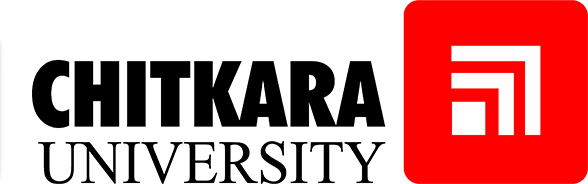
**Capstone Project - 1**

Project Report Semester-IV (Batch-2022)

**Customer Feedback Prediction**



# Supervised By: Submitted By:

Mr. Mohd. Talib Sir

Vartika Yadav 2210992515 (G-29)

Vardhman Jain 2210992513 (G-29)

Taniya 2210992448 (G-29)

**Department of Computer Science and Engineering** **Chitkara University Institute of Engineering & Technology,**

**Chitkara University, Punjab**

**Table Of Contents**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Title** | **Page No.** |
| **1.** | Introduction | 3 |
| **2.** | Problem Statement | 4 |
| **3.** | Software Interaction | 5 |
| **3.** | Exploratory Data Analysis | 6-10 |
| **5.** | Data Preprocessing | 11 |
| **6.** | ML - Models | 12-15 |
| **7.** | Conclusion | 16 |

**Introduction**

In the dynamic landscape of the airline industry, deciphering and predicting customer satisfaction stand as pivotal endeavors. This project endeavors to tackle this critical challenge by leveraging the advanced capabilities of Artificial Intelligence (AI) and Machine Learning (ML) on comprehensive survey data collected from passengers. By delving deep into this data, the project seeks to unveil intricate patterns and discern the underlying factors that significantly influence passenger satisfaction levels. Armed with these actionable insights, airlines can strategically refine their services, personalize customer interactions, and address pain points effectively, thus elevating the overall travel experience and fostering enduring loyalty among passengers.

By proactively addressing customer needs and preferences identified through AI/ML analysis, airlines can not only enhance customer satisfaction but also gain a competitive edge in the market. Through targeted improvements in service delivery, product offerings, and customer engagement strategies, airlines can cultivate stronger relationships with passengers, leading to increased brand loyalty and advocacy. Ultimately, this project aims to empower airlines to adapt and innovate in response to evolving passenger expectations, thereby driving sustained growth and success in an increasingly customer-centric industry landscape.

**Problem Statement**

In the dynamic realm of the airline industry, accurately anticipating and comprehending customer satisfaction stands as a formidable challenge. This project serves as a pioneering initiative aimed at addressing this critical issue through the strategic application of Artificial Intelligence (AI) and Machine Learning (ML) methodologies to meticulously curated survey data. By harnessing the analytical power of AI/ML, the project aims to uncover nuanced patterns and discern pivotal factors intricately linked to passenger satisfaction levels. Armed with these invaluable insights, airlines are poised to proactively refine and optimize their service offerings, thereby cultivating a more seamless and enriching travel experience for passengers.

At its core, this aims represents a transformative paradigm shift, empowering airlines to transcend conventional approaches and embrace data-driven strategies for customer-centric enhancement. By decoding the multifaceted layers of customer satisfaction, airlines can pivot towards a more agile and responsive operational framework, tailoring services to precisely meet the evolving needs and preferences of their clientele. This holistic approach not only bolsters customer satisfaction levels but also fosters enduring loyalty and advocacy, ultimately positioning airlines at the forefront of innovation and excellence in the competitive aviation landscape.

**Software interaction:**

- **Jupyter Notebook** serves as the primary Integrated Development Environment (IDE) for its interactive and collaborative features.

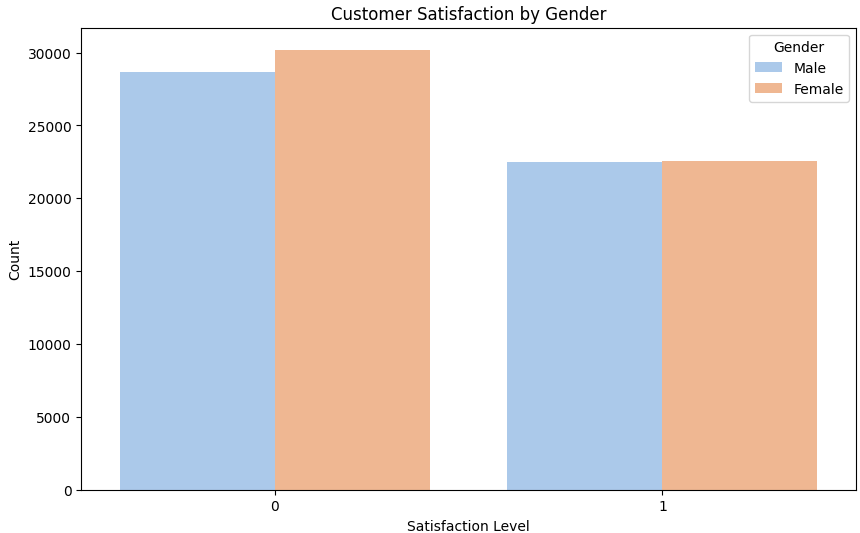
- **Pandas** and **NumPy** play integral roles in data manipulation, pre-processing, and mathematical operations, respectively.

- **Automated Exploratory Data Analysis (EDA**) is facilitated through data prep, streamlining insights discovery.

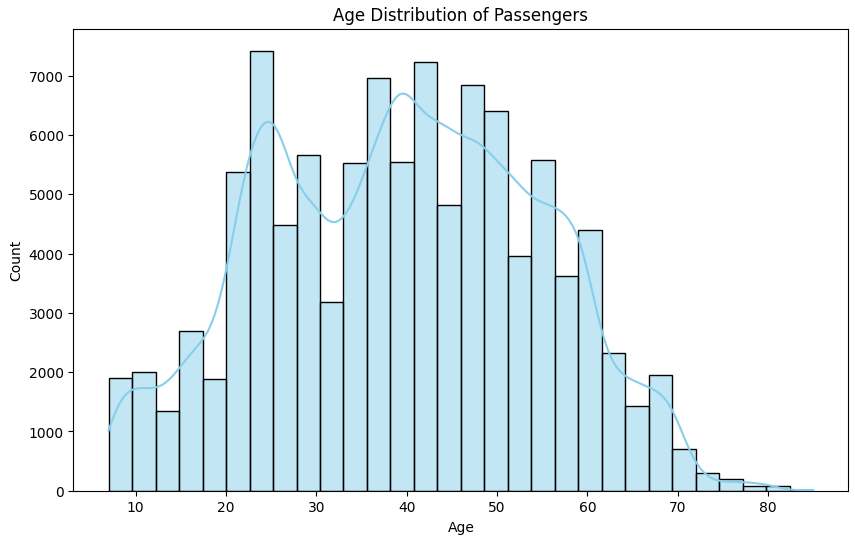
- **Matplotlib**, **Seaborn,** and **Plotly** are utilized for visualization, enabling the creation of compelling graphical representations.

- **GitHub** serves as the cornerstone for version control, ensuring collaboration, change tracking, and repository integrity.

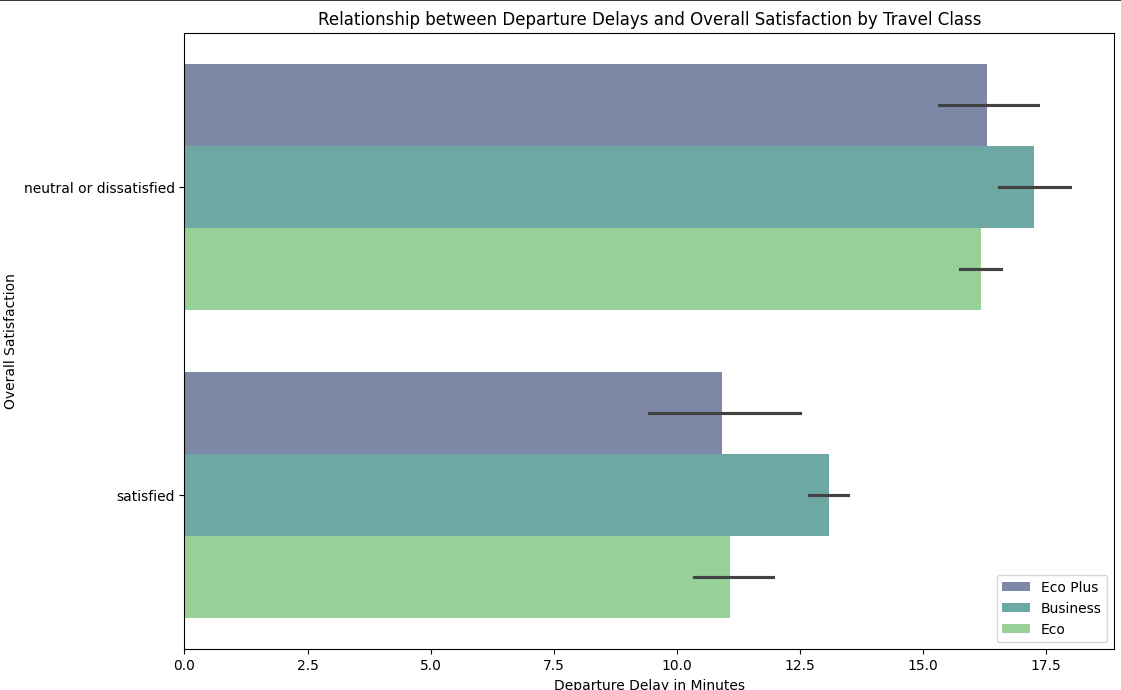
**Exploratory Data Analysis**



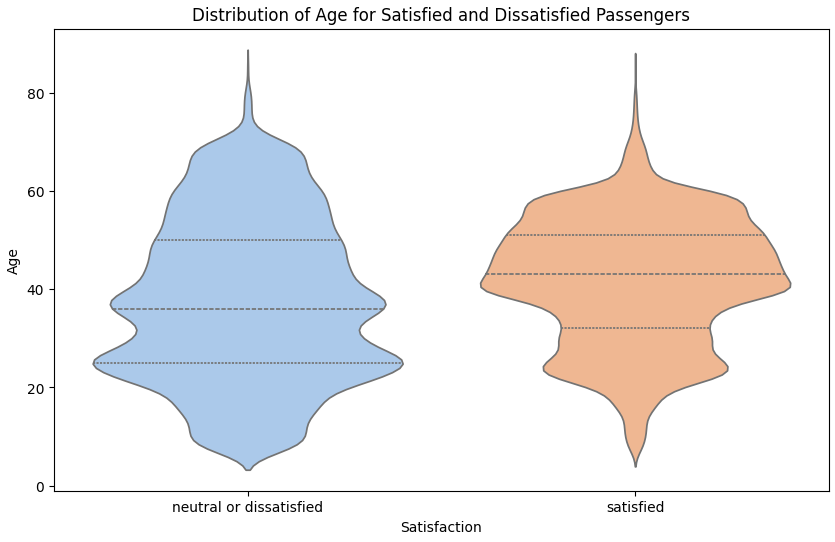
Customer Satisfaction V/S Gender

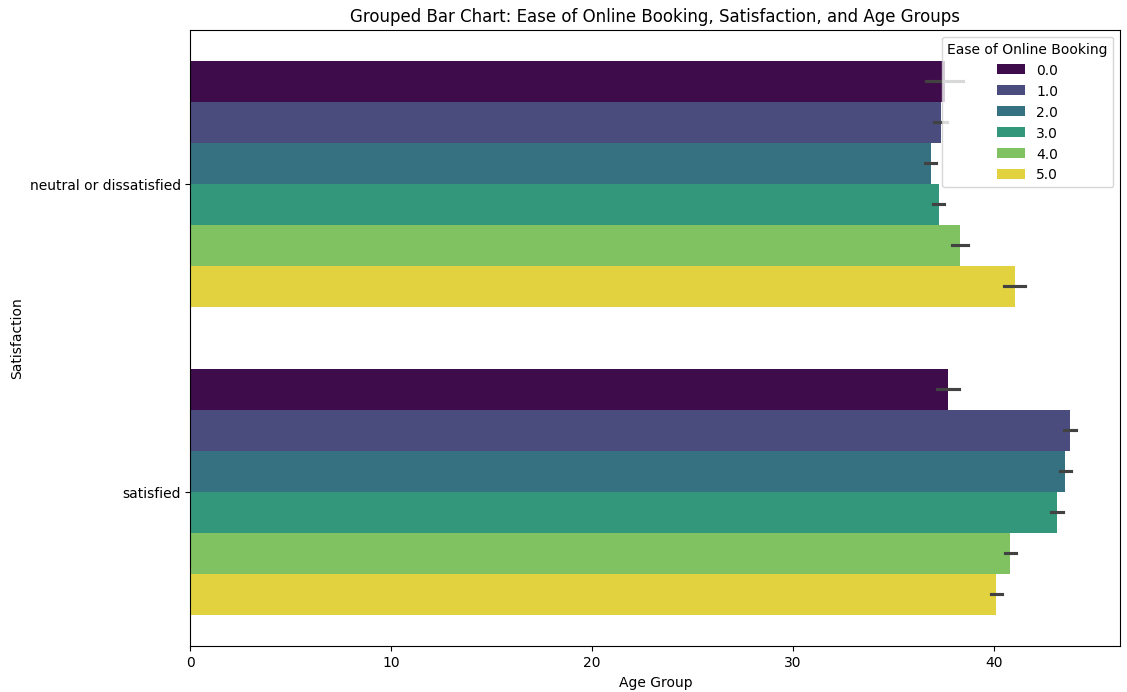


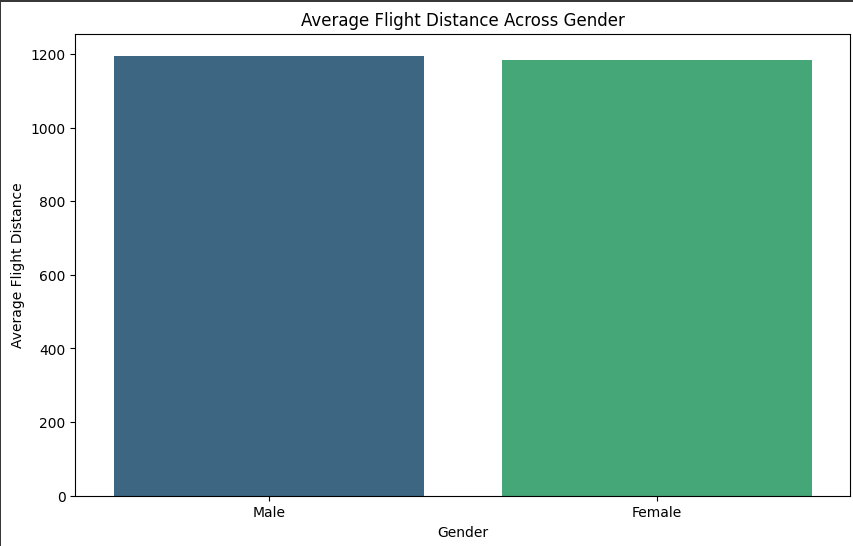
Customer Satisfaction V/S Customer Age



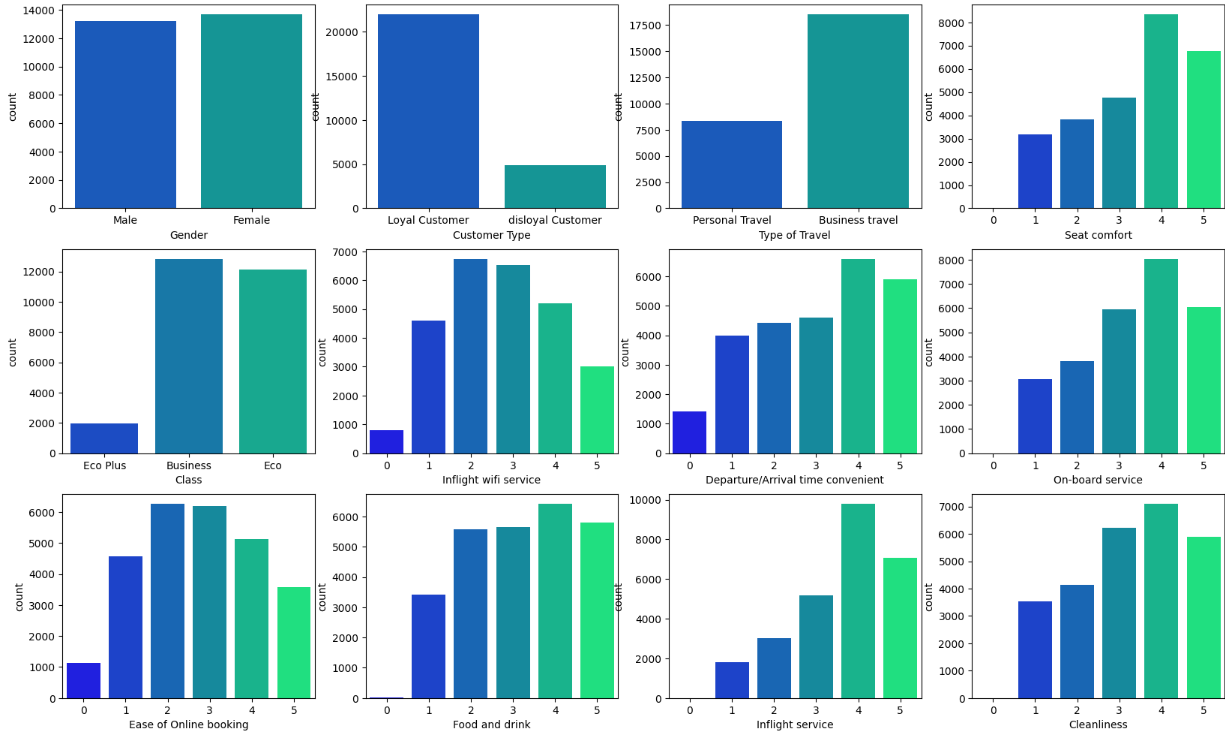
Relationship b/w departure delays and Customer Satisfaction

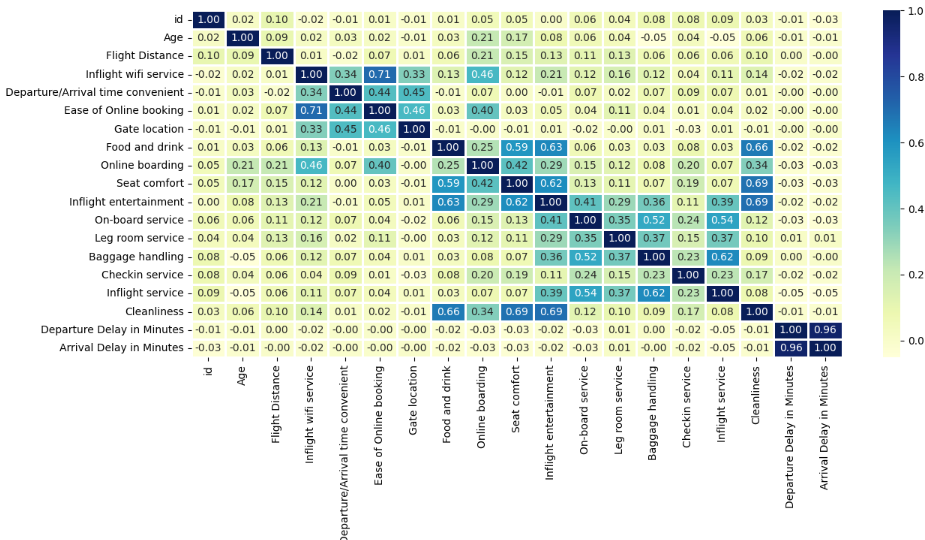


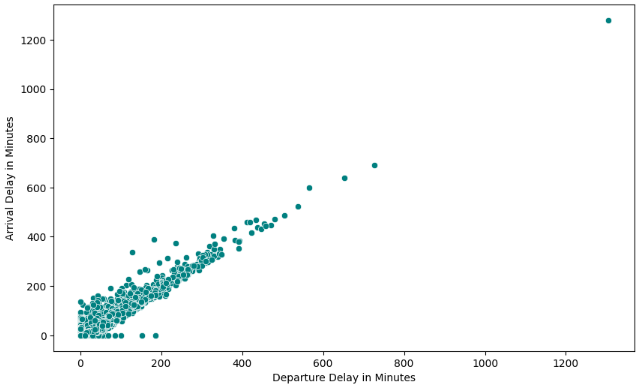








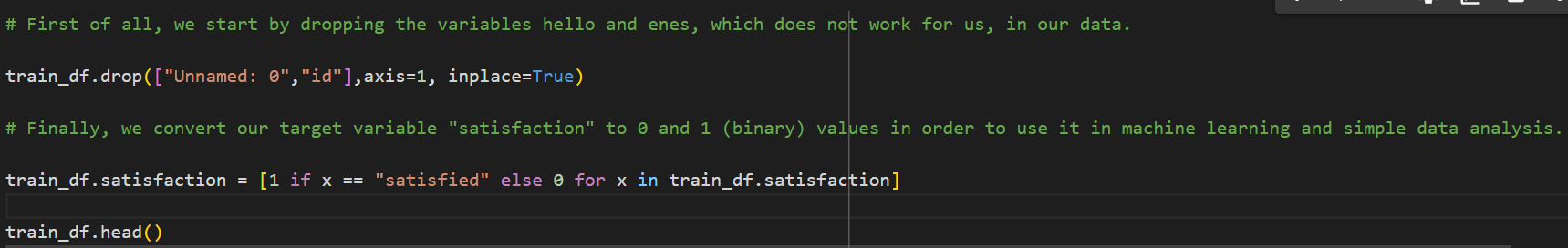




**Data Preprocessing**

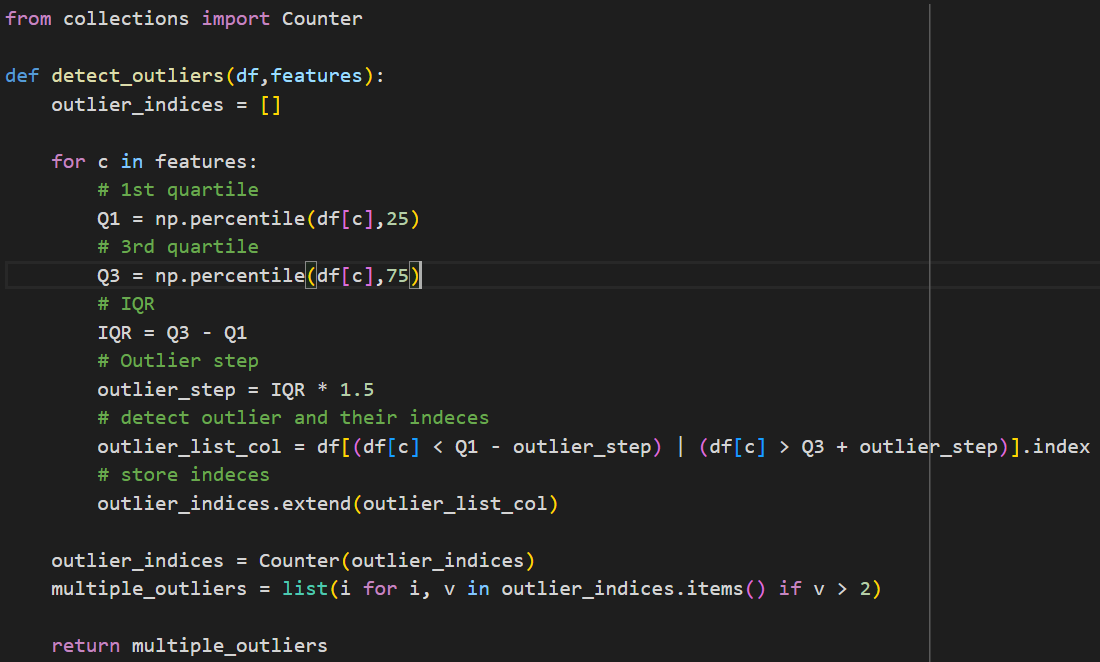
1. **Handling Missing Values:**

The code drops two columns ("Unnamed: 0" and "id") using drop() function. Then, it transforms the target variable "satisfaction" into binary values. Next, it standardizes column names for consistency. However, there's no explicit handling of missing values in the provided snippet. Additional steps such as imputation or deletion might be needed for robust data preprocessing.



1. **Handling Outliers:**

This function identifies outliers within the DataFrame `df` based on the provided `features`. It calculates quartiles and the Interquartile Range (IQR) for each feature, then identifies outlier indices falling beyond 1.5 times the IQR from the quartiles. Outlier indices occurring more than twice across features are stored and returned as `multiple\_outliers`.



**3. Handling Categorical Values:**

This code transforms categorical columns in `train\_df` into binary indicators, assigning 1 to rows matching the first unique value and 0 otherwise. Then, it converts the "satisfaction" column in `test\_df` to binary, representing "satisfied" as 1 and others as 0.

**ML – MODEL 1**

**Logistic Regression:**

This script implements logistic regression with hyperparameter tuning using GridSearchCV, evaluates the model's performance on test data, and plots the ROC curve with the calculated AUC:

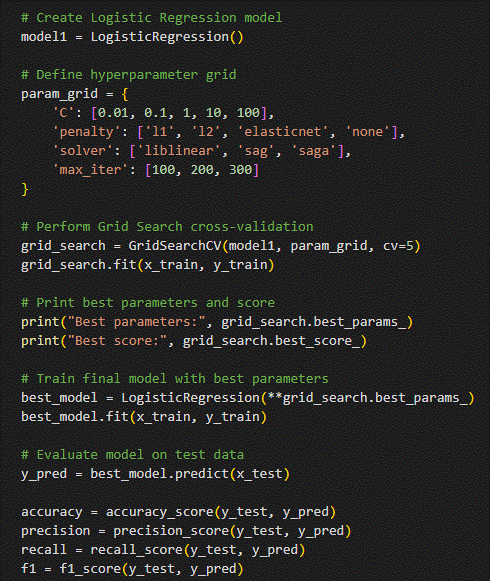
1. Import necessary libraries and modules: `RocCurveDisplay`, `LogisticRegression`, `GridSearchCV`, and evaluation metrics.

2. Create a Logistic Regression model (`model1`) and define the hyperparameter grid (`param\_grid`).

3. Perform Grid Search cross-validation (`grid\_search`) to find the best parameters.

4. Train the final model (`best\_model`) with the best parameters and evaluate its performance on the test data.

5. Plot the ROC curve and calculate the AUC.



**ML – MODEL 2**

**Decision Tree Classifier:**

1. Import the DecisionTreeClassifier from sklearn.tree.

2. Instantiate a DecisionTreeClassifier object as dt.

3. Define a parameter grid containing potential hyperparameters for tuning.

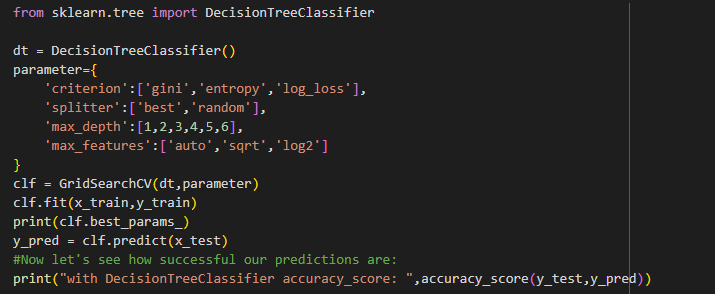
4. Instantiate a GridSearchCV object (clf) with the DecisionTreeClassifier and the defined parameter grid.

5. Fit the GridSearchCV object to the training data (x\_train, y\_train).

6. Print the best parameters found by the grid search.

7. Make predictions (y\_pred) on the test data using the best model found.

8. Evaluate the model's performance using the accuracy\_score function and print the result.



**ML – MODEL 3**

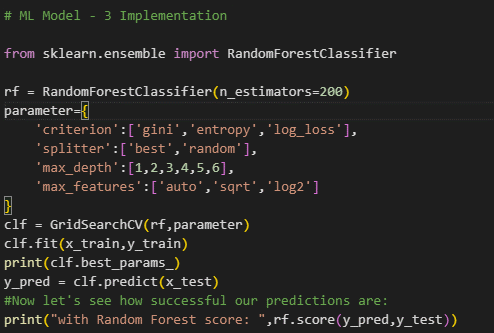
**Random Forest:**

- The code implements a Random Forest Classifier, a model that constructs multiple decision trees and aggregates their predictions for improved accuracy.

- GridSearchCV is employed for hyperparameter tuning, optimizing the model's performance by testing various parameter combinations.

- The best parameters found during the grid search are printed, aiding in understanding the model's configuration.

- Predictions are generated for the test data using the trained model.



**ML – MODEL 4**

**KNN:**

- The script initiates a K-Nearest Neighbors (KNN) Classifier, a straightforward algorithm used for classification tasks.

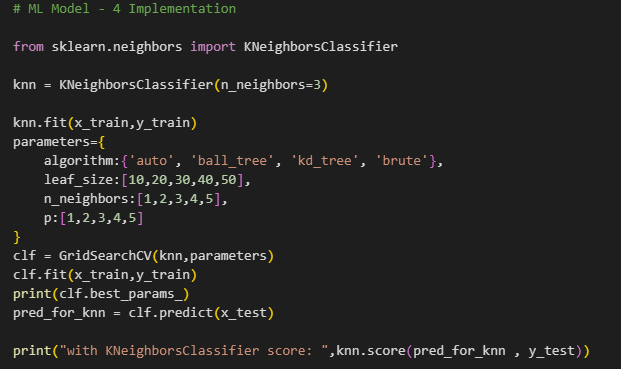
- Utilizing GridSearchCV, it explores various hyperparameter combinations to enhance the KNN model's performance.

- The script trains the model on the training data, learning patterns in the features and corresponding labels.

- Upon completion of the grid search, it prints the best parameters discovered, offering insight into the optimal configuration of the KNN model.

- Predictions are then generated for the test data, allowing assessment of the model's generalization capability.

- Finally, the accuracy score of the KNN model on the test data is printed, providing a quantitative measure of its predictive performance.



**Conclusion**

The implementation of machine learning models such as Logistic Regression, Decision Tree Classifier, Random Forest Classifier, and K-Nearest Neighbors (KNN) Classifier holds promise for predicting airline passenger satisfaction.

Through thorough data preprocessing, including feature engineering and handling missing values, and employing hyperparameter tuning techniques like GridSearchCV, these models can effectively learn from the provided data and make accurate predictions.

However, it's crucial to ensure proper evaluation metrics are employed to assess model performance accurately. By integrating these techniques into an ensemble model or deploying the best-performing individual model, airlines can enhance their understanding of passenger satisfaction factors and optimize service delivery.