Flight Delay Prediction For Aviation Industry Using Machine Learning

Model deployment :

Saving Best Model :

For the Neural Network models there are three conditions that I needed to make sure that the dataset fulfils, and these are:

1. Data has to be purely numerical
2. Data cannot contain missing values
3. Data has to be Normalised

Doing the MLP Deep Neural Network was more difficult and time consuming due to the high number of tests needed and the size of the dataset, which was reduced from +7 million to around +4 million rows by limiting the study to the top 20 destination cities. At the end, close to 50 different model architectures were tested and are documented in their respective notebooks with the best one being the one from Figure 6 (Model_5), which contains a model summary, the compiler characteristics, and the fitting which was done for only 25 epochs. Unfortunately so far, my computer hasn't been able to run the 50 epochs which is one of my goals to try to reach convergence as there are slight indications that it will happen with a higher number of epochs as the accuracy and loss plot from Figure 17 suggest.

```
model_5 = Sequential()

model_5.add(Dense(50, activation='tanh', input_shape=(63,)))

model_5.add(Dense(30, activation='tanh'))

model_5.add(Dense(15, activation='tanh'))

model_5.add(Dense(5, activation='relu'))

model_5.add(Dense(1, activation='sigmoid'))

model_5.summary()
```
executed in 53ms, finished 01:23:42 2020-10-15

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_47 (Dense)             (None, 50)                3200
_________________________________________________________________
dense_48 (Dense)             (None, 30)                1530
_________________________________________________________________
dense_49 (Dense)             (None, 15)                465
_________________________________________________________________
dense_50 (Dense)             (None, 5)                 80
_________________________________________________________________
dense_51 (Dense)             (None, 1)                 6
=================================================================
Total params: 5,281
Trainable params: 5,281
Non-trainable params: 0
_________________________________________________________________
```

```
model_5.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```
executed in 34ms, finished 01:23:53 2020-10-15

```
results5 = model_5.fit(X_train, y_train, epochs=25, batch_size=32, validation_split=0.1)
```
executed in 7h 19m 25s, finished 08:43:39 2020-10-15

Figure 6. MLP Deep Neural Network Model_5 architecture

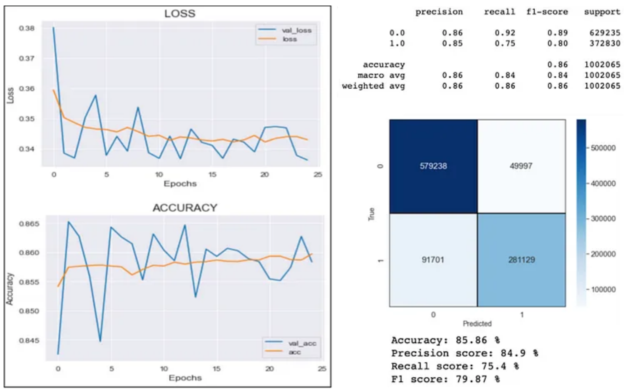


Figure 7. Model 5 performance summary with the loss and accuracy plot (left), classification report (upper right), and confusion matrix followed by other metrics calculated (lower right).

The Deep Neural Networks managed to get an improvement in accuracy of more than 15%, while the Recall and Precision have also dramatically improved. In general, all the models tested have simple architectures with small variations between them, and by looking at the Accuracy and Loss plots from Figure 7, we can infer that what I chose to be my best model is still under-fitting, so metrics could definitely improve. For a summary of all the models performance see Figure 8. There, you will notice that the models from the MLP_NN_Part_II have increased all their metrics compared to the rest. The reason for this is because a feature selection was done and proved that the months of the year were causing noise, therefore they were all dropped.

		Inbalanced data			
Notebook	Models	Precision	Recall	Accuracy	F1
MLP_NN_Part_I	Model 1	79	54	66	47
	Model 2	79	54	66	47
	Model 3	68	62	69	62
	Model 4	70	62	69	61
	Model 5	79	54	66	47
	Model 6	68	63	69	63
	Model 7	69	63	70	63
MLP_NN_Part_II	Model 1	89	83	87	85
	Model 2	88	84	87	85
	Model 3	87	80	84	82
	Model 4	87	80	84	82
	Model 5	86	84	86	84
	Model 6	87	80	84	82
	Model 7				
MLP_NN_Part_III	Model 1				
	Model 5	69	63	70	63
	Model 6	69	62	70	62
	Model 7	71	61	61	70

Figure 8. Deep Neural Networks summary table

This proofs that with the data available, it is possible to generate a model that will help predict if a flight will be delayed or not with an accuracy of 86%, an 84% of

Recall and 86% Precision, and that's not bad at all, specially if it is before it even comes up on the departure board. How is this possible though? well, this is possible because I basically analysed airlines performance during a year, I looked at the relationship between the delays (regardless of the reason) with the origin and destination cities, I looked at the elapsed time and relationship with every other feature, I also separated the cities and looked at daily, weekly and monthly patterns to better understand when any of those flights might be delayed, and many other relationships that you can find between the 28 categories. Now with more time on my hands more detail could have been achieved by looking at times of the day for example, and this would translate as better metrics. So that will probably be my way forward alongside looking at all the 10 year dataset, which you might see in another blog in the near future.

As discussed, weather condition plays an important role in proper and timely functioning of flights. We propose a flight delay prediction system which focuses mainly on predicting delay of a flight based on the weather situation. To make the system more scalable it is necessary to choose analgorithm which considers all the parameters to be independent.Supervised learning as the name indicates a presence of supervisor as teacher. Essentially supervised learning could be a learning that within which we tend to teach or train the machine exploitation data which is well tagged which means some data is already labeled with correct answer. After that, machine is given new set of examples(data) so supervised learning algorithmic rule analyses the coaching knowledge(set of training examples) and produces an correct outcome from tagged dataUsing supervised machine learning approach, the labeled data gives it authenticity. Naïve bayed model is one of the algorithm which is proven to be efficient for real time prediction as well as the fact that it considers every attribute to be independent from each other makes it an apt algorithm for the concerned projectthe algorithm. The attributes considered for calculations and taken by the API are as follows weather, temperature, humidity, Rain in mm, Visibility and Month number. As discussed that supervised machine learning is based on having a set of correct labeled data form which the algorithm bases its prediction. We use a CSV file for storing that data as a flat file format is easier to edit , update and retrieve it for calculations.

Model Evaluation

Due to having an "Imbalance data", accuracy was not enough to measure the model's performance.

- Confusion Matrices
- Precision
- Recall
- F1

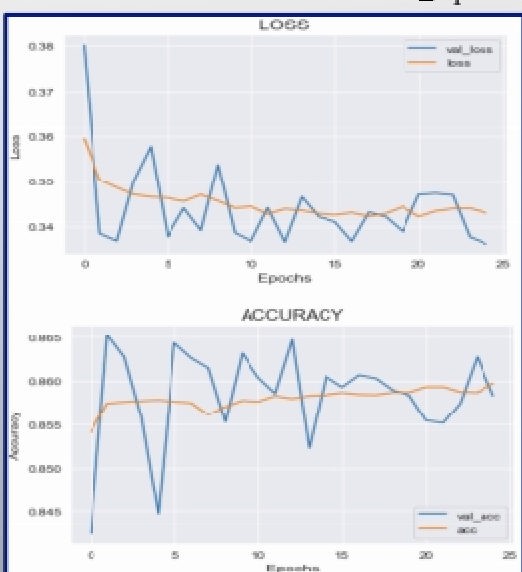
MLP Neural Network

Notebook	Models	Inbalanced data Precision	Recall	Accuracy	F1
MLP_NN_Part_I	Model 1	79	54	66	47
	Model 2	79	54	66	47
	Model 3	68	62	69	62
	Model 4	70	62	69	61
	Model 5	79	54	66	47
	Model 6	68	63	69	63
	Model 7	69	63	70	63
MLP_NN_Part_II	Model 1	89	83	87	85
	Model 2	88	84	87	85
	Model 3	87	80	84	82
	Model 4	87	80	84	82
	Model 5	86	84	86	84
	Model 6	87	80	84	82
	Model 7				
MLP_NN_Part_III	Model 1				
	Model 5	69	63	70	63
	Model 6	69	62	70	62
	Model 7	71	61	61	70

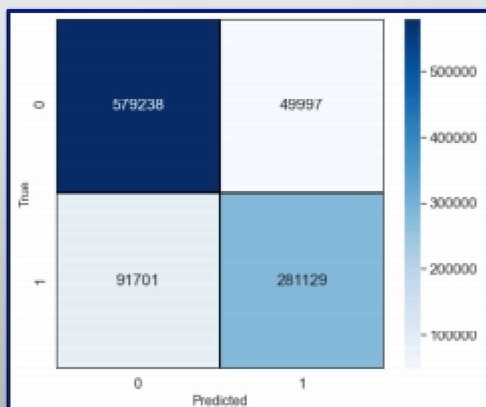
	Algorithm	Inbalanced data Precision	Recall	Accuracy	Balanced data Precision	Recall	Accuracy
1	Baseline Tree (without DEF_DELAY)	64.00	51.00	63.25	57.00	57.00	57.76
2	Bagged Tress (without DEP_DEPAY(	66.00	51.00	63.33	61.00	52.00	63.68
3	Random Forest (without DEP_DELAY)	74.00	50.00	62.89	57.00	57.00	56.76
4	Random with Bootstrat Weighting	57.00	57.00	56.57	57.00	57.00	57.25
5	AdaBoost_V1 (with DEP_DELAY)	84.00	82.00	81.72			
6	AdaBoost_V2 (without DEF_DELAY)	65.00	54.00	64.54			
7	Gradient Boosted Trees (with DEP_DELAY)	85.00	79.00	83.09			
8	Gradient Boosted Trees (without DEP_DELAY)	70.00	57.00	66.84			
9	XGBoost (with DEP_DELAY)	88.00	82.00	86.30	87.00	83.00	85.65
10	XGBoost (without DEP_DELAY)	71.00	61.00	69.37	69.00	63.00	69.68

Model Evaluation – Best Model

MLP Neural Networks - Model_5 performance evaluation:



Loss and Accuracy plot (25 epochs)




```
              precision    recall  f1-score   support

         0.0       0.86      0.92      0.89    629235
         1.0       0.85      0.75      0.80    372830

    accuracy                           0.86   1002065
   macro avg       0.86      0.84      0.84   1002065
weighted avg       0.86      0.86      0.86   1002065
```

Confusion Matrix & Classification Report

```
Accuracy: 85.86 %
Precision score: 84.9 %
Recall score: 75.4 %
F1 score: 79.87 %
```

Final Performance Metrics

Machine Learning: Predicting Flight Delays