Data Collection and Preprocessing Phase

Date	24 April 2024
Team ID	team-739704
Project Title	Identifying Airline Passenger Satisfaction Using Machine Learning
Maximum Marks	6 Marks

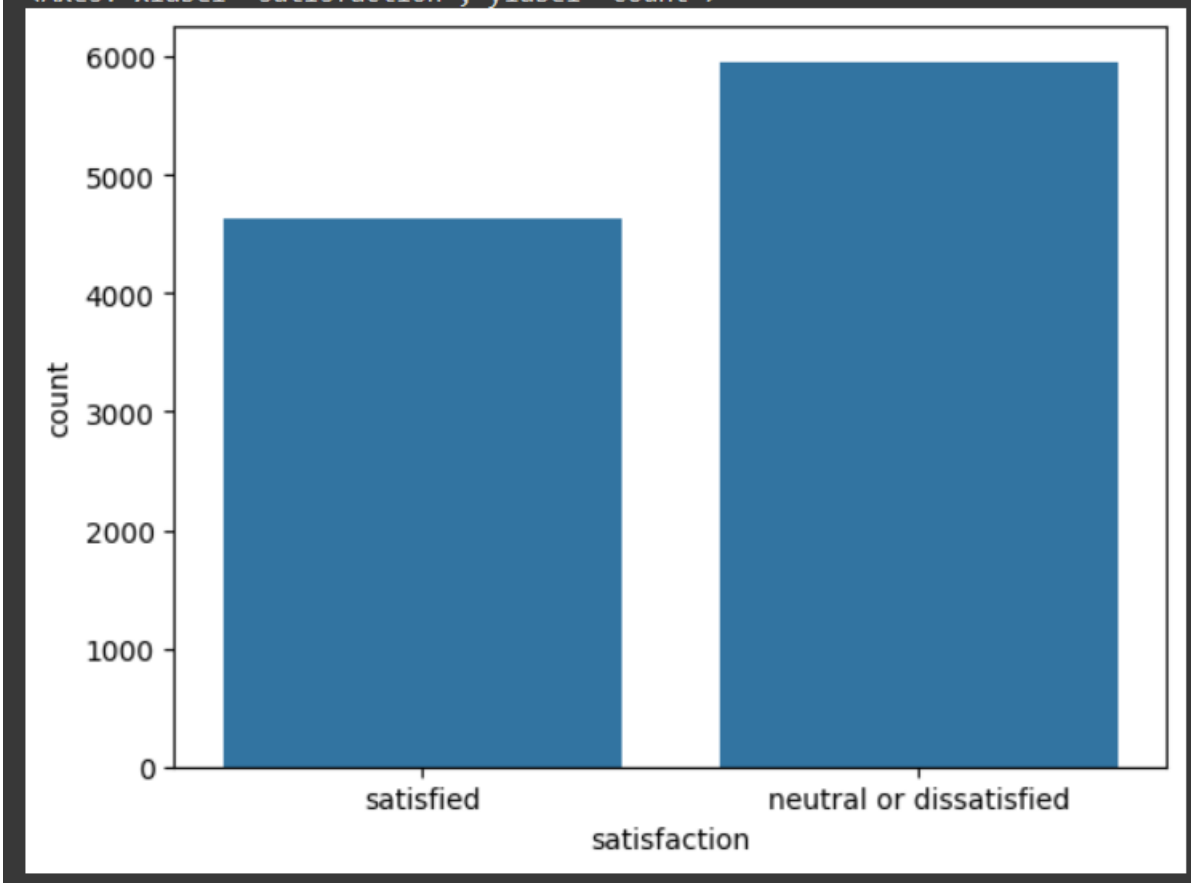
Data Exploration and Preprocessing Template

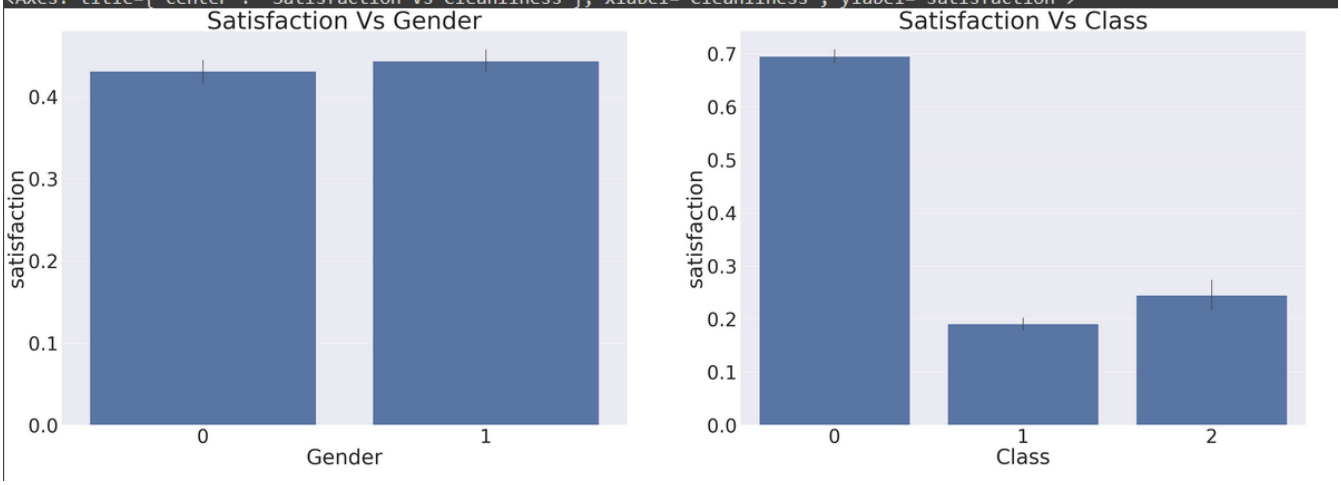
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	`data.describe()`

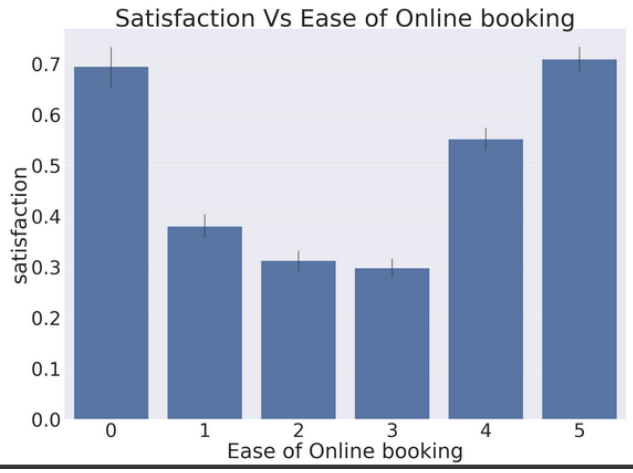
	Gender	Age	Class	Flight Distance	Inflight wifi service	Departure/ Arrival time convenient	Ease of Online booking	Gate location	1
count	10580.000000	10580.000000	10580.000000	10580.0	10580.000000	10580.000000	10580.000000	10580.000000	
mean	0.497448	39.798677	0.592439	0.0	2.723913	3.059735	2.755577	2.976560	
std	0.500017	15.144005	0.622437	0.0	1.337066	1.534992	1.409658	1.281976	
min	0.000000	7.000000	0.000000	0.0	0.000000	0.000000	0.000000	1.000000	
25%	0.000000	27.000000	0.000000	0.0	2.000000	2.000000	2.000000	2.000000	
50%	0.000000	40.000000	1.000000	0.0	3.000000	3.000000	3.000000	3.000000	
75%	1.000000	51.000000	1.000000	0.0	4.000000	4.000000	4.000000	4.000000	
max	1.000000	85.000000	2.000000	0.0	5.000000	5.000000	5.000000	5.000000	

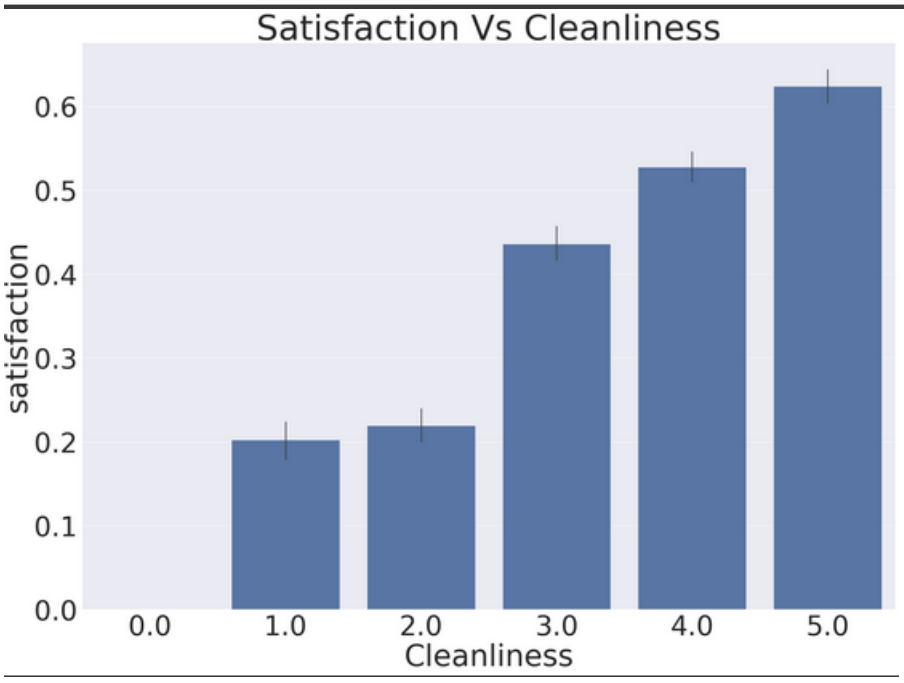
8 rows × 21 columns

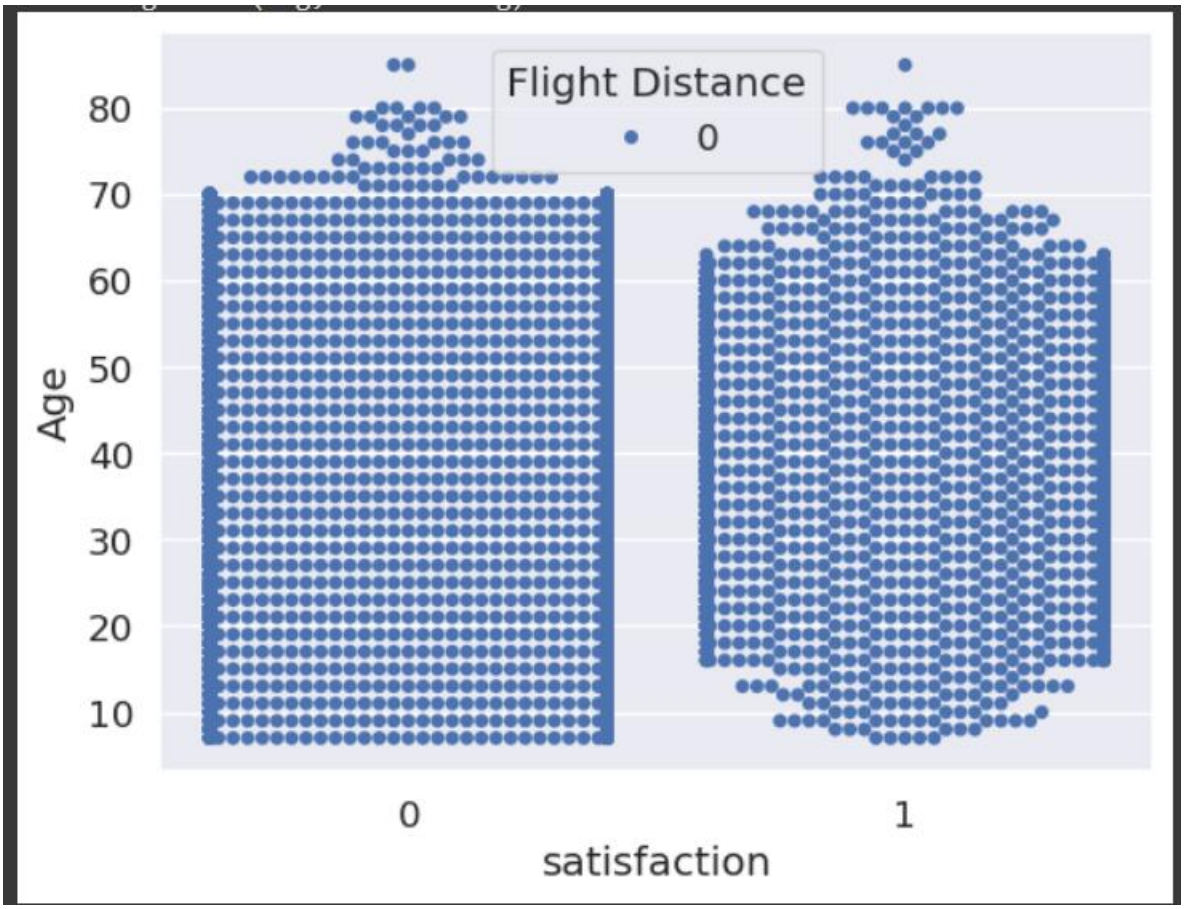
Univariate Analysis	```sns.countplot(x="satisfaction",data=data)```
	```<Axes: xlabel='satisfaction', ylabel='count'>```
	 

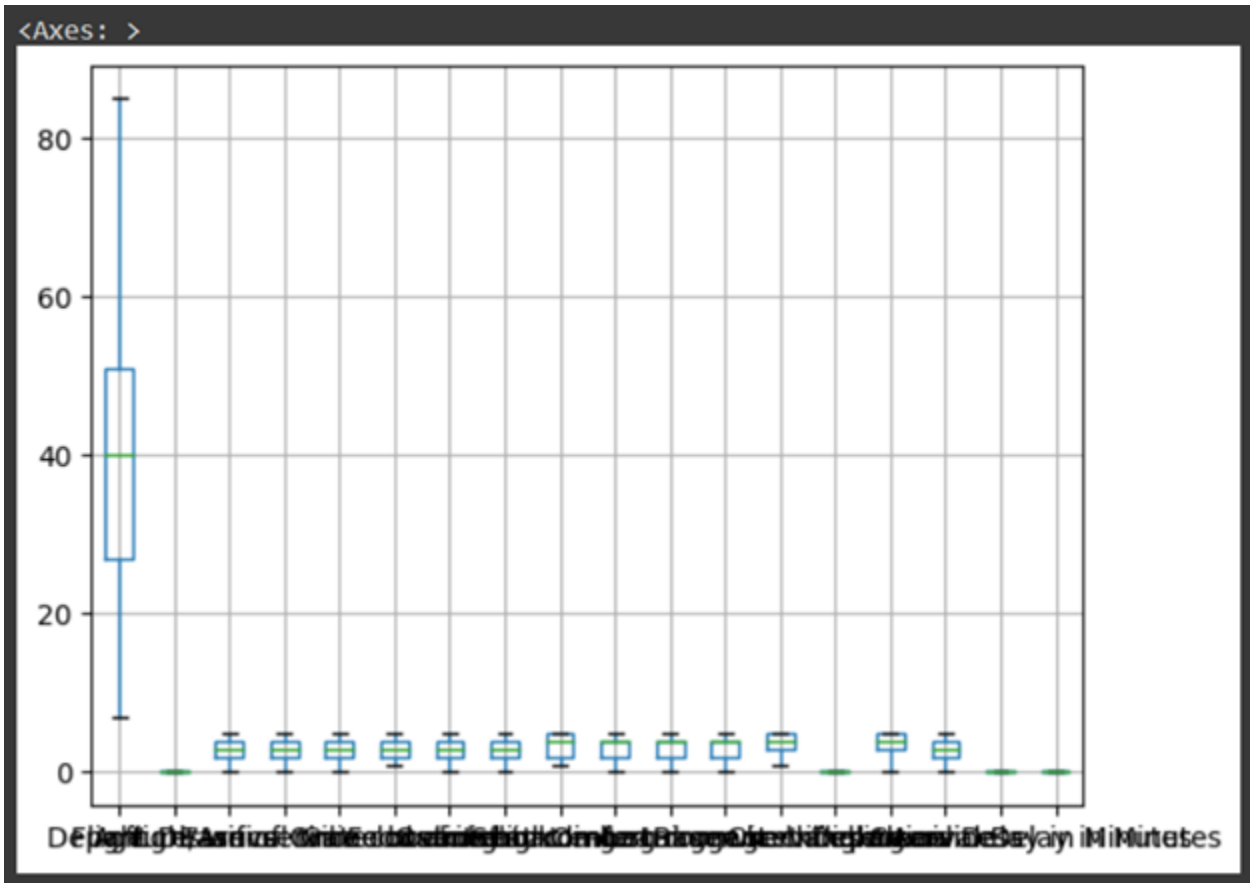
Bivariate Analysis	```<Axes: title={'center': 'Satisfaction Vs Cleanliness'}, xlabel='Cleanliness', ylabel='satisfaction'>``` 


Satisfaction Vs On-board service
satisfaction
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
1
2
3
4
5
On-board service


Satisfaction Vs Ease of Online booking
satisfaction
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
1
2
3
4
5
Ease of Online booking


Satisfaction Vs Cleanliness
satisfaction
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.0
1.0
2.0
3.0
4.0
5.0
Cleanliness

Multivariate Analysis	
Outliers and Anomalies	``` data['Flight Distance']=np.where(data['Flight Distance']>0.1,0,data['Flight Distance']) data['Flight Distance']=np.where(data['Flight Distance']<0.1,0,data['Flight Distance']) data['Checkin service']=np.where(data['Checkin service']>0.1,0,data['Checkin service']) data['Checkin service']=np.where(data['Checkin service']<0.1,0,data['Checkin service']) data['Departure Delay in Minutes']=np.where(data['Departure Delay in Minutes']>0.1,0,data['Departure Delay in Minutes']) data['Departure Delay in Minutes']=np.where(data['Departure Delay in Minutes']<0.1,0,data['Departure Delay in Minutes']) data['Arrival Delay in Minutes']=np.where(data['Arrival Delay in Minutes']>0.1,0,data['Arrival Delay in Minutes']) data['Arrival Delay in Minutes']=np.where(data['Arrival Delay in Minutes']<0.1,0,data['Arrival Delay in Minutes']) ```



Data Preprocessing Code Screenshots

Loading Data

```
data=pd.read_csv("/content/test.csv")
```

```
data.head()
```

	Unnamed: 0	id	Gender	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/ Arrival time convenient	Ease of Online booking
0	0	19556	Female	52	Business travel	Eco	160	5	4	3
1	1	90035	Female	36	Business travel	Business	2863	1	1	3
2	2	12360	Male	20	Business travel	Eco	192	2	0	2
3	3	77959	Male	44	Business travel	Business	3377	0	0	0
4	4	36875	Female	49	Business travel	Eco	1182	2	3	4

5 rows × 24 columns

Handling Null values	`data.dropna(inplace=True)`
	
	`data.isnull().sum()`
	
	Gender 0
	Age 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

Data Transformation	`from sklearn.preprocessing import LabelEncoder`
	
	`le=LabelEncoder()`
	
	`data['Gender'] = le.fit_transform(data['Gender'])`
	`data['Class'] = le.fit_transform(data['Class'])`
	`data['satisfaction'] = le.fit_transform(data['satisfaction'])`

Save Processed Data	`import pickle`
	`import warnings`
	
	`with open("mod.pkl","wb") as f:`
	`  pickle.dump(random,f)`