Ubiquitous Sensing and Smart City
Assignment Four

made by

Aisha Hagar
Khadija Hesham
Mohammed Elnamory

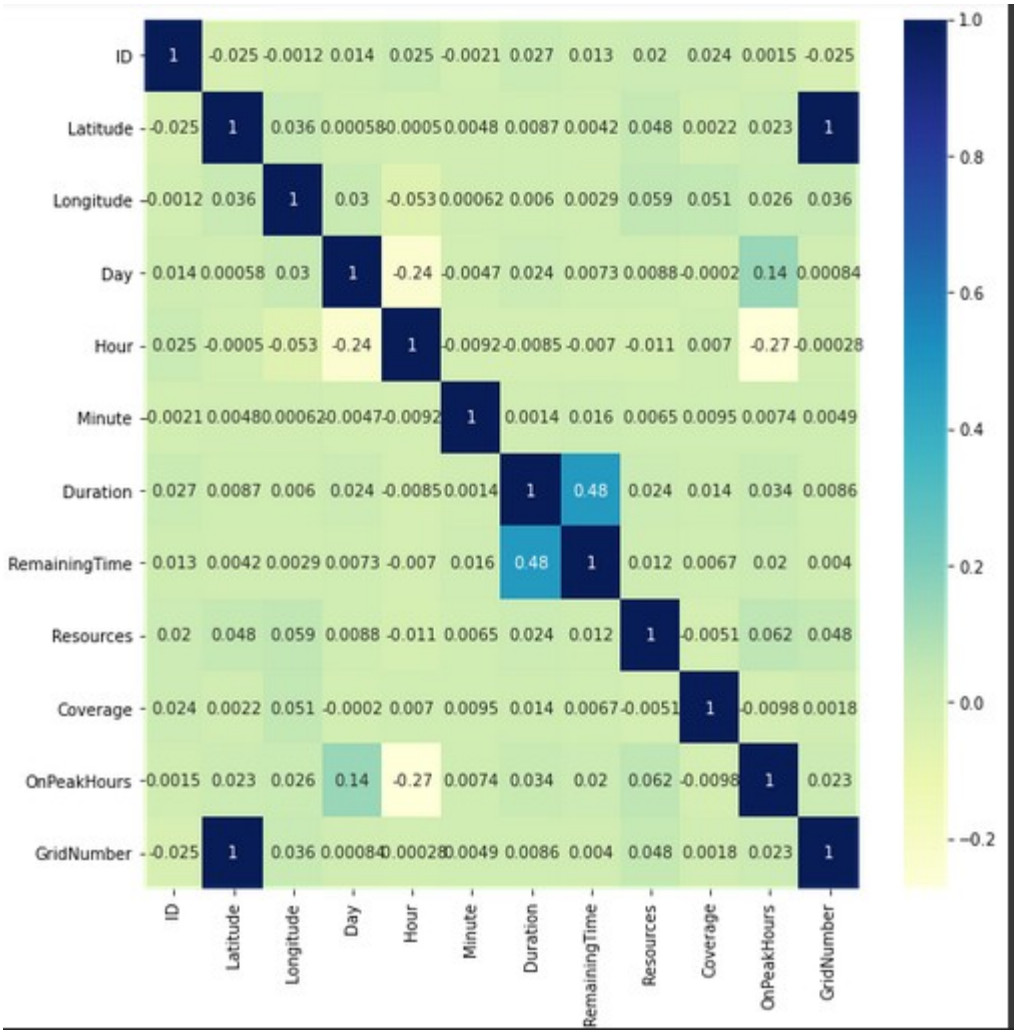
Supervised by

Prof. Burak Kantarci

Table of content

1. Data Visualization:

Here we are visualizing the correlation matrix between variable, and it's shown below all variables are not correlated or not highly correlated.
Except for Latitude and Grid-Number features, so we are dropping one of them.
We also should drop the id feature.



We have investigated our data, and it has no missing values.

```
dataset.isnull().sum()

ID                0
Latitude          0
Longitude         0
Day               0
Hour              0
Minute            0
Duration          0
RemainingTime     0
Resources         0
Coverage          0
OnPeakHours       0
GridNumber        0
dtype: int64
```

We can get some statistical insights from the features remaining.
The remaining time minimum value is 10 while the maximum value is 60 with unexpected increasing rate.
On the other hand, we can see that the number of resources quantiles are increasing in reasonable rate.

	Latitude	Longitude	Day	Hour	Minute	Duration	RemainingTime	Resources	Coverage	OnPeakHours
count	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000	14484.000000
mean	45.484035	-75.217603	2.513946	12.348177	29.480185	44.219829	27.109914	5.838097	65.292184	0.182822
std	0.058989	0.054501	1.704509	6.538839	17.353324	14.511027	14.993890	2.878052	20.311306	0.386534
min	45.365600	-75.334116	0.000000	0.000000	0.000000	10.000000	10.000000	1.000000	30.000000	0.000000
25%	45.434521	-75.264506	1.000000	7.000000	14.000000	30.000000	10.000000	3.000000	48.000000	0.000000
50%	45.484917	-75.220013	2.000000	13.000000	29.000000	50.000000	20.000000	6.000000	65.000000	0.000000
75%	45.541527	-75.173048	4.000000	18.000000	45.000000	60.000000	40.000000	8.000000	82.000000	0.000000
max	45.584678	-75.088915	6.000000	23.000000	59.000000	60.000000	60.000000	10.000000	100.000000	1.000000

3. Modeling:

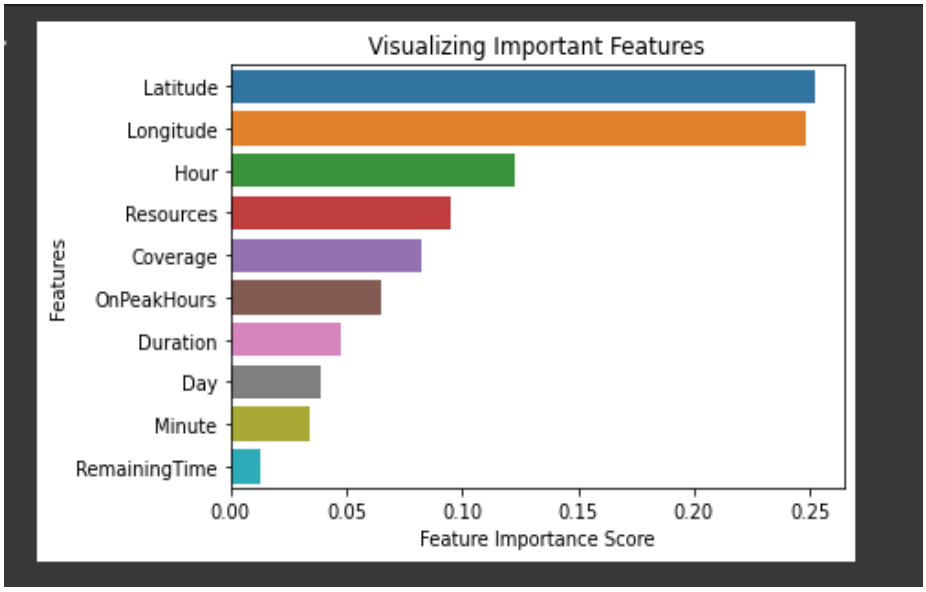
3.1 Random Forest:

We have used the random forest algorithm with 100 estimators and default learning rate.
Random forest has shown a high accuracy of 99% with precision value of 100%.
Hence it has a very high true positive rate.

```
Accuracy: 0.9910251984811874
```

	precision	recall	f1-score	support
0	1.00	0.93	0.97	390
1	0.99	1.00	0.99	2507
accuracy			0.99	2897
macro avg	0.99	0.97	0.98	2897
weighted avg	0.99	0.99	0.99	2897

Features importance by the algorithm:



We have dropped the least necessary features such as Duration and what below in importance.

Retrain after feature selection:

The accuracy witnessed almost no change but the precision value dropped a little.

```
Accuracy: 0.9948222298929927
```

	precision	recall	f1-score	support
0	0.99	0.98	0.98	406
1	1.00	1.00	1.00	2491
accuracy			0.99	2897
macro avg	0.99	0.99	0.99	2897
weighted avg	0.99	0.99	0.99	2897

3.2 Adaboost:

We have used the random forest algorithm with 50 estimators and learning rate of 1.
Adaboost has no high accuracy as random forest algorithm, and it has a very low recall value.
Hence, too few relevant items are selected by the classifier.

```
Accuracy: 0.9395926820849154
```

	precision	recall	f1-score	support
0	0.84	0.70	0.77	406
1	0.95	0.98	0.97	2491
accuracy			0.94	2897
macro avg	0.90	0.84	0.87	2897
weighted avg	0.94	0.94	0.94	2897

3.3 Naive Bayes:

We have used the gaussian naive bayes algorithm
Accuracy is keeping dropping, here it has a value of 83% , and a very little values of precision and recall has been resulted.
Hence too few relevant items are selected and too few selected items are relevant.
This algorithm fails in predicting fake tasks efficiently.

```
Accuracy: 0.8346565412495686
```

	precision	recall	f1-score	support
0	0.40	0.37	0.39	406
1	0.90	0.91	0.90	2491
accuracy			0.83	2897
macro avg	0.65	0.64	0.65	2897
weighted avg	0.83	0.83	0.83	2897

3.4 Voting:

We used hard voting by the combination of RF, NB and Adaboost classifiers.

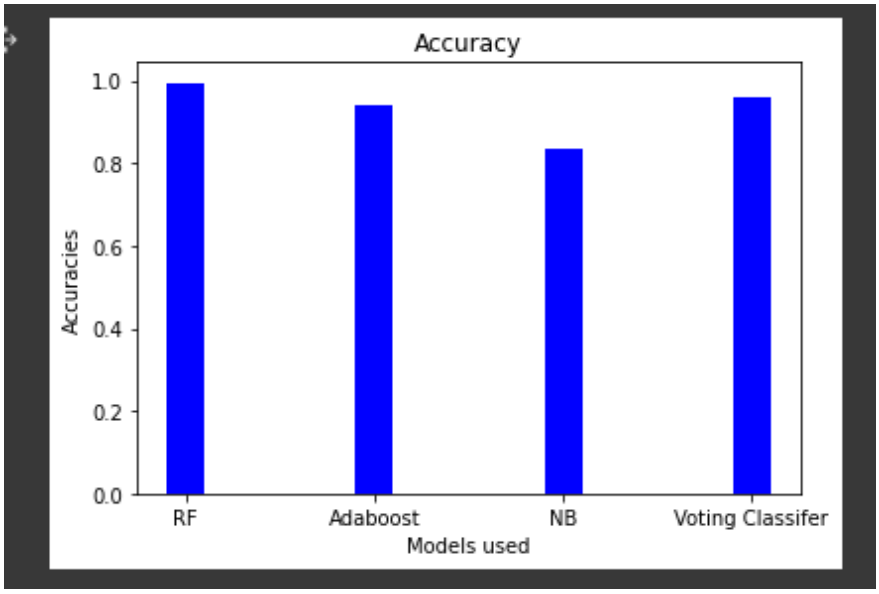
```
Accuracy: 0.9554711770797376
```

Accuracy here is reasonable with a value of 95%.
Recall value is acceptable as it has a 93% score, but precision is very low.
Hence, model is good at differentiating between relevant items but fails in predicting true label.

	precision	recall	f1-score	support
0	0.71	0.93	0.81	290
1	0.99	0.96	0.97	2607
accuracy			0.96	2897
macro avg	0.85	0.95	0.89	2897
weighted avg	0.96	0.96	0.96	2897

3.5 Comparison:

It’s shown here the random forest algorithm and the voting algorithm have the highest accuracy amone the rest of algorithms, and naive bayes algorithm has the lowest accuracy value.



4. References:

[1] sklearn library