Project Report

Exploratory Factor Analysis

Name:Keshav Anand

Course: AI and ML

(Batch-4)

Duration: 12 months

Problem Statement: Implementing Exploratory Factor Analysis technique on a given dataset.

Prerequisites

What things you need to install the software and how to install them:

Python 3.6 This setup requires that your machine has latest version of python. The following url https://www.python.org/downloads/ can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: https://www.pythoncentral.io/add-python-to-path-python-is-not- recognized-as-an-internal-or-external-command/. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.

Second and easier option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url https://www.anaconda.com/download/ You will also need to download and install below 3 packages after you install either python or anaconda from the steps above Sklearn (scikit-learn) numpy scipy if you have chosen to install python 3.6 then run below commands in command prompt/terminal to install these packages pip install -U scikit-learn pip install numpy pip install scipy if you have chosen to install anaconda then run below commands in anaconda prompt to install these packages conda install -c scikit-learn conda install -c anaconda numpy conda install -c anaconda scipy

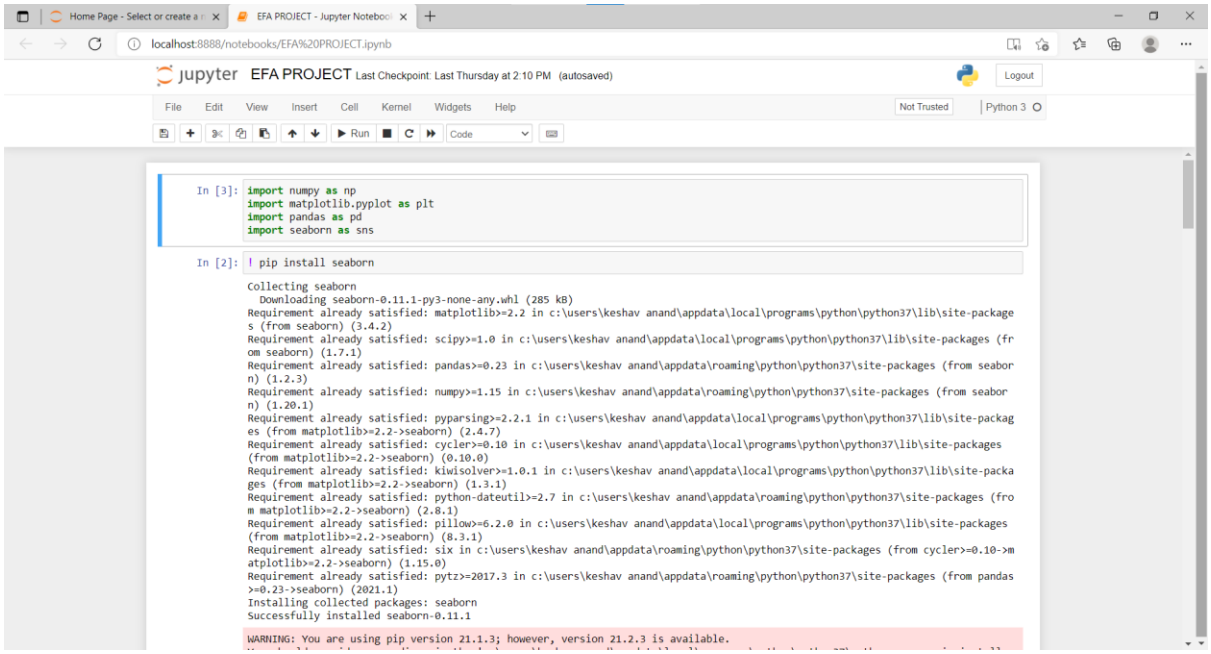
Dataset used:

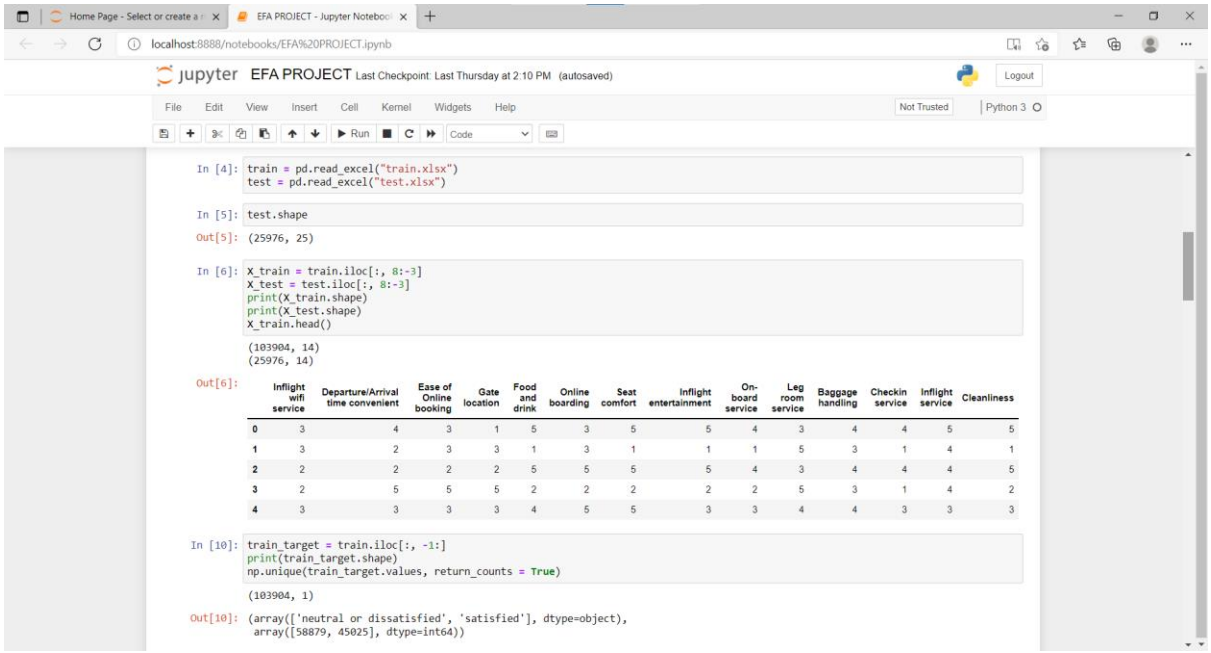
The dataset used is Airline Passenger Satisfaction dataset which is available on a website called Kaggle.com.

Method used for detection:

EFA (Exploratory Factor Analysis)

Screenshots of Source Code and Output:





Home Page - Select or create a n | EFA PROJECT - Jupyter Notebook

localhost:8888/notebooks/EFA%20PROJECT.ipynb

jupyter **EFA PROJECT** Last Checkpoint: Last Thursday at 2:10 PM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

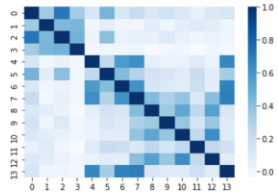
Run Code

In [11]:
```
# Zero centering the data
X = X_train.values
X_mean = np.mean(X, axis = 0)
X_n = X - np.matrix(X_mean)
X_n = X_n.T
print(X_n.shape)
```

```
(14, 103904)
```

In [12]:
```
c1 = np.cov(X_n)
c2 = np.corrcoef(X_n)
```

In [13]:
```
ax = sns.heatmap(c2, cmap = "Blues")
```



In [18]:
```
eig_val, eig_vec = np.linalg.eig(c1)
eig_sorted = np.sort(eig_val)[::-1]
arg_sort = np.argsort(eig_val)[::-1]
```

In [19]:
```
print(eig_val)
```

```
[6.52815927 4.47255915 3.43215579 1.98504987 1.61162741 1.18057993
```

Home Page - Select or create a n | EFA PROJECT - Jupyter Notebook

localhost:8888/notebooks/EFA%20PROJECT.ipynb

jupyter **EFA PROJECT** Last Checkpoint: Last Thursday at 2:10 PM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Run Code

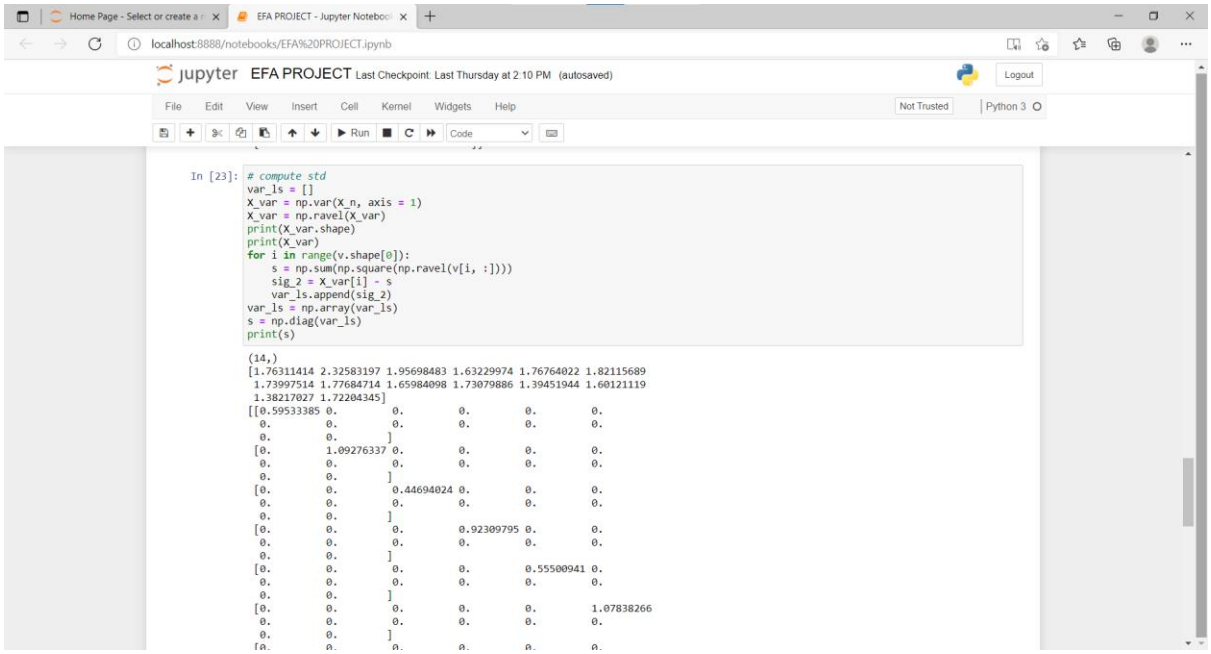
In [20]:
```
eig_vec_ls = []
eig_val_ls = []
```

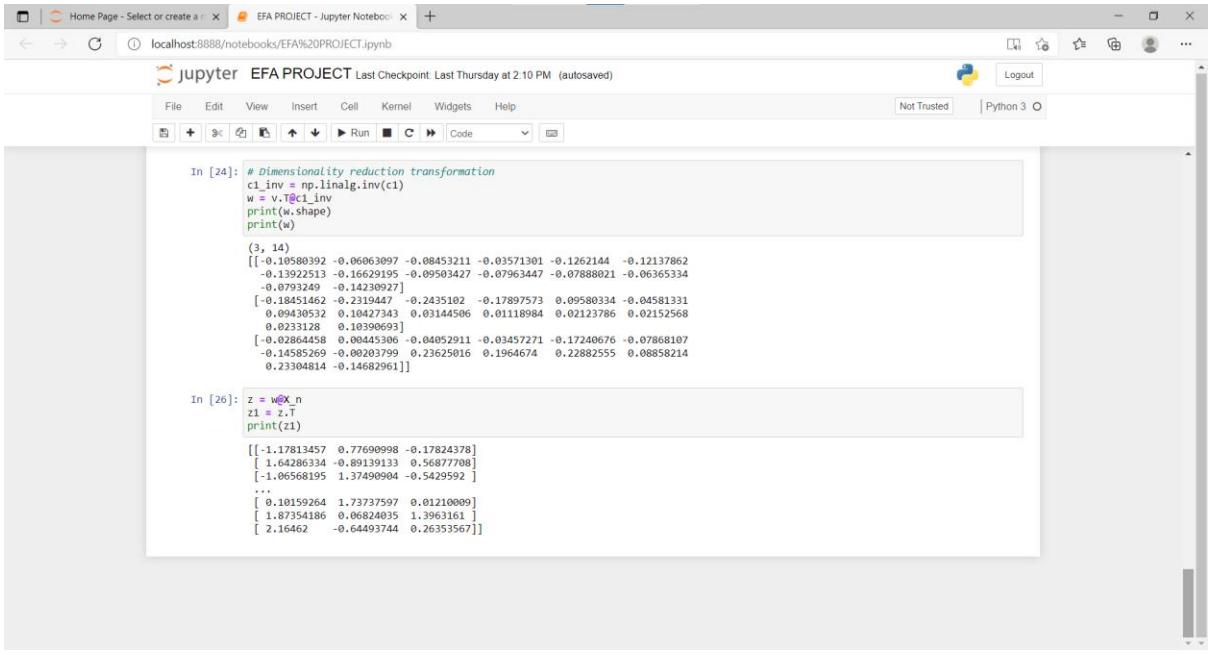
In [21]:
```
imp_vec = arg_sort[:3]
for i in imp_vec:
    eig_vec_ls.append(eig_vec[:, i])
    eig_val_ls.append(eig_val[i])
print(eig_vec_ls)
print(eig_val_ls)
```

```
[array([-0.27033179, -0.15491373, -0.21598175, -0.09124768, -0.32248111,
       -0.31012556, -0.35572388, -0.42488029, -0.24281505, -0.20346816,
       -0.20154101, -0.16263595, -0.20267719, -0.36360393]), array([-0.39021937, -0.49052653, -0.51498574, -0.37850549,  0.2026
0897,
       -0.09688795,  0.1994409 ,  0.22052189,  0.06650134,  0.02366476,
        0.04491473,  0.04552343,  0.04930291,  0.2197468 ]), array([-0.05306717,  0.00824977, -0.07508455, -0.06404967, -0.3194
0211,
       -0.14576517, -0.27020783, -0.0037756 ,  0.4376789 ,  0.36397705,
        0.42392401,  0.16410797,  0.43174682, -0.27201769])]
[6.528159273595747, 4.472559148142883, 3.432157866762826]
```

In [22]:
```
# Estimate V
eig_val_arr = np.array(eig_val_ls)
lambda_1 = np.diag(eig_val_arr)
print(lambda_1)
eig_vec_mat = np.matrix(eig_vec_ls).T
v = eig_vec_mat@np.sqrt(lambda_1)
print(v)
```

```
[[6.52815927 0.         0.        ]
 [0.         4.47255915 0.        ]
 [0.         0.         3.43215579]]
[[-0.69070483 -0.82525254 -0.09831266]
 [-0.39580866 -1.03738639  0.01528359]
 [-0.55183905 -1.08911378 -0.13910223]
 [-0.23314023 -0.80047953 -0.11865892]
 [-0.8239477   0.42848608 -0.59172685]
```


Home Page - Select or create a
EFA PROJECT - Jupyter Notebook
localhost:8888/notebooks/EFA%20PROJECT.ipynb
jupyter EFA PROJECT Last Checkpoint: Last Thursday at 2:10 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [23]:
# compute std
var_ls = []
X_var = np.var(X_n, axis = 1)
X_var = np.ravel(X_var)
print(X_var.shape)
print(X_var)
for i in range(v.shape[0]):
s = np.sum(np.square(np.ravel(v[i, :])))
sig_2 = X_var[i] - s
var_ls.append(sig_2)
var_ls = np.array(var_ls)
s = np.diag(var_ls)
print(s)
(14,)
[1.76311414 2.32583197 1.95698483 1.63229974 1.76764022 1.82115689
1.73997514 1.77684714 1.65984098 1.73079886 1.39451944 1.60121119
1.38217027 1.72204345]
[[0.59533385 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 1.09276337 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 0. 0.44694024 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 0. 0. 0.92309795 0. 0.
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 0. 0. 0. 0.55500941 0.
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 0. 0. 0. 0. 1.07838266
0. 0. 0. 0. 0. 0.
0. 0.]
[0. 0. 0. 0. 0. 0.


Home Page - Select or create a
EFA PROJECT - Jupyter Notebook
localhost:8888/notebooks/EFA%20PROJECT.ipynb
jupyter EFA PROJECT Last Checkpoint: Last Thursday at 2:10 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [24]:
# Dimensionality reduction transformation
c1_inv = np.linalg.inv(c1)
w = v.T@c1_inv
print(w.shape)
print(w)
(3, 14)
[[-0.10580392 -0.06063097 -0.08453211 -0.03571301 -0.1262144 -0.12137862
-0.13922513 -0.16629195 -0.09503427 -0.07963447 -0.07888021 -0.06365334
-0.0793249 -0.14230927]
[-0.18451462 -0.2319447 -0.2435102 -0.17897573 0.09580334 -0.04581331
0.09430532 0.10427343 0.03144506 0.01118984 0.02123786 0.02152568
0.0233128 0.10390693]
[-0.02864458 0.00445306 -0.04052911 -0.03457271 -0.17240676 -0.07868107
-0.14585269 -0.00203799 0.23625016 0.1964674 0.22882555 0.08858214
0.23304814 -0.14682961]]
In [26]:
z = w@X_n
z1 = z.T
print(z1)
[[-1.17813457 0.77690998 -0.17824378]
[1.64286334 -0.89139133 0.56877708]
[-1.06568195 1.37490904 -0.5429592]
...
[0.10159264 1.73737597 0.01210009]
[1.87354186 0.06824035 1.3963161]
[2.16462 -0.64493744 0.26353567]]