



uOttawa

Assignment 4 Solution

Ubiquitous Sensing for Smart Cities

Group 6:

Abdelrahman Basha

Ahmed Samir

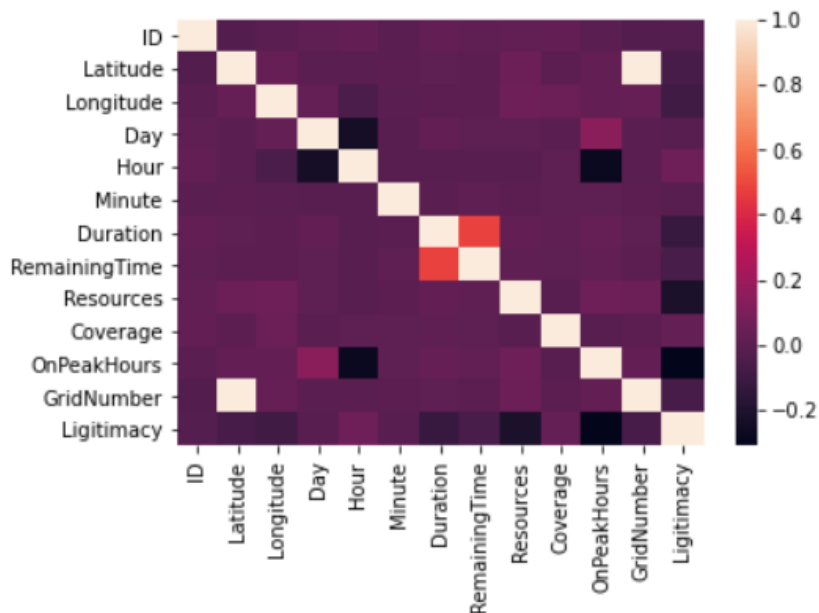
Gehad Abo Kamar

First we need to import the dataset

```
np.random.seed(0)
df = pandas.read_csv('MCSDatasetNEXTCONLab.csv')
```

Then we draw the correlation between the data to remove the high correlation

```
df_corr = df.corr()
sns.heatmap(df_corr)
```



We can see that there is a correlation between 'Latitude' column and 'GridNumber'

So we need to drop the 'Latitude'.

Then we dropped the 'ID' column.

By seeing the 'Ligitimacy' we see and see it's distribution, We can see the '1' values are equal to 12587 and the '0' values are equal to 1897.

So we need to use the SMOTE algorithm to balance the data

Finally, splitting the data to 80% training and 20% testing

```

1 X = df.drop(['ID', 'Ligitimacy', 'Latitude'], axis=1)
2 y= df.Ligitimacy
3 X.head()
4
5 #SMOTE
6 counter = Counter(y)
7 print(counter)
8 oversample = SMOTE()
9 X, y = oversample.fit_resample(X, y)
10 scaler = preprocessing.MaxAbsScaler().fit(X)
11 counter = Counter(y)
12 print(counter)

```

```

Counter({1: 12587, 0: 1897})
Counter({1: 12587, 0: 12587})

```

```

1 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

```

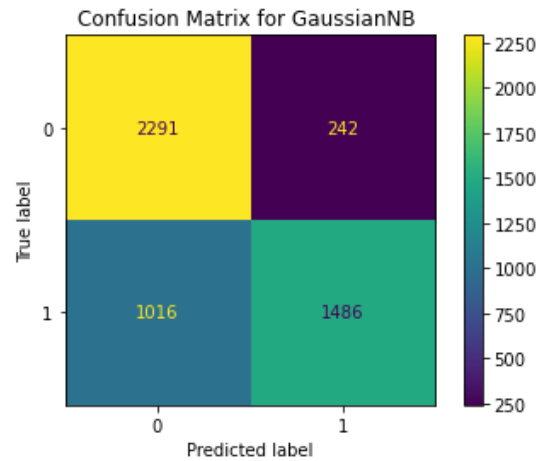
Then we passed these data to **5 different models** ('Naive Bayes', 'Random Forest', 'AdaBoost', 'SVM', 'Decision tree').

-Naive Bayes results:

Accuracy : -> 0.7501489572989076

Classification Report for GaussianNB :

	precision	recall	f1-score	support
0	0.69	0.90	0.78	2533
1	0.86	0.59	0.70	2502
accuracy			0.75	5035
macro avg	0.78	0.75	0.74	5035
weighted avg	0.78	0.75	0.74	5035

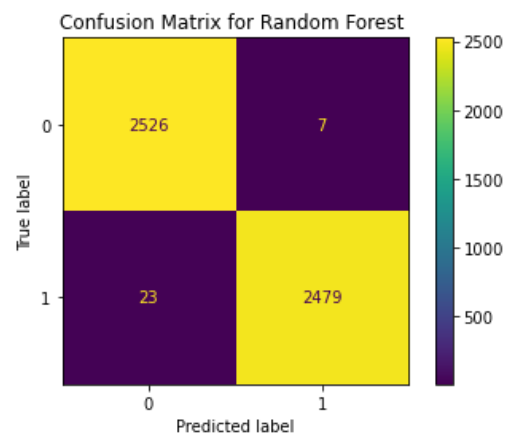


-Random Forest results:

Accuracy : -> 0.9940417080436942

Classification Report for Random Forest :

	precision	recall	f1-score	support
0	0.99	1.00	0.99	2533
1	1.00	0.99	0.99	2502
accuracy			0.99	5035
macro avg	0.99	0.99	0.99	5035
weighted avg	0.99	0.99	0.99	5035

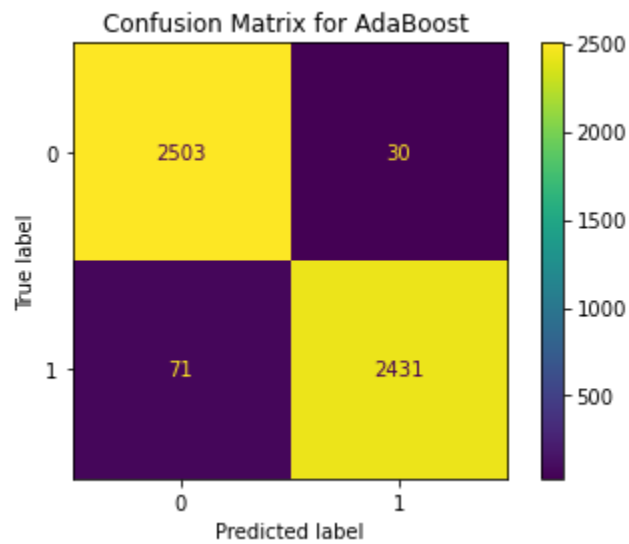


-AdaBoost results:

Accuracy : -> 0.9799404170804369

Classification Report for AdaBoost :

	precision	recall	f1-score	support
0	0.97	0.99	0.98	2533
1	0.99	0.97	0.98	2502
accuracy			0.98	5035
macro avg	0.98	0.98	0.98	5035
weighted avg	0.98	0.98	0.98	5035

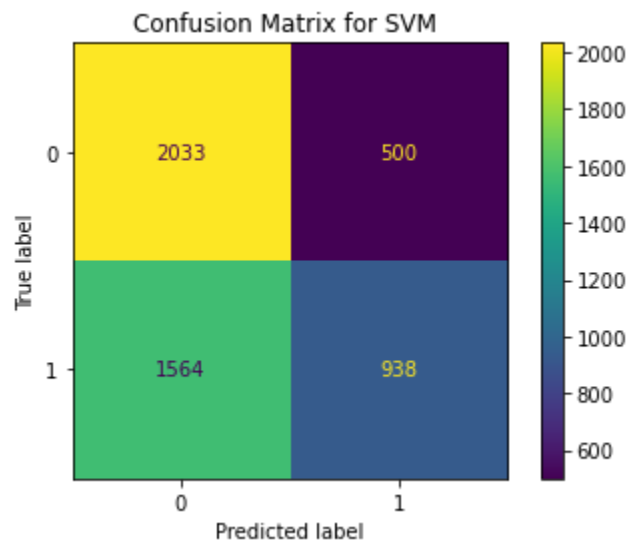


-SVM results:

Accuracy : -> 0.5900695134061569

Classification Report for SVM :

	precision	recall	f1-score	support
0	0.57	0.80	0.66	2533
1	0.65	0.37	0.48	2502
accuracy			0.59	5035
macro avg	0.61	0.59	0.57	5035
weighted avg	0.61	0.59	0.57	5035

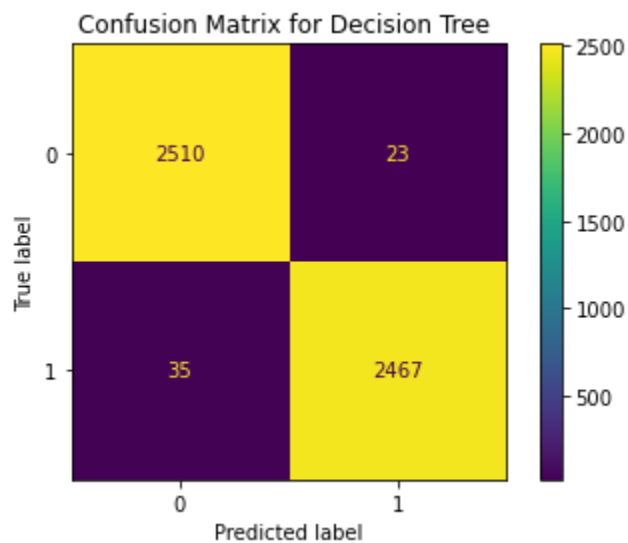


-Decision Tree results:

Accuracy : -> 0.988480635551142

Classification Report for Decision Tree :

	precision	recall	f1-score	support
0	0.99	0.99	0.99	2533
1	0.99	0.99	0.99	2502
accuracy			0.99	5035
macro avg	0.99	0.99	0.99	5035
weighted avg	0.99	0.99	0.99	5035

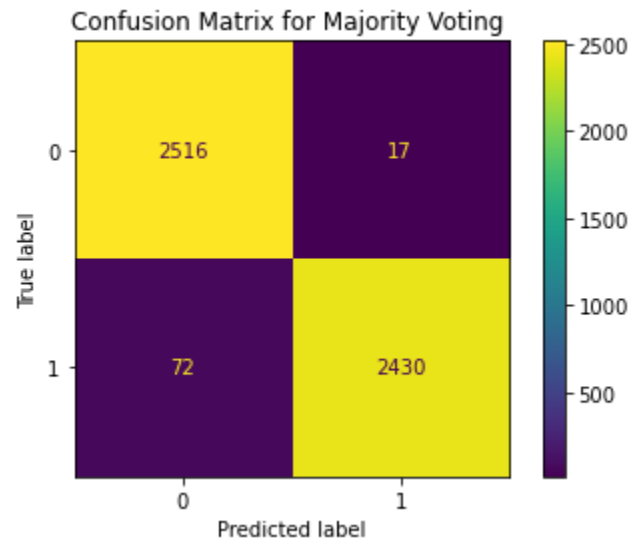


-Majority Voting 1 (Naive bayes, Adaboost, Random forest) results:

Accuracy : -> 0.9823237338629592

Classification Report for Majority Voting :

	precision	recall	f1-score	support
0	0.97	0.99	0.98	2533
1	0.99	0.97	0.98	2502
accuracy			0.98	5035
macro avg	0.98	0.98	0.98	5035
weighted avg	0.98	0.98	0.98	5035

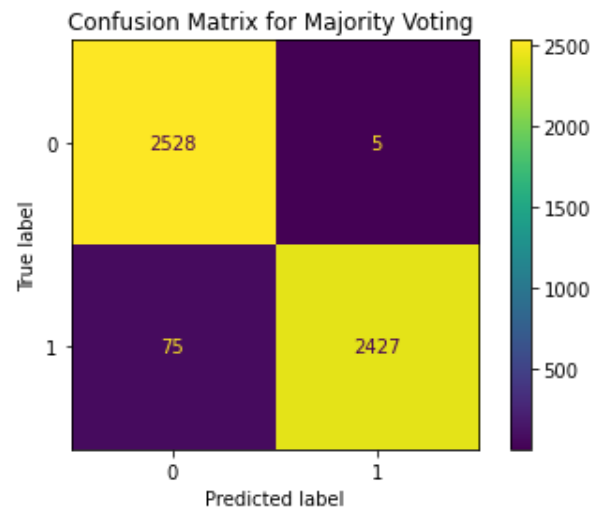


-Majority Voting 2 results (SVM, NB, DT, Adaboost, RF):

Accuracy : -> 0.9841112214498511

Classification Report for Majority Voting :

	precision	recall	f1-score	support
0	0.97	1.00	0.98	2533
1	1.00	0.97	0.98	2502
accuracy			0.98	5035
macro avg	0.98	0.98	0.98	5035
weighted avg	0.98	0.98	0.98	5035



Model	NB	RF	AdaBoost	SVM	DT	MV1	MV2
Accuracy	75%	99.4%	98%	59%	98.8%	98.2%	98.4%

From the previous results we can conclude that:

- The most achieving models are Random forest and decision tree.
 - The ensemble models accuracy are not the best performance, but their performance are way better than the average of the all models.
 - SVM and Naive Bayes are the lowest performing models.
-

For the accuracy values plotting you can see the plotting below for all models.

