Project Proposal Group 16 Prediction of Flight Delays

Group Members:
Qianjing Liu A0280514L
Teong Qi En A0114257L
Elisabeth Angelina A0057445B
Sipan Yin A0280545A

Abstract

Flight delays are a significant challenge in the aviation sector, disrupting operations and impacting passenger satisfaction. This report presents the development of a predictive flight delay detection system using real-time data and advanced analytics. Drawing insights from a dataset of US domestic flights in 2018, the report outlines key preprocessing steps, feature engineering techniques, and model selection criteria. Challenges such as data quality and interpretability are addressed through robust data governance and explainable AI methods. Implementation of predictive flight delay detection systems promises to enhance operational efficiency and customer satisfaction in the aviation sector.

1. Introduction

Flight delays disrupt passenger itineraries and challenge operational efficiency in the aviation sector. These delays, stemming from diverse factors like weather conditions and technical glitches, offer a unique opportunity for innovation through predictive analytics. From the data center of the Bureau of Transportation Statistics (BTS) which is exposed by the United State department of transport, there are 79.49% flights on time while others are delayed, canceled and diverted. (Bureau of Transportation Statistics, 2024) Almost 20% flights are delayed is a huge number since every day, FAA's Air Traffic Organization (ATO) provides service to more than 45,000 flights and 2.9 million airline passengers across more than 29 million square miles of airspace. (Quora, 2024)

Our solution, a predictive flight delay detection system, leverages real-time data to forecast delays, enabling airlines to optimize schedules, manage resources, and improve communication, thereby transforming a common challenge into an opportunity for enhanced service provision. Specifically, our solution aims to benefit the following parties:

Customers:

- Transparency: Immediate updates on flight statuses enable passengers to make informed decisions, reducing the stress associated with travel disruptions.
- Enhanced Experience: Efficient handling of delays leads to smoother rebooking and reduced waiting times, significantly improving the overall travel experience.
- Increased Trust: Reliable and effective communication fosters a sense of trust and loyalty among passengers towards the airline.

To Airport Services:

- Operational Workflow Optimization: Anticipating flight delays allows for better resource allocation, including gate assignments and ground staff scheduling, enhancing airport efficiency.
- Improved Passenger Flow Management: With advance notice of delays, airport services can manage passenger flows more effectively, reducing congestion and improving the passenger experience at the airport.
- Enhanced Coordination: Real-time delay predictions facilitate improved coordination between airlines and airport services, ensuring a smoother response to operational disruptions.

Third-Party Insurance Providers:

- Product Differentiation: Access to predictive delay information allows insurers to offer innovative, tailored products, such as dynamic pricing based on predicted flight reliability, thereby attracting more customers.

By delivering tangible benefits across the ecosystem, including passengers, insurance providers, and airport services, this project positions the airline as a leader in customer-centric and efficient operations within the competitive US domestic aviation market.

2. Data pre-processing

Data set depicts US domestic flights in 2018 comprising of 300000 rows × 28 columns. Table 1 shows a summary of the column information

COLUMN NAME	DESCRIPTION
FL_DATE	Airline Identifier
OP_CARRIER	Flight Number
OP_CARRIER_FL_NUM	Starting Airport Code
ORIGIN	Destination Airport Code
DEST	Planned Departure Time
CRS_DEP_TIME	Actual Departure Time
DEP_TIME	Total Delay on Departure in minutes
DEP_DELAY	The time duration elapsed between departure from the origin airport gate and wheels off
TAXI_OUT	The time point that the aircraft's wheels leave the ground
WHEELS_OFF	The time point that the aircraft's wheels touch on the ground

.

WHEELS_ON	The time duration elapsed between wheels-on and gate arrival at the destination airport
TAXI_IN	Planned arrival time
CRS_ARR_TIME	Actual Arrival Time
ARR_TIME	Total Delay on Arrival in minutes
ARR_DELAY	Flight Cancelled (1 = cancelled)
CANCELLED	Reason for Cancellation of flight: A - Airline/Carrier; B - Weather; C - National Air System; D - Security
CANCELLATION_CODE	Aircraft landed on airport that out of schedule
DIVERTED	Planned time amount needed for the flight trip
CRS_ELAPSED_TIME	AIR_TIME+TAXI_IN+TAXI_OUT
ACTUAL_ELAPSED_TIME	The time duration between wheels_off and wheels_on time
AIR_TIME	Distance between two airports
DISTANCE	Delay caused by the airline in minutes
CARRIER_DELAY	Delay caused by weather
WEATHER_DELAY	Delay caused by air system
NAS_DELAY	Delay caused by security
SECURITY_DELAY	Delay caused by aircraft
LATE_AIRCRAFT_DELAY	Airline Identifier

Table 1: Overview of dataset

2.1 EDA

There are a total of 29 columns, EDA and feature engineering were performed. There are four aspects we considered. Descriptive statistics, data types and missing value, unique counts, and correlation matrix.

Descriptive Statistics: We did descriptive statistics to show the number of non-null, mean, standard deviation, min, max, first quantile, second quartile and third quantile. In appendix 1(descriptive statistics for columns), we can have the insight that column ‘unnamed: 27’ is full of NaN. Additionally, there is null issue in this dataset from the observation from count.

Data Types and Missing Value: When we check the info of the dataframe, the output shown in the appendix 2 demonstrates few insights. First, there are 20 features that are float, 4 features are integers and 5 features are objects. Second, the non-null count shown in appendix 2 is the same as the count in appendix 1. Third, column unnamed: 27’ is full of NaN.

Unique Counts: This step can help us have a clearer view of how many unique counts for each column, which will help the future analysis especially for the one-hot encoding process. The unique counts for the dataset is shown in appendix 3.

Correlation matrix: A covariance matrix is a square matrix that displays the covariance between each pair of variables in a dataset. In statistical terms, covariance provides a measure of the strength and direction of the relationship between two variables. The correlation matrix is similar to the covariance matrix but goes a step further by standardizing the covariance values. This standardization scales the values to lie between -1 and 1, which represent perfect negative and positive linear relationships, respectively. In a correlation matrix, each element represents the Pearson correlation coefficient between two variables, which measures the linear relationship between them. To have a direct view of all the variables, our group built a heatmap in cool warm color to show the correlation between each pair of the features in appendix 4, while the squares that are more intense in color (deep blue or deep red) indicate stronger correlations. From appendix 4, there appears to be a strong positive correlation between ACTUAL_ELAPSED_TIME and CRS_ELAPSED_TIME, which is expected since these variables are likely to be closely related (the scheduled vs actual time of flights).

2.2 Feature Engineering

As the goal of the project is to predict the flight that will delay over 15 minutes. First, we drop the null column ‘Unnamed: 27’ which is invaluable and handle missing ‘DEP_TIME’ for cancelled or diverted flights. Second, we created a target column for classification categories ‘CLASSIFICATION’ with defining the data where 'DEP_DELAY' >= 15, the ‘CLASSIFICATION’ = 'DEPARTURE_DELAYED' while the flight is cancelled and diverted, the ‘CLASSIFICATION’ is 'CANCELLED_DIVERTED'.

By defining functions of convert_time_to_minutes and get_time_of_day, the dataset can calculate the 'DEP_DELAY' correctly.

convert_time_to_minutes: If ‘time_va’l is null return NaN; handle the time formatted as HHMM (e.g. 1517); handle time formatted as HH:MM (e.g. 15:17), return NaN for other formats. Applied the function to convert time columns to minutes since midnight. Time columns include 'CRS_DEP_TIME', 'DEP_TIME', 'CRS_ARR_TIME' and 'ARR_TIME'.

get_time_of_day: Function to categorize time of day. Create the new column 'TIME_OF_DAY’

For better prediction, we converted the FL_DATE column to datetime and extracted related features to new columns ‘DAY_OF_WEEK’, ‘DAY_OF_MONTH’ and ‘IS_WEEKEND’. Furthermore, we also defined public holidays and eve of public holidays for January 2018 by

assuming 1st and 15th January are holidays. Additionally, we created a new column ‘'FLIGHT_CATEGORY' to classify the flight distance with categories 'Short-Haul', 'Medium-Haul' and 'Long-Haul'. What’s more important, to avoid data leakage, there are few columns that have a strong correlation with our target column, so we dropped those columns.

We analyzed the columns 'ORIGIN' and 'DEST' by checking whether the city is the capital of the country. We saved the processed Data frame to a new CSV file. The size of the dataset(df_processed) is 300000 rows × 21 columns.

2.3

2.4 One Hot Encoding

From the previous step, we created a new column ‘TIME_OF_DAY’ which has categories '6-10am', '10-2pm', '2-6pm', '6-10 pm' and ‘10pm-6am'. We used one-hot encoding to embed this feature.

COLUMN	DESCRIPTION
Numerical Columns	'CRS_ARR_TIME', 'CRS_ELAPSED_TIME', 'DISTANCE', 'DAY_OF_MONTH'
Categorical Columns	'OP_CARRIER', 'DAY_OF_WEEK', 'IS_WEEKEND', 'IS_PUBLIC_HOLIDAY', 'IS_EVE_PUBLIC_HOLIDAY', 'ORIGIN', 'DEST'
Target	‘CLASSIFICATION’

We label encoded categorical columns and scaled numerical columns.

2.5 Train Test Split and SMOTE to Deal with Imbalanced data

```
SCHEDULED              221017
DEPARTURE_DELAYED       65515
CANCELLED_DIVERTED      13468
Name: CLASSIFICATION, dtype: int64
```

Figure 1: Imbalanced dataset

- X is df_processed without ‘CLASSIFICATION’
- Y is ‘CLASSIFICATION’ column in df_processed

We split the dataset to train and test set by having 20% into the testing dataset and 80% to the training dataset. Imbalanced data will cause the model to have a higher weight on learning the majority part but kind of ignore the minor part which causes the model to be biased, so we need to solve this issue to make our model more robust and accurate. Since the ‘CLASSIFICATION’ of the DEPARTURE_DELAYED is much less than the SCHEDULED, we would like to use SMOTE implementing on both training dataset and testing dataset to solve the imbalanced data issue.

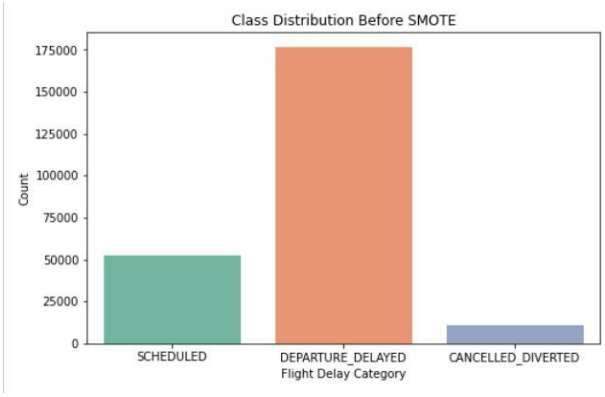


Figure 2: Class distribution before SMOTE

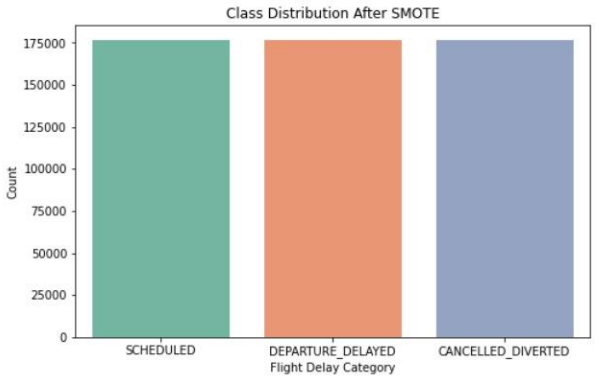


Figure 3: Class distribution before SMOTE

From Figure 2 and Figure 3, after SMOTE was applied, the data distribution is more balanced.

3. Model Selection

Several models were considered and assessed. For the evaluation, we would like to implement cost-sensitive learning. Cost-sensitive learning is a subfield of machine learning that addresses classification problems where the misclassification costs are not equal (Elk01, FernandezGarciaG+18, LS08). Cost-sensitive problems occur in many disciplines such as medicine (e.g., disease detection), engineering (e.g., machine failure detection), transport (e.g., traffic-jam detection), finance (e.g., fraud

detection), and so forth. They are often related to the class-imbalance problem since in most of these problems, the goal is to detect events that are rare. The training datasets therefore typically contain fewer examples of the event of interest. (Fraud Detection Handbook, 2024)

In evaluating the accuracy of our predictive model, we prioritize the true negative rate as a key parameter. When constructing our business matrix, we assign different weights to each outcome to reflect their respective impacts.

Selling to an Insurance Company:

Insurance companies are often concerned with risk management and cost optimization. They might be particularly interested in minimizing the costs associated with false negatives and false positives in specific classes:

False Negatives for CANCELLED_DIVERTED: Missing a prediction where a flight is cancelled or diverted could be costly in terms of payouts for claims related to travel disruptions.

False Positives for DEPARTURE_DELAYED: Incorrectly predicting a delay might lead to unnecessary rebooking costs or compensations.

In this context, the cost matrix could be adjusted to impose higher penalties for misclassifying cancellations and diversions. Precision (minimizing false positives) in predicting cancellations might be prioritized, along with recall (minimizing false negatives) for the same category.

cost_matrix_insurance =

np.array([[0, 20, 50], # Cost of predicting actual SCHEDULED as each class

[50, 10, 80], # Cost of predicting actual DEPARTURE_DELAYED as each class

[100, 80, 10]]) # Cost of predicting actual CANCELLED_DIVERTED as each class

For example, for the first row, 0 means cost of correctly predicting ‘SCHEDULE’ as ‘SCHEDULE’. 20 means cost of incorrectly predicting ‘SCHEDULE’ as ‘DEPARTURE_DELAYED’. 50 means costs of incorrectly predicting ‘SCHEDULE’ as ‘CANCELLED_DIVERETED’. The number in the diagonal shows the costs of correct prediction for each categories. We assigned small positive number for correctly predict DEPARTURE_DELAY and CANCELLED_DIVERT, while assigned non costs for correctly predicting ‘SCHEDULE’ as ‘SCHEDULE’.

For Insurance Companies: Focus on high recall for CANCELLED_DIVERTED to ensure all potential claims are anticipated. Precision for DEPARTURE_DELAYED to avoid unnecessary costs.

Selling to Individual Customers:

Individual customers may prioritize accurate information on flight delays more than cancellations, especially if they are planning connections or important events based on flight timings. Here, the impact of a false positive (unnecessarily adjusting plans for a delay that doesn't happen) or a false negative (not being informed about a delay) could be seen as more disruptive personally.

For individual customers, you might emphasize a balanced view but could still adjust the cost matrix to reflect higher penalties for inaccuracies in predicting delays.

cost_matrix_customer =

np.array([[0, 30, 70], # Cost of predicting actual SCHEDULED as each class

[60, 10, 90], # Cost of predicting actual DEPARTURE_DELAYED as each class

[100, 90, 20]]) # Cost of predicting actual CANCELLED_DIVERTED as each class

For Individual Customers: Balanced precision and recall for DEPARTURE_DELAYED to ensure customers receive timely and accurate information about flight changes that could affect their plans.

3.1 Linear Regression

It assumes a linear relationship between the logit of the outcome and each predictor variable. The logit function is the logarithm of the odds of the dependent variable being true. The output of logistic regression is a probability that the given input point belongs to the positive class, which is computed using the logistic function, also known as the

sigmoid function. The logistic function outputs a value between 0 and 1, which is interpreted as the probability.

From the data set, accuracy of 0.68 was obtained.

3.1.1 CONFUSION MATRIX AND COST/BENEFIT ANALYSIS

Confusion Matrix from model's prediction:	[[594 332 1796] [627 3041 9237] [1897 5274 37202]]
	cost_matrix_insurance = np.array([[0, 20, 50], [50, 10, 80], [100, 80, 10]])
	cost_matrix_customer = np.array([[0, 30, 70], [60,10, 90], [100, 90, 20]])
Total Cost/Benefit for Insurance Companies:	$2,443,440
Total Cost/Benefit for Customer:	$1,880,800

The logistic regression model achieved an accuracy of approximately 68.06%. The total benefit for the insurance companies is $2,443,440 while the total benefit for the customer is $1,880,800.

3.2 XGBoost

XGBoost is a type of ensemble tree method that uses the boosting technique. It combines multiple weak learners (decision trees) to create a strong learner in a sequential manner, where each new tree corrects errors made by the previous ones.

From the data set, accuracy of 0.74 was obtained.

3.2.1 CONFUSION MATRIX AND COST/BENEFIT ANALYSIS

Confusion Matrix from model's prediction:	[[1259 479 984] [473 4612 7820] [1009 4571 38793]]
	cost_matrix_insurance = np.array([[20, 10, 40], [10, 30, 100], [50, 20, 30]])
	cost_matrix_customer = np.array([[10, 50, 10], [20, 10, 50], [10, 40, 20]])
Total Cost/Benefit for Insurance Companies:	$2,149,700
Total Cost/Benefit for Customer:	$1,608,660

The Xgboost model achieved an accuracy of approximately 74.44%. The total benefit for the insurance companies is $2,149,700 while the total benefit for the customer is $1,608,660.

3.3 Explainable AI and LIME

The bar chart (Figure 4) displays the importance of different features in two different models we used: logistic regression and xgboost. Feature importance is typically measured to understand which features contribute most to the predictions of the model. Here, the importance is likely calculated using the "weight" method of XGBoost, which counts the number of times a feature appears in the trees of the model. The most important feature according to the chart is CRS_ARR_TIME, followed by CRS_DEP_TIME, and CRS_Elapsed_TIME. These features have the highest importance scores, suggesting they have the most significant impact on the model's predictions.

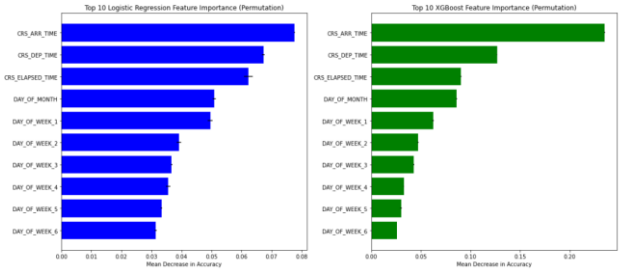


Figure 4: Top 10 Feature Importance of Logistic Regression and

Additionally, we applied LIME to show the feature importance in Figure 5. We generated explanations for both logistic regression and XGBoost models. For logistic regression, the prediction probabilities are displayed along with the feature contributions. The outputs shown are the prediction probabilities from both models and the feature impact on those predictions. The tables detail how much each feature is contributing to the model's predictions. Features with higher absolute values are more influential in the model's decision-making process for the specific instance. The difference in feature importance level is caused by a few reasons. First, the model structure and complexity is different. Logistic Regression is a linear model that assumes a linear relationship between the input features and the log-odds of the dependent variable. It's generally less flexible in capturing complex patterns because it's constrained to linearity. XGBoost, on the other hand, is an implementation of gradient boosted decision trees designed for speed and performance. It is non-linear and can model complex interactions between features. This allows it to capture more intricate patterns in the data, which might not be possible with logistic regression.

Figure 5: LIME Feature Importance for logistic regression and XGBoost

4. Addressing Challenges and Limitations

4.1 Data Quality and Integration

Implementing predictive flight delay detection systems presents several challenges and considerations that must be carefully addressed to ensure their effectiveness and successful integration into airline operations. One of the primary challenges is ensuring the quality and integration of data from various sources, including airline systems, weather forecasts, and air traffic control. The accuracy and completeness of this data are crucial for training accurate predictive models. To overcome this challenge, airlines must implement robust data governance practices, validation processes, and integration frameworks. This involves conducting regular data quality checks, validating data against known standards, and implementing automated data cleansing procedures to maintain data integrity and consistency.

4.2 Model Interpretability and Transparency:

Another significant challenge lies in interpreting complex machine learning models and explaining predictions to stakeholders, including airline executives, regulators, and passengers. Predictive models often employ sophisticated algorithms that can be difficult to interpret, making it challenging to understand how and why certain predictions are made. To address this challenge, airlines can utilize explainable AI techniques such as LIME or SHAP to provide transparent insights into model decisions and feature contributions. By generating explanations for individual predictions and visualizing feature importance, airlines can enhance transparency and build trust with stakeholders.

4.3 Dynamic Nature of Flight Delays:

The dynamic nature of flight delays presents another challenge, as delays can be influenced by a wide range of factors, including weather conditions, air traffic congestion, and operational disruptions. Predicting these delays accurately requires continuously updating and refining predictive models using real-time data streams and adaptive algorithms. By incorporating new data sources, adjusting model parameters, and leveraging advanced forecasting techniques, airlines can adapt to changing conditions and improve forecast accuracy over time.

4.4 Regulatory and Compliance Considerations:

Regulatory and compliance considerations also play a crucial role in implementing predictive flight delay detection systems. Airlines must adhere to strict regulatory requirements and privacy regulations while collecting, storing, and analysing sensitive passenger data. This involves collaborating with regulatory authorities, data protection agencies, and industry stakeholders to ensure compliance with data privacy laws and regulations. Airlines must implement robust data security measures, obtain necessary permissions and consent for data usage, and maintain transparency in data handling practices to protect passenger privacy and ensure regulatory compliance.

4.5 Change Management and Adoption:

Finally, overcoming resistance to change and ensuring buy-in from key stakeholders, including airline staff, ground crews, and IT teams, is essential for successful implementation. Airlines must provide comprehensive training, communication, and support to employees to foster a culture of data-driven decision-making. By educating staff on the benefits of predictive analytics, addressing concerns about job displacement or workflow changes, and involving employees in the implementation process, airlines can encourage ownership and participation in the adoption of predictive analytics tools and processes.

In summary, addressing these challenges through proactive strategies and effective management is essential for the successful implementation of predictive flight delay detection systems. By overcoming obstacles and integrating predictive analytics into airline operations, airlines can drive operational excellence, enhance customer satisfaction, and maintain a competitive edge in the market.

5. Future Works

5.1 Advanced Modelling Techniques:

While the current predictive flight delay detection system has demonstrated promising results, there is scope for further improvement through the exploration of advanced modelling techniques. Future research could focus on implementing deep learning architectures such as recurrent neural networks (RNNs) or convolutional neural networks (CNNs) to capture complex temporal and spatial dependencies in flight delay data. Additionally, ensemble methods such as stacking or boosting could be investigated to combine the strengths of multiple models and enhance prediction accuracy.

5.2 Incorporation of Additional Data Sources:

Expanding the predictive model to incorporate additional data sources could improve its predictive capabilities and robustness. Future works could explore integrating data from sources such as weather forecasts, air traffic congestion patterns, airport operations data, and social media sentiment analysis. By leveraging a diverse range of data inputs, the predictive model can capture a more comprehensive understanding of the factors influencing flight delays and improve forecast accuracy.

5.3 Real-Time Decision Support Systems:

Developing real-time decision support systems that integrate predictive flight delay detection models with operational workflows could streamline airline operations and enhance responsiveness to disruptions. Future works could focus on building interactive dashboards and alerting mechanisms that provide actionable insights to airline staff, ground crews, and airport services in real-time. By enabling proactive decision-making and resource allocation, these systems can minimize the impact of flight delays on passengers and improve overall operational efficiency.

5.4 Continuous Model Monitoring and Refinement

To ensure the long-term effectiveness of the predictive flight delay detection system, continuous model monitoring and refinement are essential. Future works could establish mechanisms for monitoring model performance in production environments, detecting drifts or deviations from expected behaviour, and triggering model retraining or recalibration as needed. By adopting a proactive approach to model maintenance and refinement, airlines can sustain high prediction accuracy and adapt to evolving operational dynamics.

5.5 Integration with Predictive Maintenance

Exploring synergies between predictive flight delay detection and predictive maintenance systems could unlock additional value for airlines. Future works could investigate ways to integrate flight delay predictions with aircraft maintenance schedules, enabling proactive maintenance interventions to prevent potential technical issues that may lead to flight delays. By leveraging predictive analytics across both operational and maintenance domains, airlines can optimize resource allocation, improve aircraft reliability, and enhance overall service reliability for passengers.

In summary, future works in the field of predictive flight delay detection should focus on leveraging advanced modelling techniques, incorporating additional data sources, developing real-time decision support systems, implementing continuous model monitoring and refinement, and exploring integration with predictive maintenance systems. By advancing research and innovation in these areas, airlines can further enhance operational efficiency, customer satisfaction, and competitiveness in the aviation industry.

6. Conclusion

Two machine learning models, Logistic Regression and XGBoost, were evaluated. The XGBoost model demonstrated superior accuracy (approximately 74.44%) compared to the Logistic Regression model (approximately 68.66%), indicating its stronger capability in handling complex patterns in the data.

We meticulously calculate the cost/benefit outcomes for two major stakeholder groups—insurance companies and individual customers—using custom cost matrices to quantify the financial impact of accurate or erroneous predictions.

Our project effectively demonstrates the potential of utilizing advanced predictive analytics to tackle the pervasive problem of flight delays within the US domestic aviation sector. By harnessing real-time data to predict delays, the proposed system is positioned to enhance operational efficiency and customer service for airlines, airport services, and insurance providers.

Our key achievement is providing innovation solutions and comprehensive benefit analysis. The system utilizes a predictive model to forecast flight delays, enabling proactive measures in scheduling, resource management, and passenger communication. The project clearly outlines how different stakeholders, including passengers, airport services, and third-party insurance providers, stand to gain from the implementation of this system. These benefits range from improved operational efficiency to enhanced customer trust and satisfaction.

Descriptive Statistics:

	OP_CARRIER_FL_NUM	CRS_DEP_TIME	DEP_TIME	DEP_DELAY
count	300000.000000	300000.000000	287601.000000	287274.000000
mean	2571.684417	1328.406513	1338.732167	14.048581
std	1894.613086	488.553170	500.873534	52.328898
min	1.000000	1.000000	1.000000	-49.000000
25%	905.750000	915.000000	924.000000	-5.000000
50%	2016.000000	1320.000000	1332.000000	-1.000000
75%	4045.000000	1733.000000	1744.000000	12.000000
max	7439.000000	2359.000000	2400.000000	2007.000000

	TAXI_OUT	WHEELS_OFF	WHEELS_ON	TAXI_IN
count	287450.000000	287450.000000	287129.000000	287129.000000
mean	18.050628	1363.860731	1473.859011	7.750461
std	10.822997	501.220433	527.214785	6.369986
min	1.000000	1.000000	1.000000	1.000000
25%	11.000000	940.000000	1056.000000	4.000000
50%	15.000000	1345.500000	1512.000000	6.000000
75%	21.000000	1759.000000	1913.000000	9.000000
max	165.000000	2400.000000	2400.000000	258.000000

	CRS_ARR_TIME	ARR_TIME	...	ACTUAL_ELAPSED_TIME	AIR_TIME
count	300000.000000	287129.000000	...	286532.000000	286532.000000
mean	1492.678703	1477.993397	...	138.304751	112.513245
std	514.563771	532.080050	...	73.558811	71.659207
min	1.000000	1.000000	...	17.000000	8.000000
25%	1107.000000	1101.000000	...	85.000000	60.000000
50%	1520.000000	1516.000000	...	120.000000	94.000000
75%	1919.000000	1918.000000	...	169.000000	143.000000
max	2359.000000	2400.000000	...	728.000000	674.000000

	DISTANCE	CARRIER_DELAY	WEATHER_DELAY	NAS_DELAY
count	300000.000000	64592.000000	64592.000000	64592.000000
mean	802.440747	21.893191	4.759459	13.655298
std	598.163022	66.593280	36.148859	31.991234
min	31.000000	0.000000	0.000000	0.000000
25%	362.000000	0.000000	0.000000	0.000000
50%	641.000000	1.000000	0.000000	1.000000
75%	1041.000000	19.000000	0.000000	18.000000
max	4983.000000	2007.000000	1682.000000	1317.000000

	SECURITY_DELAY	LATE_AIRCRAFT_DELAY	Unnamed: 27	flight_id
count	64592.000000	64592.000000	0.0	300000.000000
mean	0.122430	27.537435	NaN	150000.500000
std	4.002186	52.298122	NaN	86602.684716
min	0.000000	0.000000	NaN	1.000000
25%	0.000000	0.000000	NaN	75000.750000
50%	0.000000	5.000000	NaN	150000.500000
75%	0.000000	34.000000	NaN	225000.250000
max	593.000000	1648.000000	NaN	300000.000000

[8 rows x 24 columns]

Appendix 1: Column Descriptive Statistic

Appendix

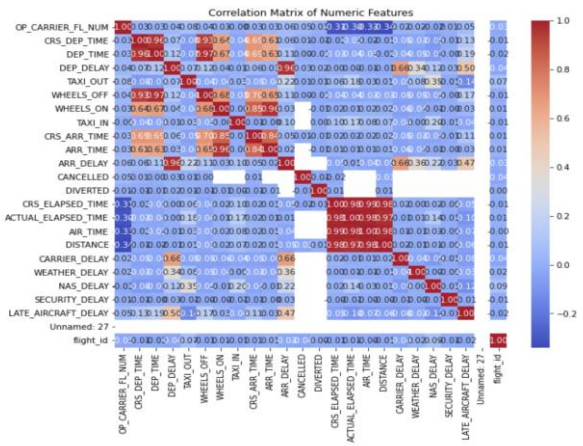
Github link: https://github.com/LiuQianjing111/5153-project/blob/main

```
Data Types and Missing Information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300000 entries, 0 to 299999
Data columns (total 29 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   FL_DATE              300000 non-null  object
 1   OP_CARRIER           300000 non-null  object
 2   OP_CARRIER_FL_NUM    300000 non-null  int64
 3   ORIGIN               300000 non-null  object
 4   DEST                 300000 non-null  object
 5   CRS_DEP_TIME         300000 non-null  int64
 6   DEP_TIME             287601 non-null  float64
 7   DEP_DELAY            287274 non-null  float64
 8   TAXI_OUT             287450 non-null  float64
 9   WHEELS_OFF           287450 non-null  float64
 10  WHEELS_ON            287129 non-null  float64
 11  TAXI_IN              287129 non-null  float64
 12  CRS_ARR_TIME         300000 non-null  int64
 13  ARR_TIME             287129 non-null  float64
 14  ARR_DELAY            286414 non-null  float64
 15  CANCELLED            300000 non-null  float64
 16  CANCELLATION_CODE    12626 non-null   object
 17  DIVERTED             300000 non-null  float64
 18  CRS_ELAPSED_TIME     300000 non-null  float64
 19  ACTUAL_ELAPSED_TIME  286532 non-null  float64
 20  AIR_TIME             286532 non-null  float64
 21  DISTANCE             300000 non-null  float64
 22  CARRIER_DELAY        64592 non-null   float64
 23  WEATHER_DELAY        64592 non-null   float64
 24  NAS_DELAY            64592 non-null   float64
 25  SECURITY_DELAY       64592 non-null   float64
 26  LATE_AIRCRAFT_DELAY  64592 non-null   float64
 27  Unnamed: 27          0 non-null       float64
 28  flight_id            300000 non-null  int64
dtypes: float64(20), int64(4), object(5)
memory usage: 66.4+ MB
None
```

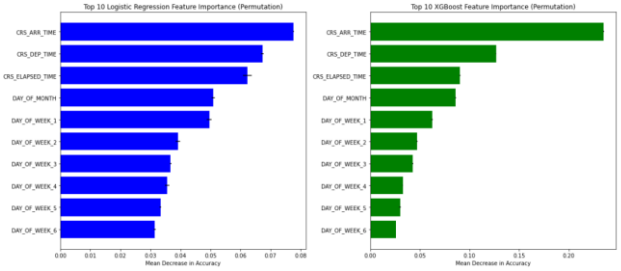
Appendix 2: Data Types and Missing Information

```
Unique Counts in Each Column:
FL_DATE: 17 unique values
OP_CARRIER: 18 unique values
OP_CARRIER_FL_NUM: 6965 unique values
ORIGIN: 333 unique values
DEST: 333 unique values
CRS_DEP_TIME: 1214 unique values
DEP_TIME: 1394 unique values
DEP_DELAY: 853 unique values
TAXI_OUT: 154 unique values
WHEELS_OFF: 1391 unique values
WHEELS_ON: 1439 unique values
TAXI_IN: 123 unique values
CRS_ARR_TIME: 1338 unique values
ARR_TIME: 1439 unique values
ARR_DELAY: 887 unique values
CANCELLED: 2 unique values
CANCELLATION_CODE: 3 unique values
DIVERTED: 2 unique values
CRS_ELAPSED_TIME: 454 unique values
ACTUAL_ELAPSED_TIME: 602 unique values
AIR_TIME: 571 unique values
DISTANCE: 1395 unique values
CARRIER_DELAY: 638 unique values
WEATHER_DELAY: 426 unique values
NAS_DELAY: 331 unique values
SECURITY_DELAY: 73 unique values
LATE_AIRCRAFT_DELAY: 454 unique values
Unnamed: 27: 0 unique values
flight_id: 300000 unique values
```

Appendix 3: Unique Counts for Each Columns



Appendix 4: Heatmap(Correlation Matrix)



Appendix 5: Top ten features for logistic regression and xgboost

Appendix 6: Top ten features for logistic regression and xgboost

References

Bureau of Transportation Statistics. (2024). Causes of flight delays. Retrieved from https://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp?20=E

Fraud Detection Handbook. (2024). Cost-Sensitive Learning. Retrieved from https://fraud-detection-handbook.github.io/fraud-detection-handbook/Chapter_6_ImbalancedLearning/CostSensitive.html

Quora. (2024). How many airplanes fly over the United States each day? Retrieved from https://www.quora.com/How-many-airplanes-fly-over-the-United-States-each-day