# 1 DATA MINING BUAN-670-DB

# 2 HW-1: PCA

### 2.0.1 DATA LOADING

[6]:
```
import pandas as pd

DATA = Path("C:/Users/admin/Downloads")
universities_df = pd.read_csv(DATA / "Universities.csv")

universities_df.info(), universities_df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1302 entries, 0 to 1301
Data columns (total 20 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   College.Name                 1302 non-null   object
 1   State                        1302 non-null   object
 2   Public..1…Private..2.    1302 non-null   int64
 3   X..appli..rec.d              1292 non-null   float64
 4   X..appl..accepted            1291 non-null   float64
 5   X..new.stud..enrolled        1297 non-null   float64
 6   X..new.stud..from.top.10.    1067 non-null   float64
 7   X..new.stud..from.top.25.    1100 non-null   float64
 8   X..FT.undergrad              1299 non-null   float64
 9   X..PT.undergrad              1270 non-null   float64
 10  in.state.tuition             1272 non-null   float64
 11  out.of.state.tuition         1282 non-null   float64
 12  room                         981 non-null    float64
 13  board                        804 non-null    float64
 14  add..fees                    1028 non-null   float64
 15  estim..book.costs            1254 non-null   float64
 16  estim..personal..            1121 non-null   float64
 17  X..fac..w.PHD                1270 non-null   float64
 18  stud..fac..ratio             1300 non-null   float64
 19  Graduation.rate              1204 non-null   float64
```

```
dtypes: float64(17), int64(1), object(2)
memory usage: 203.6+ KB
```

[6]:

```
(None,
                        College.Name State  Public..1…Private..2.  \
 0         Alaska Pacific University    AK                      2
 1  University of Alaska at Fairbanks    AK                      1
 2     University of Alaska Southeast    AK                      1
 3  University of Alaska at Anchorage    AK                      1
 4        Alabama Agri. & Mech. Univ.    AL                      1

    X..appli..rec.d  X..appl..accepted  X..new.stud..enrolled  \
 0            193.0              146.0                   55.0
 1           1852.0             1427.0                  928.0
 2            146.0              117.0                   89.0
 3           2065.0             1598.0                 1162.0
 4           2817.0             1920.0                  984.0

    X..new.stud..from.top.10.  X..new.stud..from.top.25.  X..FT.undergrad  \
 0                       16.0                       44.0            249.0
 1                        NaN                        NaN           3885.0
 2                        4.0                       24.0            492.0
 3                        NaN                        NaN           6209.0
 4                        NaN                        NaN           3958.0

    X..PT.undergrad  in.state.tuition  out.of.state.tuition    room   board  \
 0            869.0            7560.0                7560.0  1620.0  2500.0
 1           4519.0            1742.0                5226.0  1800.0  1790.0
 2           1849.0            1742.0                5226.0  2514.0  2250.0
 3          10537.0            1742.0                5226.0  2600.0  2520.0
 4            305.0            1700.0                3400.0  1108.0  1442.0

    add..fees  estim..book.costs  estim..personal..  X..fac..w.PHD  \
 0      130.0              800.0             1500.0           76.0
 1      155.0              650.0             2304.0           67.0
 2       34.0              500.0             1162.0           39.0
 3      114.0              580.0             1260.0           48.0
 4      155.0              500.0              850.0           53.0

    stud..fac..ratio  Graduation.rate
 0              11.9             15.0
 1              10.0              NaN
 2               9.5             39.0
 3              13.7              NaN
 4              14.3             40.0  )
```

### 2.0.2 4.2.a. Remove all categorical variables. Then remove all records with missing numerical measurements from the dataset.

**From the Universities.csv dataset the following columns are categorical:**

- College.Name (object)
- State (object)
- Public..1...Private..2. is a numeric code, but it represents a categorical distinction (public/private).

Lets treat the "Public..1...Private..2." as categorical and remove it along with the other two. Then we will drop the rows with any missing numerical values.

Now lets clean the data accordingly.

[8]:
```
# Dropping the categorical columns by using this and mentioning what columns we␣
↪have to drop.
categorical_cols = ['College.Name', 'State', 'Public..1...Private..2.']
numerical_df = universities_df.drop(columns=categorical_cols)

# Dropping the rows with any missing values and cleaning the dataset.
cleaned_df = numerical_df.dropna()

# After cleaning the data lets see the result shape and preview
cleaned_df.shape, cleaned_df.head()
```

[8]:
```
((471, 17),
     X..appli..rec.d  X..appl..accepted  X..new.stud..enrolled  \
 0             193.0              146.0                   55.0
 2             146.0              117.0                   89.0
 9             805.0              588.0                  287.0
 11            608.0              520.0                  127.0
 21           4414.0             1500.0                  335.0

     X..new.stud..from.top.10.  X..new.stud..from.top.25.  X..FT.undergrad  \
 0                        16.0                       44.0            249.0
 2                         4.0                       24.0            492.0
 9                        67.0                       88.0           1376.0
 11                       26.0                       47.0            538.0
 21                       30.0                       60.0            908.0

     X..PT.undergrad  in.state.tuition  out.of.state.tuition    room   board  \
 0             869.0            7560.0                7560.0  1620.0  2500.0
 2            1849.0            1742.0                5226.0  2514.0  2250.0
 9             207.0           11660.0               11660.0  2050.0  2430.0
 11            126.0            8080.0                8080.0  1380.0  2540.0
 21            119.0            5666.0                5666.0  1424.0  1540.0

     add..fees  estim..book.costs  estim..personal..  X..fac..w.PHD  \
```

```
0       130.0               800.0            1500.0           76.0
2        34.0               500.0            1162.0           39.0
9       120.0               400.0             900.0           74.0
11      100.0               500.0            1100.0           63.0
21      418.0              1000.0            1400.0           56.0

    stud..fac..ratio  Graduation.rate
0               11.9             15.0
2                9.5             39.0
9               14.0             72.0
11              11.4             44.0
21              15.5             46.0  )
```

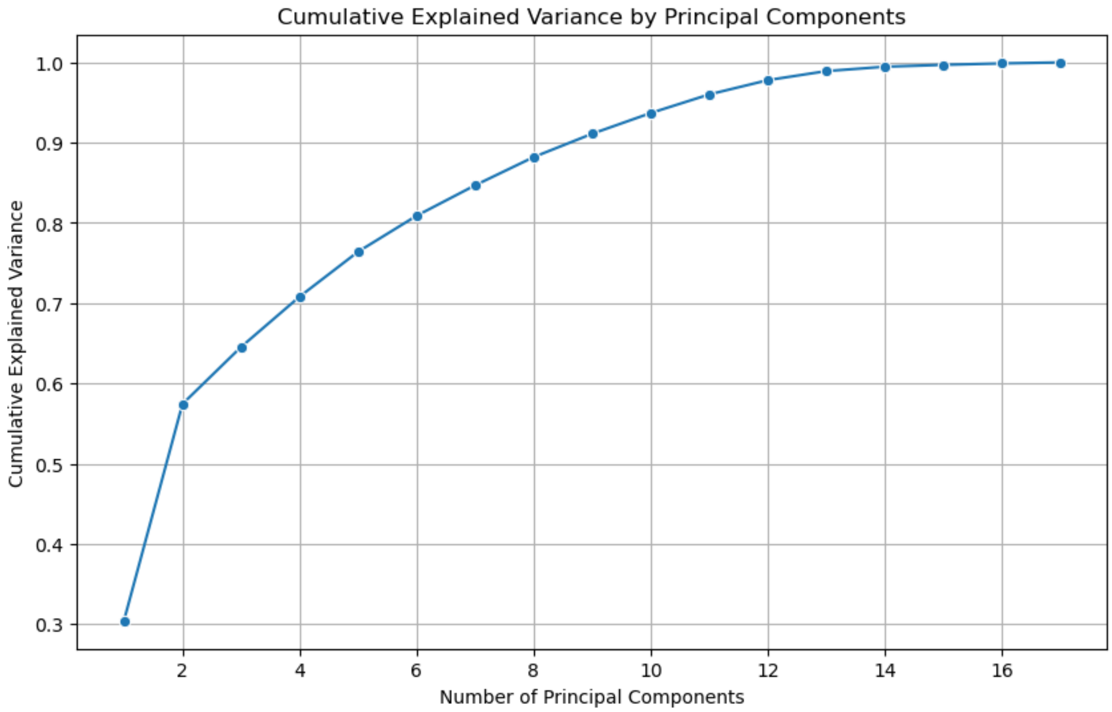
**Now we have a cleaned dataset with only numerical variables and 471 complete records example i can say as no missing values in the dataset.**

### 2.0.3 4.2.b. Conduct a principal components analysis on the cleaned data and comment on the results. Should the data be normalized? Discuss what characterizes the components you consider key.

**Here we will normalize the data it is important for PCA becuase when variables are on different scales then we will Perform PCA. Last we will Interpret the components to identify which ones are key and what they represent.**

[9]:
```python
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Standardizing the data
scaler = StandardScaler()
scaled_data = scaler.fit_transform(cleaned_df)

# Applying the PCA
pca = PCA()
pca_components = pca.fit_transform(scaled_data)

# The portion of the data variability captured by each component
explained_variance = pca.explained_variance_ratio_
cumulative_variance = np.cumsum(explained_variance)

# This is for plotting explained variance
plt.figure(figsize=(10, 6))
sns.lineplot(x=range(1, len(explained_variance)+1), y=cumulative_variance,␣
 ↪marker="o")
plt.title('Cumulative Explained Variance by Principal Components')
plt.xlabel('Number of Principal Components')
```

```
plt.ylabel('Cumulative Explained Variance')
plt.grid(True)
plt.show()

# This show the top 5 components variance explained in data.
explained_variance[:5], cumulative_variance[:5]
```




[9]:
```
(array([0.30441431, 0.27004192, 0.07096712, 0.06270505, 0.05603243]),
 array([0.30441431, 0.57445623, 0.64542335, 0.7081284 , 0.76416083]))
```

**PCA Results Summary:**

- Normalization was important because features like tuition fees, enrollment numbers, and student ratios have very different scales. If we didn't scale them, PCA would focus more on the variables with the largest values, making the results biased.
- Explained Variance: >* The PC1 explains 30.4% >* The PC2 explains 27.0% >* By combining the first 2 components explain 57.4% of the total variance. >* The first 5 components explain 76.4% of the variance.

**This suggests that 2 to 5 principal components capture most of the structure in the data.**

[ ]: