Artificial Intelligence Project Report

Airline Passenger Satisfaction Prediction

احمد ايمن محمود خليفة عبد الله ID:2023170026
ابراهيم احمد شكري عبد الفتاح ID:2023170009
ابراهيم ياسر متولي حسني ID:2023170016
مصطفي عبده عبد الباقي عبد الغني ID:202317604
احمد قبيصي لطفي احمد ID:2023170049
احمد حسام مصطفي سند عبد الظاهر ID:2023170029
ابراهيم محمد ابراهيم ID:2023170013

Data Analysis & Visualization

Before diving into preprocessing, we performed exploratory data analysis (EDA) to understand the structure of the dataset, detect potential issues, and identify patterns and relationships between features and passenger satisfaction.

We started by separating the dataset into **numerical** and **categorical** features. This allowed us to visualize and analyze them appropriately.

Numerical Features

While exploring numerical features, we found the **gender distribution** to be fairly balanced—approximately **51% male** and **49% female**—which suggests that the dataset is not biased in terms of gender representation.

We also examined **departure** and **arrival delays**. Most flights in the dataset had **no significant delays**, leading to a **highly skewed distribution** for both features. This skewness is important to keep in mind as it may affect model learning and performance.

Categorical Features

In the categorical data, we observed interesting trends in the **service-related ratings**:

- **Food and drink** ratings were almost evenly distributed across scores of 2, 3, 4, and 5.
- **Wi-Fi service** and **ease of online booking** consistently received low ratings, with a majority of responses clustered around **2 out of 5**.

These insights suggested a clear potential connection between certain features (especially low-rated services) and overall **passenger dissatisfaction**.

We also used bar plots, histograms, and box plots to visualize the distributions and relationships between features. For example:

- Passengers with lower ratings for in-flight services (e.g., Wi-Fi, food) were more likely to be dissatisfied.
- Delayed arrivals were also more common among dissatisfied passengers.

This analysis helped guide our feature selection and preprocessing decisions.

Data Preprocessing

After completing the data analysis, we moved on to preparing the data for machine learning models. This step involved cleaning, transforming, and encoding the data to ensure optimal model performance.

Correlation & Feature Reduction

We first checked the correlation between numerical features. **Departure delay** and **arrival delay** were found to be highly correlated. To reduce redundancy, we **dropped the departure delay** feature. We kept arrival delay because we hypothesized that it has a stronger impact on satisfaction—passengers are typically more frustrated by late arrivals than late departures, especially when they have tight schedules.

Additionally, we removed irrelevant features such as the **ID** column and an **unlabeled feature**, as they provided no predictive value.

Handling Outliers

We identified and removed **outliers** from the dataset. Since only about **70 rows** out of **103,000** were outliers, eliminating them had no negative impact on data quality and helped improve overall model stability.

Missing Values

- For **numerical features**, we replaced missing values with the **median**, as the presence of skewed distributions made the median a more robust choice than the mean.

- For **categorical features**, missing or null-like values (represented as `0`) were replaced with the **mode**, representing the most frequent and likely correct category.

Feature Transformation

Several numerical features were **heavily skewed**, which can negatively affect model performance. To address this, we applied a **log transformation** to normalize the distributions. This transformation helped stabilize variance and improve model training.

We also experimented with **feature engineering**, where we attempted to combine all service-related ratings into a **single summed score**. However, this approach significantly reduced model performance, likely due to loss of granularity and feature-specific importance. As a result, we decided to discard this idea and retain the individual rating features.

Encoding Categorical Variables

We encoded categorical features to make them suitable for machine learning models:

- **Binary features** were label encoded using **0 and 1**.
- One categorical feature with **three unique values** was encoded into **0, 1, and 2**.

Feature Scaling

We noticed that the **distance traveled** feature had values that were much larger than other features. To ensure this did not skew the model's learning, we applied **normalization** to scale the distance feature to a smaller, comparable range.

Models and Hyperparameter Tuning

To build an effective classification system for predicting airline passenger satisfaction, we experimented with **four machine learning models**:

- **Logistic Regression**
- **Support Vector Machine (SVM)**
- **K-Nearest Neighbors (KNN)**
- **Random Forest Classifier**

All four models are designed for **binary classification tasks**, making them well-suited for predicting satisfaction outcomes (Satisfied / Not Satisfied). Each model was selected based on its unique characteristics and potential strengths when dealing with our dataset.

Model Selection and Justification

1. Logistic Regression

We included Logistic Regression as a **baseline model** due to its simplicity, interpretability, and effectiveness in linearly separable classification problems. It also performs well when there is a **clear**

relationship between the independent variables and the output class, and it is computationally efficient.

2. Support Vector Machine (SVM)

SVM was selected because of its **ability to handle high-dimensional feature spaces** and its effectiveness in datasets with **clear margins between classes**. Given that our dataset includes multiple service and demographic features, SVM could potentially construct optimal decision boundaries to separate satisfaction levels.

3. K-Nearest Neighbors (KNN)

We used KNN because of its **non-parametric nature** and its **simplicity in implementation**. KNN works well when decision boundaries are **non-linear** and when **local patterns** in the data are important. Since customer satisfaction could depend on combinations of multiple features (e.g., flight delay + poor service), KNN was a logical choice to test.

4. Random Forest

We included Random Forest for its **ensemble learning capabilities** and **robustness against overfitting**. It can handle both **categorical and numerical data** well and can also provide insights into **feature importance**, which is valuable for interpretation. Random Forest is often a top performer in classification tasks with structured data.

Hyperparameter Tuning

K-Nearest Neighbors (KNN)

We conducted **manual hyperparameter tuning** by testing values of **K from 1 to 15**. For each K value, we measured the model's performance using cross-validation. After plotting the accuracy against the number of neighbors, we observed that the performance **plateaued after K=5** and even showed signs of minor fluctuations beyond that point. Therefore, based on a **reversed Elbow Method**, we chose **K = 5** as the optimal number of neighbors.

Evaluation

To assess the performance of our models, we used the following standard classification metrics:

- **Accuracy**: Proportion of total correct predictions.
- **Precision**: Ability of the model to return only relevant results (True Positives / (True Positives + False Positives)).
- **Recall**: Ability of the model to identify all relevant cases (True Positives / (True Positives + False Negatives)).
- **F1-Score**: Harmonic mean of precision and recall.
- **Confusion Matrix**: Provides a full breakdown of true positives, false positives, true negatives, and false negatives.

We trained and tested all models on the same preprocessed dataset using an 80/20 train-test split and 5-fold cross-validation to ensure generalizability of results.

Strengths and Weaknesses of Each Model

Logistic Regression

- **Strengths**: Fast, simple, interpretable, works well with linearly separable data.
- **Weaknesses**: Limited in capturing complex relationships; underperforms with nonlinear patterns.

SVM

- **Strengths**: Great for high-dimensional data; performs well with clear class boundaries.
- **Weaknesses**: Training can be slow; sensitive to parameter tuning; less interpretable.

K-Nearest Neighbors

- **Strengths**: Simple, non-parametric; works well with local structure and nonlinear data.
- **Weaknesses**: Computationally expensive for large datasets; sensitive to noisy features; performance highly dependent on the choice of K and scaling.

Random Forest

- **Strengths**: Highest performance; handles both categorical and numerical features; resistant to overfitting; gives feature importance.
- **Weaknesses**: Less interpretable than linear models; slower training compared to simpler models.

Summary and Insights

- The **Random Forest classifier outperformed all other models**, achieving the highest accuracy and F1-score. Its ability to handle nonlinear patterns, feature interactions, and diverse input types made it the best fit for our dataset.
- **SVM** also performed well, but required careful tuning and took longer to train.
- **Logistic Regression**, while being the simplest model, served as a strong baseline.
- **KNN** gave decent results but showed sensitivity to feature scaling and computational inefficiency with large datasets.

From this analysis, we conclude that **ensemble methods like Random Forest are best suited** for structured customer satisfaction prediction tasks, especially when dealing with mixed data types and complex interactions.

In future iterations, combining multiple models (e.g., through stacking or boosting) or testing neural networks could further enhance performance.