Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 10, 2025

PREDICTING FLIGHT DELAYS USING ML: A HYBRID APPROACH FOR RESOURCE-CONSTRAINED SETTINGS

Sirat Fatima[*1], **Abia Maryam**[*2]

[*1,2]*A-levels Students*

[*1]siratf2007@gmail.com, [*2]abiamaryam20@gmail.com

DOI:https://doi.org/10.5281/zenodo.17310950

Keywords
Machine learning; Flight delay; Classification; Regression; Hybrid modeling; Operational data; Low-resource settings; Aviation prediction; Pakistan aviation

Article History
Received: 15 July 2025
Accepted: 25 September 2025
Published: 10 October 2025

Corresponding Author: *
Sirat Fatima
Abia Maryam

Abstract
Flight delays pose serious operational and economic challenges to airlines and passengers worldwide. In low-resource settings such as Gilgit-Baltistan and Skardu, delays on domestic routes to remote areas are even more critical due to unpredictable weather, challenging terrain, and limited infrastructure. These regions often lack large-scale data systems and advanced computational resources, making accurate prediction especially difficult. This study explores a practical, low-resource approach to predicting flight delays using only basic operational data. Both classification and regression models were evaluated, and results showed that classification methods, particularly tree-based ensemble models, performed better at identifying potential delays. To support real-world applications, a hybrid implementation pipeline is proposed: a classifier first predicts whether a flight is likely to be delayed, followed by a regressor that estimates the expected delay duration. A conceptual deployment plan is also outlined, suggesting how this system could be used to send early passenger notifications, assist ground staff in smaller airports, and dynamically adjust aircraft turnaround buffers. By focusing on regional constraints and practical deployment, this work offers a feasible and scalable solution for improving flight reliability, operational efficiency, and passenger communication in underdeveloped aviation networks. Future integration into national airline systems could further support policy efforts to enhance service quality in remote regions.

INTRODUCTION

Air transportation plays an essential role in the Pakistani economy, enabling the rapid movement of people and goods across long distances. Pakistan International Airlines (PIA) and other domestic carriers operate flights to remote regions such as Gilgit-Baltistan and Skardu, which are vital for tourism, and access to healthcare and other essential services. These areas serve as gateways to popular tourist destinations and are crucial to the country in terms of jobs and resources. In 2017, approximately 1.72 million tourists visited Gilgit-Baltistan, contributing around PKR 300 million to the local economy [1].

Despite its importance, the industry continues to struggle with one of its most persistent operational challenges: flight delays. Media reports and operational data indicate that delays are especially common on routes to northern areas, primarily due to unpredictable weather, difficult terrain, and limited ground infrastructure [8]. These regions often experience sudden fog, strong winds, and rapidly changing visibility conditions, making flights particularly vulnerable to last-minute disruptions. These disruptions have become increasingly complex to manage, particularly as air traffic grows and weather patterns become more unpredictable due to climate change. This has far-reaching effects; not only does it cause passenger dissatisfaction and inconvenience, but it also results in economic losses annually. Accurately predicting flight delays is, therefore, a task of growing importance for stakeholders across the aviation industry.

Globally, machine learning (ML) methods have shown promise in predicting flight delays by learning patterns from large datasets. Unlike traditional approaches, ML models can capture intricate relationships among multiple variables without requiring rigid assumptions. Over the past decade, various algorithms have been explored for this purpose, with ensemble methods such as Random Forest and XGBoost standing out due to their ability to handle high-dimensional data and model nonlinear relationships effectively [2].

Despite these advances, many ML approaches depend on high-quality data and advanced computational resources, which are conditions often lacking in Pakistan's regional aviation context. This challenge is particularly relevant for remote areas where delays are frequent, but reliable operational data and modern infrastructure remain limited.

This study aims to explore whether a low-resource, operationally feasible approach using only basic flight data can effectively predict delays. By evaluating both classification and regression models, the study develops a hybrid prediction framework: first, a classifier identifies flights at risk of delay, and then a regressor estimates the expected delay duration. A conceptual deployment plan is also proposed, outlining practical applications such as early passenger notifications, dynamic ground crew scheduling at smaller airports, and flexible turnaround buffer adjustments.

This research is motivated by the pressing need to improve the domestic aviation industry in Pakistan's underdeveloped regions. Accurate delay forecasts can help airlines better manage limited resources, reduce last-minute cancellations, and support tourism-dependent local economies. By focusing on a low-resource design, this approach remains accessible for smaller airports and domestic carriers that lack extensive data systems or advanced computing capabilities.

In summary, this research aims to advance predictive modeling of flight delays while emphasizing regional constraints and needs. The study highlights practical strengths and limitations and offers actionable insights to support more reliable air travel in Pakistan's challenging northern and mountainous regions.

3. Literature Review

In recent years, researchers have increasingly turned to machine learning (ML) techniques to model and predict delays using large-scale historical flight data. This literature review synthesizes findings from key recent studies relevant to our project and analyzes their methodologies, findings, and contributions to the broader discourse on delay prediction. Collectively, these works inform and validate our approach, which centers on a hybrid classification-regression framework.

3.1 Use of XGBoost in Delay Prediction

The study titled "Flight Delay Prediction Using Machine Learning Algorithm XGBoost" [3] evaluates the effectiveness of the XGBoost algorithm in classifying flight delays. The authors used a dataset structurally similar to ours—U.S. domestic flight records—and aimed to predict whether a flight would be delayed based on features such as departure time, airline carrier, and weather conditions.

A key finding was that XGBoost outperformed other algorithms due to its gradient boosting architecture, which iteratively reduces error by combining weak learners. It effectively handles sparse, noisy data and reduces both bias and variance, which are factors especially relevant in high-dimensional aviation datasets. The study highlighted the importance of preprocessing and feature engineering and used the standard 15-minute delay threshold for classification.

The authors further stressed XGBoost's strength in modeling complex nonlinear relationships, which guided our choice to include it in our model comparison phase.

3.2 Comparative Analysis of Traditional and Ensemble Models

The study by Khaksar and Sheikholeslami [9], titled *Airline delay prediction by machine learning algorithms* and published in *Scientia Iranica*, offers a detailed comparison between traditional models (e.g., logistic regression, decision trees) and ensemble approaches such as random forests and support vector machines. While traditional models offer interpretability and simplicity, the study concludes that ensemble methods consistently outperform them in predictive accuracy, especially when applied to large-scale, multi-dimensional datasets.

An important contribution of this paper is its discussion of computational limitations, which the researchers mitigated by downsampling their dataset. The study also tackles the issue of class imbalance—a persistent challenge in delay prediction, where on-time flights significantly outnumber delayed ones. Their application of balanced sampling influenced our strategy in ensuring reliable performance across skewed data.

3.3 Hybrid Approaches: Combining Classification and Regression

A recent study by Jha et al. [5] introduces a hybrid approach that incorporates both classification and regression modeling. The authors aimed to first predict whether a flight would be delayed (classification) and then determine the length of the delay in minutes (regression), using a dataset structurally similar to ours. Their data pipeline included key steps such as feature transformation, categorical encoding, and outlier removal, all of which are integral to this paper's preprocessing workflow. Notably, they concluded that no single model performs best in all situations, highlighting the value of using multiple models in parallel.

This two-stage framework mirrors the operational realities for airlines, where knowing whether a flight will be delayed and estimating its duration are both important for optimizing flight schedules and gate assignments. By combining the tasks of delay classification and duration prediction, the authors established a flexible model architecture that we implement and extend in this study.

3.4 Temporal and Spatial Features

Beyond these core studies, additional literature offers relevant theoretical frameworks that enrich the understanding of flight delays. An emerging trend is the incorporation of time-based and spatial features into model design. Variables such as time-of-day, day-of-week, month, and airport pairings help models identify patterns in flight operations, including peak hours and seasonal weather variation. Studies show that temporal binning (e.g., grouping hours into time blocks) and spatial encoding improve generalization across routes and seasons [10;11].

3.5 Key Insights for Our Research

From reviewing this literature, several insights have directly shaped our methodology:

- Feature Engineering is Crucial: All studies emphasized the value of domain-specific, well-engineered features. Our inclusion of time-based variables, route identifiers, and binary delay flags aligns with these best practices.
- Ensemble Models Perform Best: Both XGBoost and random forest algorithms have shown strong predictive capabilities in past research.
- Computational Efficiency Matters: Memory and processing constraints, especially in platforms like Google Colab, guided our decision to downsample the dataset while retaining statistical representativeness.
- Hybrid Modeling Reflects Operational Needs: Predicting both delay occurrence and delay magnitude provides a more complete picture of flight performance, aligning with real-world decision-making processes in airline management.

By aligning our work with these findings and applying a decade's worth of data (2009–2018), our study aims to reinforce and extend the current body of research with practical, scalable, and interpretable insights into airline delay prediction.

4. Data Description

This section provides a structured overview of the dataset used in this study, its origin, characteristics, and limitations. We aim to ensure transparency about the data quality and constraints under which modeling decisions were made.

4.1 Dataset Source and Collection

The dataset used in this research originates from the publicly available "Airline Delay and Cancellation Data (2009–2018)" hosted on Kaggle [6]. It was originally compiled by the U.S. Bureau of Transportation Statistics (BTS) and includes roughly 58.6 million records of domestic flights operated within the United States over ten years from January 2009 to December 2018.

U.S. domestic flight data was used to demonstrate the feasibility of low-resource predictive modeling due to the unavailability of local aviation datasets. While features such as scheduled departure hour, route, and distance are generally applicable and can be directly adapted to Pakistan's context, others, like operating carrier codes, reflect U.S.-specific carriers and would need to be replaced with local airline data for practical deployment. Future work would involve retraining models on Pakistan's domestic flight data to capture carrier-specific operational behaviors and regional delay patterns fully.

Due to memory constraints on platforms like Google Colab, the full dataset was not used. Instead, a stratified sample was created by selecting 30,000 records per year, ensuring diversity. The result was a combined dataset of 300,000 rows.

4.2 Data Scope and Structure

The dataset is structured such that each row represents a single domestic flight segment. Initially, the raw data comprised 28 columns covering flight times, delays, airline codes, and airport information.

Following cleaning and filtering, the number of retained columns was reduced to 18. Further feature engineering—including time decomposition, categorical encoding, and route generation—expanded the final dataset to 25 columns. Rows missing key variables were dropped, resulting in the final sample consisting of 269,428 rows and 25 columns.

Key variables, retained and derived, include:

- Flight Information: Flight date, carrier code, flight number
- Location Data: Origin and destination airport codes
- Timing Variables: Scheduled and actual departure/arrival times
- Delay Metrics: Departure and arrival delays (in minutes)
- Operational Attributes: Distance, scheduled elapsed time
- Engineered Features: Departure and arrival hour bins, day of week, route identifier, weekend indicator, delay classification label

All times are expressed in local airport time, and delays are measured in minutes, with negative values indicating early arrivals.

4.3 Initial Cleaning and Filtering

Several preprocessing steps were applied to ensure the dataset was suitable for supervised learning tasks:

- Records missing values in either ARR_DELAY or DEP_DELAY were dropped, reducing the dataset from 300,000 to 290,598 rows.
- Flights marked as canceled or diverted were removed, as they lack complete delay data.
- Redundant or irrelevant columns, such as taxi times, wheel timestamps, and aircraft ID, were excluded.
- Scheduled departure and arrival times were standardized into four-digit integer formats for feature extraction.

The resulting dataset was consistent, free from major gaps, and ready for supervised machine learning.

4.4 Target Variable Definitions

Two supervised learning targets were defined: a classification label based on a 15-minute delay threshold, and a regression target based on raw arrival delay in minutes. The classification label followed the Federal Aviation Administration's definition of a delayed flight [7].

4.5 Data Limitations and Constraints

Although the dataset is publicly available and covers a wide range of flight records, some limitations were encountered:

- Computational Constraints: The full dataset could not be processed on standard free-tier platforms like Google Colab. This led to downsampling, which might slightly limit generalization.
- Sparse Weather Data: An effort was made to add historical weather data using the Meteostat API. While the code functioned correctly, most weather values were missing, especially for earlier years and smaller airports. Due to this lack of coverage, weather features were excluded from the final dataset. Accessing more complete or paid sources was not feasible in a high school research setting.
- Real-Time Limitations: The dataset is based on past data and does not support live or real-time predictions. However, it remains valuable for learning patterns and building predictive models.

5. Feature Modelling

This section outlines the transformation of raw flight records into a structured form suitable for supervised machine learning tasks. The focus was on selecting relevant features, handling categorical and numerical variables, and preparing the dataset for effective model training.

5.1 Input Feature Selection

Feature selection was guided by the feature's relevance to the study's goals and data availability for that specific feature. From the original set of 28 columns, only those directly related to flight scheduling, aircraft identity, airport location, and delay timings were retained. The remaining 18 columns formed a refined dataset for feature engineering and encoding processes.

5.2 Time and Route-Based Feature Engineering

To improve the usefulness of the dataset for machine learning, several new variables were created from existing ones:

- Dep_Hour and Arr_Hour: Extracted from the scheduled departure and arrival times. These represent the hour of the day the flight was planned to depart or arrive, and help capture delay patterns at different times.
- DayOfWeek and Month: Derived from the flight date to represent weekly and seasonal trends.
- IsWeekend: A binary variable indicating whether the flight occurred on a weekend.
- Route: A new column combining origin and destination airport codes to represent unique flight paths.
- Distance_Bin: This feature was derived by categorizing the distance variable into ranges (e.g., short, medium, long) based on industry-standard flight length classifications. Whilst sources use three broader categories, this paper further divides them into six, allowing for better performance in model training.

After feature engineering, the dataset expanded from 18 to 25 columns.

5.3 Categorical Variable Encoding

Several of the input features in the dataset were categorical, such as route and distance_bin. Since most machine learning models require numerical input, these features were converted into a suitable format using label encoding. Although one-hot encoding is often more effective in representing categorical variables as it doesn't impose ordinal relationships, it was avoided in this study. This is because it significantly increased the dataset's dimensionality, and due to memory constraints, it frequently resulted in system crashes.

5.4 Target Variables

Two prediction tasks were defined:

- Classification: Flights were labeled as "delayed" if arrival delay exceeded 15 minutes, based on the FAA definition. A new binary column, IS_LATE, was added for this.
- Regression: The exact arrival delay in minutes was used as a continuous target.

Columns containing information generated after departure were deliberately excluded to prevent data leakage and ensure realistic pre-departure predictions. This reduced the column count from 25 to 15. The class distribution for the classification task was imbalanced, with approximately 75% of flights being on time or early. This imbalance was taken into account during model evaluation.

6. Modelling Approach

6.2 Regression Model

The regression task in this study involved predicting the number of minutes by which a flight would be delayed upon arrival. Early arrivals, represented by negative values, were replaced with zero to simplify the task and focus only on actual delays.

Care was taken to remove any columns that could lead to data leakage. These included actual departure and arrival times, delay durations, and other outcome-based features. Rarely occurring flight routes were grouped under a common label to reduce memory usage.

After encoding, any missing values caused by inconsistent feature columns between the training and test sets were filled with zeros. This approach helped preserve the full dataset without discarding valuable samples. Standard scaling was then applied to all numerical features to ensure they were on the same scale for models that are sensitive to the features' magnitudes.

Four regression models were implemented to compare performance. Linear Regression was used as a baseline to evaluate how well a simple model could perform. Random Forest Regressor was included for its ability to model non-linear patterns and interactions across features. XGBoost Regressor was selected due to its speed and strong performance on structured data, and replaced the originally planned gradient boosting implementation. Finally, a Neural Network Regressor was added to explore whether a deep learning model could capture hidden patterns in the data, particularly after scaling.

Models were evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2).

6.3 Classification Model

In addition to estimating delay duration, a secondary classification task was also conducted to determine whether a flight would be delayed by more than 15 minutes. A new binary column called IS_LATE was created for this purpose, where 1 was assigned to indicate a delay and 0 otherwise.

Modeling was conducted using the cleaned and engineered dataset described in earlier sections.

One of the main challenges was that the number of delayed flights was much smaller than the number of on-time flights. This imbalance made it harder for the models to learn how to detect delays. To fix this, the training data was balanced using random oversampling, which duplicated delay examples so that both classes were represented equally. The split between training and test sets was done in a way that kept the proportion of delayed and on-time flights consistent.

Three classification models were evaluated: Logistic Regression, Random Forest Classifier, and Gradient Boosting Classifier. Logistic Regression was used as a simple and interpretable baseline. Random Forest Classifier, an ensemble of decision trees, was chosen for its ability to handle large datasets and find relations between variables. Gradient Boosting Classifier was included for its strong performance on structured datasets. All models were trained using standard settings from the scikit-learn library.

Model performance was evaluated using Accuracy, Precision, Recall, F1-Score, and ROC AUC. These results, along with a comparative analysis between the different models, are presented in the next section.

7. Results and Evaluation

This section presents the observed performance of different models for flight delay prediction, structured into classification and regression tasks. To assess the performance of our machine learning models for flight delay prediction, we employed a set of standard evaluation metrics tailored to each task type. For classification, we used Accuracy, Precision, Recall, F1-score, and ROC AUC. For regression models, we reported Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) to measure predictive precision and model fit.

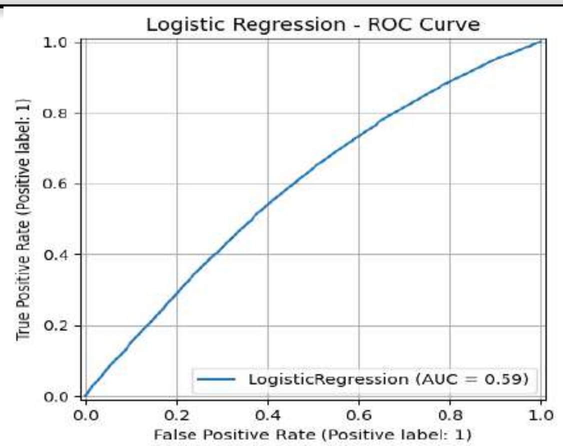
7.1 Baseline Model Performance

Classification: Logistic Regression

The logistic regression model served as a baseline to classify whether a flight would be late (>15 minutes delay). Throughout the results, 'class 1' represents the delayed class.

Table 7.2.1 Logistic Regression Performance

Metric	Accuracy	Precision	Recall	F1-Score	ROC AUC
Test Set Score	0.57	0.30 (class 1)	0.58 (class 1)	0.40 (class 1)	0.59



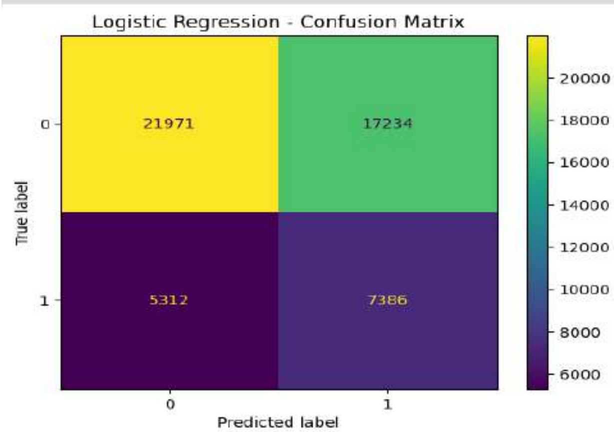


Figure 7.1.1: Confusion matrix for the Logistic Regression classifier. The matrix shows the number of correct and incorrect predictions, highlighting challenges in identifying delayed flights despite oversampling.

Figure 7.1.2: Receiver Operating Characteristic (ROC) curve for the Logistic Regression classifier. The curve illustrates the trade-off between true positive rate and false positive rate across different thresholds, with an AUC of 0.59 indicating moderate discrimination ability.

Regression: Linear Regression
The Linear Regression model was trained on clipped delay values (ARR_DELAY ≥ 0). Despite preprocessing, performance remained limited.

- R^2: 0.013
- train R^2: 0.012
- MAE: 21.0
- RMSE: 37.5

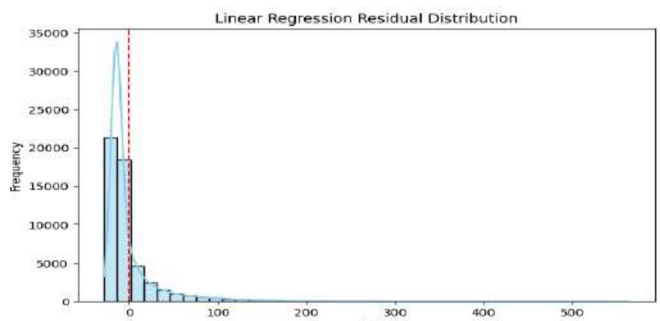
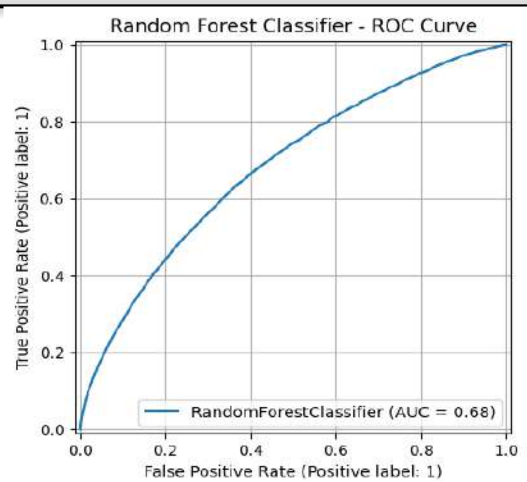


Figure 7.1.3: Residual distribution for the Linear Regression model. Most residuals are concentrated near zero, suggesting reasonable accuracy on small delays. The positive skew indicates frequent underestimation of larger delays, highlighting the model's limitations in capturing extreme cases.

7.2 Advanced Model Results
Classification: Ensemble Models
Tree-based ensemble classifiers were trained on the same dataset used for the Logistic Regression baseline.

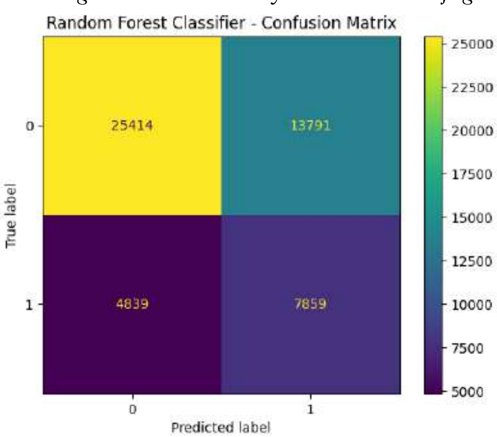
Table 7.3.1 — Ensemble Classifier Performance

Model	Accuracy	Precision (Class 1)	Recall (Class 1)	F1-Score (Class 1)	ROC AUC
Random Forest Classifier	0.64	0.36	0.62	0.46	0.68
Gradient Boosting Classifier	0.64	0.36	0.63	0.46	0.69



Both Gradient Boosting and Random Forest classifiers yielded comparable performance, with test accuracies of 0.64 and F1-scores of 0.46. Due to their similarity, only Gradient Boosting metrics are visualized.

Figure 7.2.1: ROC curve for the Random Forest classifier. The curve shows an AUC of 0.68, indicating moderate ability to distinguish between delayed and on-time flights.



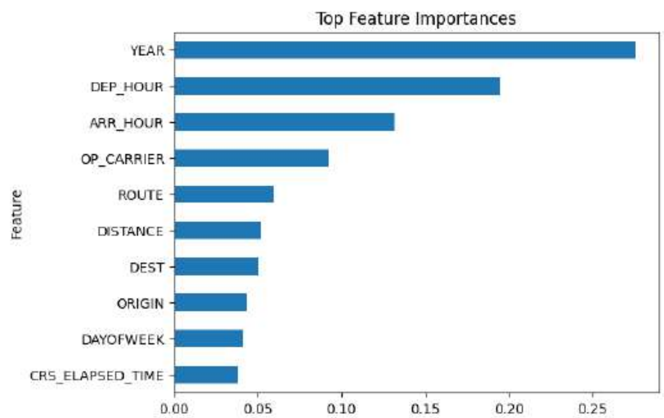


Figure 7.2.2: Top feature importances for the Random Forest classifier. Year, departure hour, and arrival hour were the most influential variables, reflecting the importance of temporal patterns.

Figure 7.2.3: Confusion matrix for the Random Forest classifier. The model performs better at predicting on-time flights, with a notable number of delayed flights misclassified, highlighting challenges in handling class imbalance.

Regression

Ensemble and neural regression models were applied to predict ARR_DELAY and evaluated against the baseline. Ensemble models like Random Forest and XGBoost slightly outperformed the linear regression baseline, achieving marginally better R^2 scores and slightly lower RMSE and MAE values. The Neural Network Regressor, however, performed poorly. It had the lowest R^2 score and a very high MAE, suggesting it was not able to make stable or accurate predictions.

Table 7.3.2 — Regressor Performance

Model	Train R^2	Test R^2	RMSE (Test)	MAE (Test)
Random Forest Regressor	0.153	0.102	35.79	19.38
XGBoost Regressor	0.155	0.115	35.53	19.2
Neural Network	0.037	0.036	20.67	37.1

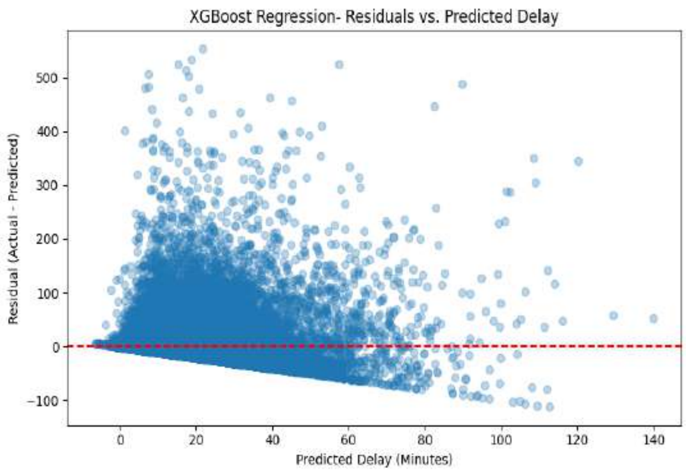


Figure 7.3.1: Residuals versus predicted delays for XGBoost Regressor. The plot shows increasing spread with higher predicted delays, indicating reduced accuracy for larger delay estimates.

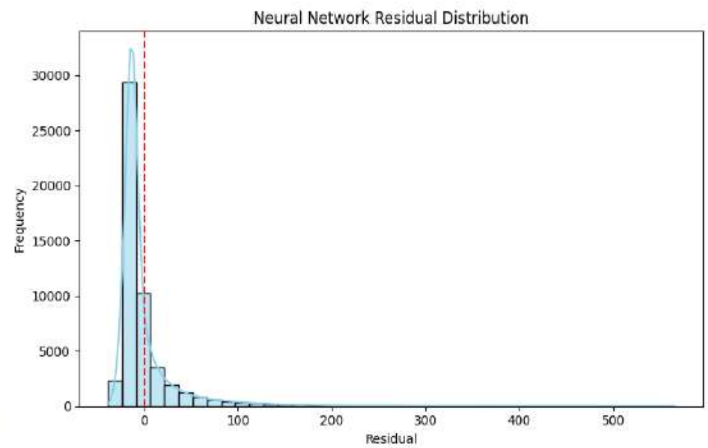


Figure 7.3.2: Residual distribution for the Neural Network Regressor. Most residuals are tightly clustered near zero; the right-skew indicates tendencies to underestimate larger delays.

Across both tasks, tree-based ensemble models offered the most balanced results, although overall accuracy remained limited. These comparisons help highlight the practical strengths and weaknesses of each approach and prepare for a deeper analysis in the following Discussion section.

8. Discussion

8.1 Interpretation of Results

The results of this study show that classification models performed more reliably than regression models when predicting flight delays using operational flight data alone. Among all classification models, ensemble methods such as Gradient Boosting and Random Forest achieved the best performance in terms of accuracy, F1-score, and ROC AUC. These models were particularly effective at handling the class imbalance problem, especially when combined with oversampling. In contrast, the logistic regression baseline showed limited ability to correctly identify delayed flights, despite having high precision for the majority class.

Regression models struggled to predict delay durations with high accuracy. Although XGBoost Regressor showed the best performance among the regression models, the overall R^2 scores remained low. The Neural Network Regressor performed the worst, with low R^2 and unusually high MAE, likely due to poor generalization and a limited feature set. These findings suggest that predicting the presence of a delay is more feasible than estimating the delay's exact length using only basic flight-level features.

8.2 Practical Applications and Recommendations

The results suggest that classification models are better suited for real-time decision-making in airline operations, where speed and reliability are critical. These models can be used to build alert systems that notify passengers early about potential delays, allowing them to rebook flights or adjust travel plans in advance. For airport staff, such predictions can support rapid decisions about gate assignments and crew scheduling.

Although regression models were less accurate, they can still help estimate approximate buffer times or plan ground operations. For example, if a flight is predicted to be delayed by at least 30 minutes, ground staff can adjust turnaround schedules or reallocate resources to reduce further delays.

A combined system that uses both classification and regression could be particularly effective. A classifier could first determine whether a flight is at risk of delay, and if so, a regression model could then estimate how long the delay might last. Such a two-stage pipeline approach mirrors what some real-world airline management systems aim to implement for improved operational efficiency.

8.3 Model Behavior and Methodological Characteristics

Classification models, especially tree-based ones like Gradient Boosting and Random Forest, were more stable and easier to understand. This is partly because they predict simple binary outcomes, which reduces the impact of extreme or unusual values. During training, these models produced consistent results across multiple runs and were less likely to overfit. They also handled categorical variables well, and their performance improved further with oversampling to correct the class imbalance.

Regression models, on the other hand, were more sensitive to data variation. Because they try to predict exact delay durations, they were more affected by skewed values. Their results changed more depending on how tuning parameters like learning rate or model depth were set. The Neural Network Regressor was especially unstable. Even after normalizing the input data, its performance varied and showed signs of bias. This may be due to the limited number of input features and the difficulty of learning patterns from noisy data. Overall, regression models were more fragile and harder to work with, especially without access to richer information.

8.4 Limitations and Future Work

This study faced several limitations that affected model performance and generalisation. One key limitation was the absence of real-time and external features such as weather conditions and air traffic congestion. These variables often play a critical role in delay prediction, but were not available in the dataset used. While feature engineering was applied to the available data, such as converting time fields and encoding categorical variables, the overall feature space remained limited in scope and detail.

Another challenge was the limited access to computing power, which made it hard to fully tune and test more complex models. Running large grid searches or training deeper neural networks was not possible, so many models were used with basic settings. This likely affected how well they performed. For example, the neural network showed unstable results, which might have been improved with more tuning.

In the future, adding more types of data like weather reports, airport crowding levels, or aircraft history could improve the accuracy of the models. It may also be useful to include patterns over time, such as whether a previous flight by the same plane was delayed. Finally, combining classification and regression in one system and tuning each part more carefully could make the models more useful in real-world airline operations.

Table 8.1 summarizes the main strengths and limitations of each task type based on this study.

Aspect	Classification Models	Regression Models
Prediction Output	Binary (Delayed / On-Time)	Continuous (Delay in Minutes)
Practical Use	Alerts, decision triage, and resource allocation	Buffer planning, simulations, scheduling impact analysis
Accuracy with Given Data	Higher — better suited to limited features	Lower — struggles without contextual/external inputs
Interpretability	High	Moderate

Robustness to Noise	Higher — more stable with class imbalance and label noise	Lower — sensitive to outliers and distribution skew
Suitability for Real-Time	High — supports fast operational decision-making	Low — better for offline or planning use

9. Practical Implementations

Despite the growing global interest in machine learning-based delay prediction systems for air traffic management, their practical implementation in low-resource aviation environments remains largely underexplored. Most predictive models are optimized for high-tech ecosystems with access to real-time weather feeds and dedicated data science teams. However, this does not reflect the operational realities of many regional airports in Pakistan, where limited technical resources, basic infrastructure, and smaller staff sizes present significant barriers to adoption.

This study proposes a lightweight, hybrid ML architecture specifically designed for real-world integration in such settings. The following components outline the proposed implementation pathway.

9.1. Low-Cost Two-Stage Predictive System

At the core of the system lies a two-step machine learning pipeline designed for minimal computational load and feature requirements:

Step 1: Delay Classification**:** A binary classifier predicts whether a flight is likely to be delayed.

Step 2: Delay Duration Estimation**:** If a delay is predicted, a regression model estimates the expected delay in minutes.

The regression model is only activated for flights flagged in Step 1. Moreover, the model was trained using a curated set of features such as scheduled departure time, flight route, and carrier code; deliberately avoiding real-time external inputs like local weather APIs, which are often unavailable at smaller Pakistani airports.

9.2. Proactive Passenger Communication System

One of the most direct applications of the classification model output is in enhancing the passengers' experience through real-time notifications. The classifier's binary output can be integrated with existing SMS systems and mobile apps to send automated alerts when a flight is at risk of delay.

For instance:

" Flight PK306 is likely to be delayed. Please check for updates before leaving for the airport."

This functionality allows passengers to make informed travel decisions, reducing:

1. Early or unnecessary airport arrivals
2. Terminal overcrowding
3. Dissatisfaction due to a lack of information

Such alerts are particularly useful in contexts where smartphone access is variable, as they can function effectively over traditional SMS platforms, where implementation can be achieved via local telecom integration.

9.3. Operations Dashboard for Ground Crew and Airport Staff

To help airport operation teams act on model predictions, we suggest a central monitoring dashboard that combines real-time classification and regression results. The dashboard is easy to use, requires minimal technical setup, and can run on existing airport systems or simple local networks.

Key Dashboard Features:

- Flight Number and Route
- Scheduled Departure Time
- Delay Risk Level (Low/Medium/High)
- Predicted Delay Duration (in minutes)
- Suggested Gate Reassignment
- Ground Crew Shift Recommendations

The interface may use a color-coded system:

- Red: High delay risk (> 45 mins)
- Yellow: Moderate risk (15–45 mins)
- Green: On-time/Low risk

This dashboard can be built using lightweight open-source frameworks such as Streamlit, Dash, or Flask, or integrated into Google Sheets for non-technical environments. It equips staff with decision-ready insights without relying on proprietary systems or cloud licenses.

9.4. Dynamic Turnaround Buffer Management

Aircraft turnaround time (TAT), the interval between arrival and subsequent departure, is often tight, especially in congested or understaffed regional airports. Delays in one flight can create a ripple effect, impacting multiple schedules. Our proposed solution introduces a dynamic buffer logic based on the regression model's delay duration output.

Buffer Adjustment Rules:
Let D be the predicted delay (in minutes). Then:

Predicted Delay (D)	D ≤ 20 mins	20 < D ≤ 45 mins	45 < D ≤ 60 mins	D > 60 mins
Buffer Adjustment Action	No change	Increase TAT buffer by 25%; alert the standby crew	Increase TAT buffer by 40%; consider gate reassignment	Trigger backup flight slotting; notify cross-functional teams

These rules can be added as a simple logic layer in existing schedules or dashboards. Automating buffer adjustments in this way helps reduce chain delays and makes turnaround operations more reliable without needing extra staff.

9.5. Feasibility in Low-Resource Settings

The design philosophy behind this system emphasizes efficiency and deployability, keeping in view the limitations at many Pakistani airports:

- No reliance on GPUs or real-time cloud infrastructure
- Can be hosted on locally available servers or desktop machines
- Compatible with open-source software stacks (e.g., Python, SQLite, Flask)
- Requires only historical flight schedules and airline metadata, no dependence on live radar or third-party APIs

Even in offline or low-connectivity settings, batch predictions can be prepared in advance and shared with staff as printouts or CSV files.

While this study demonstrates feasibility using U.S. flight data, future work and practical app development will focus on incorporating localized Pakistani flight data and regional weather information. This would further improve prediction accuracy in Pakistan's domestic aviation sector.

9.6. Policy and Administrative Alignment

Beyond technical design, this system supports broader national and organizational goals:

- Pakistan Civil Aviation Authority (PCAA): Aligns with their modernization and digital transformation objectives [4;12;13].
- Airlines (PIA and private carriers): Helps reduce fuel waste, lower delay-related penalties, and ease customer service pressures.
- Passenger welfare: Improves travel experience through more timely and transparent updates, allowing passengers to plan better and avoid unnecessary waiting at airports.

Telecom providers can help by offering affordable bulk SMS services and creating public-private partnerships to expand reach. This system can also serve as a pilot for tier-2 airports such as Multan, Skardu, Faisalabad, and Quetta, where simple and innovative tech solutions are urgently needed and practical to test.

9.7. Scalability & Future Extensions

The simplicity of this system does not preclude future upgrades. Once deployed and validated, the model could:

- Incorporate weather data where available
- Use adaptive learning from real-time updates
- Integrate with national air traffic control APIs
- Scale to other use cases like arrival delay prediction or cancellation risk

Thus, this architecture serves both as a final solution for low-resource settings and a foundation for more advanced systems where resources permit.

10. Conclusion

This study explored the use of machine learning models to predict flight delays using only basic operational flight data, focusing on applications in low-resource settings where additional data, such as weather and air traffic information, is not available. Both classification and regression approaches were tested to understand their strengths and limitations in this context.

The results showed that classification models, especially ensemble methods like Random Forest and Gradient Boosting, were more reliable at predicting whether a flight would be delayed. Regression models, while able to provide approximate delay durations, showed lower accuracy and greater sensitivity to limited features. These findings suggest that, when only simple pre-departure data is available, it is more practical to focus on predicting the occurrence of a delay rather than its exact length.

In real-world terms, classification models can help provide timely alerts to passengers and support operational decisions at airports, while regression estimates can offer additional guidance for planning ground operations. Combining both in a two-stage system by first classifying delay risk and then estimating the delay magnitude offers a practical and scalable solution, especially for regional airports in Pakistan and similar environments.

Beyond model performance, this work emphasizes the importance of creating solutions that align with real operational constraints. The proposed hybrid system and practical implementation concepts, such as dashboards and SMS alerts, show how machine learning can deliver value even without large-scale data or advanced infrastructure.

Finally, this project provided valuable experience in working through an end-to-end machine learning pipeline and proposing a practical system design. By balancing technical goals with real-world needs, this research lays the groundwork for future studies that can incorporate richer local data sources and further improve delay prediction systems in under-resourced aviation contexts.

11. Bibliography

The Express Tribune. (2019, May 20). Tourism economy Gilgit-Baltistan. *The Express Tribune*. https://tribune.com.pk/letter/1970934/tourism-economy-gilgit-baltistan

Catal, U., & Diri, B. (2021). Flight delay prediction using gradient boosting machine learning classifiers. *ResearchGate*. https://www.researchgate.net/publication/352288281_Flight_Delay_Prediction_Using_Gradient_Boosting_Machine_Learning_Classifiers

Shah, S. Z., & Tabrizchi, M. (2022). Flight delay prediction using machine learning algorithm XGBoost. *ResearchGate*. https://www.researchgate.net/publication/361603066_Flight_Delay_Prediction_Using_Machine_Learning_Algorithm_XGBoost

Tavana, M., Soheilirad, S., Santos-Arteaga, F. J., & Mina, H. (2017). Airline delay prediction and causality analysis using machine learning techniques. Scientia Iranica, 24(6), 3268–3280. https://scientiairanica.sharif.edu/article_20020_ca3a3beb96caf169b6f454c703517d0c.pdf

Tian, B., Wei, H., & Hu, S. (2020). A review of airline delay prediction models and applications. Transportation Research Record, 2674(6), 688–708. https://doi.org/10.1177/0361198120930014

Jha, R. K., Jha, S. B., Pandey, V., & Babiceanu, R. F. (2024). *Flight Delay Prediction using Hybrid Machine Learning Approach: A Case Study of Major Airlines in the United States*. arXiv. https://arxiv.org/abs/2409.00607

Mu, Y. W. (2019). Airline delay and cancellation data 2009–2018. *Kaggle*. https://www.kaggle.com/datasets/yuanyuwendymu/airline-delay-and-cancellation-data-2009-2018/data

Federal Aviation Administration. (n.d.). ASQP: Definitions of variables. *Aviation System Performance Metrics (ASPM) Help*. Retrieved July 13, 2025, from https://aspm.faa.gov/aspmhelp/index/ASQP__Definitions_of_Variables.html

Abb Takk News. (2023). Severe weather grounds six flights in northern regions. *Abb Takk News*. https://abbtakk.tv/severe-weather-grounds-six-flights-in-northern-regions/

Khaksar, H., & Sheikholeslami, A. (2017). Airline delay prediction by machine learning algorithms. *Scientia Iranica*, *24*(6), 3268–3280. https://scientiairanica.sharif.edu/article_20020.html

Asif, M., Ahmed, M., Latif, S., & Qayyum, A. (2023). Flight delay prediction using gradient boosting and SHAP analysis. *Electronics*, 13(24), 4910. https://doi.org/10.3390/electronics13244910

Zeng, Y., Hu, Y., Zhang, K., & Wang, L. (2024). Flight delay prediction using spatio-temporal features and explainable machine learning. *arXiv preprint arXiv:2407.15185*. https://arxiv.org/abs/2407.15185

Pakistan Civil Aviation Authority. (n.d.). *Home*. https://www.pcaa.gov.pk/

Reuters. (2023, March 30). Pakistan kicks off outsourcing operations, assets of three airports. https://www.reuters.com/world/asia-pacific/pakistan-kicks-off-outsourcing-operations-assets-three-airports-2023-03-30/

Simple Flying. (2023, March 30). Pakistan set to outsource its three major airports. https://simpleflying.com/pakistan-outsource-major-airports/https://scientiairanica.sharif.edu/article_20020_ca3a3beb96caf169b6f454c703517d0c.pdf