**Topic**: FLIGHT STATUS PREDICTION

**Team No:** 18

**Team Members:** Forum Karia, Vatsal Mehta

**INTRODUCTION**:

**Background**:

Flight delays and cancellations are common in the aviation industry, causing inconvenience for passengers and costing airlines money. While weather is an uncontrollable factor, inadequate planning, mechanical issues, air traffic congestion, inadequate staffing, and poor communication also contribute to delays. Airlines can address these issues by improving planning, maintaining aircraft, reducing congestion, ensuring adequate staffing and training, and providing clear communication to passengers. Addressing these issues can improve efficiency, reduce costs, and provide a better travel experience for passengers.

**Motivation**:

Exploratory data analysis and machine learning can be used to address flight delays. These techniques can analyze past flight data to forecast delays and identify underlying causes, allowing airlines to take preventative and remedial action. Additionally, EDA and ML can help airlines identify opportunities for improvement and develop more efficient operations. Overall, these techniques offer a powerful toolset to minimize the impact of delays, improve efficiency, and enhance the travel experience for passengers.

**Goals**:

The project aims to build a machine learning model that can accurately predict flight delays or cancellations based on past data. The model would consider various factors that affect flight status, enabling airlines to schedule and staff flights more efficiently and improve the travel experience for passengers. Benefits include proactive adjustments to schedules, improved passenger experience, and identification of operational inefficiencies for cost savings and revenue growth.

**METHODOLOGY**:

Our project's methodology involves the following:

1. **Data Collection:** We obtained our dataset from Kaggle.

**DESCRIPTION OF DATASET:**

The Flight Status Dataset is a sizable dataset that includes details on the departure and arrival times for major US airlines. The dataset may be helpful for examining patterns in flight delays, figuring out what causes them, and perhaps even forecasting delays for upcoming flights. We can learn a lot about the airline industry's potential for development by investigating the statistics.

Link: <https://www.kaggle.com/datasets/robikscube/flight-delay-dataset-20182022>

2. **Data preprocessing and cleaning:** Data preprocessing and cleaning involve checking for duplicates, handling missing values, and formatting data appropriately for machine learning models. These steps are essential to ensure accurate, complete, and relevant data for the model to generate accurate predictions and provide valuable insights.

Here’s what we did in our project:

1. Heatmap showing the null values:

Graphical user interface

Description automatically generated

1. Heatmap After dropping the null values:

A picture containing graphical user interface

Description automatically generated

3. **Exploratory Data Analysis:** To obtain insights and find patterns in the data, we will do exploratory data analysis after cleaning and preparing the data. This will involve visualizing the data using various plots and graphs.

1. Creating a Heatmap to see the correlation:

Timeline

Description automatically generated

1. Created a Histplot to visualize the distribution of the "ArrDelayMinutes" column of a DataFrame using a histogram, which shows how frequently each delay value occurs in the data. The kernel density estimate line can provide additional information about the shape of the distribution.

Table

Description automatically generated with medium confidence

1. Created a Jointplot showing a strong positive correlation between departure delay and arrival delay, which is expected. However, the plot also reveals that some airlines have a consistently higher or lower delay time compared to others. It could be due to the difference in the airline's policies, efficiency, or maintenance issues. This information can be helpful for travelers to choose an airline with less delay time.

Chart

Description automatically generated

1. Created a Histogram for the distribution of arrival delay times in the data, showing how frequently each delay time occurs.

Chart, histogram

Description automatically generated

1. Created a Boxplot of the distribution of arrival delay times in the data, showing the median, quartiles, and outliers of the data. We can see the spread of data to identify any extreme values or potential errors.

Chart, box and whisker chart

Description automatically generated

1. Created a Scatter plot to visualize the relationship between the departure delay and arrival delay times in the data. The scatter plot displays each data point as a point in the coordinate plane, showing how the two variables are related. This helps us to see if there is a correlation between the two variables and if any patterns or outliers exist in the data.

Chart, scatter chart

Description automatically generated

1. The first plot uses the histplot() function to create a histogram of the "DepDelay" column.

The second plot uses the boxplot() function to create a box plot of the "DepDelay" column. Overall, these two plots can be used to explore the distribution of departure delay times in the data. The histogram shows the frequency of delay times, while the box plot shows the median, quartiles, and outliers of the data. The user can identify any extreme values or potential errors and see the spread of the data.

Chart, histogram

Description automatically generated

Chart

Description automatically generated

1. The first plot creates a bar chart using sns.countplot(). It shows the number of flights for each airline, with the frequency count of each airline displayed on the y-axis.

The second plot creates a bar chart using sns.barplot(). It shows the average arrival delay time for each day of the week, with the day of the week displayed on the x-axis and the average arrival delay time displayed on the y-axis.

Overall, these plots can help identify any patterns or trends in our data related to the categorical variables. The count plot shows the frequency of each category in a single column, while the bar plot shows the average value of a numeric variable for each category in a categorical column.

Chart, waterfall chart

Description automatically generated

Chart, box and whisker chart

Description automatically generated

1. Created this to calculate and visualize the average departure delay by destination city and airline.

Chart, bar chart

Description automatically generated

1. Created a box plot of arrival delays by airline.

The y-axis represents the airline names, and the x-axis represents the arrival delay in minutes. The box plot shows the median value, interquartile range (IQR), and any outliers in the data for each airline.

Chart, box and whisker chart

Description automatically generated

1. Here we grouped the airlines by average arrival delay and create a horizontal bar plot to visualize the results.

Chart, bar chart

Description automatically generated

4. **Feature Selection:** We will choose the most relevant features to be employed in the machine learning models based on the insights obtained by EDA. To determine the most crucial qualities, this may include employing methods like principal component analysis or correlation analysis.

Selected the columns which we needed for our project out of all 61 columns in the dataset

Text

Description automatically generated

5. **Model Training:** Using the prepared dataset, we may train the machine learning models when the relevant features have been found. In this case, the dataset is divided into training and testing sets, the models are trained on the training set, and their performance is tested on the testing set.

The input data, x, and the target variable, y, are split into four different arrays: x\_train, x\_test, y\_train, and y\_test.

The testing set size is set to 0.2, which means that 20% of the data will be used for testing, while the remaining 80% will be used for training.

The random\_state parameter is set to 1 to ensure that the same random splits are generated each time the code is run, which helps to maintain consistency in the results.

Graphical user interface, text, application, email

Description automatically generated

6. **Model Evaluation:** When the models have been trained, we may assess their effectiveness using a variety of measures, including accuracy, precision, recall, and F1 score. This will assist us in determining the model or algorithm that performs the best at predicting flight status.

We have used 4 of the machine learning algorithms:

1. **Linear regression:** When using linear regression for flight status prediction, the dependent variable (output variable) would be the predicted delay time for a flight, and the independent variables (input features) would include factors such as airline, departure time, etc.

The model's performance can be evaluated using metrics such as the mean squared error (MSE) and root mean squared error (RMSE). These metrics measure the average squared difference between the predicted delay times and the actual delay times.

The formula for MSE and RMSE is:

MSE = (1/n) \* ∑(yi - ŷi)²

RMSE = √(MSE)

Chart, line chart

Description automatically generated

1. **Decision Tree:** Decision Tree is a supervised machine learning algorithm used for classification and regression tasks. It creates a tree-like model of decisions and their possible consequences based on input features. For flight status prediction, Decision Tree can be used to categorize flights into delayed and not delayed categories based on input features such as airline, departure time,etc. It is easy to interpret and can handle both categorical and numerical input features but is prone to overfitting and requires careful tuning of hyperparameters and pruning techniques.

Chart, line chart

Description automatically generated

1. **Random Forest:** Random Forest is an ensemble learning approach used for classification and regression tasks. It combines multiple decision trees trained on different subsets of data and input features to make more accurate predictions. For flight status prediction, a Random Forest model can be used to forecast the chance of an aircraft being delayed based on input features such as airline, departure time, etc. It can handle a large number of input features and missing data and provides a more accurate and reliable prediction compared to other machine learning algorithms.

Chart, line chart

Description automatically generated

1. **XGBoost:** XGBoost is an ensemble learning algorithm used for both classification and regression tasks. It constructs a series of decision trees, learning from the mistakes of previous trees to reduce bias and variance in the model. For flight status prediction, XGBoost can be used to forecast the chance of an aircraft being delayed or cancelled based on input features. It can handle both categorical and numerical data, captures intricate correlations between characteristics, works well for large datasets and can automatically select important features.

Chart, line chart, scatter chart

Description automatically generated

**Compared all the models MSE, RMSE and R2 Scores:**

* + - 1. Plot for MSE values of all models:

Chart

Description automatically generated

2.Plot for RMSE values of all models:

Chart

Description automatically generated

3.Combined:

Text, table

Description automatically generated with medium confidence

**From the above figures, it can be concluded that Linear Regression is not performing well and Decision Tree, RandomForest perform almost similarly. However, XGBoost performs the best.**

**7.** Implementing PCA to reduce dimensions

a. Standardizing data (Normalization):

Graphical user interface

Description automatically generated with medium confidence

Chart, bar chart

Description automatically generated

* 1. Implementing all the algorithms again:

First replacing the missing values with the mean:

Graphical user interface, text

Description automatically generated

* + - 1. Linear Regression After PCA:

**Chart, line chart

Description automatically generated**

* + - 1. Decision Tree Regressor after PCA:

Chart, line chart

Description automatically generated

* + - 1. Random Forest After PCA:

Chart, line chart, scatter chart

Description automatically generated

4.XGBoost after PCA:

Chart, line chart

Description automatically generated

Comparing the MSE and RMSE of all models after PCA:

1.MSE of all models:

Chart

Description automatically generated

2. RMSE of all models:

Chart, bar chart

Description automatically generated

**CONCLUSION:**

From all the above models, before PCA and after PCA, we can conclude that linear model has high MSE and RMSE compared to all other models. Random Forest and Decision tree models work almost the same and XGBoost works the best in both conditions. Hence the status of the flights can be predicted well using XGBoost model.

**References:**

1. Kaggle: <https://www.kaggle.com>  
2. StatQuest with Josh Starmer: <https://www.youtube.com/user/joshstarmer>

3. Stack Overflow: <https://stackoverflow.com/>

4.GitHub: <https://github.com/>

5.<https://scikitlearn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html>

6.<https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html>

7.<https://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

8.<https://xgboost.readthedocs.io/en/latest/python/python_api.html#module-xgboost.sklearn>

9. <https://towardsdatascience.com/decision-tree-regression-in-python-b185c10d2c52>

10.<https://towardsdatascience.com/random-forest-in-python-24d0893d51c0>

11.<https://towardsdatascience.com/xgboost-python-annotated-6c9b6aa48c4>

12.<https://scikitlearn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>