King Saud University
College of Computer and Information Sciences
Information Technology department
IT 326: Data Mining

Course Project

Airline Passenger Satisfaction

Project final Report

Group#:	4	
Section #:	52846	
	Name	ID
Members:	shooq aldawssari	443200601
	Raseel Aldawish	443203036
	Leen Alotaibi	443200417
	Leen Alqahtani	443200591

1.Problem **4**
2.Datamining task **5**
Classification goal: 5
Clustering goal: 5
Problem statement: 5
3. Data **6**
The source: 6
General information of the dataset: 6
4. Data preprocessing **7**
Dealing with missing value 7
Description: 8
outliers analysis 9
Description: 9
Feature selection 9
Description: 11
Data Transformation 11
Normalization: 11
Discretization 13
5. Data Mining Techniques **17**
Classification 17
Python Packages and Methods: 17
Clustering 17
Python Packages and Methods 17
6. Evaluation and Comparison **19**
6.1-Classification 19
● Classification [70% training, 30% test]: 19
● Classification [80% training, 20% test]: 20
● Classification [90% training, 10% test]: 21
Comparison Criteria: 22
6.2 Clustering **23**
Interpretation of Results 24
1. Silhouette Scores: **24**
2. Total Within-Cluster Sum of Square (WSS): **24**
Best K Based on Metrics 24
Conclusion 24
7. Findings **26**
8- References **29**

1.Problem

Air travel is an essential mode of transportation globally, serving millions of passengers daily. However, ensuring passenger satisfaction remains a significant challenge for airlines. Understanding and addressing the factors that influence passenger satisfaction is crucial for improving the overall travel experience and maintaining competitiveness in the airline industry.

The Problem Statement:
The dataset on airline passenger satisfaction provides valuable insights into passenger preferences, experiences, and satisfaction levels with various aspects of air travel. Our goal is to analyze the dataset and develop a predictive model that can identify key factors affecting passenger satisfaction.

2.Datamining task

Our objective in gathering the airline satisfaction dataset is to compile pertinent and precise information regarding passenger experiences, preferences, and demographics during air travel. By analyzing attributes such as age, gender, flight class, travel purpose and more, we aim to achieve the following objectives:

Classification goal:

The primary aim is to classify passengers accurately, predicting whether they are satisfied, neutral or dissatisfied with airline services based on a combination of attributes. This classification enables early identification and prevention of passenger dissatisfaction, ultimately enhancing their experience and service quality.

Clustering goal:

Our goal in clustering is to uncover inherent patterns or groupings within the dataset that illuminate the factors influencing airline satisfaction. By categorizing passengers with similar preferences or experiences, clustering facilitates research, personalized service delivery, and strategic planning for enhancing overall satisfaction levels.

Problem statement:

Our project aims to leverage data mining techniques to assist researchers in early detection, risk assessment, and enhanced understanding of factors impacting airline satisfaction. To achieve this, we will:

- Conduct exploratory data analysis to gain insights into variable distribution and characteristics.
- Identify significant variables and their relationships with satisfaction levels through statistical analysis and visualization.
- Preprocess the dataset to handle missing values, outliers, and categorical variables, ensuring data quality and consistency.
- Employ appropriate data mining techniques, such as classification algorithms, to develop a predictive model for assessing airline satisfaction, thereby contributing to service enhancement and customer relationship management.

3. Data

The source:

click here

General information of the dataset:

Number of Attributes: 24 , Number of Object: 500 , Class Label: Satisfaction

Attribute name	Data type	Possible values
id	Integer	Numeric IDs
Gender	String	Male, Female
Customer Type	String	Loyal Customer, disloyal Customer
Age	Integer	Numeric ages
Type of Travel	String	Business travel, Personal Travel
Class	String	Eco, Business, Eco Plus
Flight Distance	Integer	Numeric distances 31 to 4983
Inflight wifi service	Integer	0 to 5
Departure/Arrival time convenient	Integer	0 to 5
Ease of Online booking	Integer	0 to 5
Gate location	Integer	0 to 5
Food and drink	Integer	0 to 5
Online boarding	Integer	0 to 5
Seat comfort	Integer	0 to 5
Inflight entertainment	Integer	0 to 5

On-board service	Integer	0 to 5
Leg room service	Integer	0 to 5
Baggage handling	Integer	0 to 5
Checkin service	Integer	0 to 5
Inflight service	Integer	0 to 5
Cleanliness	Integer	0 to 5
Departure Delay in Minutes	Integer	Numeric minutes 0-1592
Arrival Delay in Minutes	Integer	Numeric minutes 0-158000
satisfaction	String	satisfied, neutral or dissatisfied

4. Data preprocessing

We undertake data preprocessing to enhance the quality and integrity of our dataset before analysis. This involves addressing missing values, outliers, and inconsistencies through techniques such as data cleaning, feature selection, and data transformation. Missing values are filled or imputed, outliers are identified and managed, and features are selected to improve model performance. Additionally, data transformation techniques such as normalization and discretization are applied to ensure uniformity and comparability across variables. By meticulously preprocessing the data, we aim to improve the accuracy and reliability of any subsequent analyses or models built upon it.

Dealing with missing value

```
##Dealing with missing value

missing_values = dataset.isnull().sum()

plt.figure(figsize=(10, 6))
missing_values.plot(kind='bar', color='blue')
plt.xlabel('Columns')
plt.ylabel('Number of Missing Values')
plt.title('Missing Values in Each Column')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

output:

```
id                                  0
Gender                              0
Customer Type                       0
Age                                 0
Type of Travel                      0
Class                               0
Flight Distance                     0
Inflight wifi service               0
Departure/Arrival time convenient   0
Ease of Online booking              0
Gate location                       0
Food and drink                      0
Online boarding                     0
Seat comfort                        0
Inflight entertainment              0
On-board service                    0
Leg room service                    0
Baggage handling                    0
Checkin service                     0
Inflight service                    0
Cleanliness                         0
Departure Delay in Minutes          0
Arrival Delay in Minutes            0
satisfaction                        0
dtype: int64
```

Description:

After conducting a thorough check for missing values in the dataset, it was found that no missing values were present. The absence of missing values in our dataset is advantageous as it ensures that all necessary information is available for analysis, leading to more reliable results.

outliers analysis

```
## outliers analysis

columns_with_outliers = [
    'Gender', 'Customer Type', 'Age', 'Type of Travel',
    'Class', 'Flight Distance', 'Inflight wifi service',
    'Departure/Arrival time convenient', 'Ease of Online booking', 'Gate location',
    'Food and drink', 'Online boarding', 'Seat comfort',
    'Inflight entertainment', 'On-board service', 'Leg room service',
    'Baggage handling', 'Checkin service' , 'Inflight service', 'Cleanliness' ,
    'Departure Delay in Minutes' , 'Arrival Delay in Minutes', 'satisfaction'
]

# Calculate the mean for each column
mean_values = dfp[columns_with_outliers].mean(numeric_only=True)

# Calculate the absolute differences from the mean for each column
differences_from_mean = abs(dfp[columns_with_outliers] - mean_values)

# Find the index of the row with the largest difference from the mean across all columns
max_difference_index = differences_from_mean.sum(axis=1).idxmax()

# Remove the row with the largest difference from the mean
df_no_outlier = dfp.drop(max_difference_index)

# Display the original DataFrame and the DataFrame after removing the row with the larges
print("Original DataFrame:")
display(df)
print("\nDataFrame after removing the row with the largest difference from the mean:")
display(df_no_outlier)
```

Description:

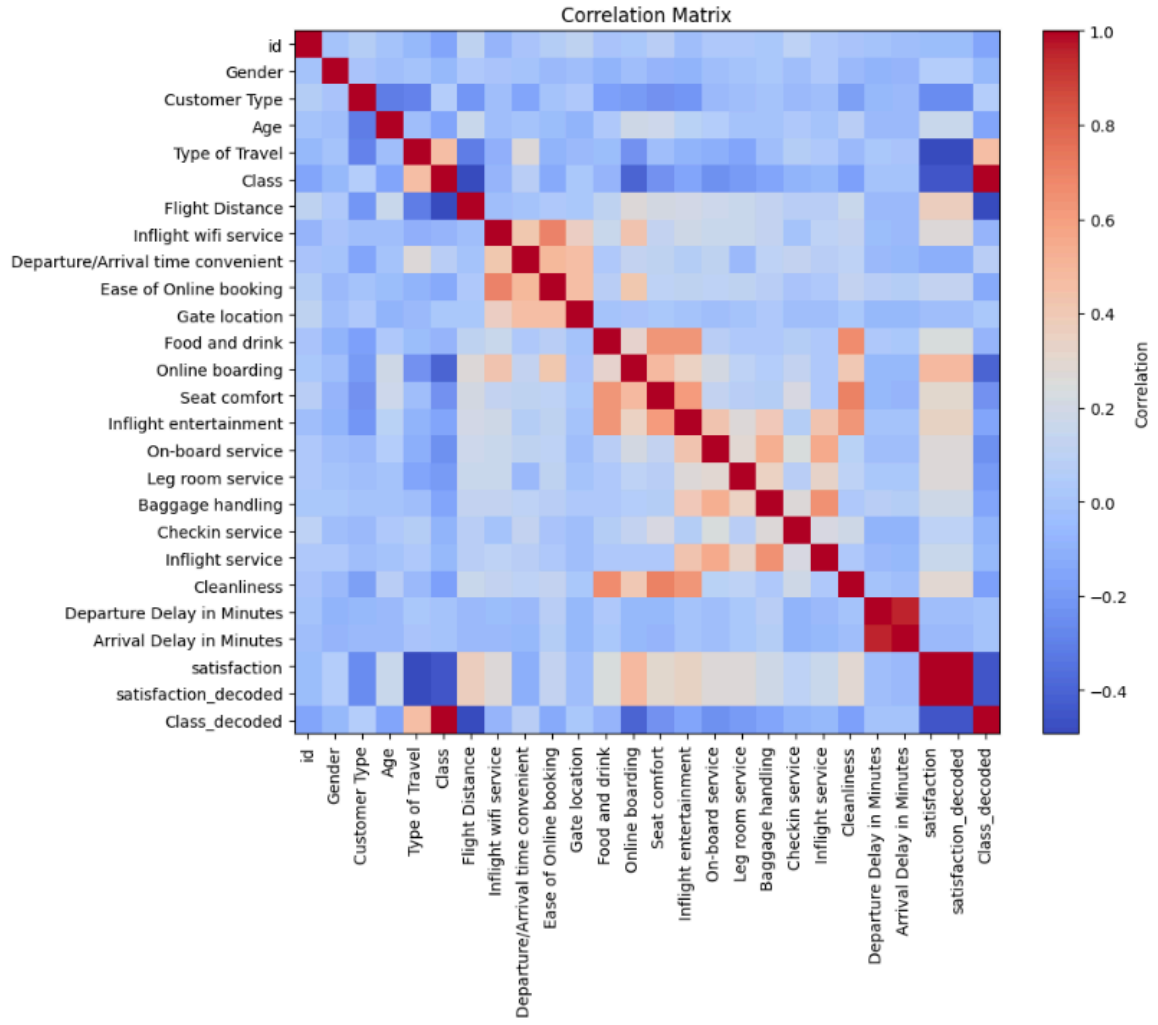
This code detects outliers in selected columns of a dataset, removes the row with the largest outlier, and displays the before and after dataframes.

Feature selection

```
# Calculate the correlation matrix for numeric columns
numeric_columns = dfp.select_dtypes(include=np.number)
correlation_matrix = numeric_columns.corr()

# Print correlation matrix
print("Correlation Matrix:")
display(correlation_matrix)
```

```
# Create a heatmap of the correlation matrix
plt.figure(figsize=(10, 8))
plt.imshow(correlation_matrix, cmap='coolwarm', interpolation='nearest')
plt.colorbar(label='Correlation')
plt.title("Correlation Matrix")
plt.xticks(range(len(correlation_matrix.columns)), correlation_matrix.columns, rotation=9
plt.yticks(range(len(correlation_matrix.columns)), correlation_matrix.columns)
plt.show()
```



Data after feature selection:

```
Updated Dataset after feature selection:
```

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Departure/Arrival time convenient	Ease of Online booking	Gate location	...	Inflight entertainment	On-board service	Leg room service	B h
0	19556	0	0	52	0	1	160	4	3	4	...	5	5	5	
1	90035	0	0	36	0	0	2863	1	3	1	...	4	4	4	
2	12360	1	1	20	0	1	192	0	2	4	...	2	4	1	
3	77959	1	0	44	0	0	3377	0	0	2	...	1	1	1	
4	36875	0	0	49	0	1	1182	3	4	3	...	2	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
495	1770	1	1	20	0	0	408	0	5	2	...	3	3	5	
496	70400	0	0	44	0	0	1162	4	4	4	...	4	4	4	
497	15078	0	0	33	0	0	239	1	1	1	...	4	4	4	
498	4927	0	0	60	0	0	1020	4	4	4	...	5	5	5	
499	53073	0	0	24	0	2	1208	5	5	5	...	4	2	2	

500 rows × 23 columns

Description:

The correlation matrix serves as a useful tool in the feature selection process by providing insights into the relationships between features and their relevance for predictive modeling tasks. We dropped the 'Arrival Delay in Minutes', 'Inflight WiFi Service', and 'Inflight Service' columns because the correlation threshold was greater than 0.6. Thus, we decided to remove one column from each highly correlated pair.

Data Transformation

Normalization:

Min-Max Normalization

```
columns_to_normalize = [6, 19]
column_data = dfp.iloc[:, columns_to_normalize]
scaler = MinMaxScaler()
normalized_data = scaler.fit_transform(column_data)
dfp.iloc[:, columns_to_normalize] = normalized_data
display(dfp)
```

- **Data before the normalization:**

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/Arrival time convenient	Ease of Online booking	...	Leg room service	Baggage handling	Checkin service	Infli ser
0	19556	0	0	52	0	1	160	5	4	3	...	5	5	2	
1	90035	0	0	36	0	0	2863	1	1	3	...	4	4	3	
2	12360	1	1	20	0	1	192	2	0	2	...	1	3	2	
3	77959	1	0	44	0	0	3377	0	0	0	...	1	1	3	
4	36875	0	0	49	0	1	1182	2	3	4	...	2	2	4	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
495	1770	1	1	20	0	0	408	5	0	5	...	5	5	3	
496	70400	0	0	44	0	0	1162	4	4	4	...	4	4	2	
497	15078	0	0	33	0	0	239	1	1	1	...	4	4	4	
498	4927	0	0	60	0	0	1020	4	4	4	...	5	5	4	
499	53073	0	0	24	0	2	1208	4	5	5	...	2	4	1	

Leg room service	Baggage handling	Checkin service	Inflight service	Cleanliness	Departure Delay in Minutes	Arrival Delay in Minutes	satisfaction	satisfaction_decoded	Class_decoded
5	5	2	5	5	50	44	1	1	1
4	4	3	4	5	0	0	1	1	0
1	3	2	2	2	0	0	0	0	1
1	1	3	1	4	0	6	1	1	0
2	2	4	2	4	0	20	1	1	1
...	...	...	...	...	...	...	...	...	...
5	5	3	4	3	0	0	1	1	0
4	4	2	4	2	0	4	1	1	0
4	4	4	4	1	0	0	1	1	0
5	5	4	5	5	5	0	1	1	0
2	4	1	4	4	1	0	1	1	2

- **Data after the normalization:**

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Departure/Arrival time convenient	Ease of Online booking	Gate location	...	Online boarding	Seat comfort	Inflight entertainment
0	19556	0	0	50-60	0	1	0.017511	4	3	4	...	4	3	5
1	90035	0	0	30-40	0	0	0.587764	1	3	1	...	4	5	4
2	12360	1	1	<20	0	1	0.024262	0	2	4	...	2	2	2
3	77959	1	0	40-50	0	0	0.696203	0	0	2	...	4	4	1
4	36875	0	0	40-50	0	1	0.233122	3	4	3	...	1	2	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
495	1770	1	1	<20	0	0	0.069831	0	5	2	...	5	3	3
496	70400	0	0	40-50	0	0	0.228903	4	4	4	...	1	2	4
497	15078	0	0	30-40	0	0	0.034177	1	1	1	...	5	1	4
498	4927	0	0	50-60	0	0	0.198945	4	4	4	...	4	4	5
499	53073	0	0	20-30	0	2	0.238608	5	5	5	...	4	4	4

On-board service	Leg room service	Baggage handling	Checkin service	Cleanliness	Departure Delay in Minutes	satisfaction
5	5	5	2	5	0.205761	1
4	4	4	3	5	0.000000	1
4	1	3	2	2	0.000000	0
1	1	1	3	4	0.000000	1
2	2	2	4	4	0.000000	1
...	...	...	...	...	...	...
3	5	5	3	3	0.000000	1
4	4	4	2	2	0.000000	1
4	4	4	4	1	0.000000	1
5	5	5	4	5	0.020576	1
2	2	4	1	4	0.004115	1

- **Description:**

The columns that are normalized are "Flight Distance" and "Departure Delay in Minutes" are chosen due to their original values have a big range and the for the result of same scale is benefical for future data mining techniques. Min-max normalization, also known as feature scaling, aims to scale the data to a fixed range, typically between 0 and 1. It ensures that all features have the same scale, which can be beneficial for algorithms that are sensitive to the scale of the features.

Discretization

```
dfp['Age'] = pd.to_numeric(df['Age'], errors='coerce')
bins = [0, 20, 30, 40, 50,60, 100]
labels = ['<20', '20-30', '30-40', '40-50','50-60', '>=60']
dfp['Age'] = pd.cut(df['Age'], bins=bins, labels=labels)
display(dfp)
```

- **Data before discretization**

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/Arrival time convenient	Ease of Online booking	...	Leg room service	Baggage handling	Checkin service	In se
0	19556	0	0	52	0	1	160	5	4	3	...	5	5	2	
1	90035	0	0	36	0	0	2863	1	1	3	...	4	4	3	
2	12360	1	1	20	0	1	192	2	0	2	...	1	3	2	
3	77959	1	0	44	0	0	3377	0	0	0	...	1	1	3	
4	36875	0	0	49	0	1	1182	2	3	4	...	2	2	4	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
495	1770	1	1	20	0	0	408	5	0	5	...	5	5	3	
496	70400	0	0	44	0	0	1162	4	4	4	...	4	4	2	
497	15078	0	0	33	0	0	239	1	1	1	...	4	4	4	
498	4927	0	0	60	0	0	1020	4	4	4	...	5	5	4	

- **Data after discretization**

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Departure/Arrival time convenient	Ease of Online booking	Gate location	...	Inflight entertainment	On-board service	Leg room service
0	19556	0	0	50-60	0	1	0.017511	4	3	4	...	5	5	5
1	90035	0	0	30-40	0	0	0.587764	1	3	1	...	4	4	4
2	12360	1	1	<20	0	1	0.024262	0	2	4	...	2	4	1
3	77959	1	0	40-50	0	0	0.696203	0	0	2	...	1	1	1
4	36875	0	0	40-50	0	1	0.233122	3	4	3	...	2	2	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
495	1770	1	1	<20	0	0	0.069831	0	5	2	...	3	3	5
496	70400	0	0	40-50	0	0	0.228903	4	4	4	...	4	4	4
497	15078	0	0	30-40	0	0	0.034177	1	1	1	...	4	4	4
498	4927	0	0	50-60	0	0	0.198945	4	4	4	...	5	5	5

- **Description:**

Discretization is the process of converting continuous data into categorical or discrete intervals. By discretizing the age column, the dataset becomes more manageable and interpretable, allowing for easier analysis and visualization. Discretizing the age column enhances the robustness and interpretability of the data analysis process.since the data for column "Age "is evenly distributed across a wide range, more bins was used ('<20', '20-30', '30-40', '40-50','50-60', '>=60')to capture the variation and few bins may oversimplify the data.

Raw Data:

	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/Arrival time convenient	Ease of Online booking	...	Inflight entertainment	On-board service	rc ser
0	19556	Female	Loyal Customer	52	Business travel	Eco	160	5	4	3	...	5	5	
1	90035	Female	Loyal Customer	36	Business travel	Business	2863	1	1	3	...	4	4	
2	12360	Male	disloyal Customer	20	Business travel	Eco	192	2	0	2	...	2	4	
3	77959	Male	Loyal Customer	44	Business travel	Business	3377	0	0	0	...	1	1	
4	36875	Female	Loyal Customer	49	Business travel	Eco	1182	2	3	4	...	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
495	1770	Male	disloyal Customer	20	Business travel	Business	408	5	0	5	...	3	3	
496	70400	Female	Loyal Customer	44	Business travel	Business	1162	4	4	4	...	4	4	
497	15078	Female	Loyal Customer	33	Business travel	Business	239	1	1	1	...	4	4	
498	4927	Female	Loyal Customer	60	Business travel	Business	1020	4	4	4	...	5	5	
499	53073	Female	Loyal Customer	24	Business travel	Eco Plus	1208	4	5	5	...	4	2	

Leg room service	Baggage handling	Checkin service	Inflight service	Cleanliness	Departure Delay in Minutes	Arrival Delay in Minutes	satisfaction
5	5	2	5	5	50	44	satisfied
4	4	3	4	5	0	0	satisfied
1	3	2	2	2	0	0	neutral or dissatisfied
1	1	3	1	4	0	6	satisfied
2	2	4	2	4	0	20	satisfied
...	...	...	...	...	...	...	...
5	5	3	4	3	0	0	satisfied
4	4	2	4	2	0	4	satisfied
4	4	4	4	1	0	0	satisfied
5	5	4	5	5	5	0	satisfied
2	4	1	4	4	1	0	satisfied

Data after preprocessing:

id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/Arrival time convenient	Ease of Online booking	...	Leg room service	Baggage handling	Checkin service	Inflight service
19556	0	0	52	0	1	160	5	4	3	...	5	5	2	5
90035	0	0	36	0	0	2863	1	1	3	...	4	4	3	4
12360	1	1	20	0	1	192	2	0	2	...	1	3	2	2
77959	1	0	44	0	0	3377	0	0	0	...	1	1	3	1
36875	0	0	49	0	1	1182	2	3	4	...	2	2	4	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1770	1	1	20	0	0	408	5	0	5	...	5	5	3	4
70400	0	0	44	0	0	1162	4	4	4	...	4	4	2	4
15078	0	0	33	0	0	239	1	1	1	...	4	4	4	4
4927	0	0	60	0	0	1020	4	4	4	...	5	5	4	5
53073	0	0	24	0	2	1208	4	5	5	...	2	4	1	4

Ease of Online booking	...	Leg room service	Baggage handling	Checkin service	Inflight service	Cleanliness	Departure Delay in Minutes	Arrival Delay in Minutes	satisfaction	satisfaction_decoded	Class_decoded
3	...	5	5	2	5	5	50	44	1	1	1
3	...	4	4	3	4	5	0	0	1	1	0
2	...	1	3	2	2	2	0	0	0	0	1
0	...	1	1	3	1	4	0	6	1	1	0
4	...	2	2	4	2	4	0	20	1	1	1
...	...	...	...	...	...	...	...	...	...	...	...
5	...	5	5	3	4	3	0	0	1	1	0
4	...	4	4	2	4	2	0	4	1	1	0
1	...	4	4	4	4	1	0	0	1	1	0
4	...	5	5	4	5	5	5	0	1	1	0
5	...	2	4	1	4	4	1	0	1	1	2

5. Data Mining Techniques

Standard Python Packages and Methods used during data mining are Pandas (import pandas as pd) and NumPy (import numpy as np). These packages are essential for data manipulation and preprocessing tasks. Matplotlib (import matplotlib.pyplot as plt): Matplotlib is used for data visualization, including plotting decision trees and elbow plots.Scikit-learn (from sklearn import): Scikit-learn is a powerful library for machine learning tasks in Python. It implements decision trees, K-Means clustering, evaluation metrics, and preprocessing techniques.

Classification

Classification is a supervised learning task because the model is trained on labeled data, where each instance is associated with a known class label. Decision trees are a popular algorithm for classification tasks. They recursively split the feature space into regions based on feature values, with each split maximizing the homogeneity of the target variable (class labels). Attribute selection measures like Information Gain (Entropy) and Gini Index are used to determine the best feature to split on at each decision tree node.

Python Packages and Methods:

- DecisionTreeClassifier from sklearn.tree: This class implements the decision tree algorithm for classification.
- plot_tree and export_text from sklearn.tree: These functions help visualize and interpret the decision tree.
- Evaluation metrics, such as accuracy, precision, and recall, can be calculated using methods from sklearn.metrics.

Clustering

Clustering is an unsupervised learning task because the model is trained on unlabeled data and aims to discover inherent patterns or structures within the data.K-Means is a popular algorithm for clustering tasks. It partitions the dataset into K clusters by iteratively assigning data points to the nearest cluster centroid and updating the centroids based on the mean of the data points in each cluster.Various metrics such as Silhouette Coefficient, Elbow Method, Calinski-Harabasz Score, and Davies-Bouldin Score can be used to assess the quality and coherence of the clusters generated by the algorithm.

Python Packages and Methods

- KMeans from sklearn.cluster: This class implements the K-Means clustering algorithm.

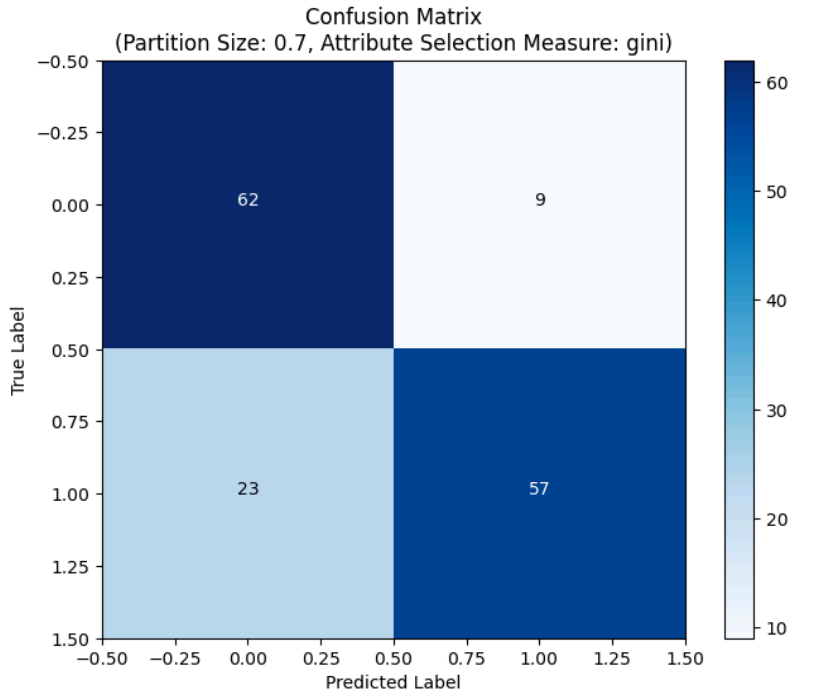
- Evaluation metrics like Silhouette Coefficient, Elbow Method (total within-cluster sum of square), Calinski-Harabasz Score, and Davies-Bouldin Score can help assess the clustering quality.
- silhouette_score from sklearn.metrics: Computes the silhouette coefficient, which measures the compactness and separation of clusters.
- calinski_harabasz_score and davies_bouldin_score from sklearn.metrics: These metrics provide additional measures of cluster quality.

6. Evaluation and Comparison

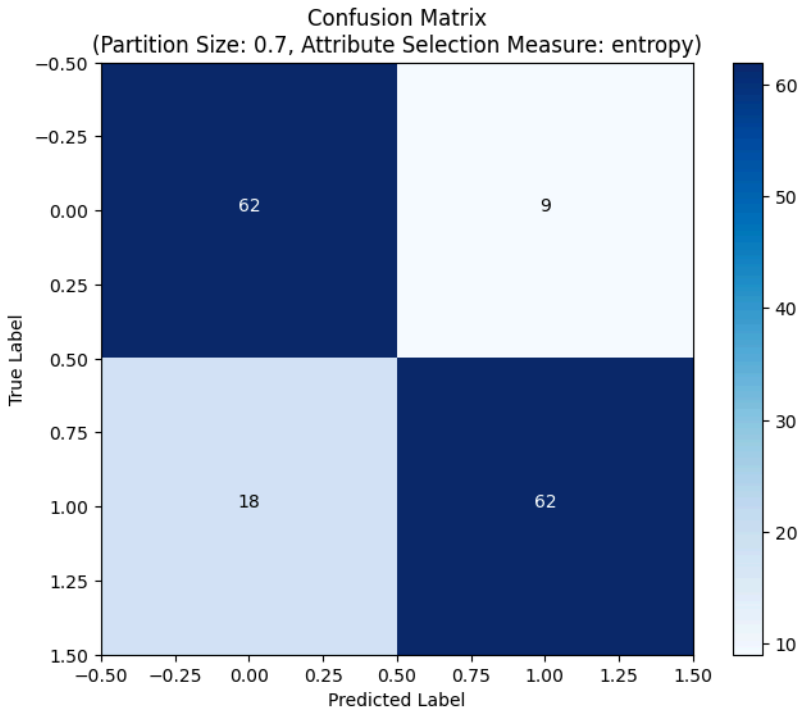
6.1-Classification

- Classification [70% training, 30% test]:

Figure(1) (Confusion matrix):

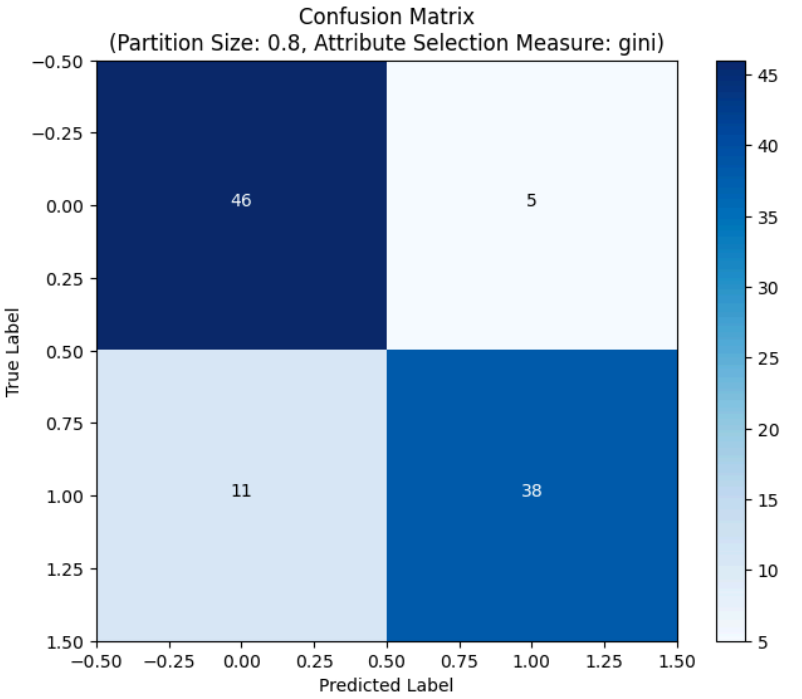


Figure(2) (Confusion matrix):

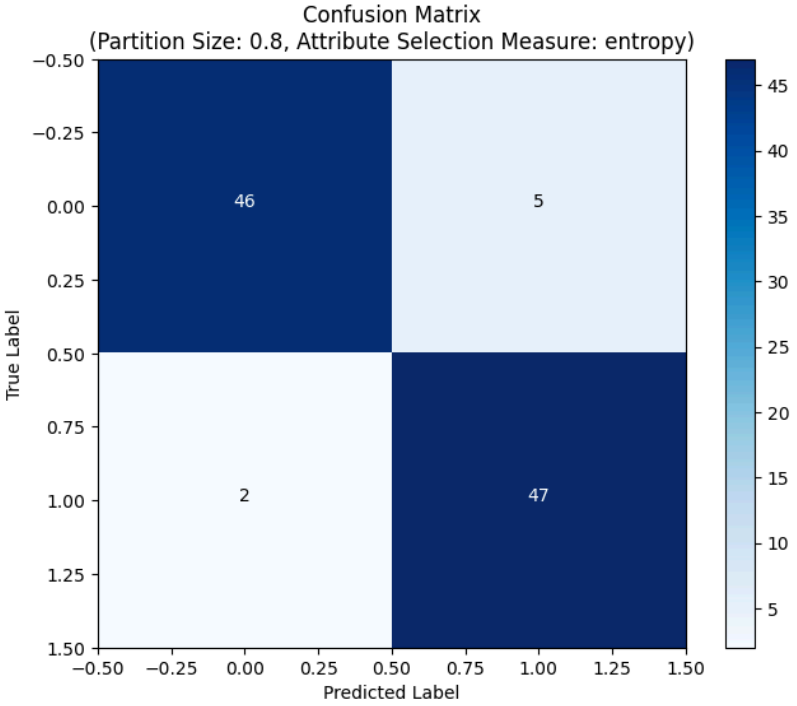


- Classification [80% training, 20% test]:

Figure(1) (Confusion matrix):

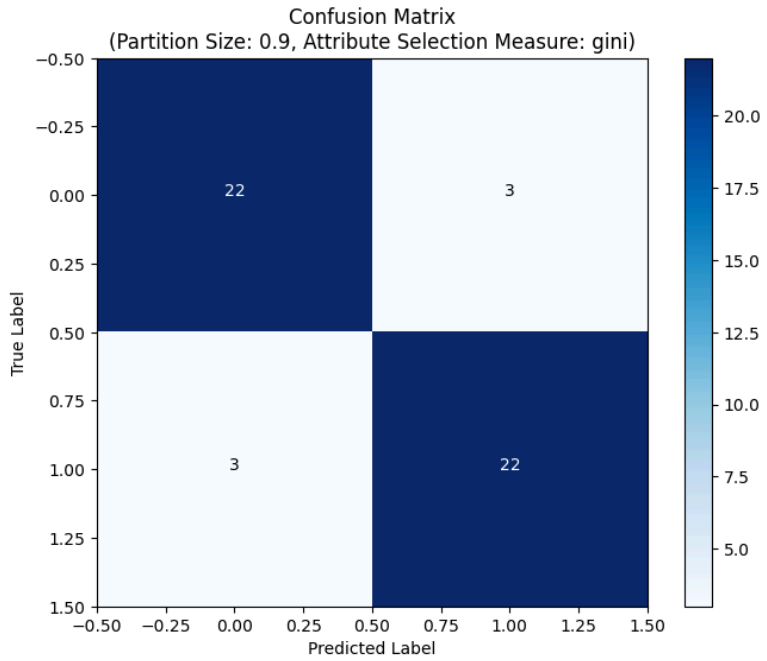


Figure(2) (Confusion matrix):

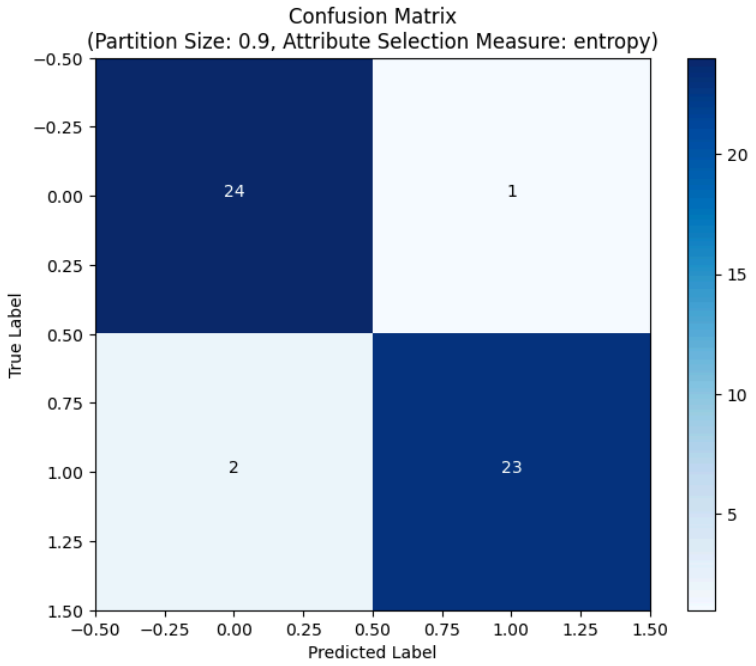


- Classification [90% training, 10% test]:

Figure(1) (Confusion matrix):



Figure(2) (Confusion matrix):



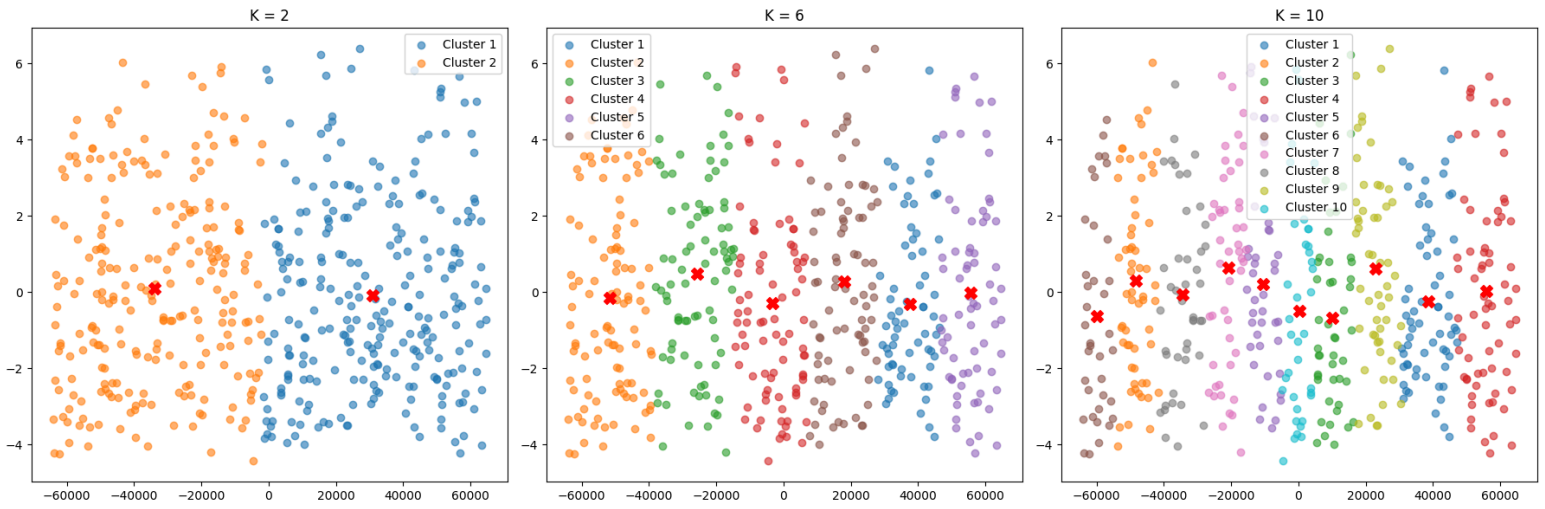
Comparison Criteria:

Partition		90% training set, 10% testing test (BEST)	80% training set, 20% testing test	70% training set, 30% testing test
Accuracy	Gini	88%	84%	78.8%
	Information Gain (entropy)	94%	93%	82.1%

6.2 Clustering

	K=2	K=6	K=10
Average Silhouette Width	Cluster 1: 0.61 Cluster 2: 0.64	Cluster 1: 0.52 Cluster 2: 0.65 Cluster 3: 0.51 Cluster 3: 0.51 Cluster 4: 0.47 Cluster 5: 0.63 Cluster 6: 0.51	Cluster 1: 0.45 Cluster 2: 0.58 Cluster 3: 0.43 Cluster 4: 0.62 Cluster 5: 0.49 Cluster 6: 0.70 Cluster 7: 0.56 Cluster 8: 0.50 Cluster 9: 0.55 Cluster 10: 0.58
Overall Average Silhouette Score	0.62	0.55	0.55
Total Within-cluster Sum of Square	177380907860.33	20189419676.24	7521376169.53

The optimal number of clusters based on majority rule is: 2



Interpretation of Results

1. Silhouette Scores:
    - K=2: We got silhouette scores of 0.61 and 0.64 for the two clusters, which are quite high. This suggests that the two clusters are well separated and cohesive.
    - K=6: The scores range from 0.47 to 0.65, indicating variability in how well different clusters are separated from each other.
    - K=10: The scores range from 0.43 to 0.70, showing even more variability. Some clusters are well defined, while others are less so.

2. Total Within-Cluster Sum of Square (WSS):
    - This value decreases as *K* increases, which is expected because more clusters mean that each cluster's centroid is likely closer to its members, reducing the overall sum of squares.
    - The sharp drop from *K*=2 to *K*=6 suggests significant improvement in cluster compactness with an increase in *K*.

Best K Based on Metrics

K=2 seems to be the best choice if we consider the balance between a high overall average silhouette score and a reasonably low WSS. The clusters at *K*=2 are not only fewer but also quite distinct and compact compared to higher *K* values.

Conclusion

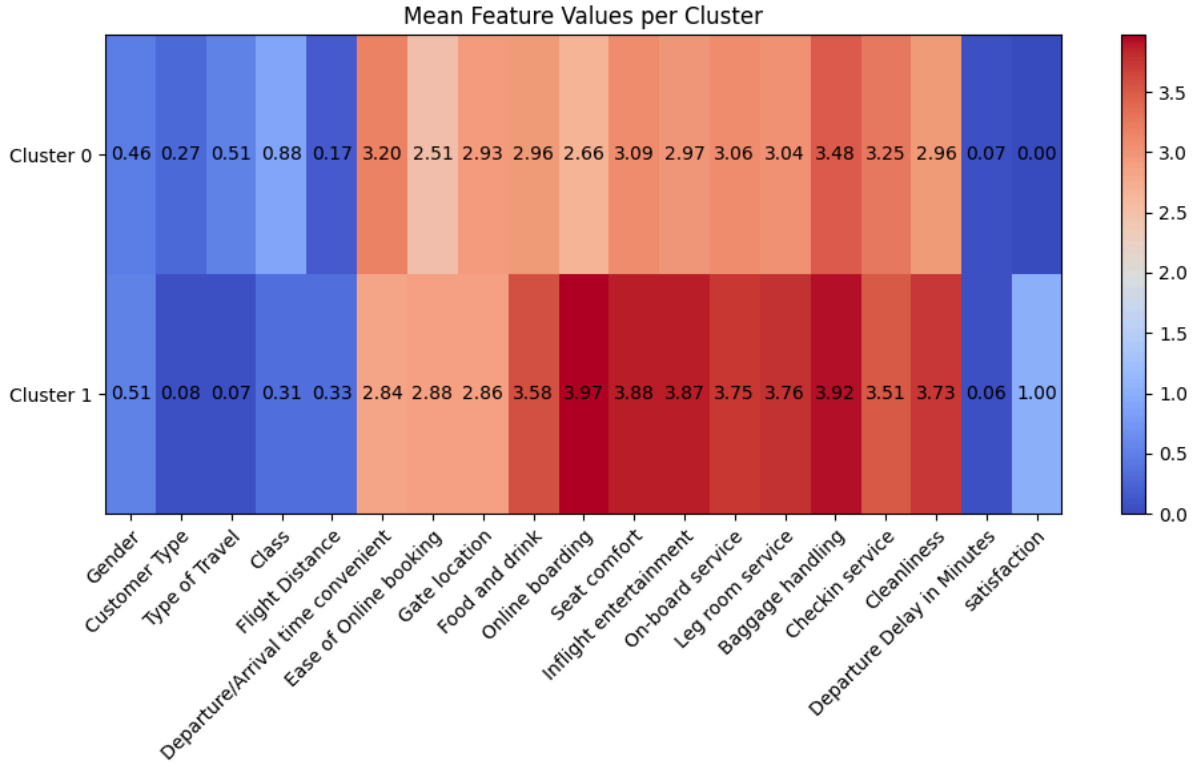
Based on the metrics:

- Best Algorithm for Each Partition: For *K*=2, the algorithm performs best as it provides the highest silhouette scores and a strong overall average silhouette score, indicating well-separated and cohesive clusters.
- Overall Best: Among all the trials, *K*=2 stands out as the optimal number of clusters. It combines high silhouette scores with significant reduction in WSS, suggesting effective clustering with just two groups.

Cluster Visualization and Interpretation.

After determining K=2 as the optimal number of clusters based on metrics like silhouette scores and Within-Cluster Sum of Squares (WSS), we applied Principal Component Analysis (PCA) to visualize the clusters in a 2D scatter plot. This plot clearly highlighted the well-separated and cohesive nature of the two clusters.

We then used a heatmap to analyze the mean values of features across each cluster, revealing distinct differences in areas like online boarding, leg room service, and cleanliness. This dual approach confirmed the separation between the clusters while providing actionable insights into key factors driving passenger satisfaction.



7. Findings

At the outset, our endeavor focused on leveraging a dataset derived from airline passenger surveys with the aim of predicting the likelihood of passenger satisfaction levels, thereby empowering airlines to implement targeted measures to enhance travelers' experiences and improve their overall quality of airline service.

To ensure the efficacy, accuracy, and optimal performance of our predictive model, we employed a series of preprocessing techniques tailored to refine the dataset. These techniques encompassed various strategies designed to enhance data integrity and quality. Utilizing visualization methods such as box plots and histograms, we gained valuable insights into the distribution and characteristics of the dataset, thereby affecting our preprocessing decisions. Through this process, we systematically addressed issues such as null values, missing data points, and outliers, which could potentially skew our predictive outcomes. Additionally, we executed data transformations, including normalization and discretization, to standardize attribute scales and facilitate seamless data handling throughout the predictive modeling phase.

Subsequently, we embarked on the application of data mining methodologies, classification and clustering techniques, to extract meaningful patterns and insights from the dataset. In the realm of classification, we opted for the decision tree algorithm to show us important information related to our problem, big factors affecting the satisfaction level . To optimize model performance, we conducted experiments with varying sizes of training and testing datasets, meticulously evaluating the impact of different partitioning schemes on model construction and performance evaluation.

We will conduct comprehensive testing of the data mining technique, Decision tree classification, using three partitions. In each partition, it will be tested by two attribute selection measures (IG (entropy) Gini index).The ‘best’ decision tree model identifies which features (attributes) are most important in determining the target, understand how different combinations of features contribute to predicting the target variable, and reveal patterns or relationships within the dataset that are not immediately apparent from descriptive statistics or exploratory data analysis.

We find that This(90% training, 10% testing) partition size achieved the highest accuracy rate, indicating its superiority in predicting passenger satisfaction levels. Therefore, the 90% training, 10% testing partition is deemed the most effective for this predictive modeling task especially with information gain(entropy) as the attribute selection measure. From the decision tree, we can assume online boarding service highly impacts the overall satisfaction of the passenger; this feature is the root of the tree. Another fact to conclude by using the

‘best’ decision tree we chose is that Passengers with a higher Ease of Online Booking rating (> 3.50) are less likely to be satisfied (class: 1), particularly if they rate Baggage Handling lower (<= 2.50) or On-board Service lower (<= 2.50). Another insight extracted from the decision tree is For passengers with lower Ease of Online Booking ratings (<= 3.50), satisfaction is influenced by cleanliness ratings. Those who rate Cleanliness higher (> 3.50) are more likely to be satisfied (class: 1), highlighting the importance of cleanliness in overall passenger satisfaction.

We find that the K=2 clustering configuration achieved the best overall performance, providing a strong combination of high silhouette scores and a significant reduction in total Within-Cluster Sum of Squares (WSS). The silhouette scores of 0.61 and 0.64 across the two clusters indicate well-separated and cohesive groups. In contrast, the scores for K=6 (0.47 to 0.65) and K=10 (0.43 to 0.70) show more variability and a mix of well-defined and less distinct clusters.

The sharp decline in WSS from K=2 to K=6 reveals notable improvements in cluster compactness with an increasing number of clusters. However, the K=2 clustering solution maintains a balance between a high overall average silhouette score and a relatively low WSS. As such, the K=2 partition emerges as the optimal solution, offering two clearly defined and compact clusters while providing the best combination of metrics for this clustering task.

From Heatmap Visualization we found that members of cluster 0 have an overall satisfaction rating of 0, indicating general dissatisfaction. Moderate ratings for online boarding (2.66) and onboard service (2.97) suggest that these aspects are not very positively received. Similarly, moderate ratings for cleanliness (3.25) may contribute to the lack of overall satisfaction. The leg room service (3.06) and seat comfort (2.97) ratings imply potential areas for improvement, while a relatively low rating for gate location (2.51) may indicate inconvenience or accessibility issues.

Members of cluster 1 are highly satisfied, as evidenced by their satisfaction rating of 1. Higher ratings for online boarding (3.97) and onboard service (3.88) indicate that efficient boarding processes and in-flight service positively influence satisfaction. High ratings for leg room service (3.75) and seat comfort (3.87) further contribute positively to passenger satisfaction. Baggage handling has a strong rating of 3.92, reflecting efficient and satisfactory luggage service. Ratings for food and drink (3.58) are higher than in Cluster 0, suggesting better quality catering. Higher cleanliness ratings (3.73) indicate that a clean environment correlates with overall passenger satisfaction.

The clear differences between the clusters show that passengers value cleanliness, leg room, and the efficiency of online boarding. Airlines can focus on improving these factors to boost passenger satisfaction, especially in areas where Cluster 0 falls short. Addressing

accessibility issues, such as gate location, and enhancing onboard services like food, drink, and entertainment could elevate satisfaction levels within Cluster 0. By implementing targeted strategies based on these insights, airlines can develop a comprehensive and customer-focused improvement plan.

8- References

1. TeejMahal20. Airline Passenger Satisfaction. Kaggle. https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction/data

2. DataCamp. (Feb 2023). Decision Tree Classification in Python. DataCamp. https://www.datacamp.com/tutorial/decision-tree-classification-python

3. scikit-learn developers. K-means Clustering with Silhouette Analysis. scikit-learn. https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html