dvanced-analytics-using-statistics

February 19, 2024

Q.Do Factor Analysis on the ‘teacher evaluation’ dataset using Python in Google Colab.

[18]:

```
!pip install factor_analyzer
```

```
Collecting factor_analyzer
  Downloading factor_analyzer-0.5.1.tar.gz (42 kB)
                           42.8/42.8 kB
928.6 kB/s eta 0:00:00
  Installing build dependencies … done
  Getting requirements to build wheel … done
  Installing backend dependencies … done
  Preparing metadata (pyproject.toml) … done
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages
(from factor_analyzer) (1.5.3)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages
(from factor_analyzer) (1.11.4)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages
(from factor_analyzer) (1.25.2)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-
packages (from factor_analyzer) (1.2.2)
Requirement already satisfied: python-dateutil>=2.8.1 in
/usr/local/lib/python3.10/dist-packages (from pandas->factor_analyzer) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-
packages (from pandas->factor_analyzer) (2023.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->factor_analyzer) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn->factor_analyzer)
(3.2.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-
packages (from python-dateutil>=2.8.1->pandas->factor_analyzer) (1.16.0)
Building wheels for collected packages: factor_analyzer
  Building wheel for factor_analyzer (pyproject.toml) … done
  Created wheel for factor_analyzer:
filename=factor_analyzer-0.5.1-py2.py3-none-any.whl size=42564
sha256=54bd894fd47f3372a7f639455287229cd9bea824b210eb2df6514ed896d61d3a
  Stored in directory: /root/.cache/pip/wheels/24/59/82/6493618e30ed1cb7a013b9e1
```

```
b0c9e17de80b04dfcef4ba8a4d
Successfully built factor_analyzer
Installing collected packages: factor_analyzer
Successfully installed factor_analyzer-0.5.1
```

[22]:

```
# Importing neccesaary libraries
import pandas as pd
import numpy as np
from factor_analyzer import FactorAnalyzer
import matplotlib.pyplot as plt
```

[24]:

```
# Loading the dataset with headers and skipping non-numeric value rows
data = pd.read_csv("Teacher_evaluation.csv", header=1, skiprows=[1])
print(data.head())
```

```
   Unnamed: 0  Expect  Entertain  Comm  Expert  Motivate  Caring  Charisma  \
0           1       2          8     1       4         7       5         4
1           2       4          8     5       3         7       7         7
2           3       2          8     2       3         6       7         1
3           4       4          8     4       2         8       7         7
4           5       3          8     5       4         8       8         7

   Passion  Friendly
0        4         8
1        6         6
2        3         7
3        5         7
4        6         7
```

[26]:

```
# Removing non-numeric columns
data = data.select_dtypes(include=[np.number])
print(data.head())
```

```
   Unnamed: 0  Expect  Entertain  Comm  Expert  Motivate  Caring  Charisma  \
0           1       2          8     1       4         7       5         4
1           2       4          8     5       3         7       7         7
2           3       2          8     2       3         6       7         1
3           4       4          8     4       2         8       7         7
4           5       3          8     5       4         8       8         7

   Passion  Friendly
0        4         8
1        6         6
2        3         7
3        5         7
4        6         7
```

[27]:

```
# Creating factor analysis object
fa = FactorAnalyzer()
fa.fit(data)
```

[27]:

```
FactorAnalyzer(rotation_kwargs={})
```

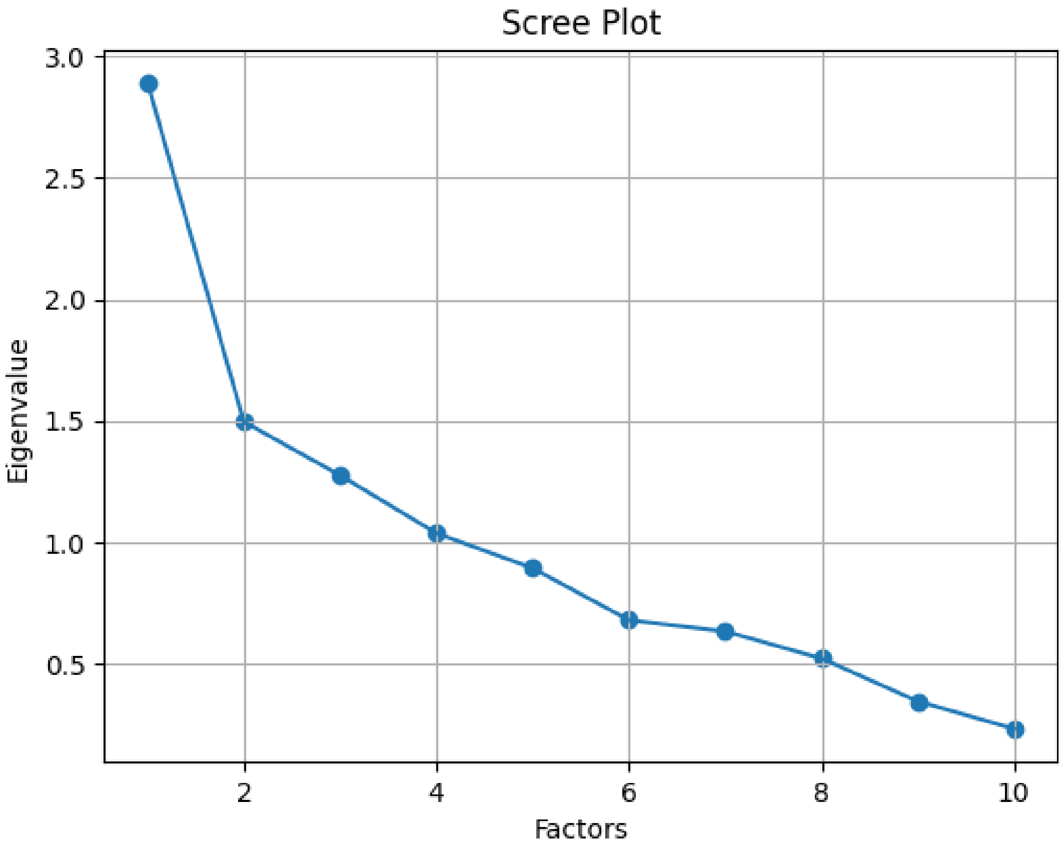
[28]:

```
# Checking Eigenvalues
eigen_values, vectors = fa.get_eigenvalues()
print('Eigenvalues:', eigen_values)
```

```
Eigenvalues: [2.8913459  1.49504406 1.2748036  1.03658883 0.89255537 0.67908327
 0.63267539 0.52124679 0.34481561 0.23184118]
```

[29]:

```
# Create scree plot
plt.scatter(range(1, data.shape[1]+1), eigen_values)
plt.plot(range(1, data.shape[1]+1), eigen_values)
plt.title('Scree Plot')
plt.xlabel('Factors')
plt.ylabel('Eigenvalue')
plt.grid()
plt.show()
```



[30]:
```
# Performing factor analysis with a specific number of factors
fa = FactorAnalyzer(n_factors=3, rotation='varimax')
fa.fit(data)
```

[30]:
```
FactorAnalyzer(rotation='varimax', rotation_kwargs={})
```

[44]:
```
# Loading the factor
loadings = fa.loadings_
print('Factor Loadings:\n', loadings)
```

```
Factor Loadings:
 [[ 1.68320693e-01 -2.90671634e-02 -2.99509614e-01]
 [ 1.74744806e-02 -3.89981435e-01 -3.96669376e-01]
 [ 5.62344975e-01  3.36316215e-01  4.54088270e-02]
 [ 8.18740075e-01 -1.05895054e-01 -8.05006128e-02]
 [ 2.64714851e-01  6.56144357e-02  2.15978932e-02]
 [ 4.12333883e-01  6.18248206e-03  4.64095161e-01]
 [ 1.66172901e-01 -5.07241937e-02  4.00384152e-01]
 [ 8.53291085e-01 -8.04076585e-03  9.70139247e-04]
 [ 5.41289381e-01  6.63796282e-02  2.33534654e-01]
 [ 1.35644153e-01  9.85378263e-01 -7.86968389e-02]]
```

[46]:
```
# Getting variance explained
variance_explained = fa.get_factor_variance()
print('Variance Explained:', variance_explained)
```

```
Variance Explained: (array([2.32241022, 1.25961043, 0.69248574]),
array([0.23224102, 0.12596104, 0.06924857]), array([0.23224102, 0.35820206,
0.42745064]))
```

[47]:
```
# Importing PCA library for PCA
from sklearn.decomposition import PCA
```

[36]:
```
# Removing non-numeric columns
data_numeric = data.select_dtypes(include=[np.number])
```

[37]:
```
# Performing PCA
pca = PCA(n_components=3)
pca.fit(data_numeric)
```

[37]:
```
PCA(n_components=3)
```

[38]:
```
# Transforming the data to the new coordinate system
data_pca = pca.transform(data_numeric)
```

[39]:

```python
# Getting the principal components
principal_components = pd.DataFrame(data_pca, columns=['PC1', 'PC2', 'PC3'])
```

[40]:

```python
# Printing the variance ratio
print('Variance Ratio:', pca.explained_variance_ratio_)
```

```
Variance Ratio: [0.9867739  0.00725419 0.00166419]
```

[43]:

```python
# Printing the principal components
print('Principal Components:\n', principal_components.head())
```

```
Principal Components:
          PC1       PC2       PC3
0  59.527681 -1.399650 -0.170280
1  58.469321  3.298496  0.118372
2  57.550827 -3.735994  1.218940
3  56.479401  2.715759  0.374383
4  55.481681  3.615700  1.696452
```

[49]:

```python
# Removing non-numeric columns for Common Factor Analysis
data_numeric = data.select_dtypes(include=[np.number])
```

[50]:

```python
# Performing Common Factor Analysis
cfa = FactorAnalyzer(rotation=None)  # No rotation for simplicity
cfa.fit(data_numeric)
```

[50]:

```
FactorAnalyzer(rotation=None, rotation_kwargs={})
```

[60]:

```python
# Loading the factor
loadings_cfa = cfa.loadings_
print('Factor Loadings:\n', loadings_cfa)
```

```
Factor Loadings:
 [[ 1.03710864e-01  9.07425635e-02  3.16057920e-01]
 [-1.40788777e-01  4.09395697e-01  3.49730116e-01]
 [ 6.24226907e-01 -1.89957868e-01  7.52276985e-02]
 [ 7.46157993e-01  3.09521379e-01  1.88345922e-01]
 [ 2.72431903e-01  1.75178185e-04  2.50300356e-02]
 [ 4.75154778e-01  6.25833079e-02 -3.94655596e-01]
 [ 2.15367793e-01  6.15408115e-02 -3.74597502e-01]
 [ 8.15719607e-01  2.17655987e-01  1.24091357e-01]
 [ 5.73506976e-01  5.24645543e-02 -1.42394184e-01]
 [ 3.44603659e-01 -9.13574581e-01  2.05408500e-01]]
```

[61]:

```python
# Getting the communalities
communalities = cfa.get_communalities()
print('Communalities:\n', communalities)
```

```
Communalities:
 [0.11888276 0.30973747 0.43140243 0.68802942 0.07484568 0.38544177
 0.19049385 0.72817127 0.35193888 0.99556285]
```

[62]:

```python
# Getting the unique variances
unique_variances = 1 - communalities
print('Unique Variances:\n', unique_variances)
```

```
Unique Variances:
 [0.88111724 0.69026253 0.56859757 0.31197058 0.92515432 0.61455823
 0.80950615 0.27182873 0.64806112 0.00443715]
```