Govardhan Seshasayee

Airline Data Project Report

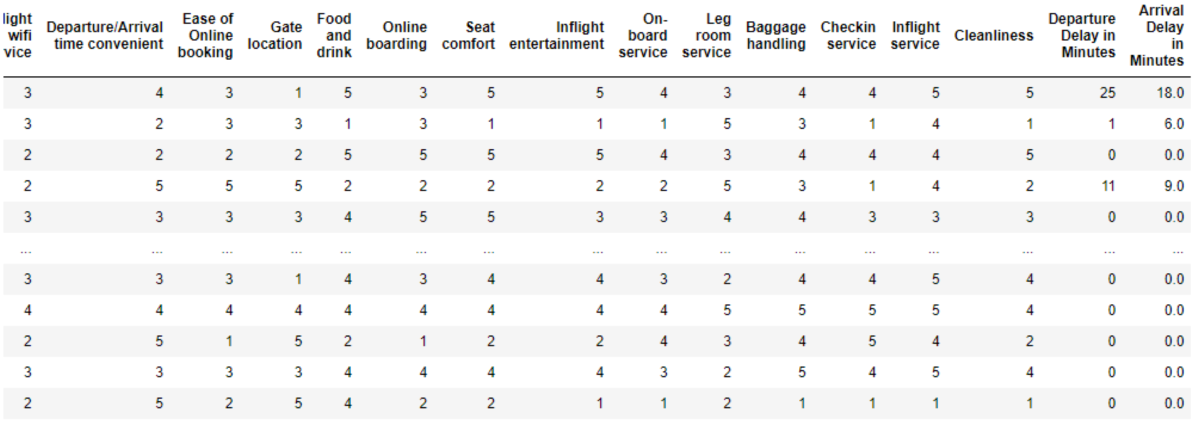
The Airline Passenger satisfaction data is considered in performing the task using EFA.

In multivariate statistics, exploratory factor analysis is a statistical method used to uncover the underlying structure of a relatively large set of variables. EFA is a technique within factor analysis whose overarching goal is to identify the underlying relationships between measured variables.

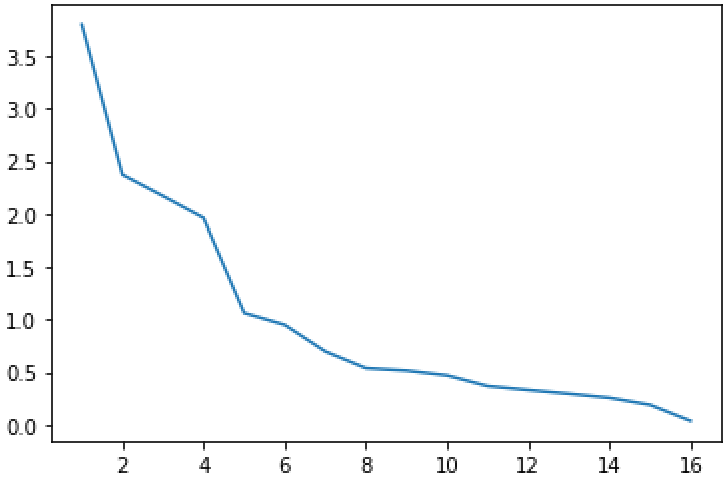
The necessary libraries are imported and the dataset given as a csv file is also imported which is already segregated into test and train data set the data set is verified and then concatenated and the missing data are checked. The bartlett sphericity is calculated using the factor analyzer library and then using that the Keiser Meyer Olkin value is obtained to see if the sample value is adequate. This value indicate if the sample is adequate or not having an value less then T indicates that the sample is inadequate.

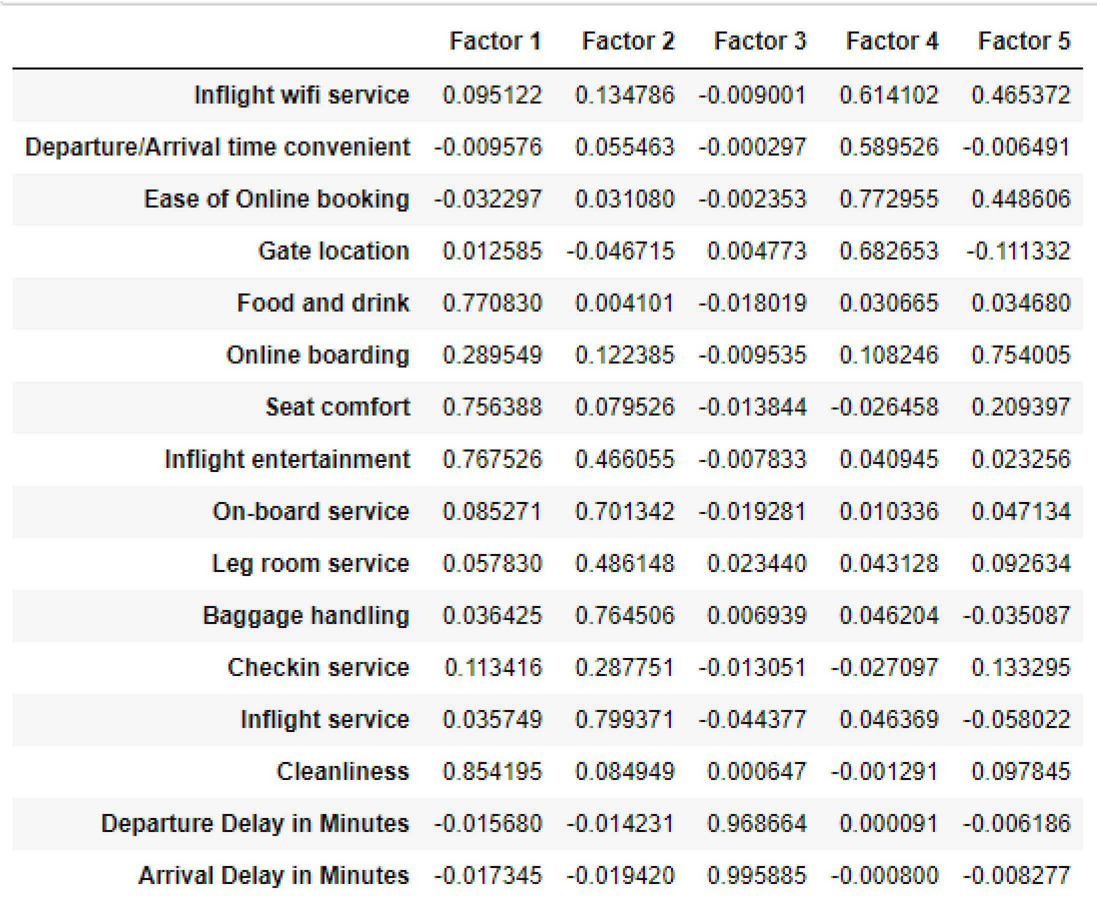
The main 5 factors that influence from the column are obtained and they are visualized.

The results obtained from the code:



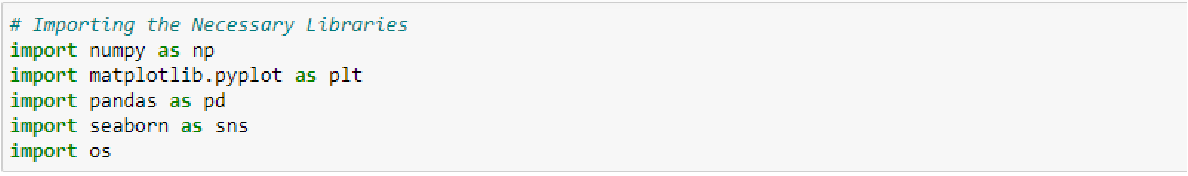
light wifi vice	Departure/Arrival time convenient	Ease of Online booking	Gate location	Food and drink	Online boarding	Seat comfort	Inflight entertainment	On-board service	Leg room service	Baggage handling	Checkin service	Inflight service	Cleanliness	Departure Delay in Minutes	Arrival Delay in Minutes
3	4	3	1	5	3	5	5	4	3	4	4	5	5	25	18.0
3	2	3	3	1	3	1	1	1	5	3	1	4	1	1	6.0
2	2	2	2	5	5	5	5	4	3	4	4	4	5	0	0.0
2	5	5	5	2	2	2	2	2	5	3	1	4	2	11	9.0
3	3	3	3	4	5	5	3	3	4	4	3	3	3	0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3	3	3	1	4	3	4	4	3	2	4	4	5	4	0	0.0
4	4	4	4	4	4	4	4	4	5	5	5	5	4	0	0.0
2	5	1	5	2	1	2	2	4	3	4	5	4	2	0	0.0
3	3	3	3	4	4	4	4	3	2	5	4	5	4	0	0.0
2	5	2	5	4	2	2	1	1	2	1	1	1	1	0	0.0



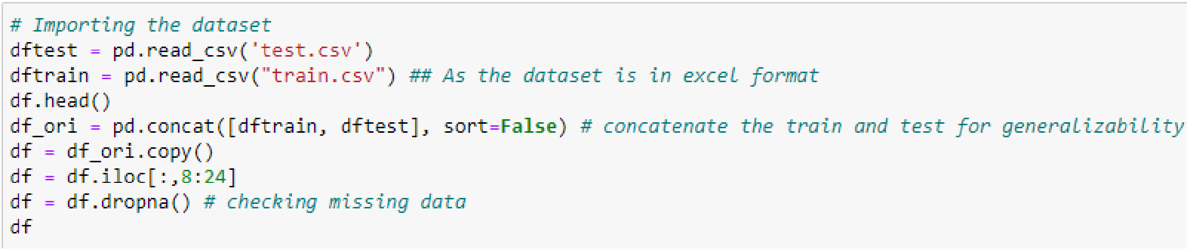


	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5
Inflight wifi service	0.095122	0.134786	-0.009001	0.614102	0.465372
Departure/Arrival time convenient	-0.009576	0.055463	-0.000297	0.589526	-0.006491
Ease of Online booking	-0.032297	0.031080	-0.002353	0.772955	0.448606
Gate location	0.012585	-0.046715	0.004773	0.682653	-0.111332
Food and drink	0.770830	0.004101	-0.018019	0.030665	0.034680
Online boarding	0.289549	0.122385	-0.009535	0.108246	0.754005
Seat comfort	0.756388	0.079526	-0.013844	-0.026458	0.209397
Inflight entertainment	0.767526	0.466055	-0.007833	0.040945	0.023256
On-board service	0.085271	0.701342	-0.019281	0.010336	0.047134
Leg room service	0.057830	0.486148	0.023440	0.043128	0.092634
Baggage handling	0.036425	0.764506	0.006939	0.046204	-0.035087
Checkin service	0.113416	0.287751	-0.013051	-0.027097	0.133295
Inflight service	0.035749	0.799371	-0.044377	0.046369	-0.058022
Cleanliness	0.854195	0.084949	0.000647	-0.001291	0.097845
Departure Delay in Minutes	-0.015680	-0.014231	0.968664	0.000091	-0.006186
Arrival Delay in Minutes	-0.017345	-0.019420	0.995885	-0.000800	-0.008277

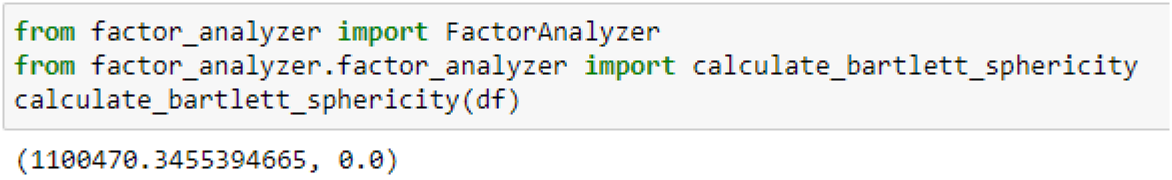
The Source code for the results:



```
# Importing the Necessary Libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import os
```

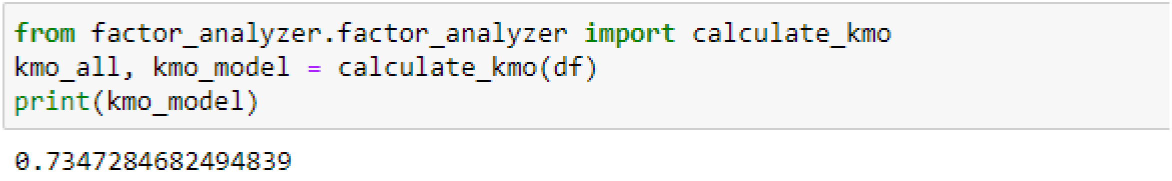


```
# Importing the dataset
dftest = pd.read_csv('test.csv')
dftrain = pd.read_csv("train.csv") ## As the dataset is in excel format
df.head()
df_ori = pd.concat([dftrain, dftest], sort=False) # concatenate the train and test for generalizability
df = df_ori.copy()
df = df.iloc[:,8:24]
df = df.dropna() # checking missing data
df
```



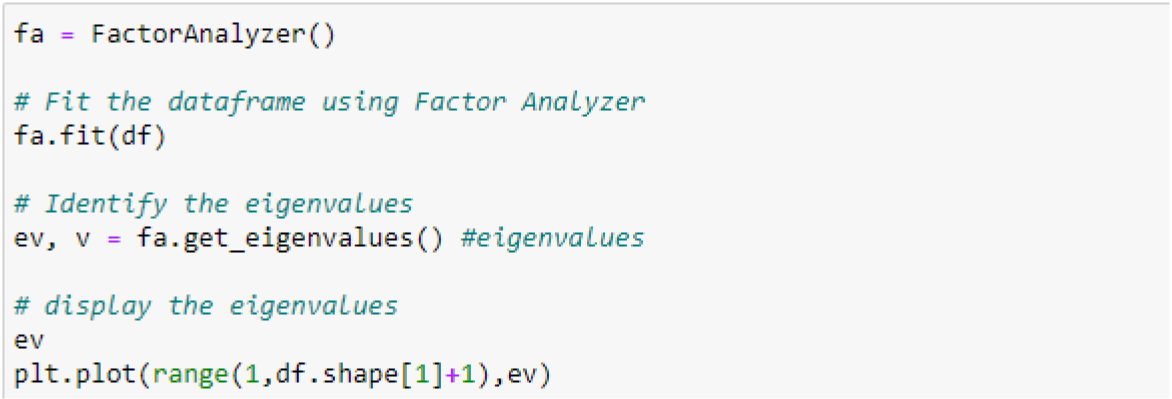
```
from factor_analyzer import FactorAnalyzer
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity
calculate_bartlett_sphericity(df)
```

```
(1100470.3455394665, 0.0)
```



```
from factor_analyzer.factor_analyzer import calculate_kmo
kmo_all, kmo_model = calculate_kmo(df)
print(kmo_model)
```

```
0.7347284682494839
```

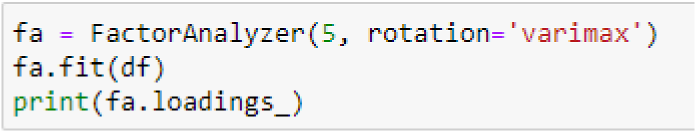


```
fa = FactorAnalyzer()

# Fit the dataframe using Factor Analyzer
fa.fit(df)

# Identify the eigenvalues
ev, v = fa.get_eigenvalues() #eigenvalues

# display the eigenvalues
ev
plt.plot(range(1,df.shape[1]+1),ev)
```



```
fa = FactorAnalyzer(5, rotation='varimax')
fa.fit(df)
print(fa.loadings_)
```

lmatrix = pd.DataFrame(fa.loadings_, index = list(df.columns), columns = ['Factor 1', 'Factor 2', 'Factor 3', 'Factor 4', 'Factor 5'])

lmatrix #loading matrix

lmatrix.sort_values('Factor 1', ascending=False)

lmatrix.sort_values('Factor 2', ascending=False)

lmatrix.sort_values('Factor 3', ascending=False)

lmatrix.sort_values('Factor 4', ascending=False)

lmatrix.sort_values('Factor 5', ascending=False)