Deliverable 4: Data Mining Part B

BILLAL ZAZAI, 8572975
YOUSSEF MAKBOUL, 8609614
GRAYDON HOPE, 300045044

RESULTS:

Random Forest Algorithm

	precision	recall	f1-score
0=resolved	0.89	0.86	0.88
1=unresolved	0.98	0.97	0.98
2=fatal	0.84	0.86	0.85
micro avg	0.91	0.90	0.91
macro avg	0.90	0.90	0.90
weighted avg	0.91	0.90	0.91
samples avg	0.90	0.90	0.90

Time: 0.9400615999999999

Decision Tree Algorithm

	precision	recall	f1-score
0=resolved	0.88	0.84	0.86
1=unresolved	0.98	0.97	0.98
2=fatal	0.83	0.82	0.82
micro avg	0.91	0.88	0.89
macro avg	0.90	0.87	0.89
weighted avg	0.91	0.88	0.89
samples avg	0.88	0.88	0.88

Time: 0.05077657199999974

Gradient Boost Algorithm

	precision	recall	f1-score
0=resolved			
1=unresolved			
2=fatal			
micro avg			
macro avg			
weighted avg			
samples avg			

Time: (Unfortunately we were not able to properly implement gradient boost algorithm on time)

COMPARISON:

From the above tables we can see that the algorithm with the highest scores(deducting gradient boost) was Random Forest. However, with these slightly higher results came longer run time as Random Forest took 89 seconds more than Decision tree. Additionally, we can also analyze that for unresolved cases the precision, recall and f1-score are the same for both Random Forest and Decision tree.

Although simple to understand, Decision Trees tend to have issues as data sets get larger, whereas, random forests tend to be the more accurate learning algorithm. Random forests and gradient boosting each excel in different areas. Random forests perform well for multi-class object detection which tends to have a lot of statistical noise. Gradient Boosting performs well when you have unbalanced data. This is also the reason why we were unable to successfully get gradient boosting to work as our classification had multiple labels being resolved, unresolved and fatal.

SUMARRY:

For this Deliverable our group decided to classify based on case type: resolved, unresolved or fatal. As a result we were dealing with a multi classification problem. To showcase this we utilized Pythons scikit learn which is a free machine learning library that features various classification, regression and clustering algorithms. Since we were dealing with a classification problem we utilized the following sci kit Models: Random Forest, Decision Trees and Gradient Boost. Each model being unique in its own use cases. During the deliverable we gained a lot of insight as to how each model reacted, trained and predicted to our Covid data and classification. The Decision tree and random forest models both have similar parameters that can be passed to help fine tune the model allowing it to cater to the developers needs, however the Random forest had additional parameters that made it more unique. The most significant parameter being the n_jobs parameter which allowed the number of jobs to run in parallel, and if -1 was provided then it would use all processors. This provided a slight advantage as this allowed the Random first to utilize more processing power to complete its task. Gradient Boost was the more different amongst the three as it didn't have any similar parameters. The most widely popular parameters for gradient boost were learning_rate and n_estimators.