PCA

April 25, 2023

1 Principal component analysis on Planets Dataset

2 Load Libraries

[1]:
```
import pandas as pd
import seaborn as sns
```

2.1 Import the Dataset

[2]:
```
df=sns.load_dataset("planets")
```

[3]:
```
df
```

[3]:
```
                method  number  orbital_period   mass  distance  year
0     Radial Velocity       1       269.300000   7.10     77.40  2006
1     Radial Velocity       1       874.774000   2.21     56.95  2008
2     Radial Velocity       1       763.000000   2.60     19.84  2011
3     Radial Velocity       1       326.030000  19.40    110.62  2007
4     Radial Velocity       1       516.220000  10.50    119.47  2009
…                 …     …             …    …       …    …
1030          Transit       1         3.941507    NaN    172.00  2006
1031          Transit       1         2.615864    NaN    148.00  2007
1032          Transit       1         3.191524    NaN    174.00  2007
1033          Transit       1         4.125083    NaN    293.00  2008
1034          Transit       1         4.187757    NaN    260.00  2008

[1035 rows x 6 columns]
```

[4]:
```
df=df.drop("method",axis=1)
```

[5]:
```
df=df.dropna()
```

[6]:
```
df
```

[6]:
```
     number  orbital_period    mass  distance  year
0         1       269.30000   7.100     77.40  2006
1         1       874.77400   2.210     56.95  2008
```

```
2          1       763.00000   2.600     19.84  2011
3          1       326.03000  19.400    110.62  2007
4          1       516.22000  10.500    119.47  2009
..       …             …      …         …    …
640        1       111.70000   2.100     14.90  2009
641        1         5.05050   1.068     44.46  2013
642        1       311.28800   1.940     17.24  1999
649        1         2.70339   1.470    178.00  2013
784        3       580.00000   0.947    135.00  2012

[498 rows x 5 columns]
```

3 Principal component analysis

[7]:
```
from sklearn.decomposition import PCA
```

3.0.1 Reduce features to 3 from 5

[8]:
```
pca=PCA(n_components=3)
```

[9]:
```
df1=pca.fit_transform(df)
```

[10]:
```
df1
```

[10]:
```
array([[-5.66504590e+02,  2.47806690e+01,  3.44946665e+00],
       [ 3.89898869e+01,  4.93234005e+00, -6.16922066e-01],
       [-7.27419368e+01, -3.22233374e+01, -3.34307005e+00],
       …,
       [-5.24453396e+02, -3.55444185e+01,  7.11772887e+00],
       [-8.33214190e+02,  1.25044259e+02, -4.72103389e+00],
       [-2.55870669e+02,  8.26531475e+01, -4.43273009e+00]])
```

[11]:
```
df2=pd.DataFrame(df1,columns=["F1","F2","F3"])
```

[12]:
```
df2
```

[12]:
```
              F1          F2        F3
0   -566.504590   24.780669  3.449467
1     38.989887    4.932340 -0.616922
2    -72.741937  -32.223337 -3.343070
3   -509.805791   58.337215  8.138673
4   -319.629461   67.231795  2.486263
..          …         …       …
493 -724.036578  -37.923212 -1.785338
494 -830.718586   -8.445276 -5.654267
495 -524.453396  -35.544418  7.117729
```

```
496 -833.214190  125.044259 -4.721034
497 -255.870669   82.653148 -4.432730

[498 rows x 3 columns]
```

4 Check Eigen vectors of PCA

[13]:
```
pca.components_
```

[13]:
```
array([[-4.75569031e-05,  9.99999255e-01,  4.57643358e-04,
        -1.11337300e-03,  1.99069706e-04],
       [-7.35009201e-03,  1.09897469e-03,  2.20090392e-02,
         9.99575157e-01,  1.76033081e-02],
       [-6.19319719e-02, -1.90509624e-05,  4.38895746e-01,
         5.66685873e-03, -8.96383200e-01]])
```

5 Check Variances of pca

[14]:
```
pca.explained_variance_ratio_
```

[14]:
```
array([9.98981761e-01, 1.00454044e-03, 8.42345589e-06])
```