UNIVERSITY OF EDINBURGH
Business School

2020-21

[CMSE11428 AND PREDICTIVE ANALYTICS & MODELLING OF DATA]

[INDIVIDUAL ASSESSMENT]

[B196824]

SUMMARY:

With more popularity for air travels and its significant role in the economy, there is an increased need to improve its service quality. The frequently faced challenges like fight delay and cancellations are not only impacting the resource management for airports/airlines, but also the customer satisfaction. It pushes consumers to modify their travel arrangements, which increases their distrust and, as a result, adversely affects the airline's business brand, reducing passenger loyalty. Airlines incur additional expenses such as aircraft manoeuvring, crew management and increased fuel use to compensate the additional delays/cancelations.

It would be a great relaxation for airlines as well as passengers if they have prior estimates of flights being delayed or cancelled. Passengers may plan their route in advance and airlines can adjust their resource supply appropriately, resulting in reduced loss and better management. The report attempts to evaluate and make effective predictions for delays and cancellations by applying various classification supervised predictive algorithms. Extensive data exploration is applied to the flight data set from 2008 and data from three airlines are selected for the analysis. Imbalanced dataset is balanced with various balancing strategies which is followed by modelling of the data. The models are then evaluated with various parameters like Area Under Curve, Sensitivity – False Positive Rate, Specificity – True Positive Rate, etc and the best performer is selected for the further analysis. Important predictors are further examined to make sound decisions which minimise delays and cancellations.

INTRODUCTION:

Airplanes are frequently used mode of transportation since they reduce the amount of time to travel. There are chances of flight delays. Flight delays are unavoidable, resulting in significant losses for airlines and have a negative impact on the economy. Due to aircraft delays, the United States government lost 31–40 billion dollars in 2007 [3]. Flight delays are influenced by a lot of factors, like demanding flight schedule, adverse weather [1], and overcapacity usage of the different components in the aviation system like airports, staff crew, etc. Flight delays and cancellations are closely related. Flight cancellations effectively lower overall demand, which can lead to shorter delays for subsequent aircraft in a queuing system. During GDPs, Xiong [2] researched this process and it was determined that airlines negotiate exchange between aircraft cancellations and flight delays determining.

Machine learning has evolved and become increasingly used for aircraft delay/cancellation prediction problems as artificial intelligence has progressed. Machine learning encompasses a wide range of fields, including probability, statistics, and computer science [4]. It helps to overcome the limits of mathematical methods and increase flight delay

forecast accuracy. Each machine learning approach (Supervised, Unsupervised, and Deep) has its own set of characteristics. The selection of appropriate model is required to result optimal results. Poorly performing algorithms waste computational power, hence algorithm selection is a critical procedure in machine learning technology. The primary goal of study is to give a useful flight delay and cancellation classification prediction approach, perform comparison analysis and evaluate metrics to overcome algorithm selection difficulties.

PROBLEM STATEMENT:

The new airline firm intends to launch its airlines shortly, and with few resources on board, the airline wishes to sustain longer with less losses and establish a strong foundation for customer satisfaction to expand its client base. Airlines is hence approaching Edinburgh PAMD Consulting Company for predicting effective delays and cancellations to increase their performance. They will not only incur less losses, but will also be in a better position to take necessary actions like changing routes, increasing the number of airlines on crowded routes, etc. The approach is to derive meaningful suggestions based on the use of several predictive models to estimate the optimal time in terms of week/month which can help predict and minimize delays/cancellations and illustrate the cascade consequences of delay correlations, if any. The focus is primarily on arrival delays since the destination site is more important to predict ahead of time so that passengers can plan their journeys and airlines can make the most use of their resources. The goal is to enhance airline planning by using classification prediction models to forecast arrival delays and cancellations in terms of binary values yes or no.

TECHNIQUES AND LITERATURE REVIEW FOR MODELS:

Data modelling has its own set of limitations; hence it's vital to have a clean, normalised, and converted data set before performing any further analytics. The likelihood of inaccuracy increases if data is not adequately cleansed and normalised. To get precise projections, the report follows the method outlined (*Figure 1*) in order:

Data Selection → Data Pre-Processing (Cleaning, Feature Section, Transformation, Normalization) → Data Split → Data Modelling → Result Analysis

Figure 1: Data Modelling process **

Note Two separate process were followed for modelling predictions for arrival delays and cancellations respectively.**

***Data Selection and Exploration Data Analysis*:** Flight data was obtained from the following source: [dataexpo/2009](). The data collected includes 29 categories/features for the year 2008 (with 1,936,758 records). The Appendix contains the whole glossary. Because huge data requires more computing time and owing to resource constraints, the data is further condensed to contain data from three airlines with medium, high, and low arrival delays (*9E, AA, AQ*) to handle data variability (*Figure 2*). The same data is used for cancelation predictions.

	UniqueCarrier	num_delays
num_delays		
0	9E	2420468.0
1	AA	8889066.0
2	AQ	15814.0

	UniqueCarrier	cancelled
cancelled		
0	9E	58
1	AA	46
2	AQ	0

Figure 2: Short data for three airlines 9E, AA and AQ

EDA was performed on the condensed dataset of Airlines for delays and cancelations and following insights were obtained by summarizing the Figure 3:

- Delays were reported in all the months, although the months of January, February, March, and June had the highest proportion of delays in comparison of others. Further for the week, high delays were observed on Fridays.
- Cancelations on the other hand was unevenly distributed with only three months showing cancelations: October, November, and December. Mondays experienced the higher cancelations in terms of weeks.

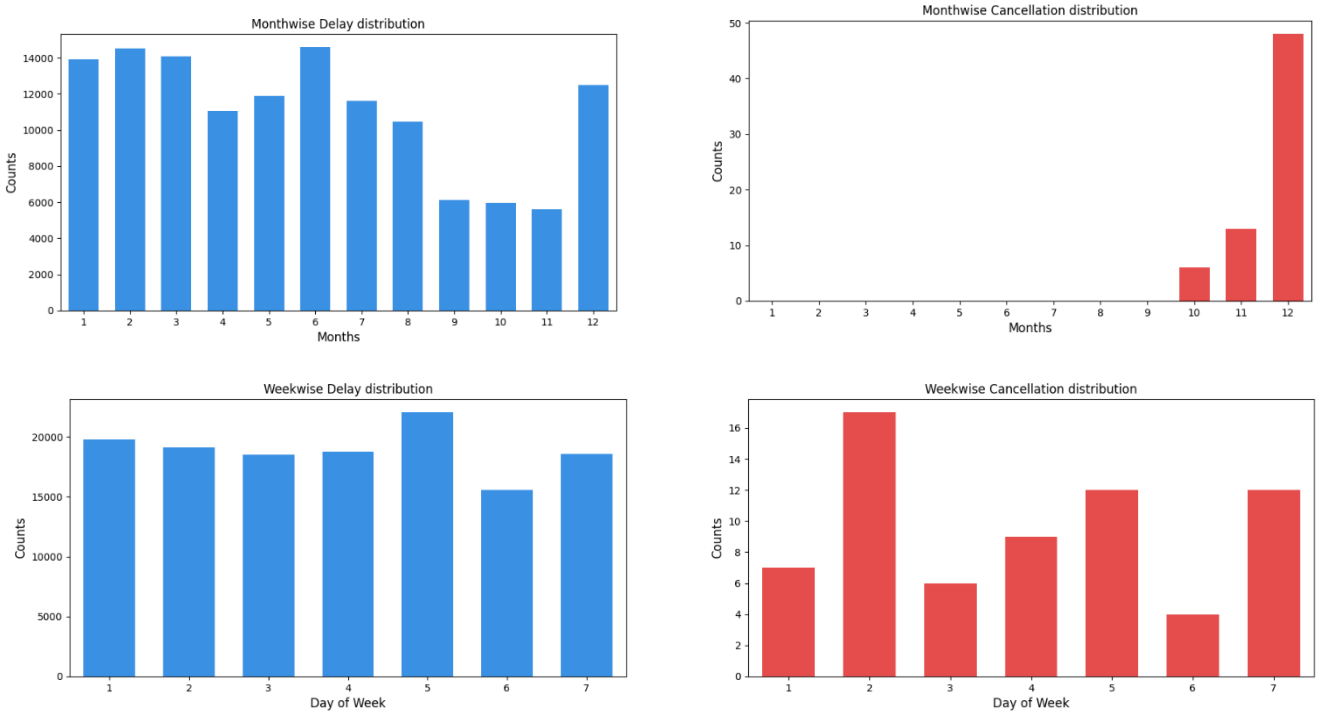

Figure 3: Arrival Delay and Cancelation counts per month and week

Data Pre-Processing-1: *Data Cleaning and Feature Selection*: During data processing, all categories were examined as potential predictors, and a co-relation matrix was generated to depict the group's co-relationship. Based on the

findings, the highly co-related factors were eliminated, because the stronger the correlation, the more biased the model gets with redundant information, making it flawed and can be avoided.

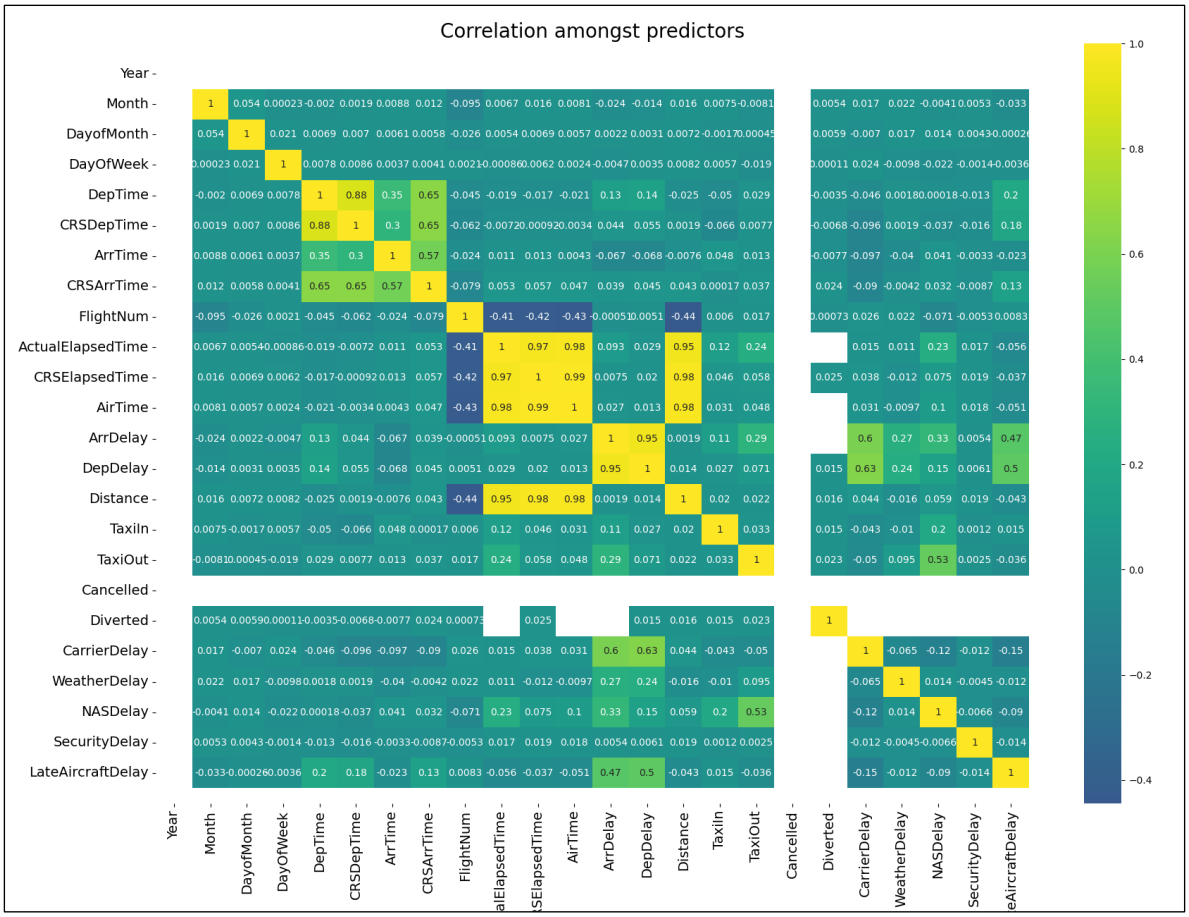

Figure 4: Correlation Matrix for potential predictors

As a part of this process, one of the correlated features (*Distance*) is selected and the other features with high correlation score more than 0.8 (marked yellow - *Fig 4*) are dropped. (*CRSElapsedTime, ActualElapsedTime, CRSDepartureTime, CRSArrivalTime, AirTime*). Since we have considered only 3 airlines data we can simple ignore *FlightNumber*.

The two additional predictors dropped are *TaxiIN* and *TaxiOut*, which indicate the time spent on the runway before and after the flight. These are of little importance to customers/airlines because they come into play in the later part of journey and the purpose is to anticipate the traits prior to departure. *Diverted* column is dropped as it can be considered as a separate classification altogether and would add more noise.

Following the completion of the above stage, the emphasis is on checking for missing or null entries. There are several numbers of missing entries associated with various delay measures like weather, security, NAS, late aircraft, and carrier. Initial thought was to impute values using mean, or with 0 (observing values ArriDelay<=15 were NA)but due to higher counts, imputing mean or 0 would be erroneous and might affect the model's accuracy, hence the ultimate choice was to remove these entries.

***Outlier Detection and Removal**:* The outlier is identified using a boxplot/ inter-quantile range for the continuous variable on the left, Distance. Outliers for distances greater than 2000 are removed to get normalised data.

Type	Counts	Percent
WeatherDelay	76168	31.7
SecurityDelay	76168	31.7
NASDelay	76168	31.7
LateAircraftDelay	76168	31.7
CarrierDelay	76168	31.7

Figure 5: Counts of Missing values for various delays

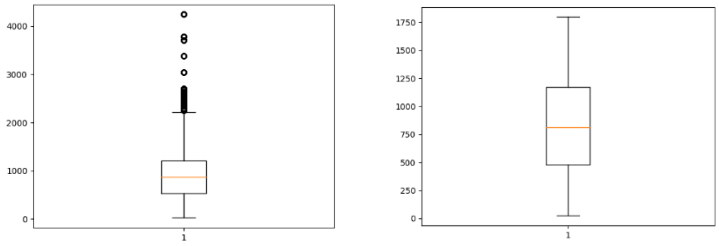

Figure 6: Pre and Post outliers' removal (Distance)

At this stage two different data sets are created for delay and cancelation respectively. The delay data set is further refined considering only non-cancelled (Cancelled=0) flights, as cancellations are irrelevant for data. The transformation of *ArrDelay* column into binary column *FlightDelayStatus* is done (delay threshold=0 if delay<=15 else 1). The other irrelevant columns were dropped. Notice the *DepDelay* attribute is retained in the delay data set, although it is highly correlated attribute, the approach is to observe the cascading effects caused from Departure delays on arrival delays. Its reasonable that if there is a Departure Delay there are high chances for Arrival delay. Figure 7 depicts the final counts for delay and cancelation data set.

```
Name: FlightDelayStatus, dtype: int64
0    201559
1    67
Name: Cancelled, dtype: int64
1    132313
0    69246
Name: FlightDelayStatus, dtype: int64
```

Figure 7: Final counts for cancelation and delay data sets

```
(244396, 8)
Month                0
DayOfWeek            0
UniqueCarrier        0
DepDelay             0
Origin               0
Dest                 0
Distance             0
FlightDelayStatus    0
```

```
(244500, 8)
Month                0
DayOfWeek            0
UniqueCarrier        0
Origin               0
Dest                 0
Distance             0
Cancelled            0
CancellationCode     0
```

Figure 8: Missing counts from delay and cancellation data sets

***Data Pre-Processing-2:** Data Transformation*: All the categorical variables for both the data sets (*Month, DayofWeek, Origin, Destination, UniqueCarrier, CancellationCode*) are now transformed into dummies. It creates a separate column for each unique category, transforming all categories and numerical data into binaries (*Figure 9*).

```
        Distance  Cancelled  CancellationCode_B  CancellationCode_C  CancellationCode_N  Month_2  Month_3  Month_4  Month_5  Month_6  Month_7  Month_8  Month_9  Month_10  Month_
138532       813          0                   0                   0                   1        0        0        0        0        0        0        0         0         0
138533       696          0                   0                   0                   1        0        0        0        0        0        0        0         0         0
138534       696          0                   0                   0                   1        0        0        0        0        0        0        0         0         0
138535       696          0                   0                   0                   1        0        0        0        0        0        0        0         0         0
138536       696          0                   0                   0                   1        0        0        0        0        0        0        0         0         0
(227371, 365)
        DepDelay  Distance  FlightDelayStatus  Month_2  Month_3  Month_4  Month_5  Month_6  Month_7  Month_8  Month_9  Month_10  Month_11  Month_12  DayOfWeek_2  DayOfWeek_3  Da
138532      30.0       813                  1        0        0        0        0        0        0        0        0         0         0         0            0            1
138534      43.0       696                  1        0        0        0        0        0        0        0        0         0         0         0            0            0
138535      13.0       696                  0        0        0        0        0        0        0        0        0         0         0         0            0            0
138536      25.0       696                  0        0        0        0        0        0        0        0        0         0         0         0            0            0
138537      36.0       696                  1        0        0        0        0        0        0        0        0         0         0         0            0            0
(201559, 363)
Process Completed
```

Figure 9: Transformed Sets

Data Pre-Processing-3*: Data-Split and Normalization:* The next step comprises of the data splits into training and test data sets. The respective target variables for both the datasets (delays and cancellations) are set as predictor variable and the actual entry is dropped out. To create a realistic model with no more train or test data overfitting, the data split for this report is set at a 70:30 ratio. Stratification is used while splitting, to ensure that the data is evenly distributed for both delay and non-delay cases. Random state is applied to assist model in learning and avoiding repetition. Normalization is carried out post the data splitting process to avoid biased train or test sets with a larger number of non-normalized values in either one. Scaling of data is done using MinMaxScaller on training data ranging from -1 to 1, so that there is some similarity in range of data values with no extremes. Later the training set is used to scale test set for obtaining a balanced split of scaled version. In addition to the preceding steps, one more process is added for cancellation prediction, which is performing SMOTE (unsupervised) technique so that the data is balanced equally, and the large difference between cancellations and non-cancellations is now handled. Fig 11 depicts that the SMOTE helps to make balance in cancellations. This marks the end of data pre-processing stage.

```
y = flights['Cancelled']
X = flights.drop('Cancelled', axis=1)
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y,
                                                    test_size=0.3, random_state=42)

# Scale the training and the test data
x_train_s = X_train.copy()
x_test_scaled = X_test.copy()
scaler = MinMaxScaler(feature_range=(-1, 1))
train_features = x_train_s[col_names]
train_features_f = scaler.fit_transform(train_features.values)
x_train_s[col_names] = train_features_f
test_features = x_test_scaled[col_names]
test_features_f = scaler.fit(train_features.values).transform(test_features.values)
x_test_scaled[col_names] = test_features_f

print("Cancelled Train Data")
print(y_train.value_counts())
sm = SMOTE()
x_train_scaled, y_train = sm.fit_resample(x_train_s, y_train)
```

Figure 10: Code Snippet for cancelation prediction for data split

```
Cancelled Train Data
0    141091
1        47
Name: Cancelled, dtype: int64
Cancelled Train Data post Smote
0    141091
1    141091
Name: Cancelled, dtype: int64
```

Figure 11: Data pre and post SMOTE

Modelling and Tunning:

The below set of classification models are studied for analysis each for delay and cancellation prediction, The reason for selective wide range of models is to forecast accurate estimates, as each model has its own limitations, one model might not perform as better as the other one.

➢ *Logistic Regression* ➢ *Decision Tree Classification Model* ➢ *Random Forest Classification Model* ➢ *ADA Booster Classification Model*	➢ *Bagging Classification Model* ➢ *XGBoost Classification Model* ➢ *Neural Network with Activation (Linear, Sigmoid, Relu)*

Logistic Model was tuned with help of Lasso technique to enhance the performance [8]. The various classification tree models are used, one of the reasons is that feature scaling, such as standardisation or normalisation, is not required in

pre-processing. The other reason is because tree-based Machine Learning methods like bagging and boosting are accessible in packages, and we can combine more models to increase accuracy [9]. The use of Neural network or Deep learning methods will help to run more iterations and tune the model for better performances. The Models have a capability to run on high degree of complexity and an enormous volume of data. Furthermore, Deep Learning may automatically extract the key characteristics from data [5]. Neural Network were permuted using different activation model for input layers, output layer was put constant as sigmoid.

Model Tuning: Hyper tuning for each model was performed using cross-validation technique with 5 folds in a grid fashion. The tuned parameters were selected based on the highest accuracy across the wide range of combinations. Refer Appendix for more information.

```
{"estimator": LogisticRegression(penalty='l1', solver='liblinear'),
 "parameters": {'grid[C]' : arange(0.0001, 1, 0.01)
   , "name": "Logistic Regression with Lasso"}},
{"estimator": DecisionTreeClassifier(random_state=44),
 "parameters": {'min_samples_leaf': [1, 5], 'max_depth': [None, 10]}
   , "name": "DecisionTreeClassifier with Stratification"},
{"estimator": RandomForestClassifier(random_state=44),
 "parameters": {'min_samples_leaf': [1, 5], 'max_depth': [None, 10], 'n_estimators': range(10, 100, 10)}
   , "name": "Random Forest Classification with Stratification"},
{"estimator": AdaBoostClassifier(random_state=44), "parameters": {'n_estimators': range(10, 100, 10)}
   , "name": "AdaBoostClassifier with Stratification"},
{"estimator": BaggingClassifier(base_estimator=DecisionTreeClassifier(), random_state=44),
 "parameters": {'n_estimators': range(40, 70, 10)}
   , "name": "BaggingClassifier with Stratification"},
{"estimator": KerasClassifier(nn_model),
 "parameters": {'no_neurons': [50, 100], 'kernel': ['relu', 'sigmoid'], 'no_layers': [1, 2],
              'learning_rate': [0.1, 0.01, 0.001], 'epochs': [10], 'verbose': [0]}
   , "name": "KerasClassifier with Stratification"}
```

Figure 12: Parameters Passed for Cross Validation

```
logi_model = LogisticRegression(penalty='none').fit(x_train_scaled, y_train)
lasso_model = LogisticRegression(penalty='l1', solver='liblinear', C=0.9901).fit(x_train_scaled, y_train)
decision_model = DecisionTreeClassifier(random_state=44).fit(x_train_scaled, y_train)
decision_model_tuned = DecisionTreeClassifier(min_samples_leaf=5, max_depth=10, random_state=44).fit(x_train_scaled,
                                                                                                     y_train)
random_model = RandomForestClassifier(random_state=44).fit(x_train_scaled, y_train)
random_forest_model_tuned = RandomForestClassifier(min_samples_leaf=5, max_depth=10, n_estimators=90,
                                                   random_state=44).fit(
    x_train_scaled, y_train)
ada_booster_model = AdaBoostClassifier(random_state=44).fit(x_train_scaled, y_train)
ada_booster_tuned = AdaBoostClassifier(n_estimators=90, random_state=44).fit(x_train_scaled, y_train)
bagging_model = BaggingClassifier(random_state=44).fit(x_train_scaled, y_train)
bagging_model_tuned = BaggingClassifier(base_estimator=DecisionTreeClassifier(), n_estimators=70,
                                        random_state=44).fit(x_train_scaled, y_train)
xgb_model = XGBClassifier(random_state=44).fit(x_train_scaled, y_train)
```

Figure 13: Tuned parameters received from the cross validation

RESULT ANALYSIS:

The models are predicted across target test values and confusion matrix along with ROC AUC curves was calculated for each model. The additional factors considered for analysis are Accuracy, Recall/Sensitivity, F1_Score, Precision. The significance of each measure changes depending on the model and the use case. The metrics for AUC, accuracy, recall (False Positive rate), and specificity are evaluated for delay prediction (True Negative rate). The rationale for considering recall and specificity is that it helps to know the model's performance in predicting the genuine event of delay in the case of actual delay and forecasting no delay when there was no delay and the same is followed for cancellations. There was inclusion of departure delay in the model to check its performance and cascading effect it can have over Arrival delays, its known to have the high correlation effects but can the question is, whether prediction models can notice it happening. Hence, we run three analytical flow which includes predicting arrival delays with departure delays, predicting arrival delays without departure delays, and predicting cancelation flights. Let's take a closer look at each of these analyses:

Arrival Delay Prediction with Departure Delay: Observing the effects, the accuracy for all the models were quite promising well over 80 %. The high accuracy was expected since any delay in departure would result in an arrival delay. This demonstrates that there is a cascading effect that the model can detect it. Precision, Recall, and F1 Score were all quite high. Surprisingly, the Specificity of Decision Tree was low around 60 percent compared to the average of other models 70 percent. The Decision Tree Classifier Hyper Tuned Model can be selected as the best performing model as it has high metrics for Accuracy, AUC, Specificity and Recall, makes the model robust.

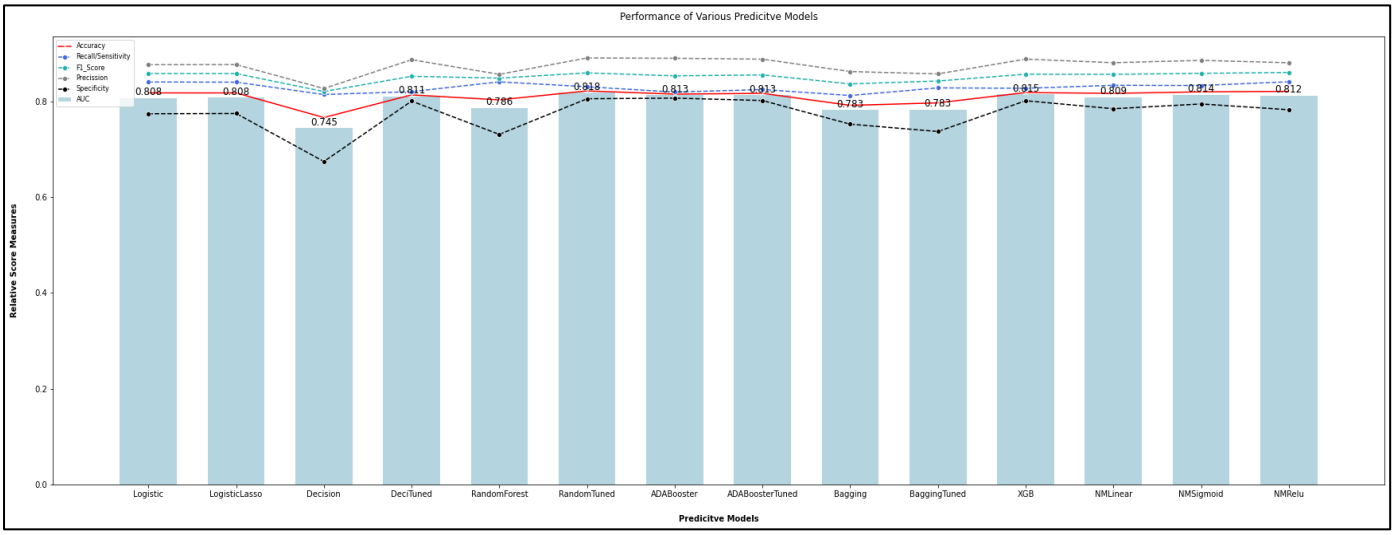

Figure 14: Performance summary for Arrival Delays - models includes Departure Delays

Arrival Delay Prediction without Departure: Based on the above results let's try to evaluate the models without departure delay and see their performance. The accuracy drops drastically to an average 60 percent indicating that the high accuracy was due to departure delay, making it a biased model. Based on the below models (Fig 15) Decision Tree outperforms the other models in terms of specificity but falls short in terms of recall and accuracy. Random Forest, Bagging and Bagging Tuned models are the closed competitors and better performers based on specificity measure. Bagging model with tunning is selected as the best performing model because of its comparable high area of AUC 54%, specificity 40%, Accuracy 61 %, Recall 80%.

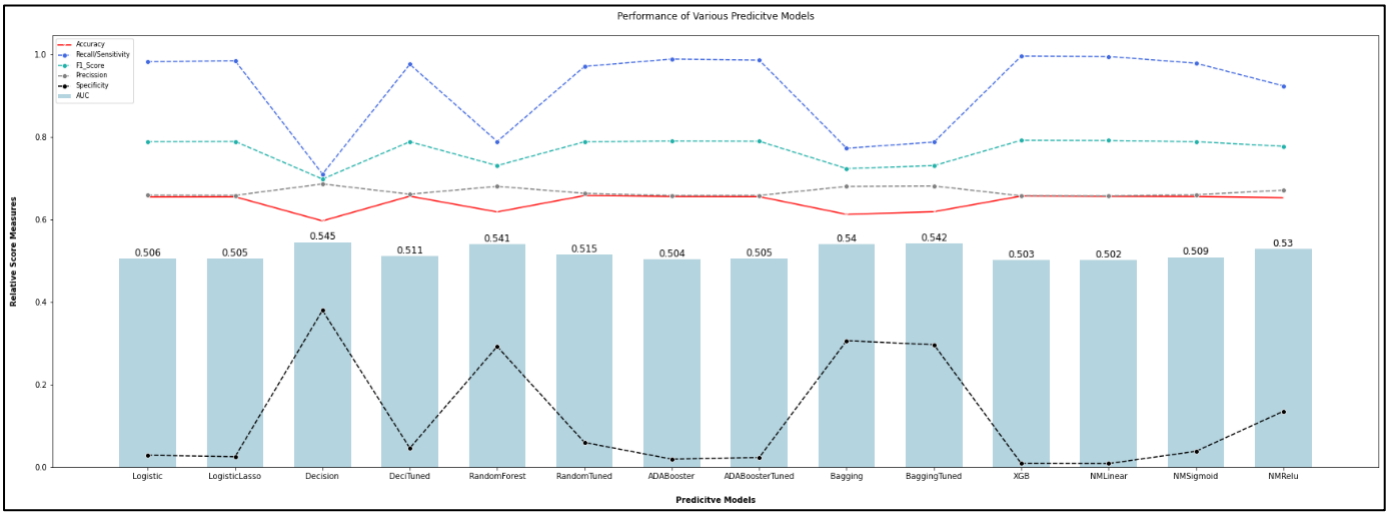

Figure 15: Performance summary for Arrival Delays - models does not include Departure Delays

Cancellation Prediction: All the model had very high accuracy, and its obvious to see that because of the uneven cancellation values and more 0s in train set, the models might end up predicting 0, considering them to be true. Hence the other metrics like specificity and recall should be given more preference. Noticeable precision is quite low in almost all the models, but negligible values are observed for the tree classifiers. Recall levels vary amongst models, with the highest found in Random Forest. Metrics are chosen based on the company and the subject being addressed. For this scenario the importance is given to both specificity and recall values. Despite the Neural Network model with Relu Activation performs the best, the next best model is considered, Logistic Regression, owing to the unavailability of a feature selection score and other important derivations to form a recommendation. The Logistic regression has high specificity and accuracy 99%, AUC 78%, precision 20% and recall 50%.

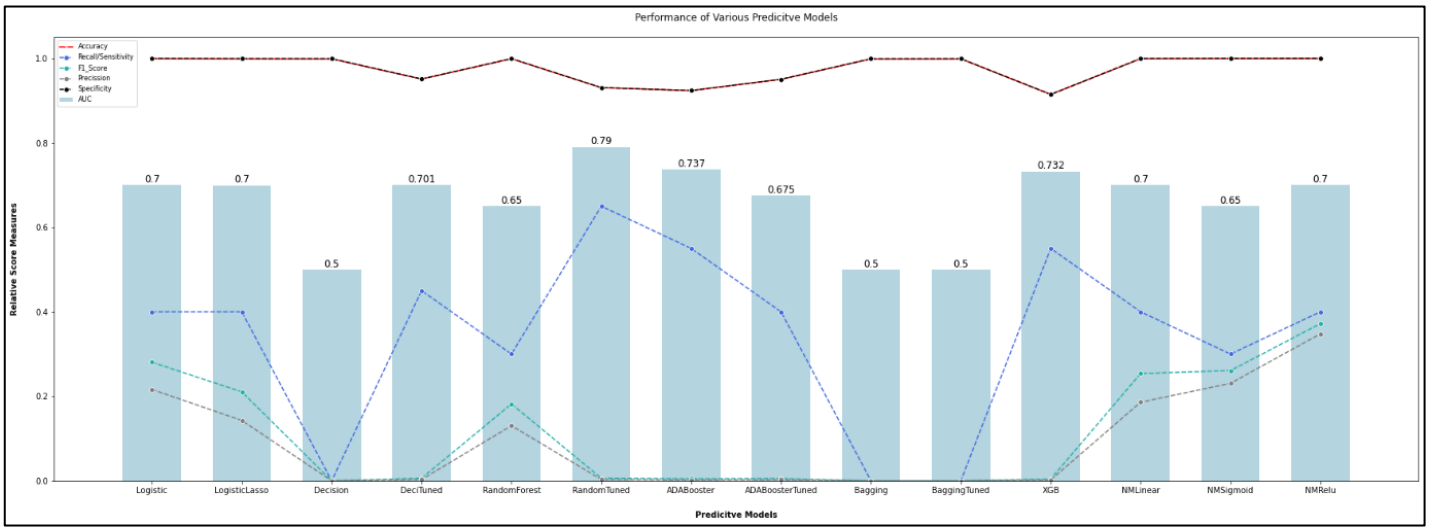

Figure 16: Performance summary for Cancelations

The model which includes departure delays is not practical since it is highly reliant on the delay and in general, departure information is not always accessible. But, again, it would be incorrect not to consider it since, even if there was a

departure delay, because the model ignored it, it may end up indicating no delay. So, it depends on the case study one tries to answer. For this report, the model without departure delay will be considered. Based on the above analysis the models selected for further evaluation are Bagging Model with Tunning (*base_estimator=DecisionTreeClassifier(), n_estimators=70*) and Logistic Regression model. Furthermore, the features importance for both the models were calculated listing top 30 features to derive useful insights.

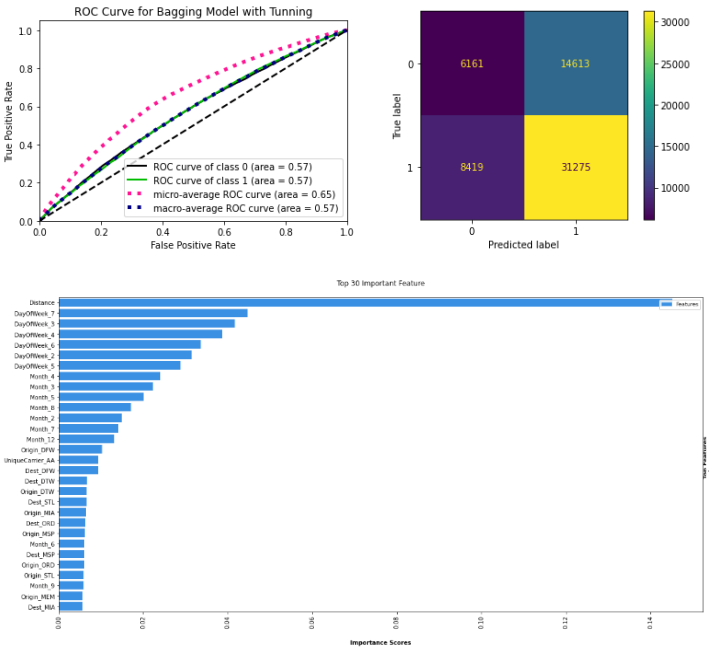

Figure 17: Bagging Classifier with Tunning – Arrival Delay Prediciton

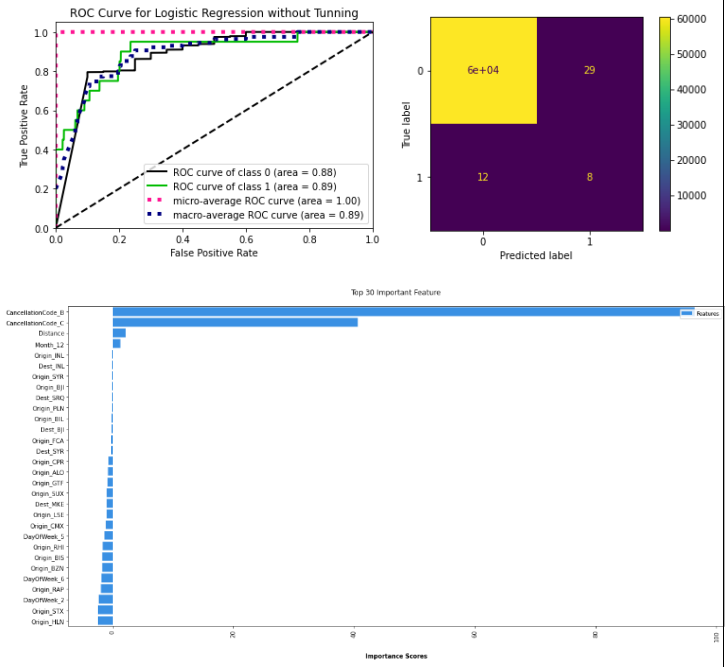

Figure 18: Logistic Regression – Cancellation Prediciton

Feature Evaluation for Delay Prediction: Based on the significance graph of characteristics, the maximum weights are given to predictors - *Distance*, Weekdays – (Sundays, Tuesdays, Wednesday, Thursday, Saturday), Months (February, March, April, May, August, December) – Origin (DFW, DTW, MIA, MSP) – Destinations (DFW, DTW, STL, ORD, MSP), etc.

Feature Evaluation for Cancelation Prediction: Notice there are negative feature importance scores, which means these features are not much of importance to the model, hence these values are ignored. It leaves to top 4 predictors, *Cancelation B, Cancelation C, Distance, Months 12* (December).

RECOMMENDATIONS:

Based on the features extracted the interesting predictor which influences both delays and cancelation is Distance between the airports indicates that the longer the flight takes or the longer the journey, the more likely it may be delayed or cancelled. Airlines can schedule more routes for shorter airport lengths with more connecting airports, so that longer

distances are not covered by a single flight. Another predictor effecting both forecasts is the month of *December*, which was observed in EDA. Airlines can roll out notifications to customer so that they are self-aware of the situation prior booking. In the months when there are greater delays, airlines can have more staff on the ground with proper management. Additionally, Airlines should provide more alternative routes and more flexible booking alternatives for passengers who experience cancellations.

Delays – Other factors contributing to delays are all days of the week except *Mondays* and *Fridays*, this can be interpreted as, airlines can make more room or have more flights on Mondays and Fridays compared to more congested days like *Sundays, Wednesdays.* Noticeably, the airports with both origin and destination have higher delays are *DFW*, *DTW and MSP*. Airlines should avoid such airports, and customers should be informed so that they may arrange other routes if they need to travel early.

Cancellations - The primary predictors for cancellation with *cancellation codes B and C* indicate that most cancellations occur due to weather and the National Air system. Weather cannot be controlled, but an anticipated weather summary may be used to arrange the airlines accordingly. The airline system should plan and route in such a way that there are fewer weather uncertainties. In the case of the National Air System, adequate co-ordination methods should be established to provide clear communication and a short wait time to avoid any cancellations.

CONCLUSIONS:

Airlines place a huge focus on customer satisfaction and with new airlines entering the market, the situation becomes more challenging. Capturing important insights would not only help them go farther but will also assist them in developing a solid consumer base. This report has performed several classifications predictive models, including trees and deep learning neural networks. The flight data was pre-processed, normalized, and scaled to produce reliable predictions. Predictions for arrival delays with and without departure delays were conducted to test the models' efficiency, and models did detect higher accuracy with departure delays, resulting in a cascade effect. To make an informed judgement, predictions for cancellations were obtained. When the various models were compared, it was discovered that the Bagging Model with tuning performed better than the other models in terms of arrival delay predictions without considering departure delays, while the Logistic Regression performed better in terms of cancellation predictions. Feature importance for each scenario were studied and useful recommendations were obtained for airlines company.

REFERENCES:

1. Dothang Truong; Mark A. Friend; Hongyun Chen. Applications of Business Analytics in Predicting Flight On-time Performance in a Complex and Dynamic System. Transportation journal 2018, 57,1, 24-52
2. Xiong, J. (2010). "Revealed Preference of Airlines Behavior under Air Traffic Management Initiatives". PhD thesis. University of California, Berkeley.
3. Rebollo JJ, Balakrishnan H. Characterization and prediction of air trafc delays. Transportation Res Part C Emerg Technol. 2014;44:231–41.
4. E. Esmaeilzadeh and S. Mokhtarimousavi, "Machine learning approach for flight departure delay prediction and analysis," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 8, pp. 145–159, 2020.
5. Yazdi, M.F., Kamel, S.R., Chabok, S.J.M. *et al.* Flight delay prediction based on deep learning and Levenberg-Marquart algorithm. *J Big Data* **7,** 106 (2020). https://doi.org/10.1186/s40537-020-00380-z
6. Zoutendijk, M. and Mitici, M., 2021. Probabilistic Flight Delay Predictions Using Machine Learning and Applications to the Flight-to-Gate Assignment Problem. *Aerospace*, 8(6), p.152.
7. TY - JOUR AU - Lambelho, Miguel AU - Mitici, Mihaela AU - Pickup, Simon AU - Marsden, Alan PY - 2019/12/11 SP - T1 - Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions VL - DO - 10.1016/j.jairtraman.2019.101737 JO - Journal of Air Transport Management ER
8. Eight to Late. 2021. *A gentle introduction to logistic regression and lasso regularisation using R*. [online] Available at: <https://eight2late.wordpress.com/2017/07/11/a-gentle-introduction-to-logistic-regression-and-lasso-regularisation-using-r/> [Accessed 12 December 2021].
9. Analytics Vidhya. 2021. *Tree-Based Machine Learning Algorithms | Compare and Contrast*. [online] Available at: <https://www.analyticsvidhya.com/blog/2021/04/distinguish-between-tree-based-machine-learning-algorithms/> [Accessed 12 December 2021].