




ADVANCED STATISTICS PROJECT



Soundarya K S C
PGP-DSBA 23.01.2022

TABLE OF CONTENTS:

Contents	Page No
1.1 State the null and the alternate hypothesis for conducting one-way ANOVA for both Education and Occupation individually.	4
1.2 Perform one-way ANOVA for Education with respect to the variable 'Salary'. State whether the null hypothesis is accepted or rejected based on the ANOVA results.	5
1.3 Perform one-way ANOVA for variable Occupation with respect to the variable 'Salary'. State whether the null hypothesis is accepted or rejected based on the ANOVA results.	5
1.4 If the null hypothesis is rejected in either (1.2) or in (1.3), find out which class means are significantly different. Interpret the result.	5
1.5 What is the interaction between the two treatments? Analyze the effects of one variable on the other (Education and Occupation) with the help of an interaction plot.	6
1.6 Perform a two-way ANOVA based on the Education and Occupation (along with their interaction Education*Occupation) with the variable 'Salary'. State the null and alternative hypotheses and state your results. How will you interpret this result?	6
1.7 Explain the business implications of performing ANOVA for this particular case study	7
2.1 Perform Exploratory Data Analysis [both univariate and multivariate analysis to be performed]. What insight do you draw from the EDA?	9
2.2 Is scaling necessary for PCA in this case? Give justification and perform scaling.	11
2.3 Comment on the comparison between the covariance and the correlation matrices from this data.[on scaled data]	13
2.4 Check the dataset for outliers before and after scaling. What insight do you derive here?	14
2.5 Extract the eigenvalues and eigenvectors. [Using Sklearn PCA Print Both]	14
2.6 Perform PCA and export the data of the Principal Component (eigenvectors) into a data frame with the original features	16
2.7 Write down the explicit form of the first PC (in terms of the eigenvectors. Use values with two places of decimals only). [hint: write the linear equation of PC in terms of eigenvectors and corresponding features]	17
2.8 Consider the cumulative values of the eigenvalues. How does it help you to decide on the optimum number of principal components? What do the eigenvectors indicate?	18
2.9 Explain the business implication of using the Principal Component Analysis for this case study. How may PCs help in the further analysis? [Hint: Write Interpretations of the Principal Components Obtained]	19

List of figures

Content	Page no
1.Pointplot	6
2.Distplot and boxplot	9
3.Pairplot	10
4.Heatmap	11
5.Boxplot_scaled	13
6.Screeplot	18

List of tables

Contents	Page no
1.Problem-1 summary of the data	3
2.Problem-2 summary of the data	7

Problem 1:

Salary is hypothesized to depend on educational qualification and occupation. To understand the dependency, the salaries of 40 individuals [[SalaryData.csv](#)] are collected and each person's educational qualification and occupation are noted. Educational qualification is at three levels, High school graduate, Bachelor, and Doctorate. Occupation is at four levels, Administrative and clerical, Sales, Professional or specialty, and Executive or managerial. A different number of observations are in each level of education – occupation combination.

	Education	Occupation	Salary
0	Doctorate	Adm-clerical	153197
1	Doctorate	Adm-clerical	115945
2	Doctorate	Adm-clerical	175935
3	Doctorate	Adm-clerical	220754
4	Doctorate	Sales	170769
5	Doctorate	Sales	219420
6	Doctorate	Sales	237920
7	Doctorate	Sales	160540
8	Doctorate	Sales	180934
9	Doctorate	Prof-specialty	248156

Info of the data

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40 entries, 0 to 39
Data columns (total 3 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Education    40 non-null    object
1   Occupation    40 non-null    object
2   Salary        40 non-null    int64
dtypes: int64(1), object(2)
memory usage: 1.1+ KB
```

To check null data or missing data

```
Education      0
Occupation     0
Salary         0
dtype: int64
```

Education and occupation columns are categorical and salary is numerical, this data has no missing values.

Value count for the column 'Occupation' and 'Education'

Doctorate	16	Prof-specialty	13
Bachelors	15	Sales	12
HS-grad	9	Adm-clerical	10
		Exec-managerial	5
Name: Education, dtype: int64		Name: Occupation, dtype: int64	

Descriptive statistics

	Education	Occupation	Salary
count	40	40	40.000000
unique	3	4	NaN
top	Doctorate	Prof-specialty	NaN
freq	16	13	NaN
mean	NaN	NaN	162186.875000
std	NaN	NaN	64860.407506
min	NaN	NaN	50103.000000
25%	NaN	NaN	99897.500000
50%	NaN	NaN	169100.000000
75%	NaN	NaN	214440.750000
max	NaN	NaN	260151.000000

As from the data obtained from descriptive statistics, salary is the numerical data, average salary is 162186.875 and the minimum and maximum salary as per the data is 50103 and 260151.

Checking for duplicates

Number of duplicate rows = 0

Education	Occupation	Salary
-----------	------------	--------

1.1 State the null and the alternate hypothesis for conducting one-way ANOVA for both Education and Occupation individually.

Hypothesis for Education:

H_0 (Null Hypothesis) = $\mu(\text{Doctorate}) = \mu(\text{Bachelors}) = \mu(\text{HS-grad})$

H_a (Alternate Hypothesis) = At least one pair of means is not equal

Hypothesis for Occupation:

H_0 (Null Hypothesis) = $\mu(\text{Adm-clerical}) = \mu(\text{Sales}) = \mu(\text{Prof-specialty}) = \mu(\text{Exec-managerial})$

H_a (Alternate Hypothesis) = At least one pair of means is not equal

1.2 Perform one-way ANOVA for Education with respect to the variable 'Salary'. State whether the null hypothesis is accepted or rejected based on the ANOVA results.

	df	sum_sq	mean_sq	F	PR(>F)
C(Education)	2.0	1.026955e+11	5.134773e+10	30.95628	1.257709e-08
Residual	37.0	6.137256e+10	1.658718e+09	NaN	NaN

The P-value is 1.257709e-08 is smaller than the level of significance 0.05. So, null hypothesis is rejected based on the p-value and mean salary is not same across all 3 education levels.

1.3 Perform one-way ANOVA for variable Occupation with respect to the variable 'Salary'. State whether the null hypothesis is accepted or rejected based on the ANOVA results.

	df	sum_sq	mean_sq	F	PR(>F)
C(Occupation)	3.0	1.125878e+10	3.752928e+09	0.884144	0.458508
Residual	36.0	1.528092e+11	4.244701e+09	NaN	NaN

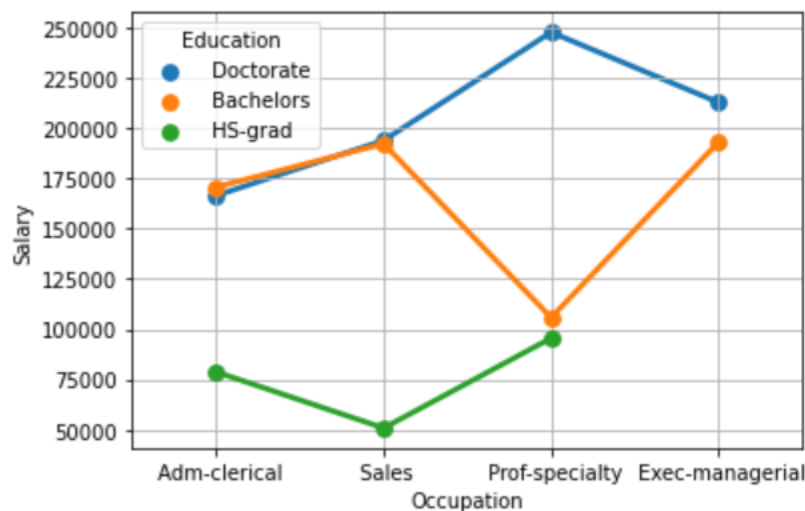
There is no sufficient evidence to reject the null hypothesis as the P-value is greater than the level of significance 0.05.

1.4 If the null hypothesis is rejected in either (1.2) or in (1.3), find out which class means are significantly different. Interpret the result.

Education		Occupation	
Bachelors	165152.933333	Adm-clerical	141424.300000
Doctorate	208427.000000	Exec-managerial	197117.600000
HS-grad	75038.777778	Prof-specialty	168953.153846
Name: Salary, dtype: float64		Sales	157604.416667
		Name: Salary, dtype: float64	

Population mean salary = 162186.875. Individual mean for the occupation lies to population mean. Hence, null hypothesis stay true for the occupation. Individual mean for education doesn't lie closer to population mean. Hence, null hypothesis is rejected for education.

1.5 What is the interaction between the two treatments? Analyze the effects of one variable on the other (Education and Occupation) with the help of an interaction plot.



From the above-mentioned graph, There is interaction between education field Doctorate and Bachelors with respect to Adm-clerical and Sales based on the salary numerical field. Apart from these, there is no interaction between other columns.

1.6 Perform a two-way ANOVA based on the Education and Occupation (along with their interaction Education*Occupation) with the variable 'Salary'. State the null and alternative hypotheses and state your results. How will you interpret this result?

	df	sum_sq	mean_sq	F	PR(>F)
C(Education)	2.0	1.026955e+11	5.134773e+10	72.211958	5.466264e-12
C(Occupation)	3.0	5.519946e+09	1.839982e+09	2.587626	7.211580e-02
C(Education):C(Occupation)	6.0	3.634909e+10	6.058182e+09	8.519815	2.232500e-05
Residual	29.0	2.062102e+10	7.110697e+08	NaN	NaN

Based on the interaction between education and occupation, in which P-value is 2.232500e-05 which smaller than significance level 0.05. We have sufficient evidence to reject null hypothesis based on the observation.

1.7 Explain the business implications of performing ANOVA for this particular case study

The business implication of performing anova is that education is the important factor impacting salary across different occupation, interaction between occupation and education has slight interaction with the salary on few observation. Salary play vital role in this data, being only categorical data to depend upon to know proper observation.

Anova gives better understanding by calculating the interaction of the variable like education and occupation to salary. The close relation with education and occupation to salary can be determined very easily with anova.

The least earned category is HS-grad from all the occupation, it understood that every business chooses or prefers a highly educated candidate who will be willing to put all his effort and stuff towards the growth of the business and effort in training when compared to doctorate or bachelors. Who posses the best professional skills, they are offered with higher salary. There is no interaction between HS-grad and Doctorate.

Problem 2:

The dataset [Education - Post 12th Standard.csv](#) contains information on various colleges. You are expected to do a Principal Component Analysis for this case study according to the instructions given. The data dictionary of the 'Education - Post 12th Standard.csv' can be found in the following file: [Data Dictionary.xlsx](#).

	Names	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.F.Ratio	perc.alumni	Expend	Grad.Rate
0	Abilene Christian University	1660.0	1232.0	721.0	23.0	52	2885.0	537.0	7440.0	3300.0	450.0	2200.0	70.0	78.0	18.1	12.0	7041.0	60.0
1	Adelphi University	2186.0	1924.0	512.0	16.0	29	2683.0	1227.0	12280.0	6450.0	750.0	1500.0	29.0	39.5	12.2	16.0	10527.0	56.0
2	Adrian College	1428.0	1097.0	336.0	22.0	50	1036.0	99.0	11250.0	3750.0	400.0	1165.0	53.0	66.0	12.9	30.0	8735.0	54.0
3	Agnes Scott College	417.0	349.0	137.0	60.0	89	510.0	63.0	12960.0	5450.0	450.0	875.0	92.0	97.0	7.7	37.0	16948.5	59.0
4	Alaska Pacific University	193.0	146.0	55.0	16.0	44	249.0	869.0	7560.0	4120.0	795.0	1500.0	76.0	72.0	11.9	2.0	10922.0	15.5

Checking for missing data

```
Names          0
Apps           0
Accept         0
Enroll         0
Top10perc      0
Top25perc      0
F.Undergrad    0
P.Undergrad    0
Outstate       0
Room.Board     0
Books          0
Personal       0
PhD            0
Terminal       0
S.F.Ratio      0
perc.alumni    0
Expend         0
Grad.Rate      0
dtype: int64
```


Checking for duplicates

Number of duplicate rows = 0

Education Occupation Salary

There are no duplicates found in the data.

Descriptive statistics of the data

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal
count	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000	777.000000
mean	2571.352638	1746.280566	660.388674	26.842986	55.796654	2935.648005	655.884170	10440.196268	4355.438224	539.425997	1323.790219
std	2422.195279	1523.286632	570.126836	15.582539	19.804778	2700.233049	716.274014	4021.712447	1090.666009	115.229712	609.505876
min	81.000000	72.000000	35.000000	1.000000	9.000000	139.000000	1.000000	2340.000000	1780.000000	275.000000	250.000000
25%	776.000000	604.000000	242.000000	15.000000	41.000000	992.000000	95.000000	7320.000000	3597.000000	470.000000	850.000000
50%	1558.000000	1110.000000	434.000000	23.000000	54.000000	1707.000000	353.000000	9990.000000	4200.000000	500.000000	1200.000000
75%	3624.000000	2424.000000	902.000000	35.000000	69.000000	4005.000000	967.000000	12925.000000	5050.000000	600.000000	1700.000000
max	7896.000000	5154.000000	1892.000000	65.000000	100.000000	8524.500000	2275.000000	21332.500000	7229.500000	795.000000	2975.000000

PhD	Terminal	S.F.Ratio	perc.alumni	Expend	Grad.Rate
777.000000	777.000000	777.000000	777.000000	777.000000	777.000000
72.774775	79.782497	14.051223	22.722008	9182.523810	65.468468
15.953120	14.473057	3.784212	12.325480	3396.496148	17.142538
27.500000	39.500000	4.000000	0.000000	3186.000000	15.500000
62.000000	71.000000	11.500000	13.000000	6751.000000	53.000000
75.000000	82.000000	13.600000	21.000000	8377.000000	65.000000
85.000000	92.000000	16.500000	31.000000	10830.000000	78.000000
103.000000	100.000000	24.000000	58.000000	16948.500000	115.500000

Info of the data

Data columns (total 18 columns):

#	Column	Non-Null Count	Dtype
0	Names	777 non-null	object
1	Apps	777 non-null	int64
2	Accept	777 non-null	int64
3	Enroll	777 non-null	int64
4	Top10perc	777 non-null	int64
5	Top25perc	777 non-null	int64
6	F.Undergrad	777 non-null	int64
7	P.Undergrad	777 non-null	int64
8	Outstate	777 non-null	int64
9	Room.Board	777 non-null	int64
10	Books	777 non-null	int64
11	Personal	777 non-null	int64
12	PhD	777 non-null	int64
13	Terminal	777 non-null	int64
14	S.F.Ratio	777 non-null	float64
15	perc.alumni	777 non-null	int64
16	Expend	777 non-null	int64
17	Grad.Rate	777 non-null	int64

dtypes: float64(1), int64(16), object(1)
memory usage: 109.4+ KB

This data has 777 rows and 18 columns where Names is categorical and others are numerical. According to above screenshots there is no missing and duplicate values.

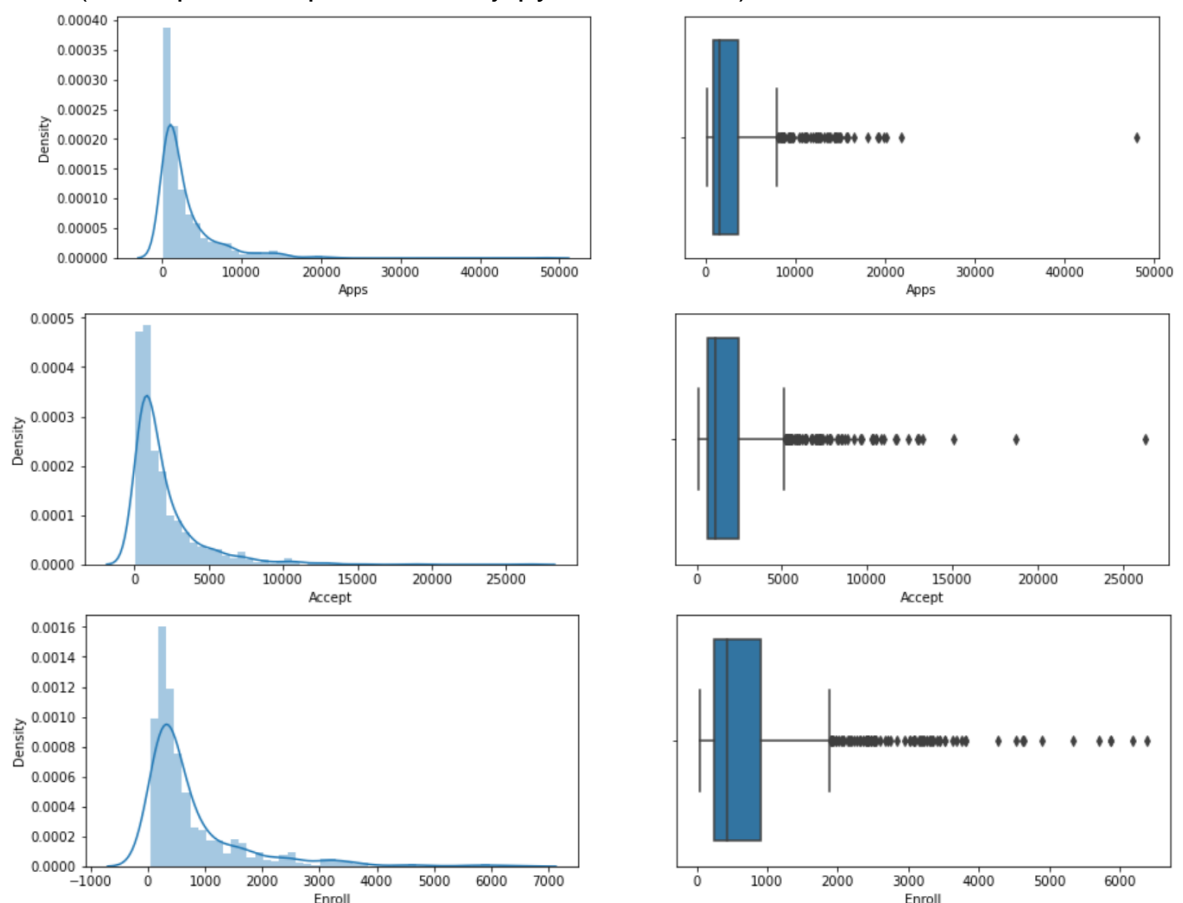
2.1 Perform Exploratory Data Analysis [both univariate and multivariate analysis to be performed]. What insight do you draw from the EDA?

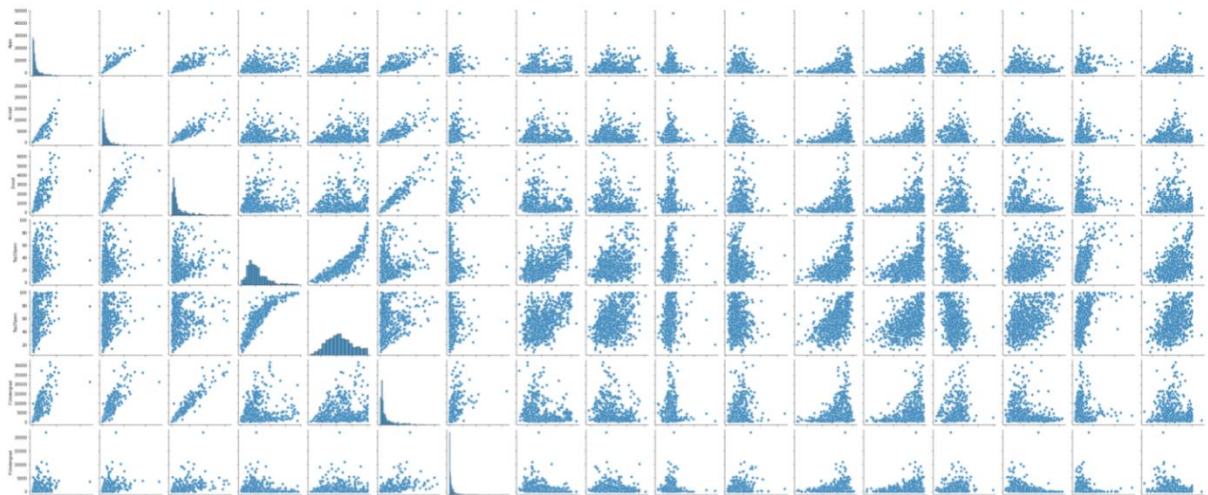
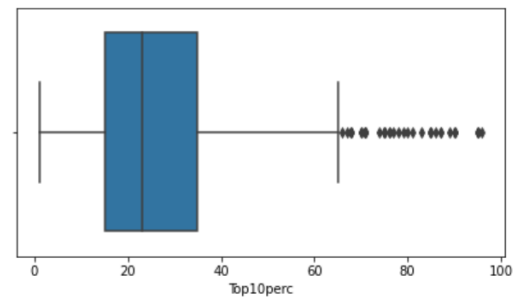
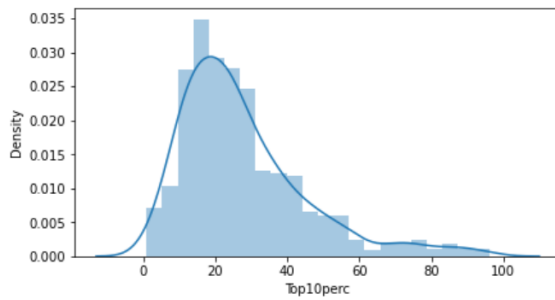
Univariate Analysis

The function is used to display information as part of univariate analysis of numeric variables. This will accept column name and number of bins as arguments.

The statistical description of the variable, histogram or distplot to view the distribution and box to view any outlier are present and if outlier present, proceed with the outlier treatment.

Distplot and boxplot of Accept, Apps, Enroll and Top25perc are only displayed here. (Other plots are presented in jupyter notebook.)





The data observed from the above plots, Top25 perc is only variables which has no outliers. Outstate, Alumni donations and graduation rate variables has less outliers.

All the variables are right skewed, except Terminal and PhD which is left skewed and graduation rate represent a normal distr

Bivariate Analysis / Multivariant analysis

Comparing multiple variables simultaneously is also another useful way to understand your data. When you have two continuous variables, a scatter plot / heatmap is usually used.

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Te
Apps	1.000000	0.943451	0.846822	0.338834	0.351640	0.814491	0.398264	0.050159	0.164939	0.132559	0.178731	0.390697	0.369491
Accept	0.943451	1.000000	0.911637	0.192447	0.247476	0.874223	0.441271	-0.025755	0.090899	0.113525	0.200989	0.355758	0.337583
Enroll	0.846822	0.911637	1.000000	0.181294	0.226745	0.964640	0.513069	-0.155477	-0.040232	0.112711	0.280929	0.331469	0.308274
Top10perc	0.338834	0.192447	0.181294	1.000000	0.891995	0.141289	-0.105356	0.562331	0.371480	0.118858	-0.093316	0.531828	0.491135
Top25perc	0.351640	0.247476	0.226745	0.891995	1.000000	0.199445	-0.053577	0.489394	0.331490	0.115527	-0.080810	0.545862	0.524749
F.Undergrad	0.814491	0.874223	0.964640	0.141289	0.199445	1.000000	0.570512	-0.215742	-0.068890	0.115550	0.317200	0.318337	0.300019
P.Undergrad	0.398264	0.441271	0.513069	-0.105356	-0.053577	0.570512	1.000000	-0.253512	-0.061326	0.081200	0.319882	0.149114	0.141904
Outstate	0.050159	-0.025755	-0.155477	0.562331	0.489394	-0.215742	-0.253512	1.000000	0.654256	0.038855	-0.299087	0.382982	0.407983
Room.Board	0.164939	0.090899	-0.040232	0.371480	0.331490	-0.068890	-0.061326	0.654256	1.000000	0.127963	-0.199428	0.329202	0.374540
Books	0.132559	0.113525	0.112711	0.118858	0.115527	0.115550	0.081200	0.038855	0.127963	1.000000	0.179295	0.026906	0.099955
Personal	0.178731	0.200989	0.280929	-0.093316	-0.080810	0.317200	0.319882	-0.299087	-0.199428	0.179295	1.000000	-0.010936	0.030613
PhD	0.390697	0.355758	0.331469	0.531828	0.545862	0.318337	0.149114	0.382982	0.329202	0.026906	-0.010936	1.000000	0.849587
Terminal	0.369491	0.337583	0.308274	0.491135	0.524749	0.300019	0.141904	0.407983	0.374540	0.099955	-0.030613	0.849587	1.000000
S.F.Ratio	0.095633	0.176229	0.237271	-0.384875	-0.294629	0.279703	0.232531	-0.554821	-0.362628	-0.031929	0.136345	-0.130530	-0.031929
perc.alumni	-0.090226	-0.159990	-0.180794	0.455485	0.417864	-0.229462	-0.280792	0.566262	0.272363	-0.040208	-0.285968	0.249009	0.249009

<AxesSubplot:>



From above heat map, there is positive linear relationship are exhibited by the some variable pairs. We can relate variable based on the positive and negative correlation.

2.2 Is scaling necessary for PCA in this case? Give justification and perform scaling.

- Yes, scaling is necessary for PCA to normalise the data. The data contains variables indicates the number of students (Apps, Accept, Enroll, F.undergrad, etc,..) and the remaining variables such as book, personal etc.. it is difficult to compare the data, we need to scale the data as necessary.
- In this case study, Standard Z- score is used which converts the group of data in our distribution such that mean is 0 and standard deviation is 1.It

converts the dataset within range of (-3,-3) provided the dataset is free from outliers or skewness.

- Before scaling the data, outliers must be checked and removed to proceed further.
- No specific method was not given. So, I applied apply(z-score) or can done with std scaler or minmax scaler.

```
def remove_outlier(col):
    sorted(col)
    Q1,Q3=col.quantile([0.25,0.75])
    IQR = Q3-Q1
    lower_range = Q1-(1.5*IQR)
    upper_range = Q3+(1.5*IQR)
    return lower_range, upper_range
```

```
lrTerminal,urTerminal=remove_outlier(edu_post['Terminal'])
edu_post['Terminal']=np.where(edu_post['Terminal']>urTerminal,urTerminal,edu_post['Terminal'])
edu_post['Terminal']=np.where(edu_post['Terminal']<lrTerminal,lrTerminal,edu_post['Terminal'])
```

```
lrRatio,urRatio=remove_outlier(edu_post['S.F.Ratio'])
edu_post['S.F.Ratio']=np.where(edu_post['S.F.Ratio']>urRatio,urRatio,edu_post['S.F.Ratio'])
edu_post['S.F.Ratio']=np.where(edu_post['S.F.Ratio']<lrRatio,lrRatio,edu_post['S.F.Ratio'])
```

```
lralumni,uralumni=remove_outlier(edu_post['perc.alumni'])
edu_post['perc.alumni']=np.where(edu_post['perc.alumni']>uralumni,uralumni,edu_post['perc.alumni'])
edu_post['perc.alumni']=np.where(edu_post['perc.alumni']<lralumni,lralumni,edu_post['perc.alumni'])
```

```
lrExpend,urExpend=remove_outlier(edu_post['Expend'])
edu_post['Expend']=np.where(edu_post['Expend']>urExpend,urExpend,edu_post['Expend'])
edu_post['Expend']=np.where(edu_post['Expend']<lrExpend,lrExpend,edu_post['Expend'])
```

```
lrGRate,urGRate=remove_outlier(edu_post['Grad.Rate'])
edu_post['Grad.Rate']=np.where(edu_post['Grad.Rate']>urGRate,urGRate,edu_post['Grad.Rate'])
edu_post['Grad.Rate']=np.where(edu_post['Grad.Rate']<lrGRate,lrGRate,edu_post['Grad.Rate'])
```

```
edu_post.shape
```

```
(777, 18)
```

```
lrapps,urapps=remove_outlier(edu_post['Apps'])
edu_post['Apps']=np.where(edu_post['Apps']>urapps,urapps,edu_post['Apps'])
edu_post['Apps']=np.where(edu_post['Apps']<lrapps,lrapps,edu_post['Apps'])
```

```
lraccept,uraccept=remove_outlier(edu_post['Accept'])
edu_post['Accept']=np.where(edu_post['Accept']>uraccept,uraccept,edu_post['Accept'])
edu_post['Accept']=np.where(edu_post['Accept']<lraccept,lraccept,edu_post['Accept'])
```

```
lr enroll,urenroll=remove_outlier(edu_post['Enroll'])
edu_post['Enroll']=np.where(edu_post['Enroll']>urenroll,urenroll,edu_post['Enroll'])
edu_post['Enroll']=np.where(edu_post['Enroll']<lr enroll,lr enroll,edu_post['Enroll'])
```

```
lr top10,ur top10=remove_outlier(edu_post['Top10perc'])
edu_post['Top10perc']=np.where(edu_post['Top10perc']>ur top10,ur top10,edu_post['Top10perc'])
edu_post['Top10perc']=np.where(edu_post['Top10perc']<lr top10,lr top10,edu_post['Top10perc'])
```

```
lr fund,urfund=remove_outlier(edu_post['F.Undergrad'])
edu_post['F.Undergrad']=np.where(edu_post['F.Undergrad']>urfund,urfund,edu_post['F.Undergrad'])
edu_post['F.Undergrad']=np.where(edu_post['F.Undergrad']<lr fund,lr fund,edu_post['F.Undergrad'])
```

```
lr pund,urpund=remove_outlier(edu_post['P.Undergrad'])
edu_post['P.Undergrad']=np.where(edu_post['P.Undergrad']>urpund,urpund,edu_post['P.Undergrad'])
edu_post['P.Undergrad']=np.where(edu_post['P.Undergrad']<lr pund,lr pund,edu_post['P.Undergrad'])
```

```
lrout,urout=remove_outlier(edu_post['Outstate'])
edu_post['Outstate']=np.where(edu_post['Outstate']>urout,urout,edu_post['Outstate'])
edu_post['Outstate']=np.where(edu_post['Outstate']<lrout,lrout,edu_post['Outstate'])
```

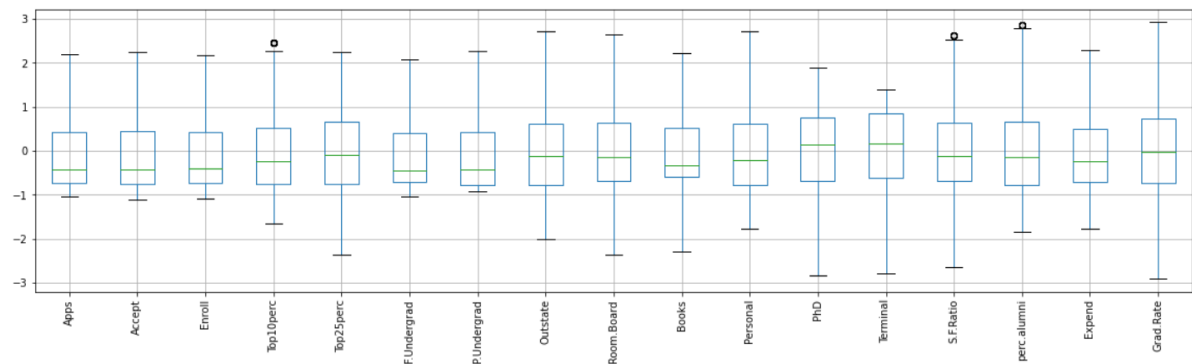
```
lr room,urroom=remove_outlier(edu_post['Room.Board'])
edu_post['Room.Board']=np.where(edu_post['Room.Board']>urroom,urroom,edu_post['Room.Board'])
edu_post['Room.Board']=np.where(edu_post['Room.Board']<lr room,lr room,edu_post['Room.Board'])
```

```
lr books,urbooks=remove_outlier(edu_post['Books'])
edu_post['Books']=np.where(edu_post['Books']>urbooks,urbooks,edu_post['Books'])
edu_post['Books']=np.where(edu_post['Books']<lr books,lr books,edu_post['Books'])
```

```
lrpersonal,urpersonal=remove_outlier(edu_post['Personal'])
edu_post['Personal']=np.where(edu_post['Personal']>urpersonal,urpersonal,edu_post['Personal'])
edu_post['Personal']=np.where(edu_post['Personal']<lrpersonal,lrpersonal,edu_post['Personal'])
```

```
lrPhD,urPhD=remove_outlier(edu_post['PhD'])
edu_post['PhD']=np.where(edu_post['PhD']>urPhD,urPhD,edu_post['PhD'])
edu_post['PhD']=np.where(edu_post['PhD']<lrPhD,lrPhD,edu_post['PhD'])
```


	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.F.Ratio	perc.alumni	Expend	Grad.Rate
0	-0.376493	-0.337830	0.106380	-0.246780	-0.191827	-0.018769	-0.166083	-0.746480	-0.968324	-0.776567	1.438500	-0.174045	-0.123239	1.070602	-0.870466	-0.630916	-0.319205
1	-0.159195	0.116744	-0.260441	-0.696290	-1.353911	-0.093626	0.797856	0.457762	1.921680	1.828605	0.289289	-2.745731	-2.785068	-0.489511	-0.545726	0.396097	-0.552693
2	-0.472336	-0.426511	-0.569343	-0.310996	-0.292878	-0.703966	-0.777974	0.201488	-0.555466	-1.210762	-0.260691	-1.240354	-0.952900	-0.304413	0.590864	-0.131845	-0.669437
3	-0.889994	-0.917871	-0.918613	2.129202	1.677612	-0.898889	-0.828267	0.626954	1.004218	-0.776567	-0.736792	1.205884	1.190391	-1.679429	1.159159	2.287940	-0.377577
4	-0.982532	-1.051221	-1.062533	-0.696290	-0.596031	-0.995610	0.297726	-0.716623	-0.216006	2.219381	0.289289	0.202299	-0.538069	-0.568839	-1.682316	0.512468	-2.916759



2.3 Comment on the comparison between the covariance and the correlation matrices from this data. [on scaled data]

```
cov_matrix = np.cov(df3.T)
cov_matrix
```

```
array([[ 1.00128866e+00,  9.56537704e-01,  8.98039052e-01,
         3.21756324e-01,  3.64960691e-01,  8.62111140e-01,
         5.20492952e-01,  6.54209711e-02,  1.87717056e-01,
         2.36441941e-01,  2.30243993e-01,  4.64521757e-01,
         4.35037784e-01,  1.26573895e-01, -1.01288006e-01,
         2.43248206e-01,  1.50997775e-01],
       [ 9.56537704e-01,  1.00128866e+00,  9.36482483e-01,
         2.23586208e-01,  2.74033187e-01,  8.98189799e-01,
         5.73428908e-01, -5.00874847e-03,  1.19740419e-01,
         2.08974091e-01,  2.56676290e-01,  4.27891234e-01,
         4.03929238e-01,  1.88748711e-01, -1.65728801e-01,
         1.62016688e-01,  7.90839722e-02],
       [ 8.98039052e-01,  9.36482483e-01,  1.00128866e+00,
         1.71977357e-01,  2.30730728e-01,  9.68548601e-01,
         6.42421828e-01, -1.55856056e-01, -2.38762560e-02,
         2.02317274e-01,  3.39785395e-01,  3.82031198e-01,
         3.54835877e-01,  2.74622251e-01, -2.23009677e-01,
         5.42906862e-02, -2.32810071e-02],
       [ 3.21756324e-01,  2.23586208e-01,  1.71977357e-01,
         1.00128866e+00,  9.15052977e-01,  1.11358019e-01,
        -1.80240778e-01,  5.62884044e-01,  3.57826139e-01,
```

```
df3.corr()
```

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Te
Apps	1.000000	0.955307	0.896883	0.321342	0.364491	0.861002	0.519823	0.065337	0.187475	0.236138	0.229948	0.463924	0.4
Accept	0.955307	1.000000	0.935277	0.223298	0.273681	0.897034	0.572691	-0.005002	0.119586	0.208705	0.256346	0.427341	0.4
Enroll	0.896883	0.935277	1.000000	0.171756	0.230434	0.967302	0.641595	-0.155655	-0.023846	0.202057	0.339348	0.381540	0.3
Top10perc	0.321342	0.223298	0.171756	1.000000	0.913875	0.111215	-0.180009	0.562160	0.357366	0.153452	-0.116730	0.544048	0.3
Top25perc	0.364491	0.273681	0.230434	0.913875	1.000000	0.181196	-0.099295	0.489569	0.330987	0.169761	-0.086810	0.551461	0.3
F.Undergrad	0.861002	0.897034	0.967302	0.111215	0.181196	1.000000	0.696130	-0.226166	-0.054476	0.207879	0.359783	0.361564	0.3
P.Undergrad	0.519823	0.572691	0.641595	-0.180009	-0.099295	0.696130	1.000000	-0.354216	-0.067638	0.122529	0.344053	0.127663	0.2
Outstate	0.065337	-0.005002	-0.155655	0.562160	0.489569	-0.226166	-0.354216	1.000000	0.655489	0.005110	-0.325609	0.391321	0.4
Room.Board	0.187475	0.119586	-0.023846	0.357366	0.330987	-0.054476	-0.067638	0.655489	1.000000	0.108924	-0.219554	0.341469	0.3
Books	0.236138	0.208705	0.202057	0.153452	0.169761	0.207879	0.122529	0.005110	0.108924	1.000000	0.239863	0.136390	0.2
Personal	0.229948	0.256346	0.339348	-0.116730	-0.086810	0.359783	0.344053	-0.325609	-0.219554	0.239863	1.000000	-0.011684	-0.0
PhD	0.463924	0.427341	0.381540	0.544048	0.551461	0.361564	0.127663	0.391321	0.341469	0.136390	-0.011684	1.000000	0.8
Terminal	0.434478	0.403100	0.354370	0.508748	0.527854	0.325054	0.130153	0.110570	0.230370	0.150318	0.234074	0.882028	1.0

- The inference is that Correlation and covariance matrix values are same for the values after scaling. correlation becomes scaled after deriving covariance.

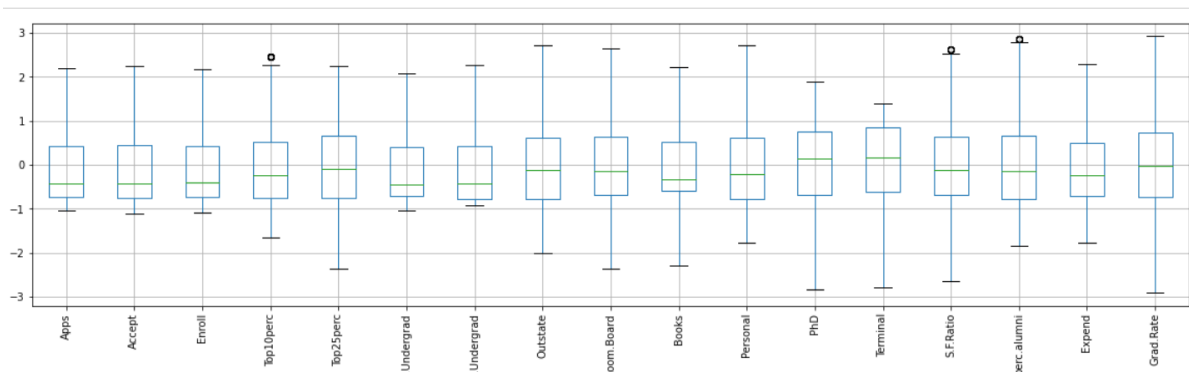
2.4 Check the dataset for outliers before and after scaling. What insight do you derive here?

Data before scaling

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.F.Ratio	perc.alumni
0	1660.0	1232.0	721.0	23.0	52	2885.0	537.0	7440.0	3300.0	450.0	2200.0	70.0	78.0	18.1	12.0
1	2186.0	1924.0	512.0	16.0	29	2683.0	1227.0	12280.0	6450.0	750.0	1500.0	29.0	39.5	12.2	16.0
2	1428.0	1097.0	336.0	22.0	50	1036.0	99.0	11250.0	3750.0	400.0	1165.0	53.0	66.0	12.9	30.0
3	417.0	349.0	137.0	60.0	89	510.0	63.0	12960.0	5450.0	450.0	875.0	92.0	97.0	7.7	37.0
4	193.0	146.0	55.0	16.0	44	249.0	869.0	7560.0	4120.0	795.0	1500.0	76.0	72.0	11.9	2.0

Data after scaling

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.
0	-0.376493	-0.337830	0.106380	-0.246780	-0.191827	-0.018769	-0.166083	-0.746480	-0.968324	-0.776567	1.438500	-0.174045	-0.123239	1.0
1	-0.159195	0.116744	-0.260441	-0.696290	-1.353911	-0.093626	0.797856	0.457762	1.921680	1.828605	0.289289	-2.745731	-2.785068	-0.0
2	-0.472336	-0.426511	-0.569343	-0.310996	-0.292878	-0.703966	-0.777974	0.201488	-0.555466	-1.210762	-0.260691	-1.240354	-0.952900	-0.0
3	-0.889994	-0.917871	-0.918613	2.129202	1.677612	-0.898889	-0.828267	0.626954	1.004218	-0.776567	-0.736792	1.205884	1.190391	-1.0
4	-0.982532	-1.051221	-1.062533	-0.696290	-0.596031	-0.995610	0.297726	-0.716623	-0.216006	2.219381	0.289289	0.202299	-0.538069	-0.0



- As the data before scaling was difficult to understand, the outliers were identified using boxplot, from which data was not easy to read or proceed with future steps with outliers. Expect Top25perc, all other found to have outliers.
- Without scaling, In boxplot it was not that easy to read Inter Quartile range, Median, Min, Max value. We can observe that there is still some outliers are present even after the scaling has been done.
- After scaling, the data was easy to understand without outliers and was easy to proceed as most of the variable was free from the outliers.

2.5 Extract the eigenvalues and eigenvectors. [Using Sklearn PCA Print Both]

Eigen Values

```
[5.6625219  4.89470815 1.12636744 1.00397659 0.87218426 0.7657541
0.58491404 0.5445048  0.42352336 0.38101777 0.24701456 0.02239369
0.03789395 0.14726392 0.13434483 0.09883384 0.07469003]
```

Eigen Vector

```
[[-2.62171542e-01  3.14136258e-01  8.10177245e-02 -9.87761685e-02
-2.19898081e-01  2.18800617e-03 -2.83715076e-02 -8.99498102e-02
1.30566998e-01 -1.56464458e-01 -8.62132843e-02  1.82169814e-01
-5.99137640e-01  8.99775288e-02  8.88697944e-02  5.49428396e-01
5.41453698e-03]
[-2.30562461e-01  3.44623583e-01  1.07658626e-01 -1.18140437e-01
-1.89634940e-01 -1.65212882e-02 -1.29584896e-02 -1.37606312e-01
1.42275847e-01 -1.49209799e-01 -4.25899061e-02 -3.91041719e-01
6.61496927e-01  1.58861886e-01  4.37945938e-02  2.91572312e-01
1.44582845e-02]
[-1.89276397e-01  3.82813322e-01  8.55296892e-02 -9.30717094e-03
-1.62314818e-01 -6.80794143e-02 -1.52403625e-02 -1.44216938e-01
5.08712481e-02 -6.48997860e-02 -4.38408622e-02  7.16684935e-01
2.33235272e-01 -3.53988202e-02 -6.19241658e-02 -4.17001280e-01
-4.97908902e-02]
[-3.38874521e-01 -9.93191661e-02 -7.88293849e-02  3.69115031e-01
-1.57211016e-01 -8.88656824e-02 -2.57455284e-01  2.89538833e-01
-1.22467790e-01 -3.58776186e-02  1.77837341e-03 -5.62053913e-02
2.21448729e-02 -3.92277722e-02  6.99599977e-02  8.79767299e-03
-7.23645373e-01]
[-3.34690532e-01 -5.95055011e-02 -5.07938247e-02  4.16824361e-01
-1.44449474e-01 -2.76268979e-02 -2.39038849e-01  3.45643551e-01
-1.93936316e-01  6.41786425e-03 -1.02127328e-01  1.96735274e-02
3.22646978e-02  1.45621999e-01 -9.70282598e-02 -1.07779150e-02
6.55464648e-01]
[-1.63293010e-01  3.98636372e-01  7.37077827e-02 -1.39504424e-02
-1.02728468e-01 -5.16468727e-02 -3.11751439e-02 -1.08748900e-01
1.45452749e-03 -1.63981359e-04 -3.49993487e-02 -5.42774834e-01
-3.67681187e-01 -1.33555923e-01 -8.71753137e-02 -5.70683843e-01
2.53059904e-02]
[-2.24797091e-02  3.57550046e-01  4.03568700e-02 -2.25351078e-01
9.56790178e-02 -2.45375721e-02 -1.00138971e-02  1.23841696e-01
-6.34774326e-01  5.46346279e-01  2.52107094e-01  2.95029745e-02
2.62494456e-02  5.02487566e-02  4.45537493e-02  1.46321060e-01
-3.97146972e-02]
[-2.83547285e-01 -2.51863617e-01  1.49394795e-02 -2.62975384e-01
-3.72750885e-02 -2.03860462e-02  9.45370782e-02  1.12721477e-02
-8.36648339e-03 -2.31799759e-01  5.93433149e-01  1.03393587e-03
-8.14247697e-02  5.60392799e-01  6.72405494e-02 -2.11561014e-01
-1.59275617e-03]
[-2.44186588e-01 -1.31909124e-01 -2.11379165e-02 -5.80894132e-01
```

- The eigen values and eigen vector are obtained from the co-variance matrix using the np.linalg.eig(covariance matrix) function
- By extracting eigen values and vector, the efficiency of the data is improved.
- Then, dimension reduction is performed.

2.6 Perform PCA and export the data of the Principal Component (eigenvectors) into a data frame with the original features

Dimension Reduction

```
array([[ -1.60249937, -1.80467545, -1.60828257, ..., -0.57688267,
        6.570952 , -0.47739307],
       [ 0.99368301, -0.07041499, -1.38279212, ..., 0.01779846,
        -1.18493014, 1.04394672],
       [ 0.03004476, 2.12212752, -0.50151255, ..., 0.32216034,
        1.32596561, -1.42543835],
       ...,
       [-0.36688624, 2.4532119 , 0.76599685, ..., 0.17522459,
        1.36851658, 0.7209176 ],
       [-0.69747582, 0.99485851, -1.02623665, ..., 0.50404279,
        -0.8227456 , 1.0518097 ],
       [ 0.71061626, -0.39608317, -0.16531057, ..., -1.45835209,
        1.20132639, 1.07308672]])
```

```
pca.components_
```

```
array([[ 2.62171542e-01,  2.30562461e-01,  1.89276397e-01,
        3.38874521e-01,  3.34690532e-01,  1.63293010e-01,
        2.24797091e-02,  2.83547285e-01,  2.44186588e-01,
        9.67082754e-02, -3.52299594e-02,  3.26410696e-01,
        3.23115980e-01, -1.63151642e-01,  1.86610828e-01,
        3.28955847e-01,  2.38822447e-01],
       [ 3.14136258e-01,  3.44623583e-01,  3.82813322e-01,
        -9.93191661e-02, -5.95055011e-02,  3.98636372e-01,
        3.57550046e-01, -2.51863617e-01, -1.31909124e-01,
        9.39739472e-02,  2.32439594e-01,  5.51390195e-02,
        4.30332048e-02,  2.59804556e-01, -2.57092552e-01,
        -1.60008951e-01, -1.67523664e-01],
       [-8.10177245e-02, -1.07658626e-01, -8.55296892e-02,
        7.88293849e-02,  5.07938247e-02, -7.37077827e-02,
        -4.03568700e-02, -1.49394795e-02,  2.11379165e-02,
        6.97121128e-01,  5.30972806e-01, -8.11134044e-02,
        -5.89785929e-02, -2.74150657e-01, -1.03715887e-01,
        1.84205687e-01, -2.45335837e-01],
       [ 9.87761685e-02,  1.18140437e-01,  9.30717094e-03,
        -3.69115031e-01, -4.16824361e-01,  1.39504424e-02,
        2.25351078e-01,  2.62975384e-01,  5.80894132e-01,
        -3.61562884e-02, -1.14982973e-01, -1.47260891e-01,
        -8.90079921e-02, -2.59486122e-01, -2.23982467e-01,
        2.13756140e-01, -3.61915064e-02],
```

```
pca.explained_variance_ratio_
array([0.33266084, 0.28755345, 0.06617164, 0.05898144, 0.05123893,
       0.04498639, 0.03436243])

df_pcaom = pd.DataFrame(pca.components_, columns=list(df3))
df_pcaom.shape
df_pcaom.head()
```

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.
0	0.262172	0.230562	0.189276	0.338875	0.334691	0.163293	0.022480	0.283547	0.244187	0.096708	-0.035230	0.326411	0.323116	-0.
1	0.314136	0.344624	0.382813	-0.099319	-0.059506	0.398636	0.357550	-0.251864	-0.131909	0.093974	0.232440	0.055139	0.043033	0.
2	-0.081018	-0.107659	-0.085530	0.078829	0.050794	-0.073708	-0.040357	-0.014939	0.021138	0.697121	0.530973	-0.081113	-0.058979	-0.
3	0.098776	0.118140	0.009307	-0.369115	-0.416824	0.013950	0.225351	0.262975	0.580894	-0.036156	-0.114983	-0.147261	-0.089008	-0.
4	0.219898	0.189635	0.162315	0.157211	0.144449	0.102728	-0.095679	0.037275	-0.069108	0.035406	-0.000475	-0.550787	-0.590407	-0.

- PCA was performed. According to explained variance ratio, the first 0.33 and the second cover 0.29 and so on.
- Then pc components are converted into dataframe and the data are obtained.

2.7 Write down the explicit form of the first PC (in terms of the eigenvectors. Use values with two places of decimals only). [hint: write the linear equation of PC in terms of eigenvectors and corresponding features]

Cumulative variance Explained

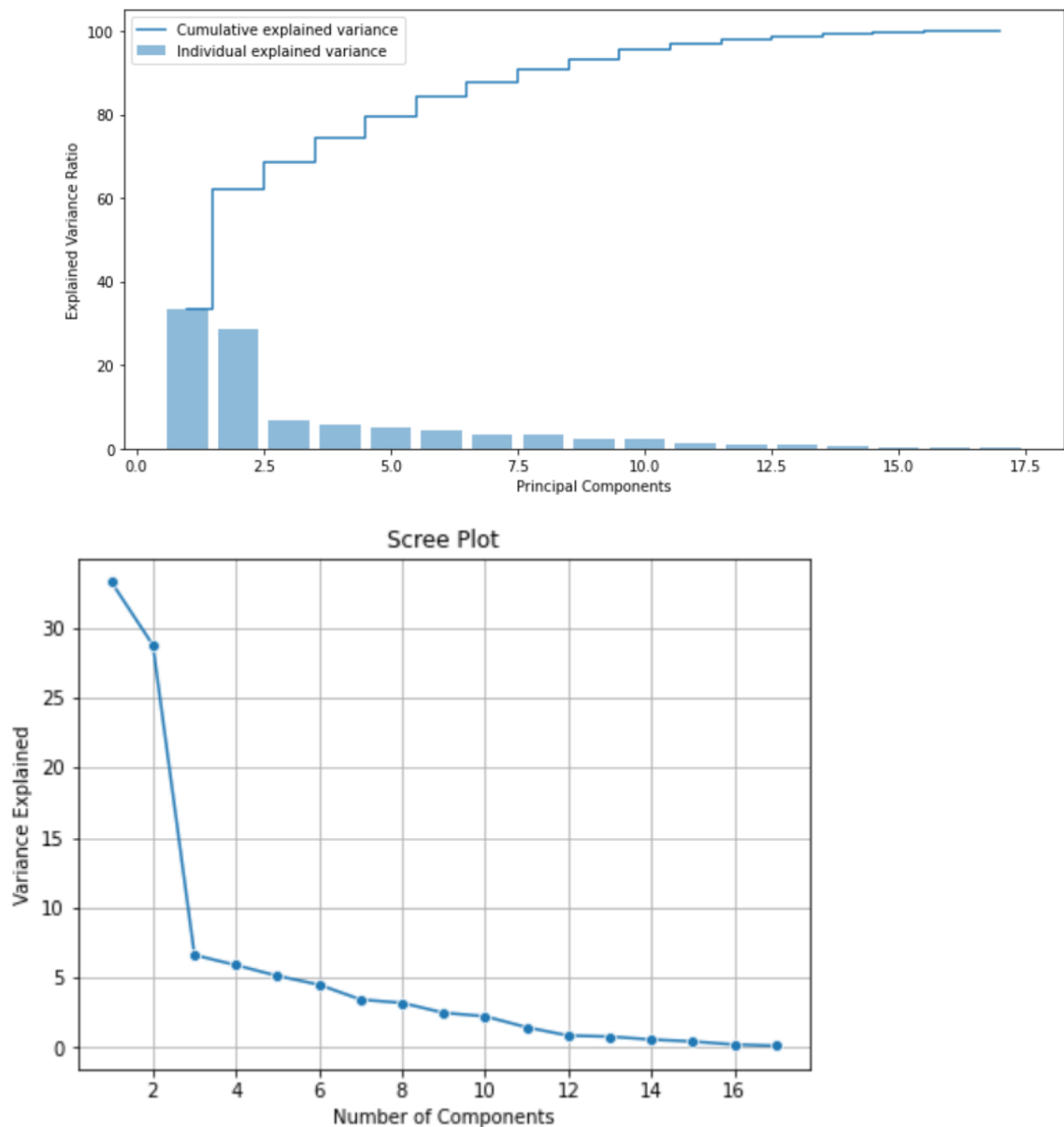
```
[ 33.27  62.02  68.64  74.54  79.66  84.16  87.6   90.79  93.28  95.52
 96.97  97.84  98.63  99.21  99.65  99.87 100. ]
```

- The first cumulative variance of 33 % is the highest among the variance, next 62 % is the variance obtained from the remaining data after 33 % has been obtained.

2.8 Consider the cumulative values of the eigenvalues. How does it help you to decide on the optimum number of principal components? What do the eigenvectors indicate?

Cumulative variance Explained

```
[ 33.27  62.02  68.64  74.54  79.66  84.16  87.6   90.79  93.28  95.52
 96.97  97.84  98.63  99.21  99.65  99.87 100. ]
```



We have plotted graph against the explained variance ratio which indicates the cumulative explained variance as well as individual explained variance for each pc, which done based on the cumulative explained variance and individual explained variance.

2.9 Explain the business implication of using the Principal Component Analysis for this case study. How may PCs help in the further analysis? [Hint: Write Interpretations of the Principal Components Obtained]

- Principal Component Analysis is basically a statistical procedure to convert a set of observation of possibly correlated variables into a set of values of linearly uncorrelated variables. It used to find the inter-relationship between variable, interpret and visualize the data.
- They are basically performed on a square symmetric matrix, pure sums of square and cross products matrix (Covariance and Correlation matrix).

- The co-variance matrix is then converted into eigen values and eigen vectors.
- The cumulative variance is the most important part of PCA as it provides us the information on the number of components we need to use for dimension reduction.
- The cumulative variance is at least 80 % then, we could say that it can successful PCA. After the component are obtained, plotting the components using scree plot to show the better visualization on how the variability is obtained.