# Enhancing Flight Services: Predictive Modeling and Sequential Pattern Analysis to Comparing 2013 Flight Attributes & Behaviours

Supervisor: Soroush Sheikh
Student Names: Eric Traccitto(218835074),
Nicholas Veronico(218567610),
Marcus Sisouphanh(217544792),
Evan Oni(218877670)

Student Emails:erictraccitto@gmail.com,
nv03@my.yorku.ca
n78uak@my.yorku.ca
Evano@my.yorku.ca

Winter 2025

## 1. Objective

This project aims to analyze flight data from 2013 to identify the key factors contributing to flight delays and develop predictive models that estimate the duration of delays. We will leverage machine learning techniques to create accurate regression-based prediction models that predict the length of flight delays rather than just classifying whether a delay will occur. Additionally, the project seeks to generate optimized scheduling strategies for flights predicted to experience delays. By incorporating predictive insights and optimization techniques, our objective is to minimize disruptions, improve scheduling efficiency, and provide actionable insights to reduce delays' impacts. The models used in this project will include Random Forest Regressors, XGBoost, and Support Vector Regression (SVR) to ensure high accuracy and robustness in delay duration estimation.

## 2. Motivation

Flight delays are affected by various external factors such as weather, air traffic congestion, airline operations, and scheduling inefficiencies, making forecasting delays problematic. Understanding these factors and their impacts on flight reliability is important for improving airline performance and passenger satisfaction. This project provides the opportunity to examine flight delay patterns and develop predictive models that estimate the delay duration. Through the use of various machine learning techniques, the goal is to identify causes of delay and optimize scheduling strategies to mitigate flight disruptions. The motivation for this project lies in the interest of better comprehension of flight operations and

improving decision-making in the aviation industry. Better accuracy in predicting flight delays and offering proactive solutions can facilitate a more effective allocation of resources, lower operating expenses and better passenger satisfaction, making the study worthwhile for both airlines and travellers.

## 3. Related Work

Flight delay prediction has been a significant area of research in the aviation industry, with numerous studies focusing on identifying causes and developing predictive models to mitigate delays. Recent advancements in machine learning have enabled researchers to analyze large-scale flight data and uncover patterns that contribute to delays. For example, G et al., 2024 explored the use of machine learning classifiers, such as Support Vector Machines (SVM) and Decision Trees, to predict flight delays. Their study incorporated weather conditions, crew management, and resource planning as key features, achieving high accuracy in delay prediction. The authors emphasized the importance of integrating real-time weather forecasting and operational data into predictive models, which aligns with the objectives of this project.

Another relevant study by Hatıpoğlu & Tosun, 2024 applied machine learning methods to predict flight delays at an airport. The authors compared the performance of various algorithms, including Random Forest and Gradient Boosting, and highlighted the role of external factors such as weather and air traffic in influencing delays. Their findings demonstrated that machine learning models can effectively capture the complex relationships between these factors and flight delays, providing actionable insights for airport operations. This study underscores the potential of data-driven approaches to improve delay prediction accuracy, which is a key focus of this project.

In addition to predictive modeling, sequential pattern analysis has been applied to understand the cascading effects of delays across flight networks. For instance, Czerny & Zhang, 2020 used sequential pattern mining to analyze how delays propagate through connecting flights, revealing that delays in early flights often lead to disruptions in subsequent flights. This insight underscores the importance of proactive measures, such as dynamic scheduling and real-time adjustments, to minimize the ripple effects of delays.

## 4 Methodology

The main goal of this project is to create a strong predictive model that accurately estimates flight delay durations while also considering various external factors such as weather conditions, air traffic congestion, airline schedules, and operational inefficiencies. By analyzing historical flight data, this project aims to uncover patterns and key factors contributing to flight delays. Apart from predicting delays, the project aims to optimize flight schedules by proposing alternative departure times for flights expected to experience significant delays. These optimized schedules can help reduce disruptions, improve resource

allocation and enhance the overall operational efficiency within the airline industry. To achieve these objectives, an extensive methodology using exploratory data analysis, feature engineering, and multiple regression models –including Random Forest Regressor, XGBoost, and Support Vector Regression(SVR) – is created to ensure high accuracy and reliability in delay predictions

### 4.1 Data Preprocessing

The preprocessing phase ensuring data quality by addressing missing values, handling outliers and standardizing key attributes

**Handling Missing and Duplicate Values:**

The dataset was first examined for missing values across all columns. A count of missing values was calculated, allowing us to identify rows with multiple missing attributes. Duplicate rows were also detected and removed to prevent redundant data from affecting our model's performance.

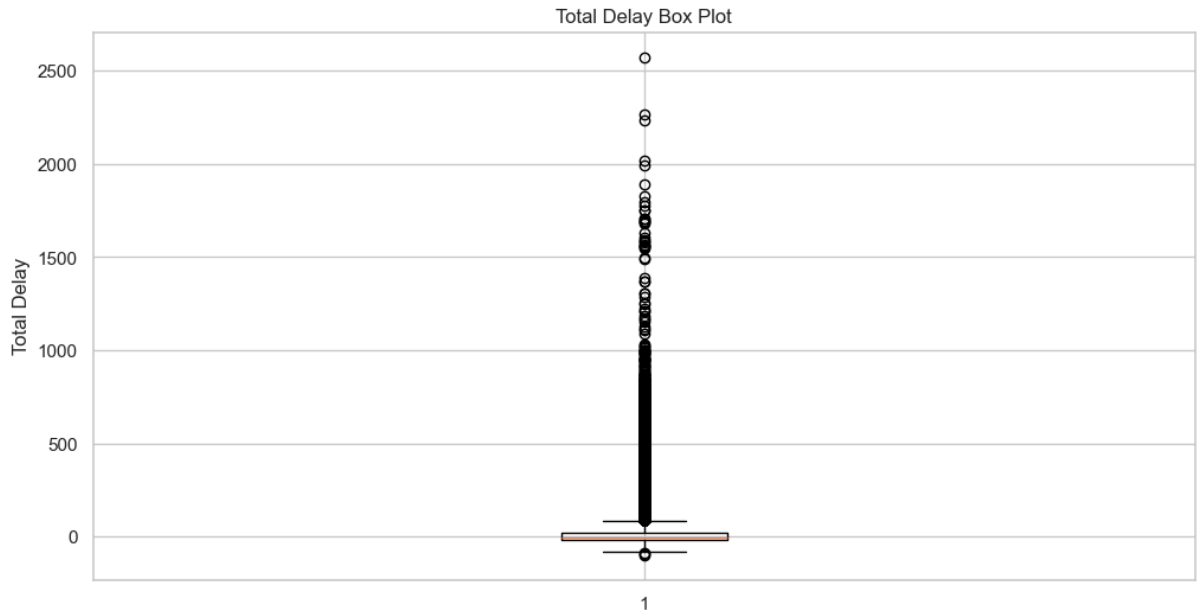
To handle missing values:

- Categorical missing values, such as Tailnum, were replaced with "private" when missing values were present
- Flights were categorized into various statuses based on departure or arrival times:
    - Cancelled (dept_time missing)
    - Incomplete (arr_time missing)
    - Completed (both arr_time and dept_time not missing)

For numerical attributes, missing values were looked at based on the flight status

- Flights that were labelled completed, missing numerical values such as dep_delay, arr_delay, and air_time were replaced with 0
- Outlier values and incorrect values like negative distances or times were either corrected or removed.
-

**Inputting Missing Air_Time values**:
For completed flights with missing air_time, the K-Nearest Neighbors (KNN) imputer algorithm was applied. The model used crucial flight attributes (distance,*dep*time, arr_time, dep_delay, arr_delay) to estimate the missing air_time values. Standardization was performed to normalize the features before imputation. After imputation, rows where air_time stayed zero were removed.




**Outlier Detection and Handling**

Outliers in numerical features were found using the Interquartile Range (IQR) method. The lower bounds and upper bounds were calculated using:

Q1-1.5 *IQR, Q3+1.5*IQR

Q1 and Q3 are the first and third quartiles, respectively. Features such as dep_delay, arr_delay, air_time, and distance were checked for outliers. We did not remove them because there was most likely a reason for them.

## Encoding Categorical Features:

To prepare categorical features for analysis, one-hot encoding was used on carrier, origin and dest attributes. This transformation converted categorical features into binary columns.

**Dataset Merge:**

Our original dataset did not contain weather data. Therefore, we found a weather dataset containing weather features for each of our airports in our dataset and merged them. The new features added were ‘precipitation’, ‘temperature’, ‘wind gust’, ‘wind direction’, ‘wind speed’, ‘pressure’, ‘humidity’, ‘visibility’, and ‘dewpoint’. The weather was merged to our flights dataset if ‘hour’, ‘month’, ‘year’, and ‘origin’ were the same.

**Imputed Missing Values from Dataset Merge**

K-Nearest Neighbors (KNN) imputer algorithm was applied for ‘precipitation’, ‘temperature’, ‘wind gust’, ‘wind direction’, ‘wind speed’, ‘pressure’, ‘humidity’, ‘visibility’, and ‘dewpoint’ to estimate missing values.

## Feature Engineering:

Feature engineering was used to enhance the dataset with meaningful insights that can improve the predictive models

- Departure and Arrival Peak Time Categories
    - To analyze flight patterns based on arrival and departure peak times, categorical features were made by segmenting flight times into specific periods
    - Departure Time Categories:
        - Early Morning(0 - 600):
        - Morning:(600-1200)
        - Afternoon(1200-1500)
        - Afternoon/Evening dep_rush(1500-2000)
        - Evening(2000-2400)

    - Arrival Time Categories:
        - Early Morning Rush(0 - 100):
        - Early Morning Slowdown:(100-700)
        - Morning(1200-1600)
        - Afternoon(1200-1600)
        - Evening Rush(1600-2400)

These categorical values were then added to the dataset for analysis on how peak-time departures and arrivals influence flight delays.

To improve the predictive power of our models, key features were engineered using datetime transformations, cyclical encoding and categorical/numerical encoding techniques These transformations allowed us to ensure our models captured essential patterns in flight delays

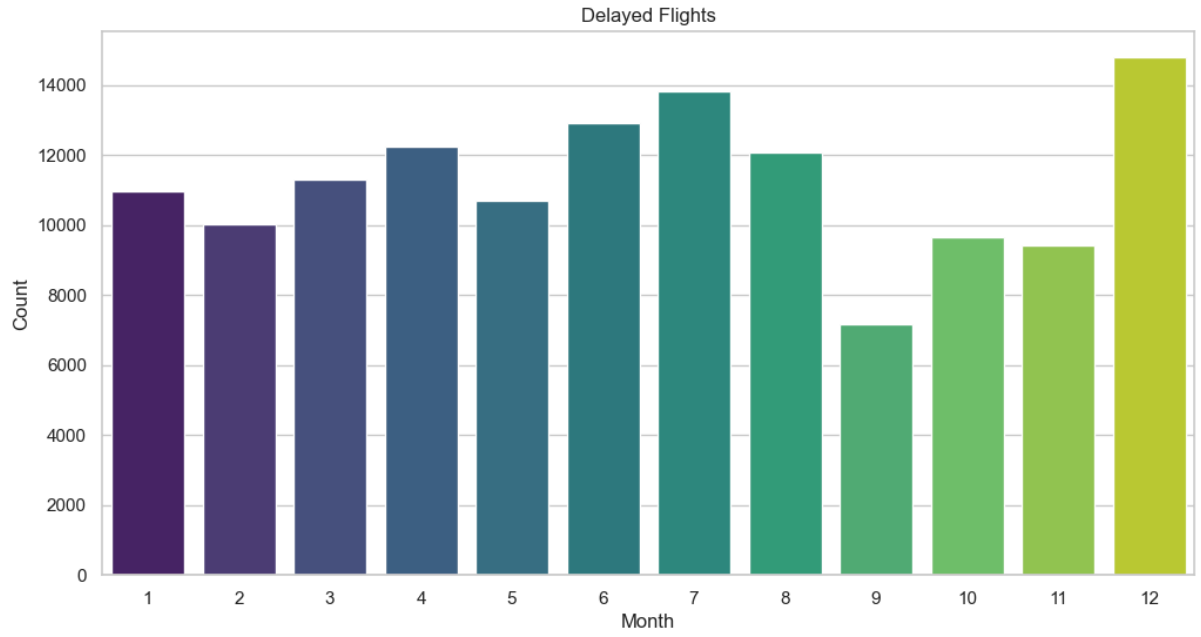
**Cyclical Encoding for Time Features**:
Due to time-related features such as month, week and day exhibit cyclic behavior, they were encoded using sine and cosine transformations to place them on a unit circle.

**Feature Extraction From DateTime**:
- Extracted hour, day_of_week, and month from time_hour_x
- Applied Cyclical encoding:
    - Day of the week(7 day cycle) :day_x, day_y
    - Month(12-month cycle): month_X, month_y

Categorical Numerical Feature Encoding:
- Ensuring data compatibility with the models is crucial, categorical attributes were one-hot encoded, while numerical features were left unchanged
- A column transformer was used to apply these transformations, ensuring the data was properly structured for training.

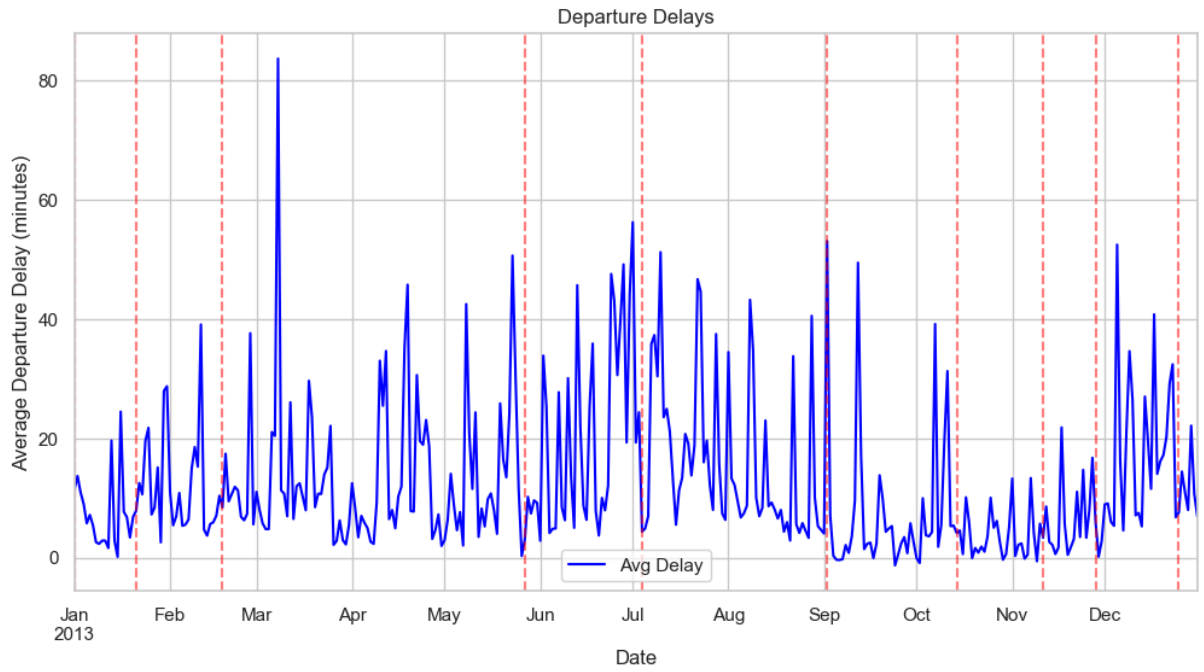
**4.2 Exploratory Data Analysis**

The objective of Exploratory Data Analysis is to identify key patterns and trends contributing to flight delays. We focus on season variations, carrier performance, air congestion and weather conditions to find out their impacts on delays.



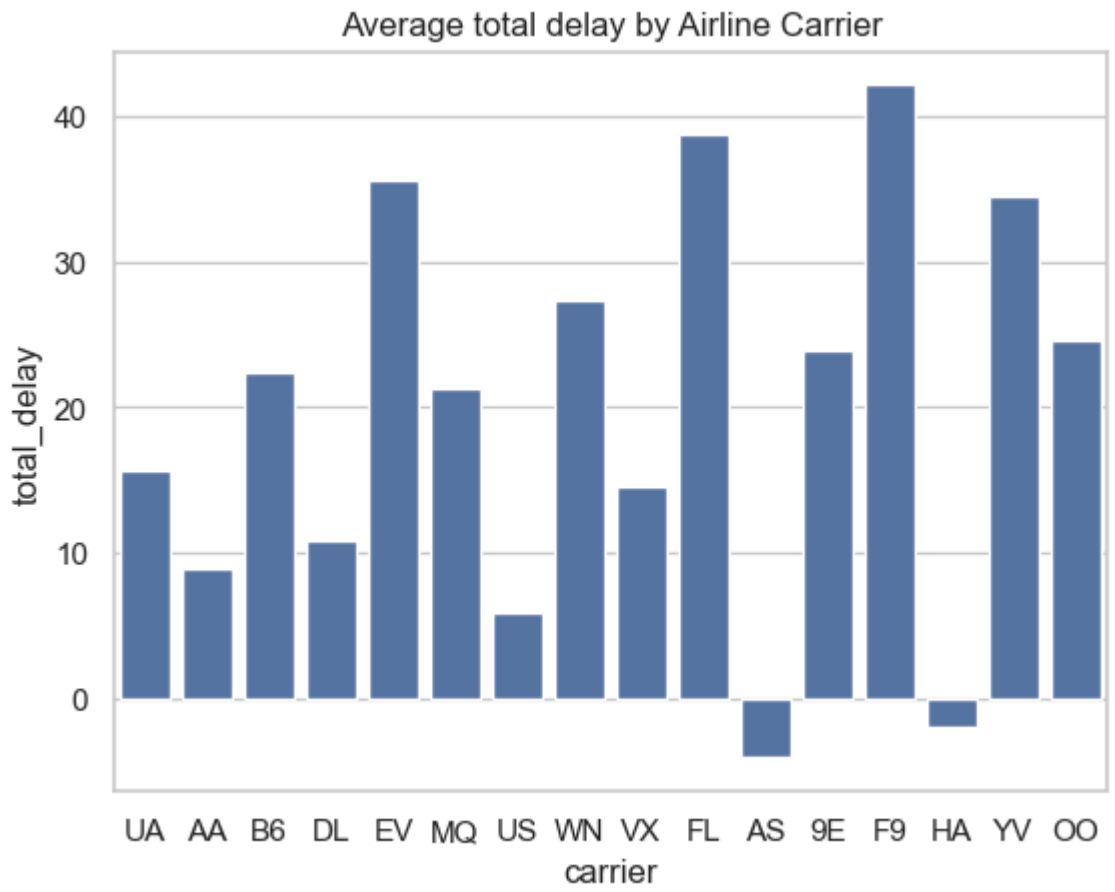

**Seasonal Trends in Flight Delays**:

- Summer Delays (June- August): The highest number of delays occur in the summer, we suspect that it could be due to increased travel during holidays and congestion. Additionally, summer is a peak season for thunderstorms, which can contribute to disruptions.
- Winter Delays (November - January): The second highest delay period occurs in winter, with December showing the highest delays. We believe that this is likely due to holiday travel rather than weather related factors, As January and November experience fewer delays.



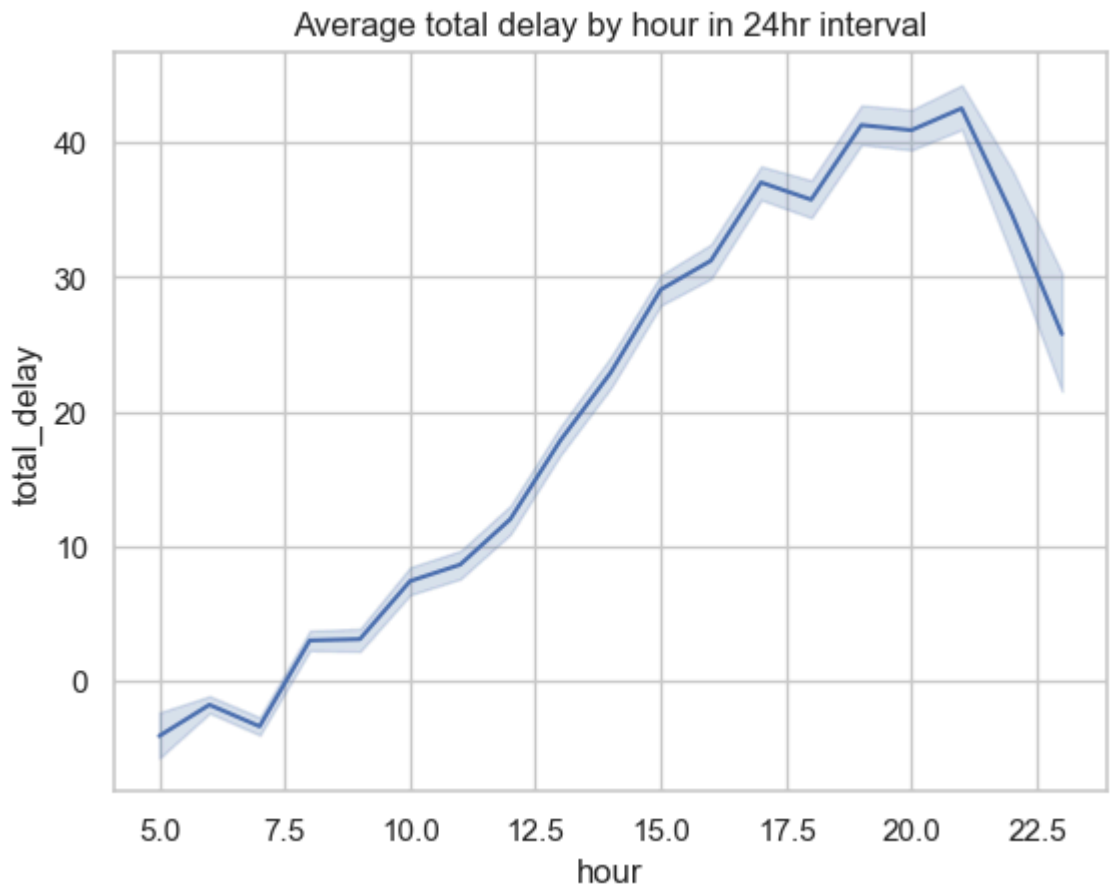

Holiday & Event-Based Delays:

- **Holidays**: Flight delays tend to spike on holidays, which align with increased travel volume
- **Non-Holiday Spikes**: some dates show unusual delay spikes, even though they aren't holidays, suggesting possible weather-related disruptions.



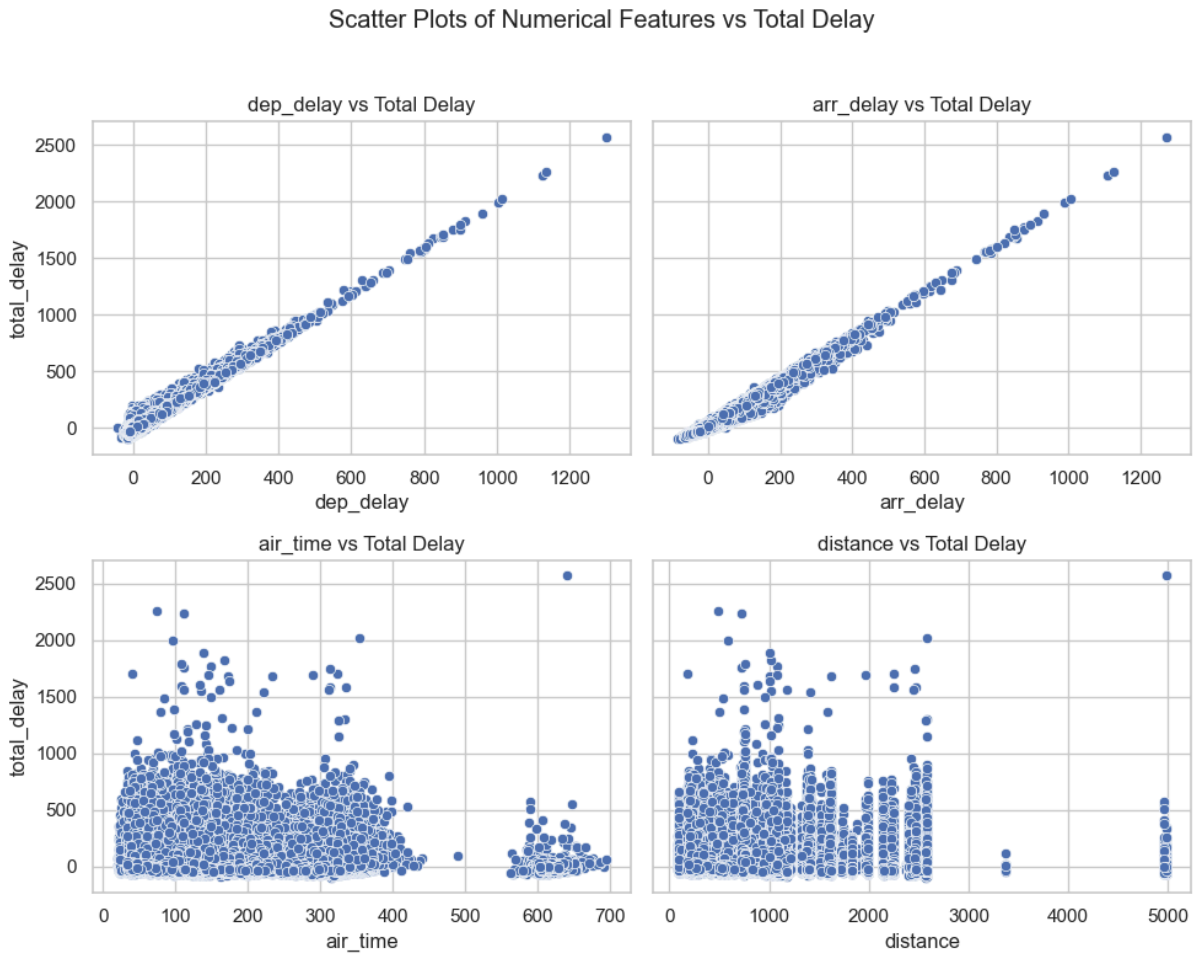

**Carrier Performance And Delays:**

- Carrier specific Delays: some carriers have consistently higher average delays, we believe this could be an indication of operational inefficiencies.
- AS & HA Carriers: interestingly, these have negative mean arrival delays, this suggests they may be rescheduling flights to reduce and minimize departure delays.




**Peak Delay Hours:**

- **Peak Delay** (10:00-19:00): Flights scheduled during these times experience the highest delays, possibly due to air traffic congestion.
- **Low Delay Hours** (01:00 06:00): Early morning flights seem to have the lowest delays, suggesting minimal congestion within these hours
- **Delay Trend Throughout the Day**: Flight delays seem to increase as the day progresses, peaking around 19:00 - 20:00, then declines after peak hours

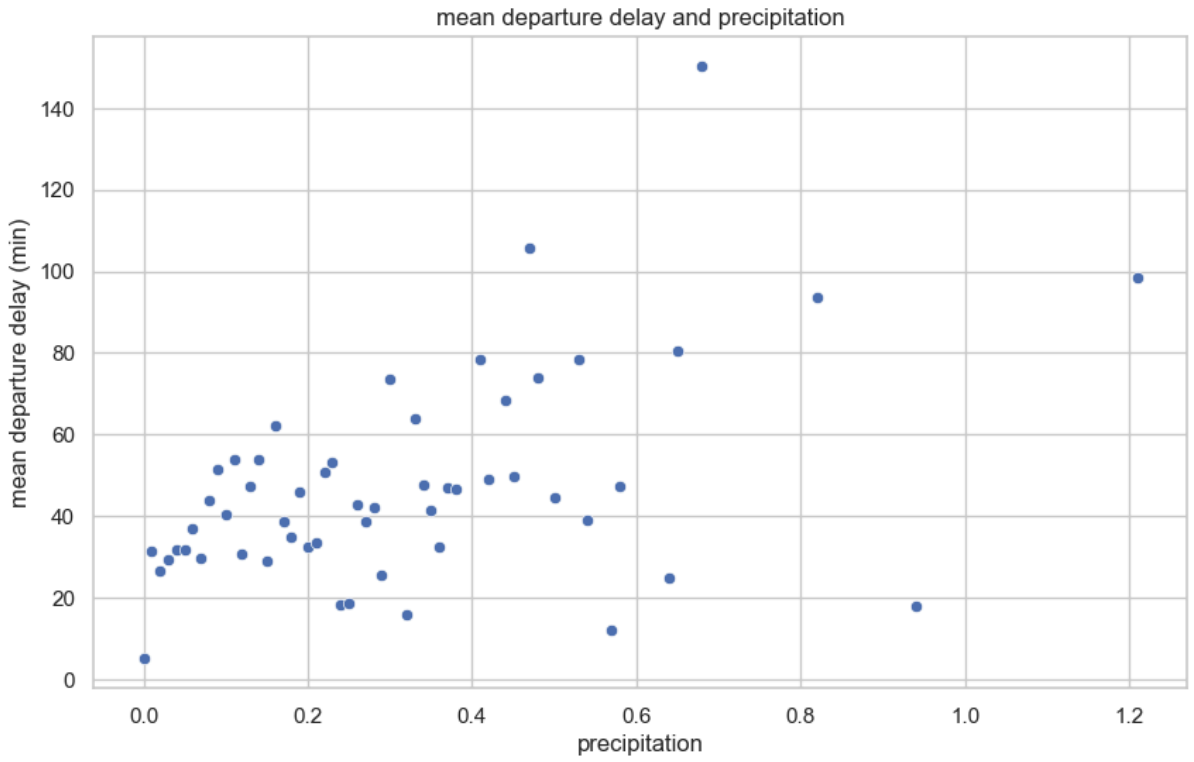



**Correlation Analysis**:

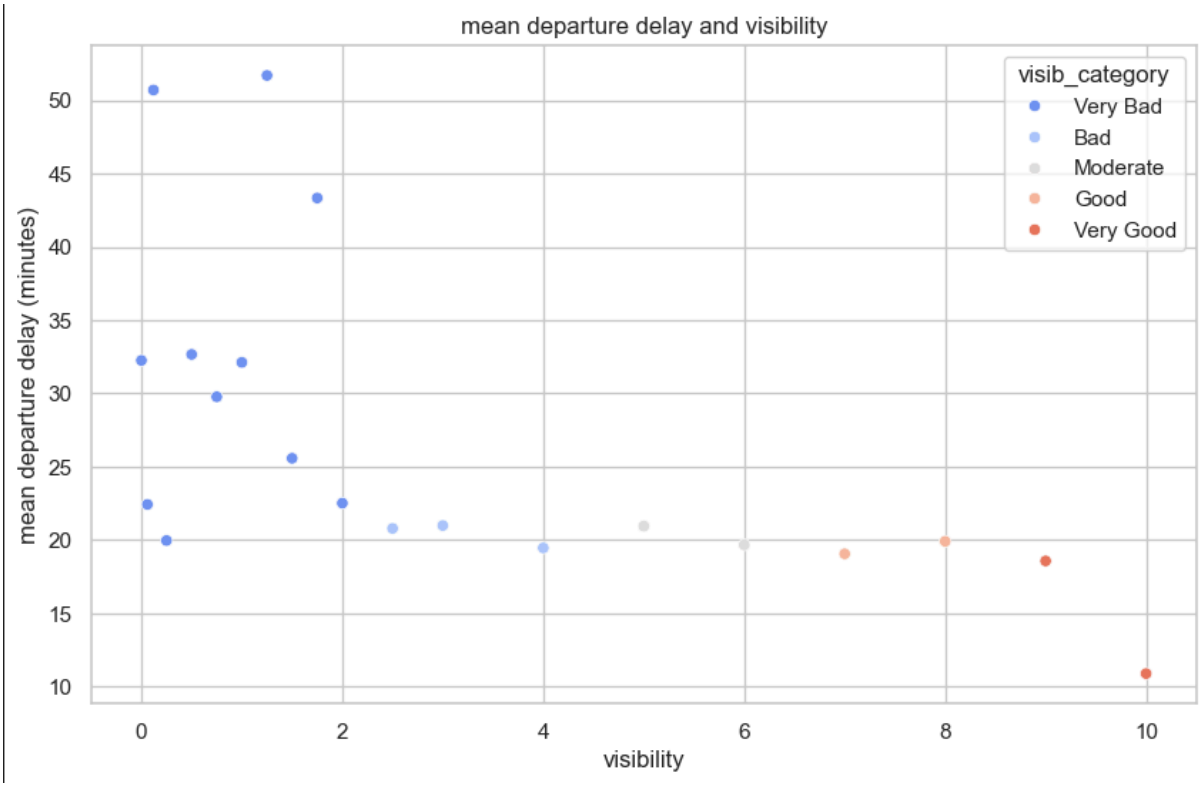
- **Strong Positive Correlations**
  - Departure Delay, Arrival Delay and Total Delay: These features are highly correlated, However, since future flights won't have these values beforehand, they are excluded from the predictive model from training, and will cause overfitting. These will be the values to predict instead.
- **Weak Correlation**

  - Air Time and Distance: These features show a weak correlation with delays, suggesting they have minimal direct impact on delays.

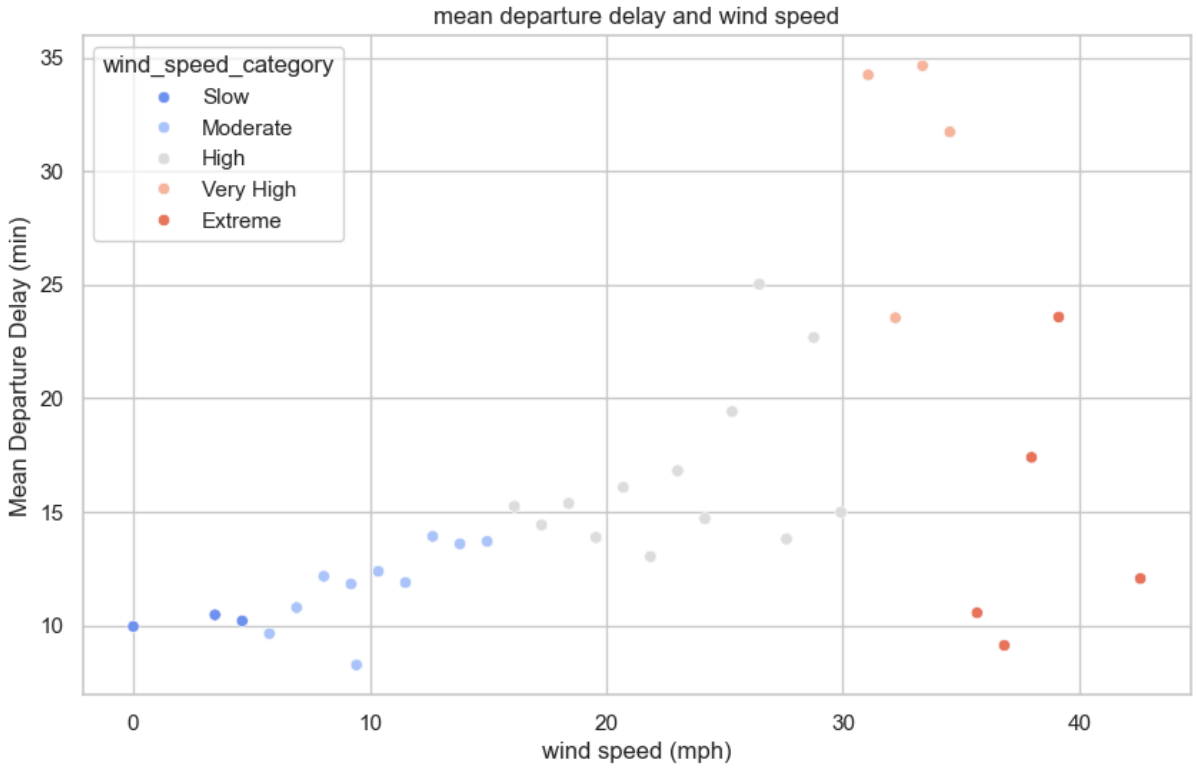
- **Weather Impact On Delays:**

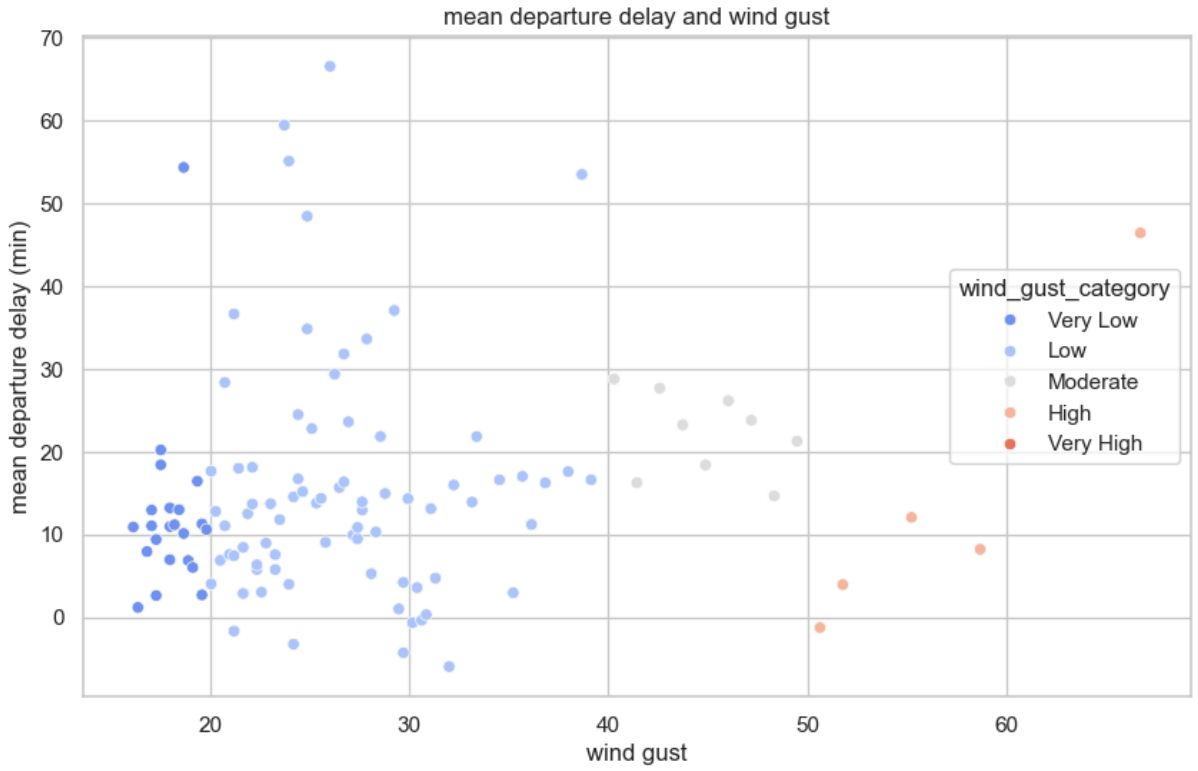



- **Precipitation:** Higher precipitation is linked to increased delay times, which indicates a strong relationship. Precipitation may impact flight stability visibility, and require the plane for maintenance therefore impacting delays

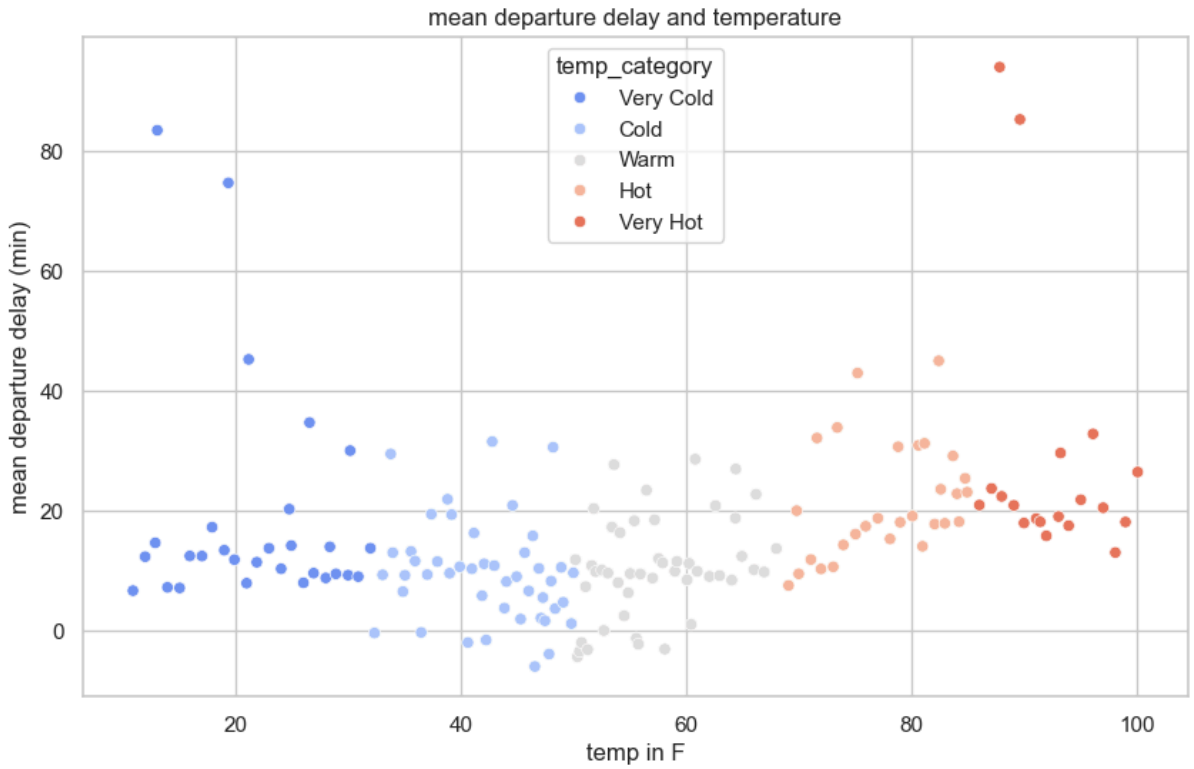



- **Visibility**: Low visibility values (0-2) strongly contribute to delays, indicating pilots have a hard time seeing therefore causing higher delays.

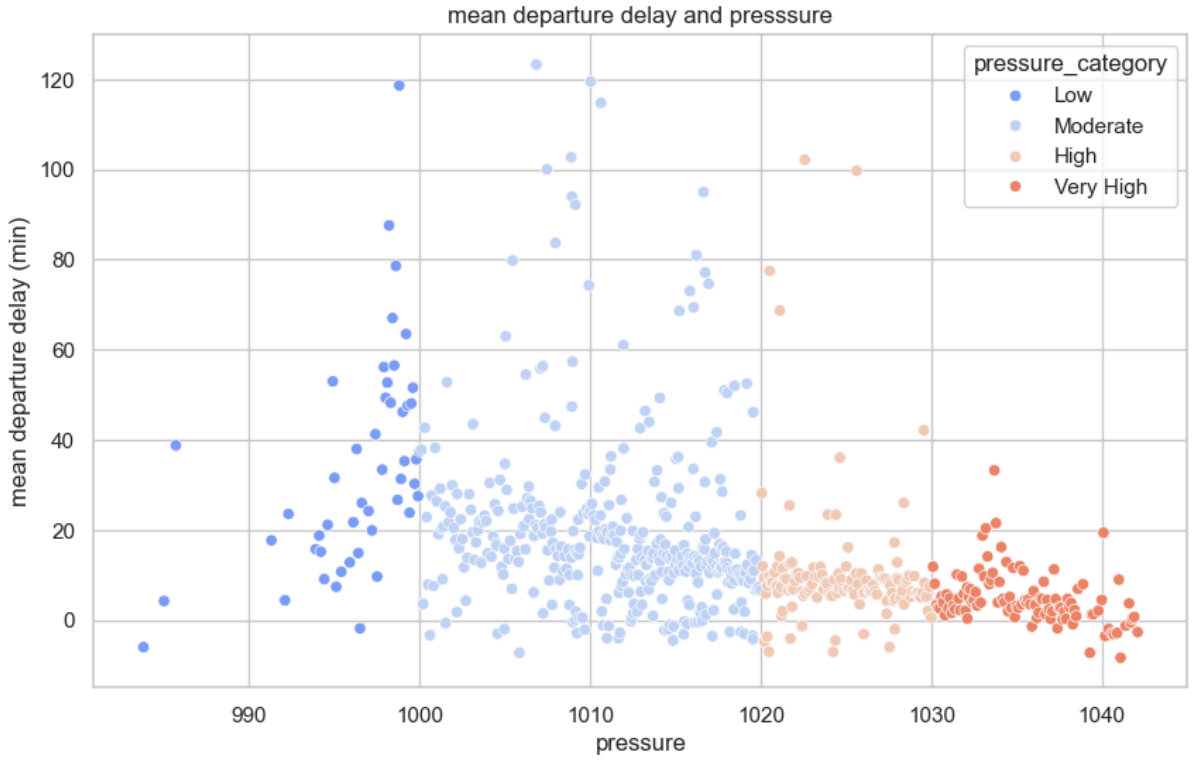



- **Wind Speed & Wind Gust:**
    - Higher wind speeds generally indicate higher delays. Wind speed can impact flight stability resulting in delays. However for extreme wind speeds delay seems to decrease. This could be due to other factors such as certain planes being able to withstand the high wind speeds, or flights being rescheduled to prevent delays

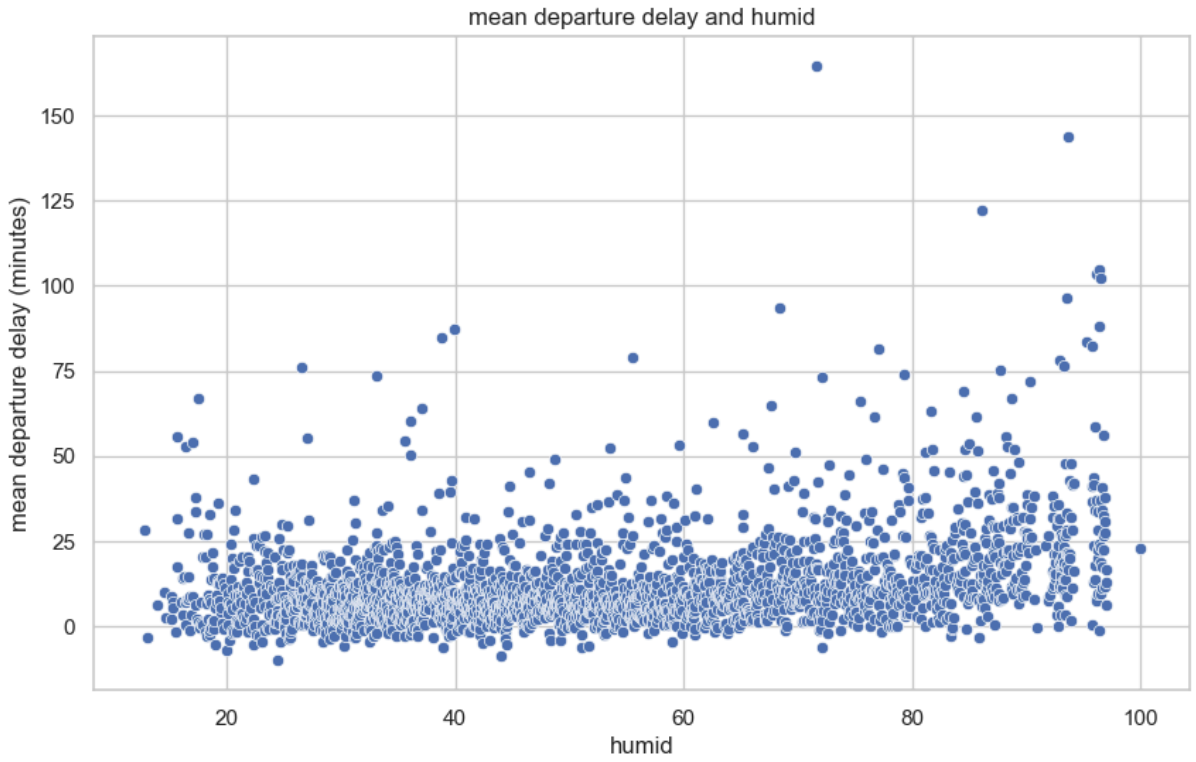



- **Temperature Extremes:**
  - Interestingly both very hot and very cold temperatures correlated with increased delays, suggesting that temperatures impact flight schedules.

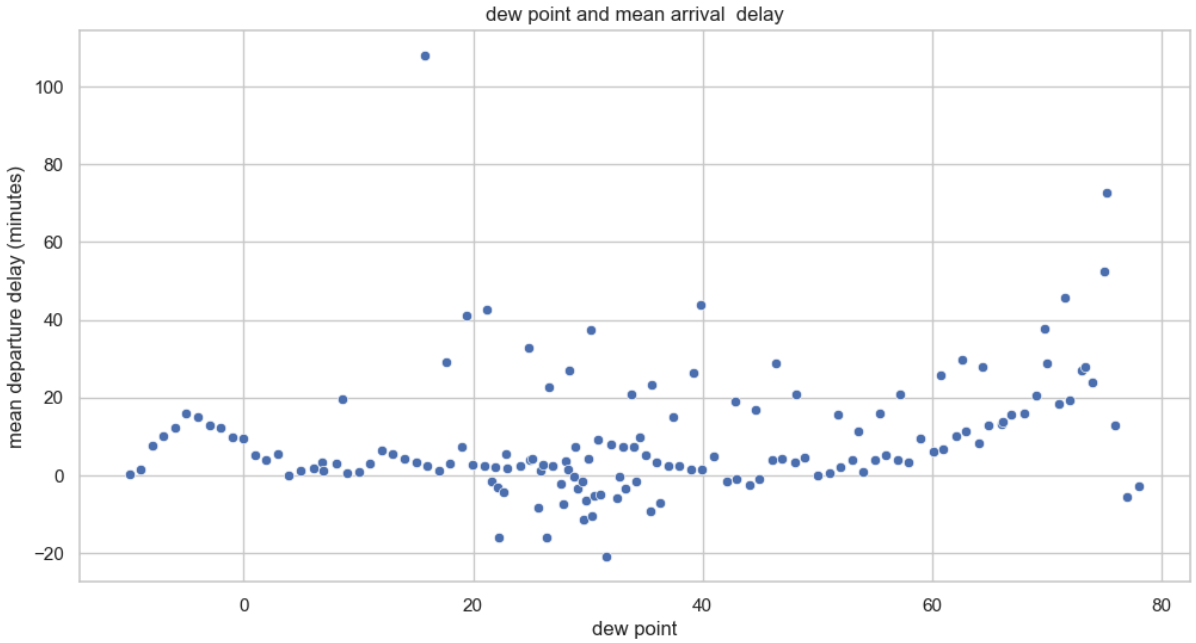



- **Pressure:** Lower pressure values seem to cause higher delays. Lower pressure may result in harsh conditions for engine parts, resulting in a delay. There are

a few outliers with large delays, but this is most likely caused due to other reasons such as congestion, weather factors, etc.




- **Humidity:** There does not seem to be a pattern between delay time and humidity values




- **Dewpoint:** There does not seem to be a pattern between delay time and dewpoint values

These insights from our EDA will help refine feature selection and ensure better accuracy in forecasting flight delays by providing insight during the predictive modelling process.

### 4.3 Model Selection

To accurately predict flight delays, various machine learning models will be explored. The selected models include:

1. **Linear Regression:** A baseline model used to provide interoperability and will help assess the dataset's linearity.
2. **Random Forest Regressor:** A powerful ensemble learning method that can handle nonlinear relationships and provide feature importance insights.
3. **XGBoost Regressor:** A gradient boosting algorithm optimized for structured data, known for its high predictive accuracy.
4. **Support Vector Regressor (SVR):** A model effective in high-dimensional spaces, useful for distinguishing between delayed and on-time flights.

These models will be compared based on metrics such as Mean Absolute Error(MAE), Root Mean Squared Error (RMSE) and $R^2$ score. The best-performing model is then selected for further tuning and optimization.

### 4.4 Model Training and Evaluation

Once the models were selected, they were trained using the historical flight data. The dataset was split into training (70%) and testing (30%) subsets to ensure robust model performance.

Several models were trained and evaluated including:

1. **Linear Support Vector Machine**:
    - A LinearSVR model is used to predict the continuous flight delay values. It was trained with StandardScaler to standardize the data and ensure correct handling of the features. Regularization parameters, like C=1.0 and epsilon =0.1, were used to ensure efficient convergence
    - The Root Mean Squared Error (RMSE) was calculated to evaluate prediction accuracy
2. **Support Vector Machine(SVM) Radial Basis Function(RBF)**:

- The SVR with the RBF kernel is used for handling more complex, non-linear relationships in the flight data. The model uses gamma='scale', C=10.0 and epsilon=0.1 as it allows the model to better fit the underlying data and stop overfitting.

- The Root Mean Squared Error (RMSE) was calculated to evaluate prediction accuracy

3. **Random Forest Regressor:**
   Random Forest ensemble method was used for regression. This model is trained with 100 estimators, ensuring that it can capture complex patterns in the flight data while avoiding the possibility of overfitting.
4. XGBoost Regressor:
   - XGBoost was trained with hyperparameters such as learning_rate=0.1, max_depth=5 and n_estimators=200. This model was also optimized using grid search for tuning hyperparameters to ensure the best performance.
   - The Poisson regression version of XGBoost is also used when modeling delay counts, using the count:poisson as the objective function, as well as an offset being applied to handle exposure values.

Each model was evaluated using the following metrics:

1. RMSE(Root Mean Squared Error): RMSE was the primary metric used to evaluate the regression models, It measured the differences between predicted and actual values. This allowed us to compare scores with lower values indicating better performance.

2. Cross-Validation
   - 5-Fold cross validation was applied to evaluate the models in terms of RMSE, this ensured that the models performed consistently across various subsets of the data, preventing overfitting.
3. Poisson Regression
   - For the models using Poisson regression, they were evaluated based on mean and standard deviation of the RMSE across the folds. We then applied an offset to adjust for exposure variables.
4. Evaluation of Best Model:
   - The best performing model will be determined based on the lowest RMSE score.

The models were trained and their performances were compared to identify the most accurate and robust model for predicting flight delays.

### 4.5 Probabilistic Analysis

To enhance the reliability of flight delay predictions, we attempted to use probabilistic modeling techniques to estimate the likelihood of a delay.

1. Poisson Regression with XGBoost:
   - We implemented a poisson regression variant of XGBoost to model delay occurrences as we assumed the flight delays follow a count-based distribution, however flight delays were in minutes.

Due to delay times (represented in minutes) not being a count value that represents an event, poisson couldn't be used. Also, our delay times contain negative values due to flights leaving and arriving before schedule. Poisson cannot handle negative values. Therefore Poisson did not work.

### 4.6 Optimization Strategies

To enhance the accuracy and reliability of our models, we applied several optimization strategies based on the characteristics of our dataset and modeling approach:

**Hyperparameter Tuning:**

- **Grid Search Optimization:** We used Grid search for fine-tuning the hyperparameters of our models to improve predictive performance
- **Poisson Regression With Exposure**: We tried to use flight distance as an exposure term, ensuring that longer flights, which usually have a higher risk of being delayed, are properly accounted for in the model, but this did not work.

**Performance and Computational Optimization:**

- Parallel Processing: Optimized data processing and model training speed to account for large data set to train models
- Cross Validation: tested each model created on 5 folds to ensure model accuracy and avoid overfitting.

## 5. Deliverables

The project will result in multiple tangible outcomes, contributing to both academic research on flight delay prediction and potential practical applications in airline scheduling and operations. The primary deliverables include:

1. **Codebase –** A well-documented repository containing implementations of machine learning models, sequential pattern analysis, data preprocessing scripts, and feature engineering techniques. The code will be structured for clarity and reproducibility.
2. **Trained Models –** The trained predictive models, including optimized configurations, will be saved as artifacts. These models can be used to analyze flight delay trends and assist in decision-making for scheduling improvements.
3. **Technical Report –** A comprehensive report detailing the methodology, data analysis, model selection, training and evaluation results, probabilistic analysis, and optimization strategies. It will include visualizations and insights derived from the study.
4. **Presentation Slides –** A professional slide deck summarizing the project's objectives, methods, findings, and implications. This will be used for academic or industry presentations.

5. **Documentation –** A structured guide explaining the project workflow, dependencies, execution steps, and insights into model interpretability. This will facilitate further research and potential real-world application.

These deliverables ensure that our findings can be replicated, extended, and potentially applied to improve flight delay management strategies in the aviation industry.

## 6. Experimental Setup

The successful execution of this project relies on access to the following resources:

- **Hardware**: A personal computer or cloud-based computing environment with sufficient computational resources for training and evaluating machine learning models. A system with a multicore CPU and a dedicated GPU (optional but beneficial) for accelerating training processes will also be helpful.

- **Software & Development Tools:**
    - Python (version 3.x)
    - Jupyter Notebook (.ipynb files) for interactive coding and documentation
    - VS Code with the Jupyter Notebook extension for local development
    - Google Colab for cloud-based model training and experimentation
    - Git & GitHub for version control and collaborative development

- **Machine Learning & Statistical Analysis:**
    - Scikit-learn for traditional machine learning models
    - Statsmodels for statistical analysis and probabilistic modeling
    - PyTorch or TensorFlow (if deep learning models are considered later on)

- **Data Processing & Visualization:**
    - Pandas and NumPy for data manipulation and preprocessing
    - Matplotlib and Seaborn for data visualization

- **Document Preparation & Reporting:**
    - LaTeX or Markdown for structured documentation
    - PowerPoint or Google Slides for presentation preparation

- **Data Access:**

    - Access to historical flight delay datasets, including features such as departure/arrival times, weather conditions, airline schedules, and external factors. Flights
    - Access to weather data during the duration of the flights WeatherAPI

- **Internet Access:** A reliable internet connection for literature review, online collaboration, accessing cloud-based tools, and retrieving external datasets.
-

All the mentioned resources are readily available, ensuring smooth execution of the project within an academic and research setting.

## 7. Impact

The proposed project has the potential to make remarkable contributions to both academic research as well as real-world applications in the following ways:

- Optimized Airline Scheduling and Reduced Delays: By identifying patterns and the root causes for delays, this project aims to contribute alternative scheduling allowing airlines to optimize their schedules, adjust departure times, and proactively manage congestion. Helping predict delays in advance helps minimize disruptions, leading to a more reliable flight network

- Data-Driven Decision-Making for Airlines and Airports: By Analyzing historical flight data and external factors(ex. Weather conditions), this project will be able to provide actionable insights to airlines and airports. These insights could be used for things such as fuel planning, crew scheduling, and overall enhancing operational efficiency

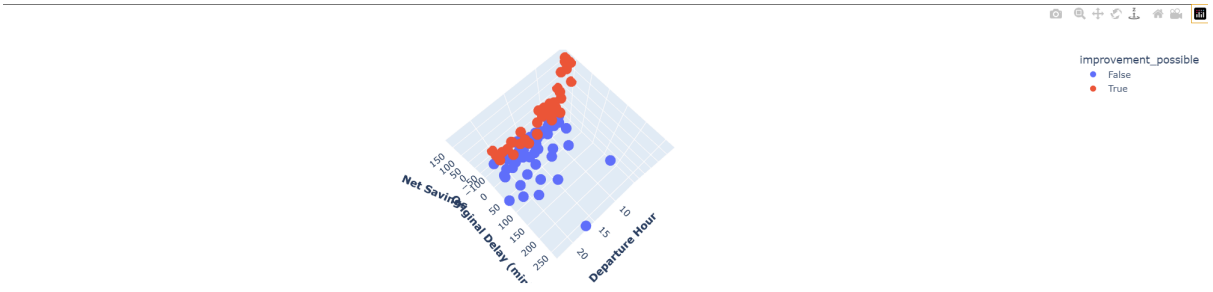
- Economic Impact and Cost savings for Airlines: Flight delays usually result in financial losses due to crew overtime, additional fuel costs, and passenger compensation. By predicting delays and optimizing schedules airlines would be able to reduce operational costs, improve profitability, and reduce resource utilization

- Adaptability to other Transport Sectors: The framework developed within this project is not only limited to aviation. The same methodology can be applied or adapted to optimize things like: train schedules, public transportation schedules, making it relevant to a wide range of transportation challenges.

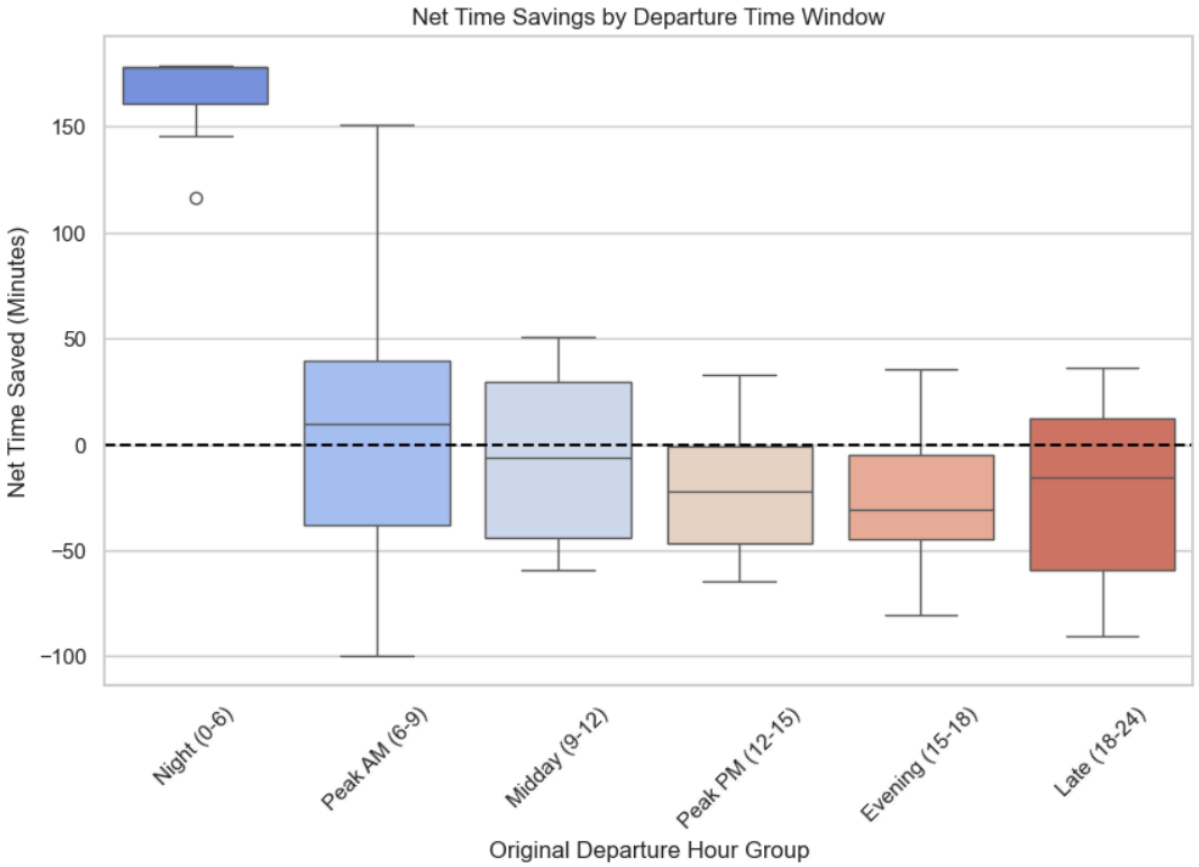
The impact of this project goes beyond theoretical research driving practical improvements in transportation planning and management.

## 8. Results

In this extensive analysis, we developed a machine-learning pipeline to optimize flight departure times and reduce total delays. We started with sophisticated feature engineering, where cyclical encoding was applied to temporal features like the day of the week and month to capture their period nature more effectively. These features were transformed into sine and cosine components. In Addition, categorical variables, including carrier, origin, destination and tail number, were processed using one-hot encoding, which resulted in a high dimensional feature space with 4,180 columns. This preprocessing set the stage for a model that could handle a variety of factors contributing to flight delays.




Our best-performing model was an XGBoost regressor, which we carefully tuned using optimal hyperparameters such as a learning rate of 0.1, a max depth of 5, and 200 estimators. Our model achieved strong predictive performance for flight delays, providing valuable insights into how departure time could be adjusted to reduce delays. The core of our analysis focused on departure time optimization. Initially, we developed a basic approach that tested shifting departure times earlier in 1-hour increments. However, this method revealed unexpected patterns, where earlier departures unbelievably sometimes increased the predicted delays. This caused us to change our approach to incorporate a more sophisticated net savings calculation, which considered both departure time change and delay reduction. The formula for this was: Net Savings = (Hours Shifted Earlier x 60) - (New Delay - Original Delay).




Through iterative testing, we discovered that shifting 6 AM flights to 5 AM produced the most significant improvements, with some flights showing net time savings of over 170 minutes. Other time slots showed either minimal benefits or adverse outcomes, which led us to further refine our approach. The final optimization model included smart constraints, blocking attempts to shift evening flights (after 2 PM) due to consistently poor performance when shifts were made. Only 1-hour shifts for the promising 6 AM time slot were allowed. This approach ensured that we were making the most effective adjustments to departure times.

In Addition to the optimization model, we developed a batch optimization analyzer to test the approach on hundreds of flights. This was complemented by visual tools that helped identify patterns in the results. Throughout the process, we traced various evaluation metrics, monitoring predicted improvements and, where available, actual operation outcomes. These metrics allowed us to identify key areas where the model could improve further, particularly in representing early morning flight characteristics and incorporating airport-specific constraints.

## 9. Conclusion and Discussion:

In Conclusion, our analysis demonstrates that strategic flight rescheduling can reduce delays, but this approach's success depends heavily on time-of-day dynamics and airport-specific conditions. The results of our study show that shifting 6 AM flights to 5 AM holds significant promise, likely due to factors such as quieter airspace, smooth aircraft turnaround, and more favourable weather conditions. However, this does not work universally, as most flights, particularly those in the afternoon and evening, face worse delays when moved earlier. This is mainly due to increased air congestion and operational constraints affecting later time slots. These findings emphasize that a one-size-fits-all approach to rescheduling is not effective. Instead, airlines should adopt a data-driven, time-segmented strategy. This would consist of prioritizing adjustments in the early morning hours, such as the Successful 6 AM to 5 AM shift while maintaining or even delaying later flights when necessary. Future efforts in optimizing flight schedules should focus on improving predictions for early-hour flights and incorporate real-time airport congestion data to enhance decision-making. The key to minimizing delays and improving efficiency in air travel lies in targeted, evidence-based rescheduling rather than arbitrary changes to flight times.

## 10. Limitations & Future Work

Despite the encouraging insights gained from our analysis, several factors must be acknowledged. First, while our optimization approach successfully reduced delays for flights departing at 6 AM by shifting them to 5 AM, the same strategy did not work uniformly across all time slots. Shifting later flights earlier often resulted in increased delays rather than reductions. This suggests that the effectiveness of rescheduling is highly dependent on multiple factors, such as time-of-day dynamics, with early morning flights benefiting from less air traffic and smoother operations, while afternoon and evening flights experience more significant operational constraints and congestion that nullify any potential gains from earlier departure times. Furthermore, it does not consider the condition of planes, so as a result rescheduling wont do anything.

Another limitation lies in the predictive modelling process itself. While the XGBoost model that we built worked, there is still further tuning that needs to be done to handle complex relationships better, especially in the case of morning flights and airport-specific delay patterns. The model also fails to include real-time levels of congestion, which would make the model more effective at recommending best departure time optimized more accurately. Additionally, when we were simulating the impact of weather-related variables, other external variables such as airline regulations, fuel constraints, airplane status, and economic factors were not simulated but also played a large role in delay behavior.

In subsequent work, we think the incorporation of real-time air traffic and weather data would significantly improve the accuracy and relevance of our predictions. In addition, another critical region of potential for improvement is integrating airline operational

constraints, such as maintenance requirements and crew scheduling, to ensure the suggested schedule changes are viable and beneficial. Implementing these improvements, our system could become an end-to-end decision-making framework for airlines to reduce delays preventively and optimize airline flight efficiency in general.

# References


1. *Columntransformer*. (n.d.). Scikit-Learn. Retrieved April 2, 2025, from https://scikit-learn/stable/modules/generated/sklearn.compose.ColumnTransformer.html
2. *Csr_matrix—Scipy v1. 15. 2 manual*. (n.d.). Retrieved April 2, 2025, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.csr_matrix.html
3. Czerny, A. I., & Zhang, H. (2020). Rivalry between airport ancillary and city-center supplies. *Transportation Research Part E: Logistics and Transportation Review*, *141*, 101987. https://doi.org/10.1016/j.tre.2020.101987
4. *Example gallery—Seaborn 0.13.2 documentation*. (n.d.). Retrieved April 2, 2025, from https://seaborn.pydata.org/examples/index.html
5. *Flights*. (n.d.). Retrieved April 2, 2025, from https://www.kaggle.com/datasets/monareyhanii/flights
6. G, R., Vijaya, K., Sadesh, S., M, A. P., V, M. P., & Kumar, M. S. (2024). Predicting flight delays and error calculation using machine learning classifiers. *2024 5th International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 1238–1244. https://doi.org/10.1109/ICESC60852.2024.10690024
7. Hatıpoğlu, I., & Tosun, Ö. (2024). Predictive modeling of flight delays at an airport using machine learning methods. *Applied Sciences*, *14*(13), 5472. https://doi.org/10.3390/app14135472
8. *Nycflights13/data-raw at main · tidyverse/nycflights13*. (n.d.). GitHub. Retrieved April 2, 2025, from https://github.com/tidyverse/nycflights13/tree/main/data-raw
9. *Plotly*. (n.d.). Retrieved April 2, 2025, from https://plotly.com/python/
10. *Weather api—Openweathermap*. (n.d.). Retrieved April 2, 2025, from https://openweathermap.org/api

11. *XGBoost Parameters—Xgboost 3.1.0-dev documentation*. (n.d.). Retrieved April 2, 2025, from https://xgboost.readthedocs.io/en/latest/parameter.html