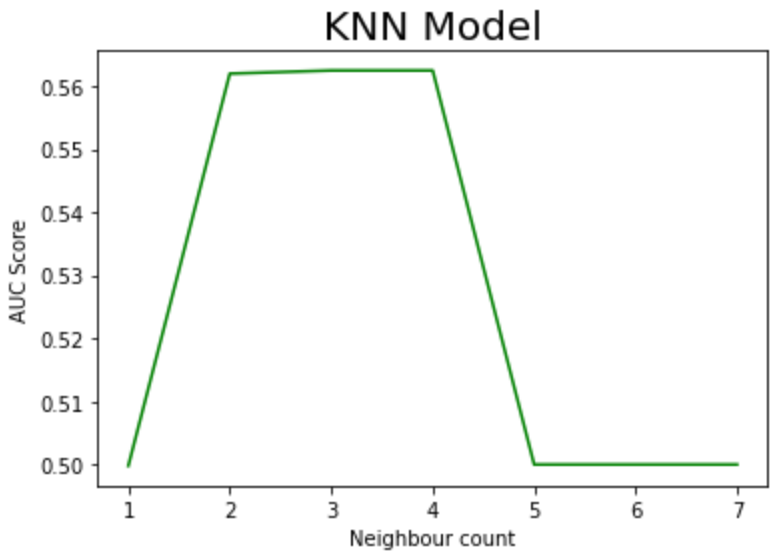
For KNN, K=2,3,4 seem to be giving the best results with Max area = 0.56

In the case, K > 4

leads to overfitting.

In case K < 2

leads to underfitting.



For Decision Tree, depth=18 to 27 seems to be giving the best result with Max area = 0.62

In the case depth > 27

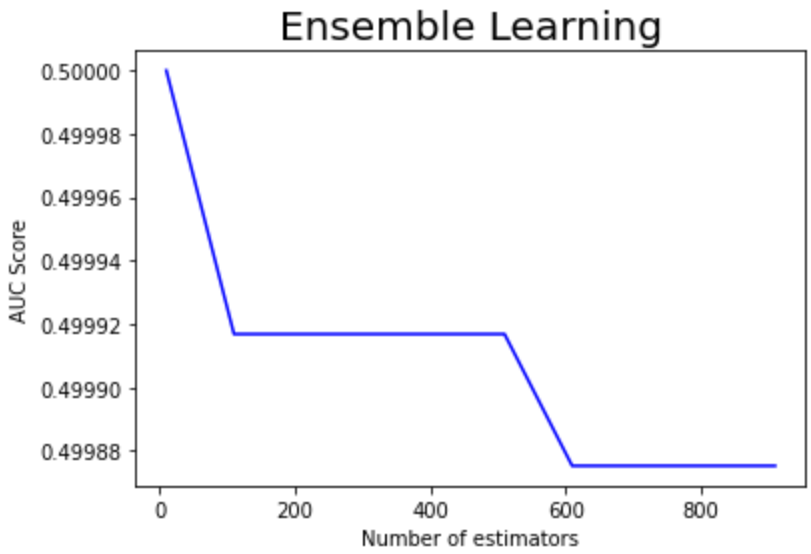
leads to overfitting.

In the case depth < 18

leads to underfitting.

For Ensemble learning ,estimators=1 to 100 seems to be giving the best result with Max area = 0.5

Overly maintained 0.5 just differ in 0.00004



4. **If you have to choose only a subset of two features to predict an earthquake,which ones would it be? Give Reasoning.**

Here from the above performed results . I found that Depth is the most important feature in predicting because along with Latitude and Longitude using Decision trees. SO we can say to predict magnitude the key factor is Depth and it indirectly also on latitude and longitude thus. s on location too.

5. **Consider test results of the best model from above analysis. Report the in-**
   put features that were used to achieve this. Try to improvise the test results
   by applying feature processing(You may come up with additional features
   by processing original ones). Report the new set of features that was used
   and also report the improvements in test results that were achieved. Please use appropriate metrics to report the results.

   Main techniques followed for Decision tree model which leads us to best model:
   - Considered only useful columns that are Latitude, Longitude, Depth, Magnitude.
   - Removing rows , columns with null values.
   - Considered median for all missing values given the max area 0.56
   - Considered most-frequent for all missing values given the max area 0.53

- Considered mean for missing values and 180 degrees as default.Given the max area= 0.58
- Combining two columns (like multiplying) but the present dataset gives best rather than this.