PROJECT REPORT

Assignment Project Report: Exploratory Factor Analysis
NAME: RISHI RAJ SINGH
AI AND ML B-4

Problem Statement:

Factor analysis is a useful technique to find latent factors that can potentially describe multiple attributes, which is sometimes very useful for dimensionality reduction. Use the Airline Passenger Satisfaction dataset to perform factor analysis. (Use only the columns that represent the ratings given by the passengers, only 14 columns). Choose the best features possible that helps in dimensionality reduction, without much loss in information.

Prerequisites:
- Software: Python 3

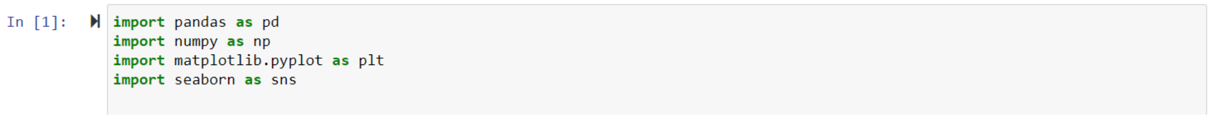
Tools:

- Pandas
- Numpy
- Matplotlib
- Seaborn

Dataset:
Airline Passenger Satisfaction dataset

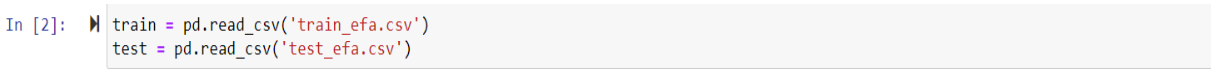
Implementation:

- Load all required libraries



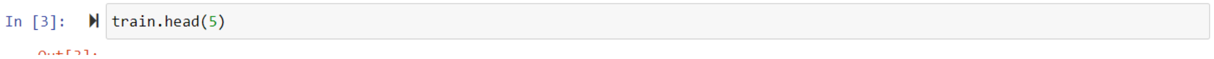
```
In [1]:  import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         import seaborn as sns
```

Loading Dataset

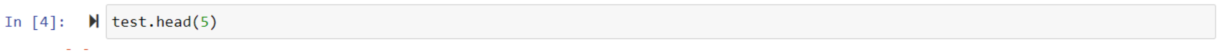


```
In [2]:  train = pd.read_csv('train_efa.csv')
         test = pd.read_csv('test_efa.csv')
```

Playing With Data

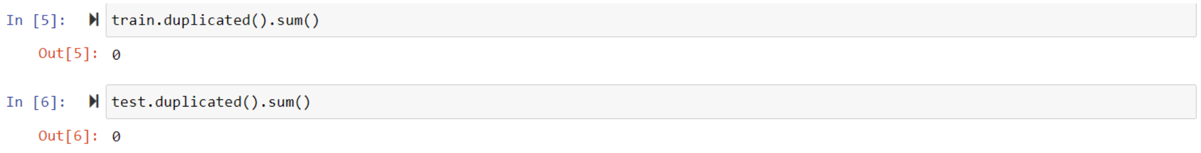


```
In [3]:  train.head(5)
   Out[3]:
```



```
In [4]:  test.head(5)
```

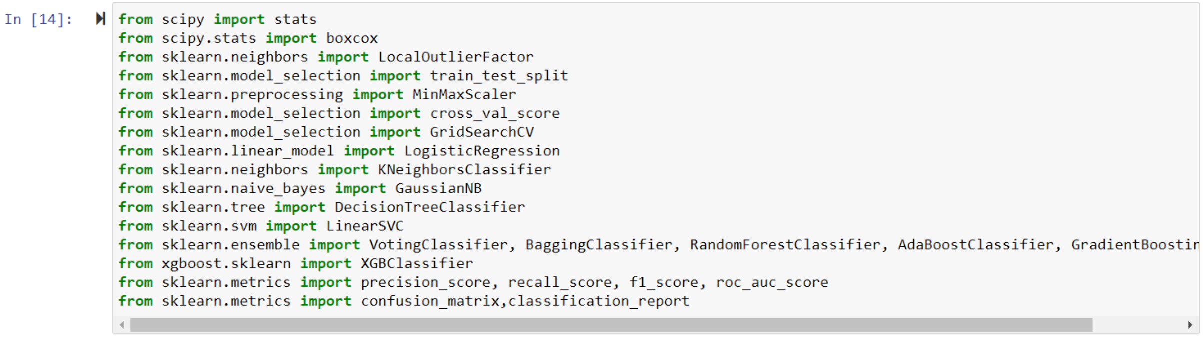
Checking Information



```
In [5]:  train.duplicated().sum()
  Out[5]: 0

In [6]:  test.duplicated().sum()
  Out[6]: 0
```

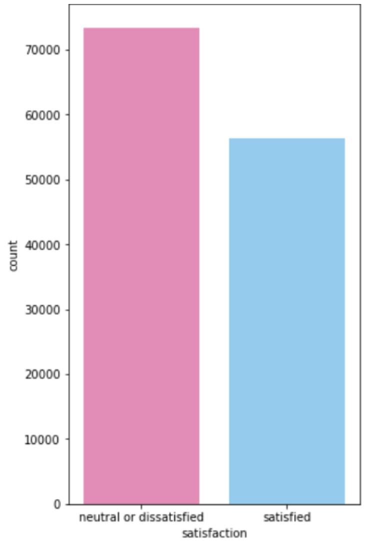
Importing ML Libraries

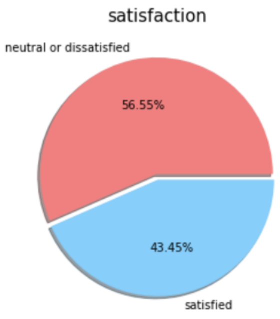


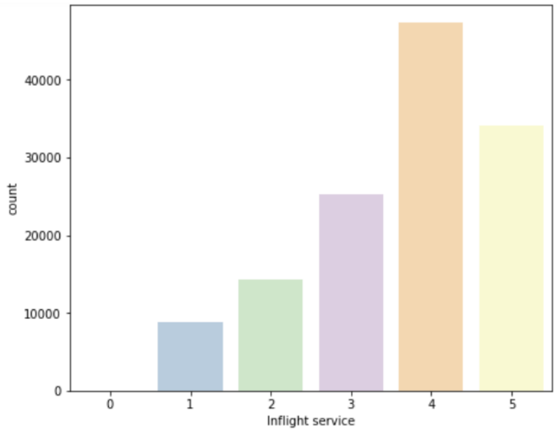
In [14]:

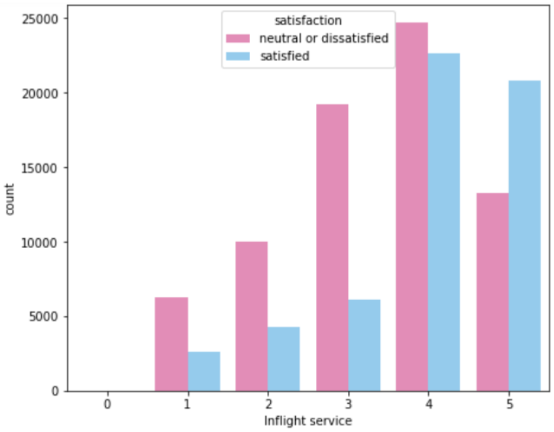
```
from scipy import stats
from scipy.stats import boxcox
from sklearn.neighbors import LocalOutlierFactor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import LinearSVC
from sklearn.ensemble import VotingClassifier, BaggingClassifier, RandomForestClassifier, AdaBoostClassifier, GradientBoostir
from xgboost.sklearn import XGBClassifier
from sklearn.metrics import precision_score, recall_score, f1_score, roc_auc_score
from sklearn.metrics import confusion_matrix,classification_report
```

- Co-relating Data and Visualising IT

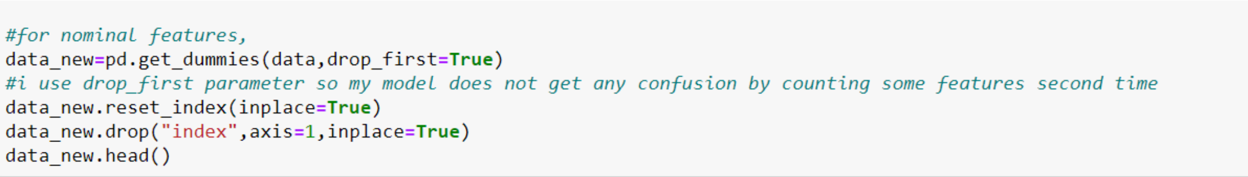




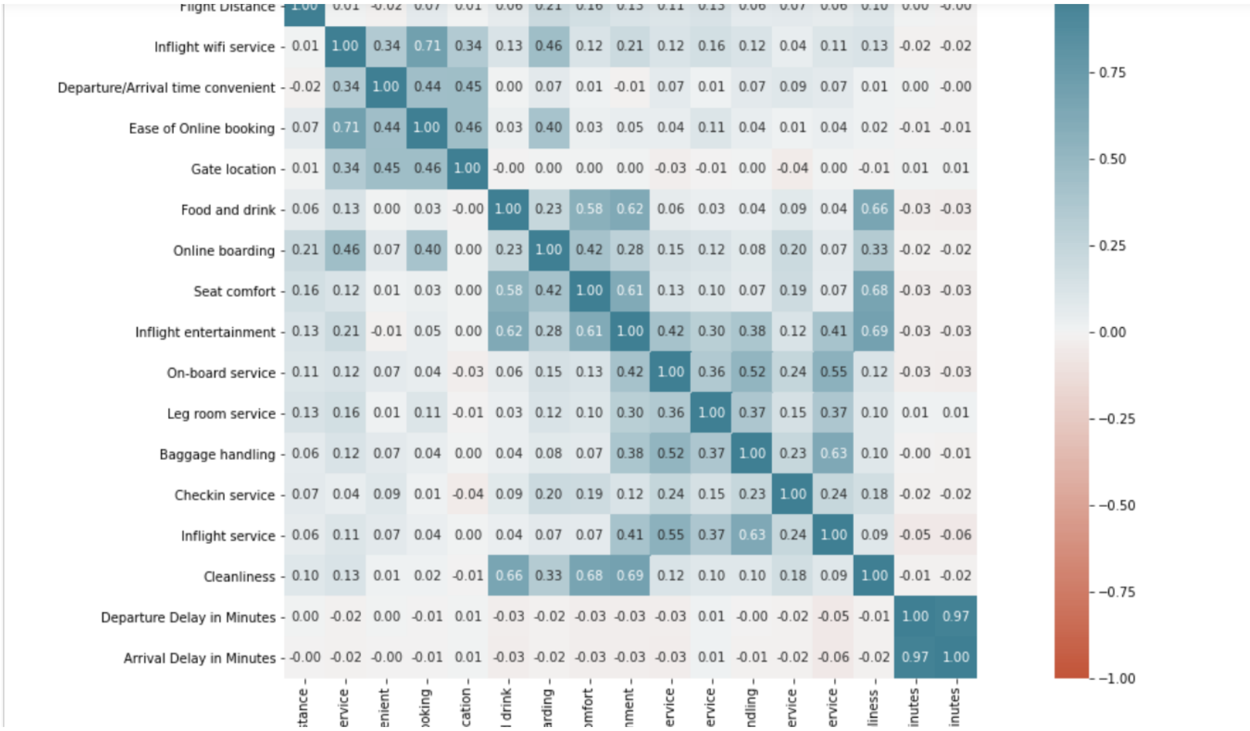




- Features And Confusion Matrix



```
#for nominal features,
data_new=pd.get_dummies(data,drop_first=True)
#i use drop_first parameter so my model does not get any confusion by counting some features second time
data_new.reset_index(inplace=True)
data_new.drop("index",axis=1,inplace=True)
data_new.head()
```



- Otsu Method

- Result After Otsu Method

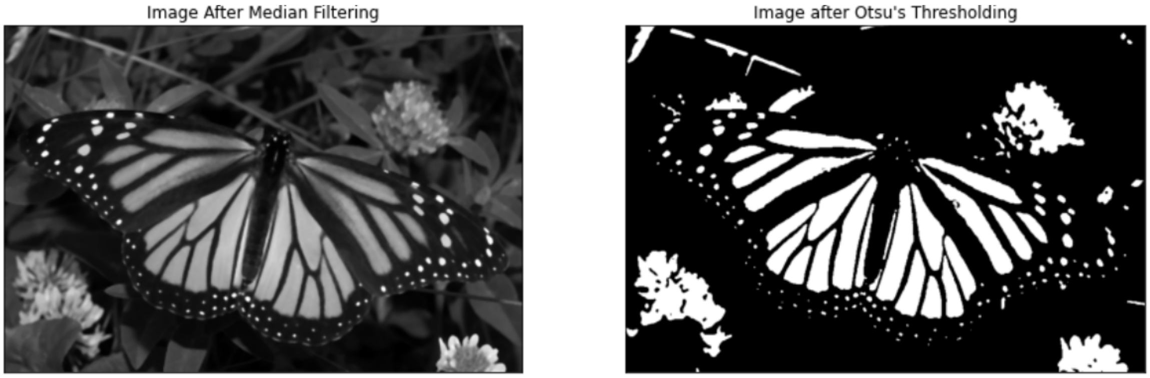


Image After Median Filtering

Image after Otsu's Thresholding