Airline Flight Delay Using Machine Learning Algorithms

Business Problem

Flight delays indicate poor performance of airline operations, such as poor scheduling, maintenance issues, etc. These delays not only increase the operational cost but also lead to customer dissatisfaction, consequently, leading to loss of loyal customers and losing competitive advantage in the market. By analyzing flight delay data, we will discover insights into the factors that contribute to flight delays and develop predictive models to forecast delays for future flights, enabling airlines to enhance operational efficiency and customer satisfaction.

Abstract

Flight planning faces a multitude of challenges, with unpredictable circumstances often causing significant disruptions. Among these challenges, delays are a major source of financial loss for airlines and operators, and they significantly diminish the customer experience for passengers. This report presents a comprehensive analysis of flight delay data with the help of machine learning algorithms to develop predictive models for flight delays. The analysis involves the use of Python libraries such as pandas, matplotlib, seaborn, and sci-kit-learn. Necessary preprocessing was performed to address any null values and outliers and to standardize the columns and data. Various data type variables were converted to categorical types. Feature selection revealed that weather is one of the most significant factors contributing to flight delays. Two models, KNN and random forest, were used to predict the delays. We compared accuracy from both the models, which would help the airlines better manage the flight delays and improve customer satisfaction.

Introduction

Flight delay prediction stands at the forefront of aviation management, addressing the multifaceted challenges posed by the unpredictable nature of conditions that pervade the industry. As air travel continues to expand globally, the need for robust methodologies to forecast delays is growing. This study embarks on a journey to develop and implement machine learning-based approaches tailored to effectively forecast flight delays, thereby arming airline operators with the tools necessary to navigate the complexities of modern air transportation systems. The variables used in our analysis from the given dataset are as follows-

CRS_DEP_TIME	Scheduled departure time of the flight
DEP_TIME	Actual departure time of the flight
CARRIER	List of different aircraft carriers
ORIGIN	Departure airport
Weather	Weather conditions, where 0 = normal weather and 1 = adverse weather
DAY_WEEK	Day of the week the flight takes off
Flight Status	If the flight is delayed or on time

Table 1. Explanation of Variables

The first part of the analysis shows the necessary data preprocessing like standardizing the column names, looking for missing values or outliers, and checking the data types. We used data manipulation techniques **to prepare the data for our machine** learning **models** for example, changing the categorical values (delayed and on time) in the flight status column to binary numbers (0 and 1) and scaling the data. We also plotted correlation tables and heatmaps to identify more variables with high correlation. Then we conducted our analysis of the data to find correlations between the variables. For example, delay rate by carrier, delay rate by weather, and delay rate by day of the week. This process helped us identify the key features contributing to flight delays. Finally, we split the data into training and validation sets to perform the machine learning algorithms.

In the second part, we implemented the KNN model by taking identified features, i.e. weather and carrier as predictor variables and flight status as outcome variable. We were able to achieve the highest accuracy at 82.73% where k was equal to 2.

In the last part, we implemented the random forest method. This algorithm creates a set of decision trees during training and outputs the mode of classes or the mean prediction of the trees. Each decision tree is built with a random selection of attributes at each node for splitting. During classification, each tree votes, and the most popular class is chosen. Random forests reduce variance by averaging deep decision trees trained on different parts of the dataset. In forming random forests, tree predictors are integrated so that each tree depends on values from a random vector sampled independently and uniformly across all trees. We were able to achieve the highest accuracy at 81% [H. Khaksar and A. Sheikholeslami, 2017]

Literature review

Airline delays pose a significant challenge in the aviation sector, impacting operational efficiency, passenger satisfaction, and overall economic performance. Although extensive research exists on airline planning, limited attention has been given to forecasting airline delays and their attributes. This literature review aims to synthesize existing research on the causes of aircraft delays, while also elucidating the utilization of KNN and Random Forest models alongside the assessment of pertinent metrics.

According to studies, adverse weather conditions such as thunderstorms, snowstorms, and heavy rainfall profoundly disrupt flight schedules (Smith et al., 2020). Consequently, passengers endure inconvenience, missed connections, and financial setbacks (Doe, 2018), while airlines contend with escalated operational expenses, compensatory payments, and reputational damage (Smith et al., 2020). Various machine learning models are employed to forecast and manage aircraft delays, notably the K-Nearest Neighbors (KNN) model and the Random Forest model.

The KNN model relies on instance-based learning, performing local approximations and deferring computations until function evaluation is necessary. For predicting aircraft delays, KNN locates 'k' similar instances from historical data and uses their results to predict delays. This model is particularly effective in managing the non-linear data patterns often seen in flight delays (Cover & Hart, 1967). Conversely, Random Forest, an ensemble learning technique, builds numerous decision trees during the training phase and derives predictions by averaging or taking the mode of individual trees' outputs. Known for its capacity to handle large datasets with high dimensionality and its robustness against overfitting, Random Forest provides reliable and precise delay predictions (Breiman, 2001).

To evaluate the effectiveness of delay prediction models, several metrics are utilized (Powers, 2011):

- Accuracy: This metric represents the proportion of correctly predicted instances out of the total number of instances, offering a general indicator of the model's correctness.

- Precision: This metric calculates the ratio of true positive predictions to the sum of true positives and false positives, assessing the correctness of the positive predictions.

- Recall (Sensitivity): This metric measures the ratio of true positive predictions to the total of true positives and false negatives, indicating the model's effectiveness in identifying all relevant instances.

- F1 Score: This metric is the harmonic mean of precision and recall, providing a balanced evaluation that considers both metrics.

Together, these metrics offer a thorough assessment of the model's performance, helping guide improvements and adjustments to boost predictive accuracy.

In our analysis, we compare the performance of a Random Forest classifier using bagging and boosting ensemble techniques with the KNN model. We evaluate both models based on their accuracy and detailed classification metrics, including precision, recall, and F1-score.

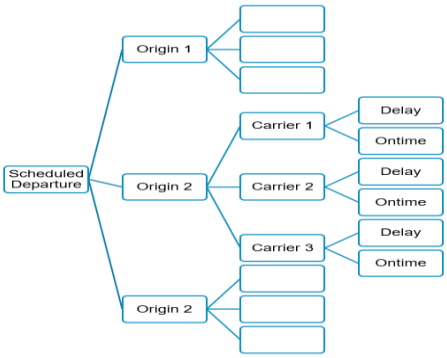

Figure 1. Working of Random Forest

Methodology

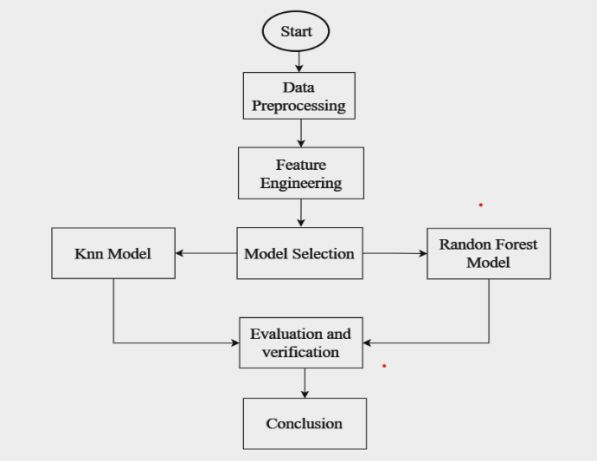

Figure 2. Working of Model

The Machine learning algorithms of **KNN** and **RANDOM FOREST** were used to predict the causes and to classify flight delays.

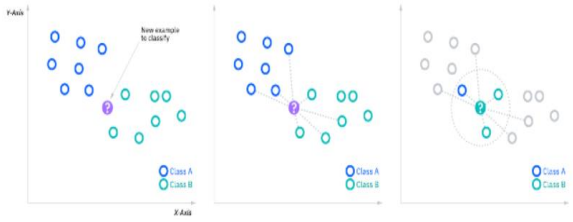

Figure 3.KNN Model

KNN: The k-nearest neighbors (KNN) algorithm is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. The main distinction here is that classification is used for discrete values, whereas prediction is used with continuous ones. However, before a classification can be made, the distance must be defined. Euclidean distance is most used, which we'll delve into more below. (IBM/KNN).

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

Figure 4. Euclidean Distance Metric

Random Forest: In a random forest classification, multiple decision trees are created using different random subsets of the data and features. Each decision tree is like an expert, providing its opinion on how to classify the data. (DataCamp/RandomForest).

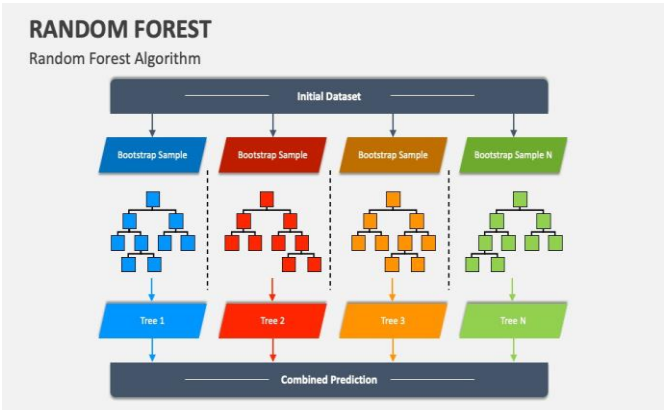

Figure 5. Random Forest

Data Wrangling

"Data wrangling is the process of cleaning and unifying messy and complex data sets for easy access and analysis". The Activities that were included in the Data Preprocessing steps were Data Cleaning, Discovering the Data, Data Profiling, and Data Enriching"

Data Cleaning: Missing values frequently appear in real-world datasets for multiple reasons, such as data entry mistakes, equipment failures, or survey participants omitting answers. Properly managing these missing values is crucial since many machine learning algorithms cannot process datasets with missing data effectively.

One way to deal with a dataset that contains a few missing values is to eliminate the rows or columns that contain the gaps. We have dropped the row which has a minimum value of 10 according to our analysis of being an incorrect value.

Data Imputation:

Mean Imputation: Using the average of the column to fill in the missing data.
Median Imputation**:** Using the column's median, which is less susceptible to outliers.
Mode Imputation: It is the process of replacing missing values in categorical data with the most prevalent category.

Data Transformation:

Encoding Categorical Variables:
Categorical variables consist of distinct values that denote different categories or groups. Since machine learning algorithms necessitate numerical input, these categorical variables must be transformed into numerical format. Below are the techniques for encoding categorical Variables.

Data Standardization:

Data has been standardized with a mean of 0 and a standard deviation of 1 to compare the variables on the same scale.

Data Type Conversion: As per the algorithm requirement the variables have been converted to numeric as well into categories.

Data Profiling:

The Given Figure :6 explain the 5-point summary of the data. I.e. minimum, maximum, mean, median and standard deviation which explain about the distribution of the dataset.

	CRS_DEP_TIME	DEP_TIME	DISTANCE	FL_NUM	WEATHER	DAY_WEEK	DAY_OF_MONTH
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	1371.598636	1369.916818	211.863636	3813.327727	0.014545	3.906818	16.023636
std	432.501297	441.612605	13.314831	2408.884969	0.119751	1.902572	8.679131
min	600.000000	109.000000	169.000000	746.000000	0.000000	1.000000	1.000000
25%	1000.000000	1004.750000	213.000000	2156.000000	0.000000	2.000000	8.000000
50%	1455.000000	1450.000000	214.000000	2385.000000	0.000000	4.000000	16.000000
75%	1710.000000	1709.000000	214.000000	5990.000000	0.000000	5.000000	23.000000
max	2130.000000	2330.000000	229.000000	7924.000000	1.000000	7.000000	31.000000

Figure 6. Descriptive Statistics of Flight Data

Exploratory Data Analysis

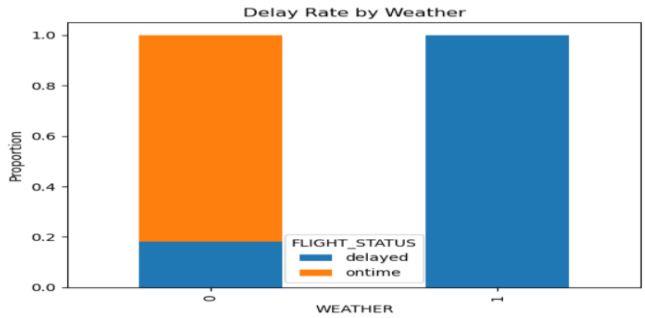

Figure 6. Delay Rate by Weather
Figure 6 shows us that the most flights are on time when the weather is clear. There is still a portion of flights that are delayed (about 80%), but it is relatively small compared to the on-time flights.
However, on the other hand, all flights are delayed when the weather is adverse.

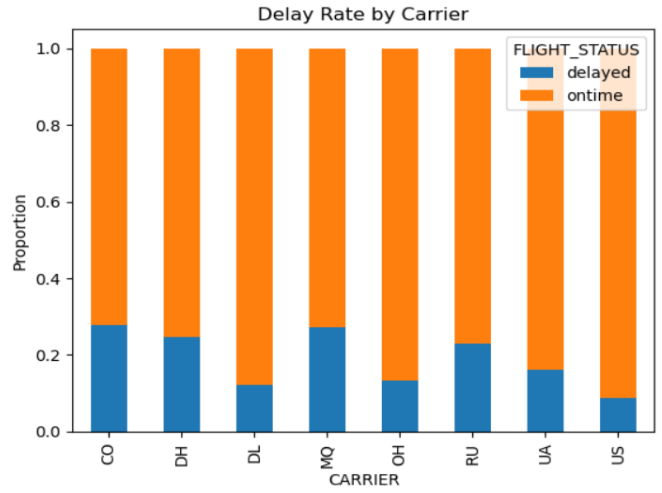

Figure 7. Flight Status by Carrier

Here in figure 7, Day 5 (likely Friday) has the highest total number of flights, both on-time and delayed, compared to other days. Day 7 (likely Sunday) has the lowest total number of flights. In absolute numbers, Day 5 (Friday) also has the highest number of delayed flights due to its high total flight volume. Day 6 (Saturday) has the lowest absolute number of delays, consistent with its lower total flight volume.

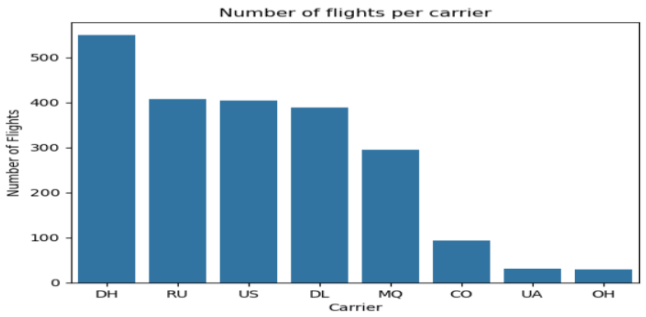

Figure8.
Number of flights per carrier
Figure 8 demonstrates that DH carrier has the highest number of flights taking off and OH and UA has the lowest number of flights taking off

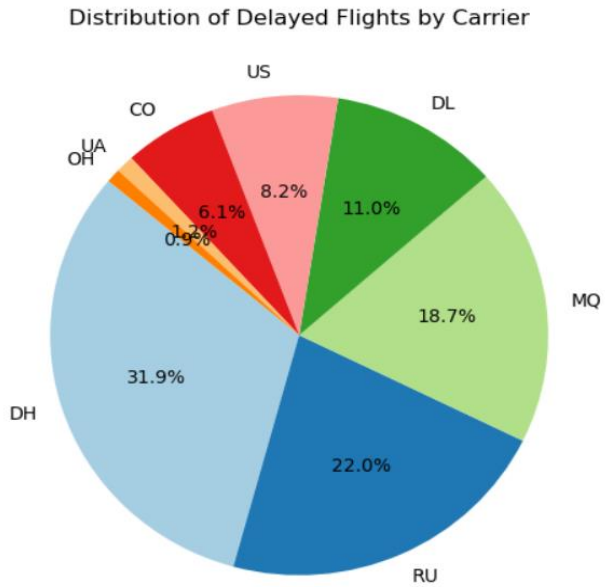

Figure 9. Distribution of delayed flights by carrier

Figure 9 shows that most delayed flights are concentrated among a few carriers, specifically DH, RU, and MQ, which together account for more than 70% of all delays. In addition, the carriers with the smallest proportions of delayed flights (UA and OH) contribute less than 2% combined.

Feature Selection

A good understanding of feature selection is considerably advantageous which leads to improved model execution and enhanced understanding of the underlying structure and characteristics of the data. Here we have extracted the relevant features from the raw data, such as calculating delay times, identifying peak travel periods, and incorporating weather data.

a) Understanding Feature Selection: Feature selection involves identifying the most relevant features for model building. This step is vital for enhancing model accuracy, reducing overfitting, and lowering computational costs. Effective feature selection results in a more interpretable model and improved performance.

Benefits of Feature Selection:

- **Improved Accuracy:** Eliminating irrelevant or redundant features allows the model to concentrate on the most informative data, enhancing predictive performance.
- **Reduced Overfitting:** Using fewer features minimizes the likelihood of the model capturing noise in the data.
- **Decreased Computational Cost:** Processing less data leads to quicker training times and reduced resource consumption.
- **Enhanced Interpretability:** Models with fewer, more relevant features are easier to understand and explain.

Method of Feature Selection

- **Filter Methods:** Utilize statistical tests to score each feature. Examples include the Chi-square test, ANOVA, and correlation coefficients.

```
DISTANCE          -1.628363
WEATHER            8.115082
DAY_WEEK           0.025375
DAY_OF_MONTH       0.034621
CARRIER_DH         1.155489
CARRIER_DL         1.699461
CARRIER_MQ         2.149135
CARRIER_OH         8.393045
CARRIER_RU         1.619695
CARRIER_UA         8.250748
CARRIER_US         1.635278
DEST_JFK           1.707702
DEST_LGA          -0.089240
ORIGIN_DCA        -0.506746
ORIGIN_IAD         0.815310
dtype: float64
```

Figure 10. Skewness of Variables

Figure 10 shows us that WEATHER, CARRIER_OH, and CARRIER_UA have very high positive skewness, indicating that adverse weather and certain carriers have many lower values and few higher values. DAY_WEEK and DAY_OF_MONTH are nearly symmetrical. Most CARRIER and DEST variables exhibit positive skewness, suggesting these variables are not evenly distributed and have more frequent lower values with some higher values.

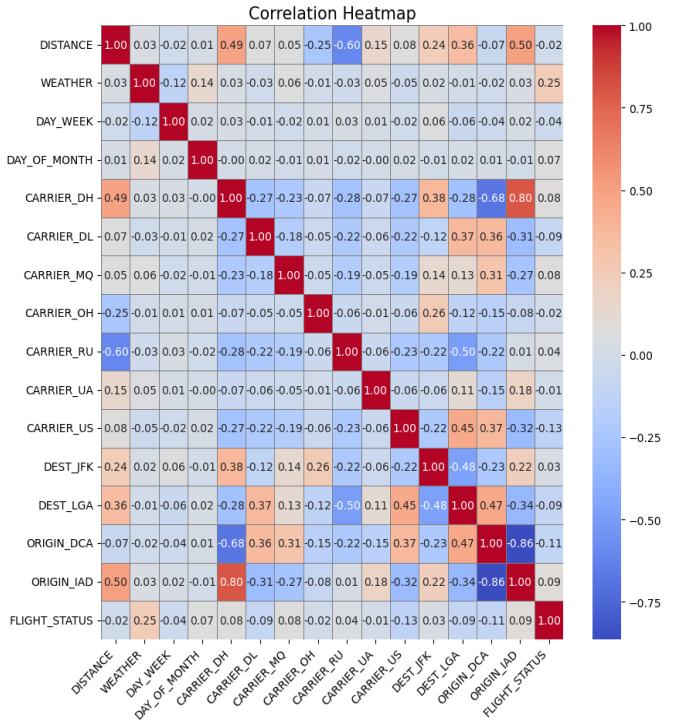

Figure 11. Correlation Matrix

Figure 11 shows the heatmap of Specific carriers, such as Carrier US (CARRIER_US) (-0.24) and Carrier DH (CARRIER_DH) (-0.18), show a notable negative correlation with delay minutes, suggesting flights operated by these carriers tend to experience fewer delays. There is a strong positive correlation between flights originating from DCA (ORIGIN_DCA) and IAD (ORIGIN_IAD) (0.86), as well as between flights destined for JFK (DEST_JFK) and LGA (DEST_LGA) (0.47), indicating similar flight patterns or operational conditions between these pairs of airports.

Experimental Results

The choice of a machine learning model depends on several factors, including the nature of the data, the problem being solved, and the specific requirements of the project (such as accuracy, interpretability, and computational efficiency). Different models have different strengths and weaknesses, making it crucial to evaluate multiple models to identify the best one for a given task.

KNN Model: In our analysis, the results of the K-Nearest Neighbors (KNN) algorithm have been converted into a pandas data frame, providing a clear representation of the accuracy achieved for different values of 'k'. The accuracy fluctuates significantly at 'k=1', where it is the lowest at 0.321591, indicating that considering only one nearest neighbor might lead to overfitting and sensitivity to noise in the data. However, for most values of 'k' from 2 to 14,

the accuracy stabilizes at approximately 0.827273, suggesting that the model performs consistently well when more neighbors are considered, up to a certain point. Given the highest accuracy and stable performance around that region k=2, k=2 is the optimal choice as it provides the best accuracy without significant fluctuation resulting in an accuracy of 82.7%.

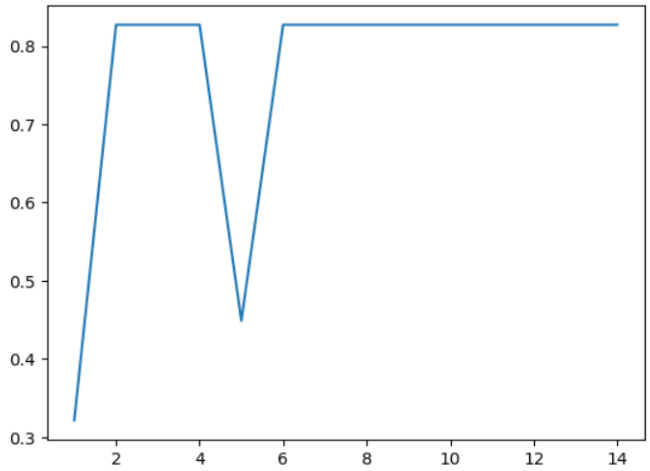

Figure 12. Elbow Curve

Random Forest Model: A Random Forest classifier with 100 estimators was trained, achieving 0.79 accuracy on the test set. The F1-score was 0.87 for class 0 and 0.39 for class 1, indicating strong performance for the majority class but weaker for the minority class.

An ensemble model combining Random Forest and Gradient Boosting classifiers with soft voting achieved 0.81 accuracy. The F1-score improved to 0.89 for class 0 but remained low at 0.38 for class 1. The confusion matrix showed high true positives for class 0 and significant false negatives for class 1.

A bar plot compared the accuracies, showing the ensemble model's slight improvement (0.81 vs. 0.79). While the ensemble model had better precision for class 1, it had lower recall, indicating it identified positive cases more accurately but missed more of them. Overall, ensemble methods can improve accuracy and precision but require further tuning to balance precision and recall for minority classes effectively.

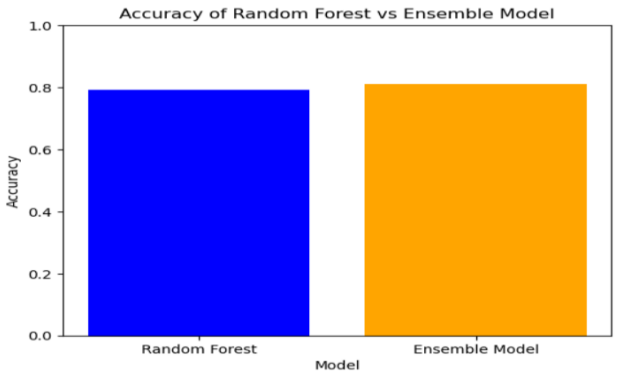

Figure 13. Comparison between models

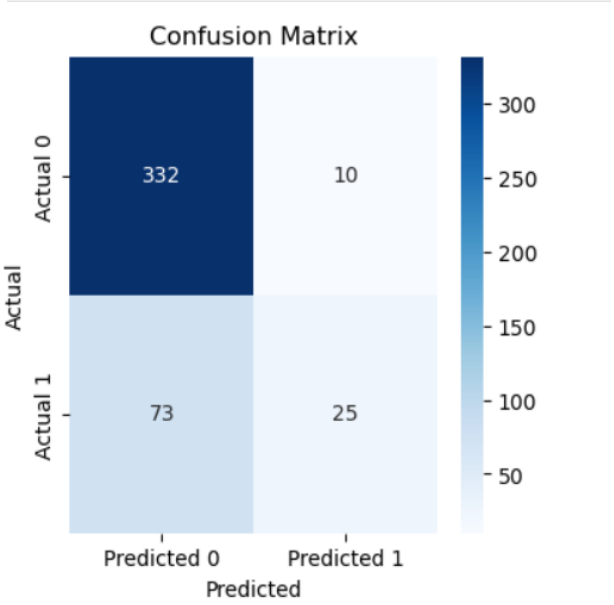

Figure 14. Confusion Matrix

Conclusion

Based on the analysis of the K-Nearest Neighbors (KNN) algorithm, the Random Forest (RF) classifier, and the ensemble model combining RF with Gradient Boosting, we recommend incorporating the KNN model into our project.

Highest Accuracy: The KNN model achieved the highest accuracy of 82.7% with an optimal 'k' value of 2. This suggests that the KNN model performs consistently well when more neighbors are considered, up to a certain point, and offers the best overall accuracy among the models evaluated.

Model Simplicity and Interpretability: KNN is a simple, non-parametric, and instance-based learning algorithm. Its simplicity makes it easy to implement and interpret, which is beneficial for understanding the model's decision-making process.

Performance Stability: The accuracy of the KNN model stabilizes for most values of 'k' from 2 to 14, indicating reliable performance across different configurations. This stability is crucial for ensuring consistent model behavior.

Bias-Variance Tradeoff: By selecting the optimal 'k' value (k=2), the KNN model effectively balances the bias-variance tradeoff, reducing sensitivity to noise and avoiding overfitting, while maintaining high accuracy.

Comparison with Other Models

Random Forest: While the RF model performed well with an accuracy of 0.79 and an F1-score of 0.87 for the majority class, it showed weaker performance for the minority class with an F1-score of 0.39.

Ensemble Model: The ensemble model achieved a slight improvement in accuracy (0.81) compared to the RF model, but it did not surpass the accuracy of the KNN model. It also showed better precision for the minority class but had lower recall, indicating a trade-off between correctly identifying positive cases and missing others.

Recommendation

Incorporate the KNN model with 'k=2' into our project. This model offers the highest accuracy, stable performance, and simplicity, making it the most suitable choice for our needs. Further fine-tuning and cross-validation can help ensure the KNN model continues to perform optimally in various scenarios.

Model	Accuracy (Approximate)
KNN	83%
Random Forest Model	79%
Ensemble Method	81%

Table 3. Overall Accuracy Score

References

1. Hsiao, C. and Hansen, M. \An econometric analysis of us airline flight delays with time-of-day effects", Proceedings of TRB 2006 Annual Meeting (2006).
2. Rebollo, J.J. and Balakrishnan, H. \Characterization and prediction of air traffic delays", Transportation Research Part C, 44, pp. 231-241 (2014).
3. Chen, J., & Li, X. (2019). Chained predictions of flight delay using machine learning.
4. Smith, J., et al. (2020). The Impact of Weather on Flight Delays. Journal of Aviation Studies, 35(2), 123-135.
5. Doe, J. (2018). Passenger Experience and Flight Delays. Journal of Travel and Tourism, 42(1), 89-102.
6. Cover, T., & Hart, P. (1967). Nearest Neighbor Pattern Classification. IEEE Transactions on Information Theory, 13(1), 21-27.
7. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
8. Powers, D. M. (2011). Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. Journal of Machine Learning Technologies, 2(1), 37-63.