AS PROJECT

by

Kajal Dusseja

PGP DSBA

Febuary’22

15-05-2022

greatlearning
Learning for Life

Table of Contents

Problem 1.. 4

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. 4

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 .... -

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 5

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

Problem 2.. **8**

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 10

2.3 Comment on the comparison between the covariance and the correlation matrices from this data.[on scaled data] 11

2.4 Check the dataset for outliers before and after scaling. What insight do you derive here?.................................. 12

2.5 Extract the eigenvalues and eigenvectors. [Using Sklearn PCA Print Both] 13

2.6 Perform PCA and export the data of the Principal Component (eigenvectors) into a data frame with the original features 14

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] 15

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?.. 17

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]................................... 18

List of Figures

Figure 1.1: Salary Dataset info……………………………………………………………………………4
Figure 1.1.1: Categories in Education……………………………………………………………………4
Figure 1.1.2: Categories in Occupation…………………………………………………………………4
Figure 1.2.1: ANOVA Table output for Education………………………………………………………5
Figure 1.3.1: ANOVA Table output for Occupation.……………………………………………………5
Figure 1.5.1: Interaction plot between Education and Occupation……………………………………6
Figure 1.6.1: Two-way ANOVA Table………………………………………………………………………6
Figure 2.1: Education dataset Info…………………………………………………………………………8
Figure 2.2: Education Dataset Summary……………………………………………………………………8
Figure 2.1.1: Histplot of various factors for Education Dataset.……………………………………9
Figure 2.1.2: Boxplot of various factors for Education Dataset………………………………………9
Figure 2.1.3: Pairplot for multivariate analysis…………………………………………………………10
Figure 2.1.4: Heatmap for multivariate analysis…………………………………………………………10
Figure 2.2.1: Scaled data………………………………………………………………………………………11
Figure 2.3.1: Covariance Matrix………………………………………………………………………………12
Figure 2.3.2: Correlation Matrix………………………………………………………………………………12
Figure 2.4.1: Outliers before Scaling…………………………………………………………………………13
Figure 2.4.2: Outliers After Scaling…………………………………………………………………………13
Figure: 2.5.1: Eigen Vectors and Eigen values……………………………………………………………14
Figure: 2.6.1: PCA on Dataset…………………………………………………………………………………15
Figure: 2.6.2: Data in dataframe after PCA…………………………………………………………………15
Figure 2.7.1: Explicit form of PCA……………………………………………………………………………17
Figure 2.8.1: Output of PCA……………………………………………………………………………………18
Figure 2.8.2: PCA Data in Data frame………………………………………………………………………18

Problem 1

Problem Statement:

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.

[Assume that the data follows a normal distribution. In reality, the normality assumption may not always hold if the sample size is small.]

Information of Salary Dataset:

```
<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)
```

Figure 1.1: Salary Dataset Info

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

Approach followed	1. Identified the different population for independent factors Education and Occupation using 'Categorical' function from pandas library. 2. Stated null and alternate hypothesis for factors Education and Occupation to conduct one-way Anova.

Output	`In [10]: DF.Education = pd.Categorical(DF.Education)`
	`DF.Education.value_counts`
	`Name: Education, dtype: category`
	`Categories (3, object): [' Bachelors', ' Doctorate', ' HS-grad']`
	Figure 1.1.1: Categories in Education

	`In [9]: DF.Occupation = pd.Categorical(DF.Occupation)`
	`DF.Occupation.value_counts`
	`Name: Occupation, dtype: category`
	`Categories (4, object): [' Adm-clerical', ' Exec-managerial', ' Prof-specialty', ' Sales']`
	Figure 1.1.2: Categories in Occupation

Inference	**Null and alternate hypothesis for Education:** H_0: $\mu_a = \mu_b = \mu_c = \ldots = \mu_k$ (All population(categories within Education that is Bachelors, Doctorate and High-School Graduate) means are equal) H_a: $\mu_a \neq \mu_b \neq \mu_c \neq \ldots \neq \mu_k$ (Not all population means are equal, or least one pair is unequal) **Null and alternate hypothesis for Occupation:** H_0: $\mu_a = \mu_b = \mu_c = \ldots = \mu_k$ (All population(categories within occupation that is Administrative and clerical, Executive and managerial, Professional, Sales) means are equal) H_a: $\mu_a \neq \mu_b \neq \mu_c \neq \ldots \neq \mu_k$ (Not all population means are equal, or least one pair is unequal)

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.

Approach followed	1. One way ANOVA identifies effect of one variable in this case education. Using the ols function from statsmodels library the ANOVA table is generated for Education with respect to Salary. 2. The ratio of Variance between samples to variance within sample denoted by F determines whether null hypothesis is rejected/accepted. 3. The default confidence level (alpha = 0.05) is 95%.
Output	See code and output below. **Figure 1.2.1: ANOVA Table output for Education**
Inference	As it can been seen from the results above F value is less than 0.05 hence the null hypothesis for Education is rejected. That is to say population means for Education factors are not equal (at least one or all).

```
In [11]: formula = 'Salary ~ C(Education)'
         model = ols(formula, DF).fit()
         aov_table = anova_lm(model)
         print(aov_table)

                 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
```

Figure 1.2.1: ANOVA Table output for Education

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.

Approach followed	1. One way ANOVA identifies effect of one variable in this case occupation. Using the ols function from statsmodels library the ANOVA table is generated for Occupation with respect to Salary. 2. The ratio of Variance between samples to variance within sample denoted by F determines whether null hypothesis is rejected/accepted. 3. The default confidence level (alpha = 0.05) is 95%.
Output	See code and output below. **Figure 1.3.1: ANOVA Table output for Occupation**
Inference	As it can been seen from the results above F value is greater than 0.05 hence the null hypothesis for Occupation is accepted. That is to say population means for Occupation are equal.

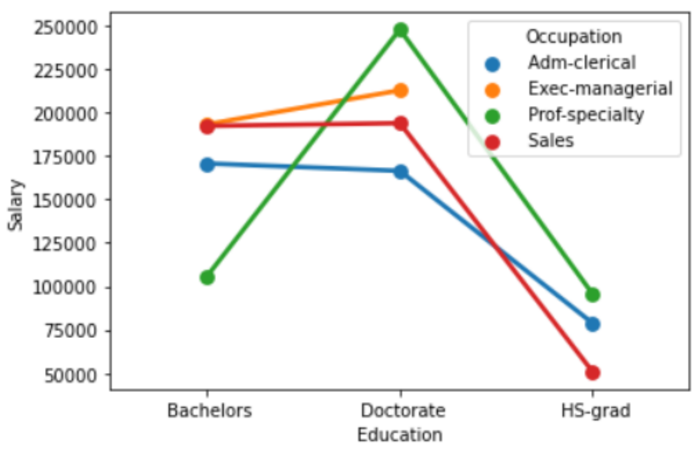
```
In [12]: formula = 'Salary ~ C(Occupation)'
         model = ols(formula, DF).fit()
         aov_table = anova_lm(model)
         print(aov_table)

                  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
```

Figure 1.3.1: ANOVA Table output for Occupation

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.

Approach followed	With the help of pointplot, an interaction plot between Education and Occupation is generated. This will help to analyze the effects between both factors.

Output of pointplot	<AxesSubplot:xlabel='Education', ylabel='Salary'>  **Figure 1.5.1: Interaction plot between Education and Occupation**
Inference	As can be seen from the pointplot, 1. For all occupations except Executive and Managerial, high school graduates have less salary than Bachelors and Doctorates. 2. A professional or specialist with bachelors has almost a similar salary as a high-school graduate. 3. Occupations of Administrative and clerical, Sales have same similar salary for Bachelors and Doctorate. 4. An Executive and Managerial with Doctorate has sightly high salary than a person with Bachelors. This implies that a person with Doctorate will earn much more as a Professional or Specialist than a person in Sales or Executive and Managerial. 5. Education does not have major effect on Occupation, it though has on Salary. A person with at least a Bachelors degree can earn a higher salary quite similar to a Doctorate in some occupations.

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?

Approach followed	1. Two-way ANOVA helps to understand the interaction between two independent variables in this case Education and Occupation and, also the influence one variable on another. 2. 3 pair of the hypothesis is defined for a two-way ANOVA. 3. One for each of the factors and one for the interaction between two factors in this case Education and Occupation.
Hypothesis	H_0: $\mu_a = \mu_b = \mu_c = ... = \mu_k$ (All population(categories within Education that is Bachelors, Doctorate and High-School Graduate) means are equal) H_a: $\mu_a \neq \mu_b \neq \mu_c \neq ... \neq \mu_k$ (Not all population means are equal, or least one pair is unequal) H_0: $\mu_a = \mu_b = \mu_c = ... = \mu_k$ (All population(categories within occupation that is Administrative and clerical, Executive and managerial, Professional, Sales) means are equal) H_a: $\mu_a \neq \mu_b \neq \mu_c \neq ... \neq \mu_k$ (Not all population means are equal, or least one pair is unequal) H_0: There is no interaction between Education and Occupation. H_a: There is interaction between Education and Occupation.

Output

```
In [13]: formula = 'Salary ~ C(Education) + C(Occupation) + C(Education):C(Occupation)'
         model = ols(formula, DF).fit()
         aov_table = anova_lm(model)
         print(aov_table)

                               df        sum_sq       mean_sq          F  \
C(Education)                  2.0  1.026955e+11  5.134773e+10  72.211958
C(Occupation)                 3.0  5.519946e+09  1.839982e+09   2.587626
C(Education):C(Occupation)    6.0  3.634909e+10  6.058182e+09   8.519815
Residual                     29.0  2.062102e+10  7.110697e+08        NaN

                                   PR(>F)
C(Education)                 5.466264e-12
C(Occupation)                7.211580e-02
C(Education):C(Occupation)   2.232500e-05
Residual                              NaN
```

Figure 1.6.1: Two-way ANOVA table

Inference

As can be seen above, at 95% confidence value of alpha is 0.05

1. P-value for Education is less than 0.05, hence null hypothesis is rejected.
2. P-value for Occupation is more than 0.05, hence null hypothesis is accepted.
3. P-value for interaction between Education and Occupation is less than 0.05, hence null hypothesis is rejected. That to say there is no interaction between factors Education and Occupation.

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

Answer

Based on observations on the dataset and ANOVA test following are business implications:

1. It is important to complete high school graduation to get a job in majority of the occupations.
2. The study reveals that degree of education matters when it comes to earning a higher salary.
3. With the Bachelors degree a person can earn as much as a person with Doctorate degree in some occupations.
4. With Doctorate degree a person can earn much more than a person with Bachelors but only in few specific occupations. This shows that it’s important to understand what degree of Education can put you on higher salary range in a specific occupation as the requirement for all the occupations is not similar.

Problem 2:

Problem Statement:

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.

Information of Education Dataset:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 777 entries, 0 to 776
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)
```

Figure 2.1: Education Dataset Info

Summary of Education Dataset:

	count	mean	std	min	25%	50%	75%	max
Apps	777.0	3001.638353	3870.201484	81.0	776.0	1558.0	3624.0	48094.0
Accept	777.0	2018.804376	2451.113971	72.0	604.0	1110.0	2424.0	26330.0
Enroll	777.0	779.972973	929.176190	35.0	242.0	434.0	902.0	6392.0
Top10perc	777.0	27.558559	17.640364	1.0	15.0	23.0	35.0	96.0
Top25perc	777.0	55.796654	19.804778	9.0	41.0	54.0	69.0	100.0
F.Undergrad	777.0	3699.907336	4850.420531	139.0	992.0	1707.0	4005.0	31643.0
P.Undergrad	777.0	855.298584	1522.431887	1.0	95.0	353.0	967.0	21836.0
Outstate	777.0	10440.669241	4023.016484	2340.0	7320.0	9990.0	12925.0	21700.0
Room.Board	777.0	4357.526384	1096.696416	1780.0	3597.0	4200.0	5050.0	8124.0
Books	777.0	549.380952	165.105360	96.0	470.0	500.0	600.0	2340.0
Personal	777.0	1340.642214	677.071454	250.0	850.0	1200.0	1700.0	6800.0
PhD	777.0	72.660232	16.328155	8.0	62.0	75.0	85.0	103.0
Terminal	777.0	79.702703	14.722359	24.0	71.0	82.0	92.0	100.0
S.F.Ratio	777.0	14.089704	3.958349	2.5	11.5	13.6	16.5	39.8
perc.alumni	777.0	22.743887	12.391801	0.0	13.0	21.0	31.0	64.0
Expend	777.0	9660.171171	5221.768440	3186.0	6751.0	8377.0	10830.0	56233.0
Grad.Rate	777.0	65.463320	17.177710	10.0	53.0	65.0	78.0	118.0

Figure 2.2: Education Dataset Summary

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

<table>
<tr><td>Approach followed</td><td>

1. For univariate analysis histplot and boxplot of each column is generated.
2. For multivariate analysis pairplot and heat map of the entire dataset is generated.
3. Inference is built for each factor from plots for univariate and multivariate.

</td></tr>
<tr><td>Output for Univariate Analysis</td><td>

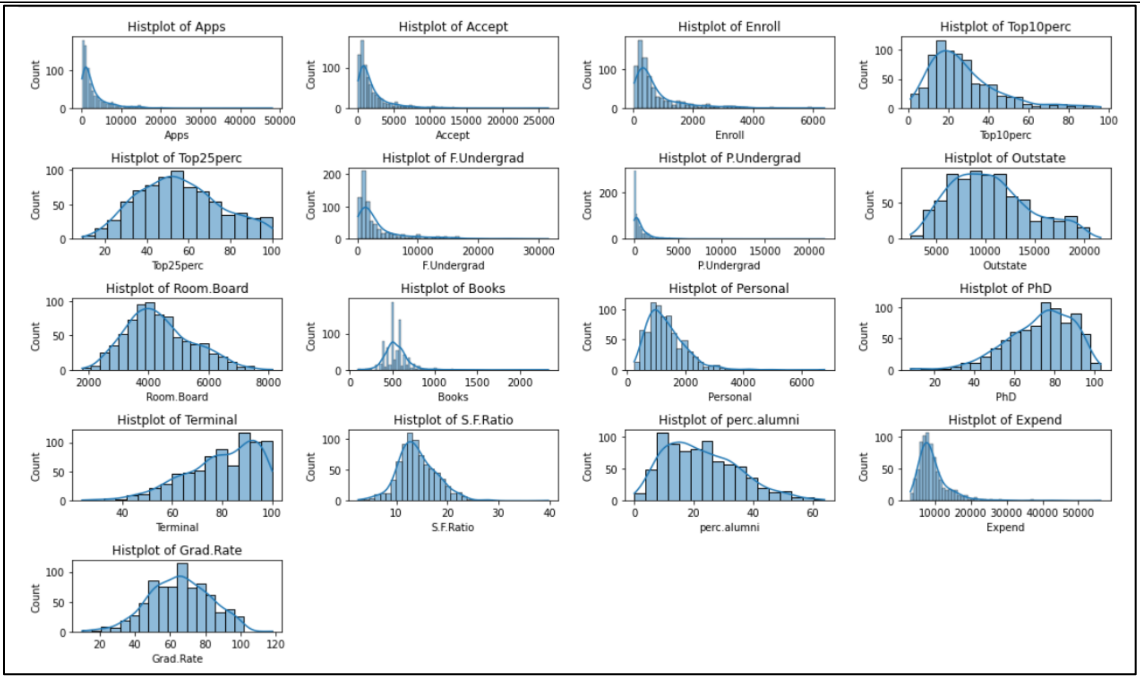


Figure 2.1.1: Histplot of different factors from Education Dataset

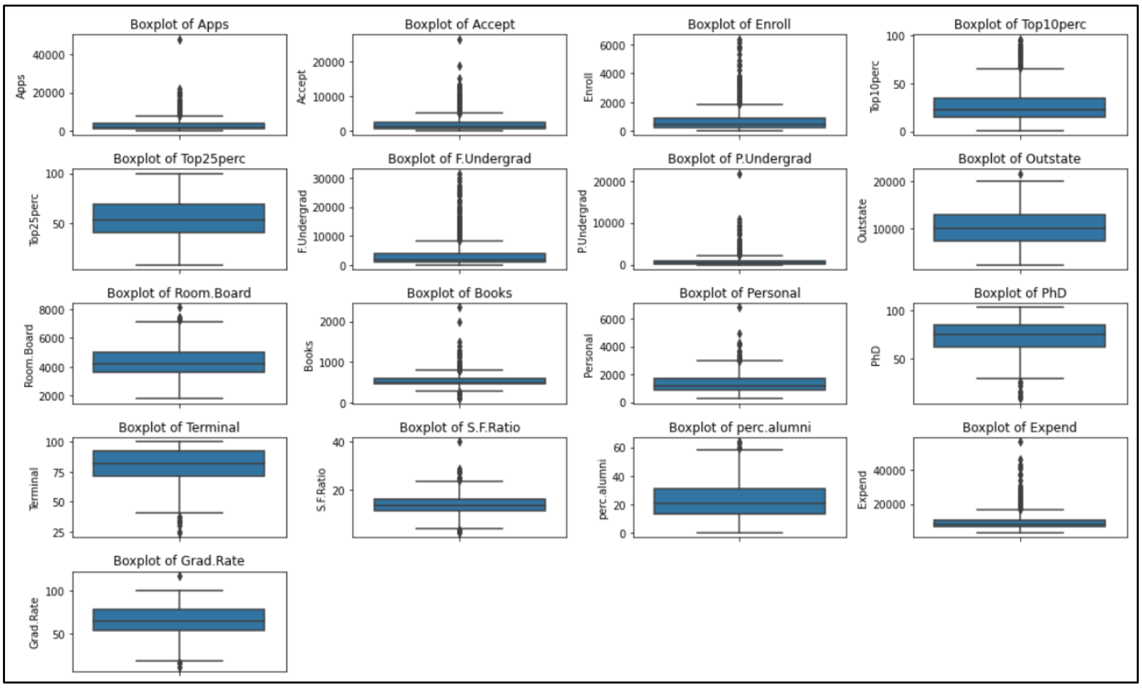


Figure 2.1.2: Boxplot of different factors from Education Dataset

</td></tr>
<tr><td>Inference</td><td>

Following inferences are derived from Univariate Analysis:

1. Majority of the features are right/left distributed. Only few features like top 10 and top 25 percentage, room board, outstate, graduation rate are normally distributed.
2. Only top 25 percentage has no outliers, rest of the features have outliers present.

</td></tr>
</table>

Output for Multivariate Analysis

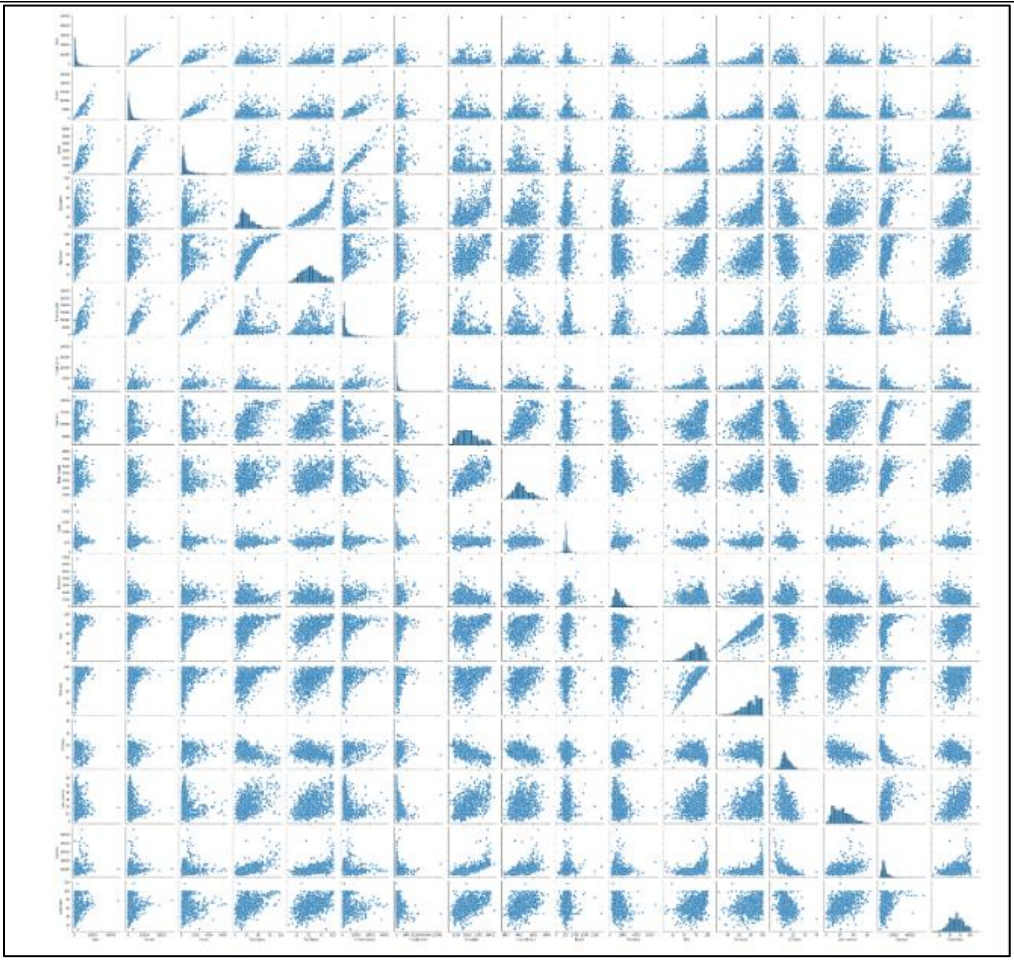


Figure 2.1.3: Pairplot for Mutivariate analysis

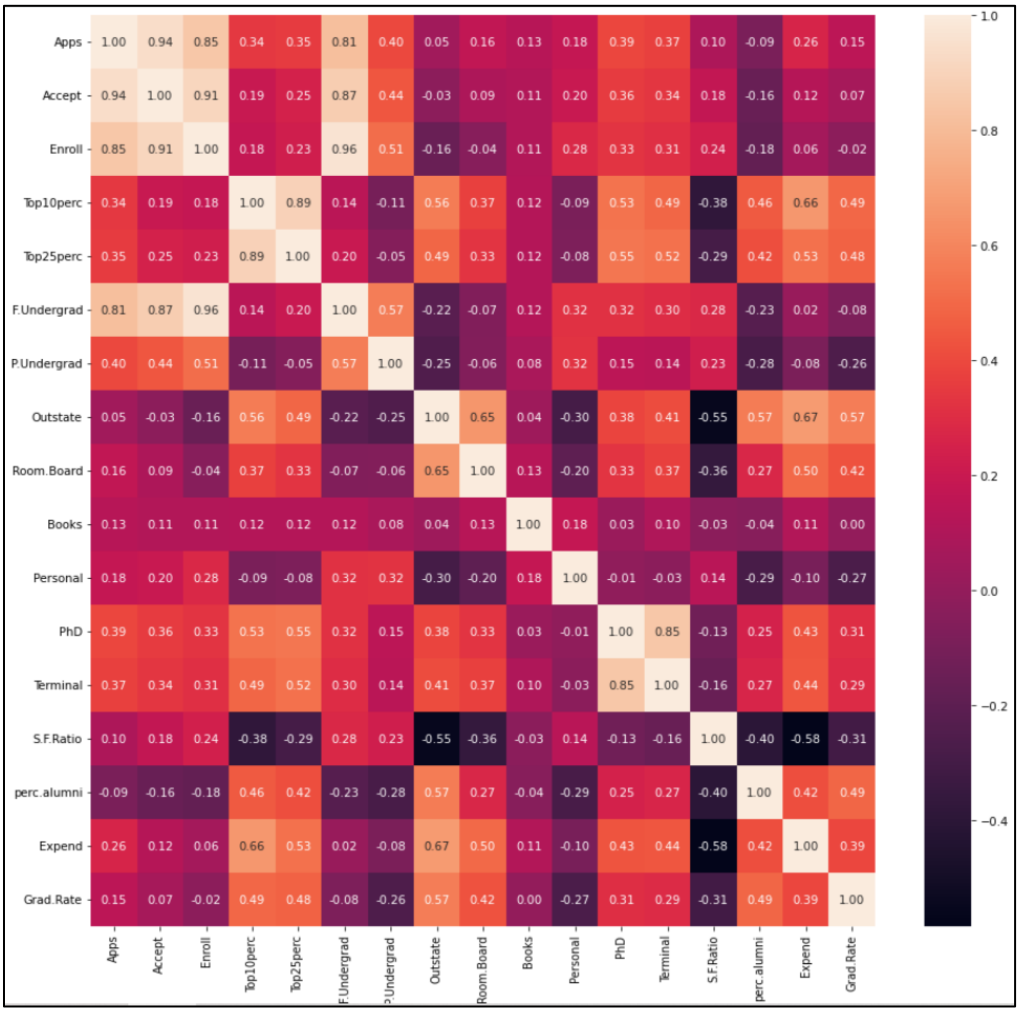


	Figure 2.1.3: Heatmap for multivariate analysis
Inference	Following inferences are derive from Multivariate analysis: 1. Application feature is highly positively correlated with application accepted, students enrolled and full-time graduates. 2. There is negative correlation between application and percentage of alumni. This shows not all students are part of alumni of their college or university. 3. The application with top 10, 25 of higher secondary class, outstate, room board, books, personal, PhD, terminal, S.F ratio, expenditure and Graduation ratio are positively correlated.

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

Answer

1. Scaling (also called data normalization) is a step of data pre-processing which helps to scale down the data for better understanding of data. Since the range of values of data may vary scaling becomes necessary step to standardize the data.
2. For Education dataset, 'Names' column is dropped before applying scaling as it's a categorical field.
3. Z-score scaling is performed on dataset which tells us how many standard deviation point is away from mean and also the direction.

Output after scaling

In [35]:
```
from scipy.stats import zscore
df_pca_scaled = DF_EDU.apply(zscore)
df_pca_scaled.head().T
```

Out[35]:

	0	1	2	3	4
Apps	-0.346882	-0.210884	-0.406866	-0.668261	-0.726176
Accept	-0.321205	-0.038703	-0.376318	-0.681682	-0.764555
Enroll	-0.063509	-0.288584	-0.478121	-0.692427	-0.780735
Top10perc	-0.258583	-0.655656	-0.315307	1.840231	-0.655656
Top25perc	-0.191827	-1.353911	-0.292878	1.677612	-0.596031
F.Undergrad	-0.168116	-0.209788	-0.549565	-0.658079	-0.711924
P.Undergrad	-0.209207	0.244307	-0.497090	-0.520752	0.009005
Outstate	-0.746356	0.457496	0.201305	0.626633	-0.716508
Room.Board	-0.964905	1.909208	-0.554317	0.996791	-0.216723
Books	-0.602312	1.215880	-0.905344	-0.602312	1.518912
Personal	1.270045	0.235515	-0.259582	-0.688173	0.235515
PhD	-0.163028	-2.675646	-1.204845	1.185206	0.204672
Terminal	-0.115729	-3.378176	-0.931341	1.175657	-0.523535
S.F.Ratio	1.013776	-0.477704	-0.300749	-1.615274	-0.553542
perc.alumni	-0.867574	-0.544572	0.585935	1.151188	-1.675079
Expend	-0.501910	0.166110	-0.177290	1.792851	0.241803
Grad.Rate	-0.318252	-0.551262	-0.667767	-0.376504	-2.939613

Figure 2.2.1: Scaled data

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

Answer

Co-variance and correlation helps to understand the relation between dimensions and the dependency within them. Positive and negative values of co-variance & correlation indicate whether they are directly/inversely proportional.

Output of Covariance

```
Covariance Matrix
%s [[ 1.00128866  0.94466636  0.84791332  0.33927032  0.35209304  0.81554018
   0.3987775   0.05022367  0.16515151  0.13272942  0.17896117  0.39120081
   0.36996762  0.09575627 -0.09034216  0.2599265   0.14694372]
 [ 0.94466636  1.00128866  0.91281145  0.19269493  0.24779465  0.87534985
   0.44183938 -0.02578774  0.09101577  0.11367165  0.20124767  0.35621633
   0.3380184   0.17645611 -0.16019604  0.12487773  0.06739929]
 [ 0.84791332  0.91281145  1.00128866  0.18152715  0.2270373   0.96588274
   0.51372977 -0.1556777  -0.04028353  0.11285614  0.28129148  0.33189629
   0.30867133  0.23757707 -0.18102711  0.06425192 -0.02236983]
 [ 0.33927032  0.19269493  0.18152715  1.00128866  0.89314445  0.1414708
  -0.10549205  0.5630552   0.37195909  0.1190116  -0.09343665  0.53251337
   0.49176793 -0.38537048  0.45607223  0.6617651   0.49562711]
 [ 0.35209304  0.24779465  0.2270373   0.89314445  1.00128866  0.19970167
  -0.05364569  0.49002449  0.33191707  0.115676   -0.08091441  0.54656564
   0.52542506 -0.29500852  0.41840277  0.52812713  0.47789622]
 [ 0.81554018  0.87534985  0.96588274  0.1414708   0.19970167  1.00128866
   0.57124738 -0.21602002 -0.06897917  0.11569867  0.31760831  0.3187472
   0.30040557  0.28006379 -0.22975792  0.01867565 -0.07887464]
 [ 0.3987775   0.44183938  0.51372977 -0.10549205 -0.05364569  0.57124738
   1.00128866 -0.25383901 -0.06140453  0.08130416  0.32029384  0.14930637
   0.14208644  0.23283016 -0.28115421 -0.08367612 -0.25733218]
 [ 0.05022367 -0.02578774 -0.1556777   0.5630552   0.49002449 -0.21602002
  -0.25383901  1.00128866  0.65509951  0.03890494 -0.29947232  0.38347594
   0.40850895 -0.55553625  0.56699214  0.6736456   0.57202613]
 [ 0.16515151  0.09101577 -0.04028353  0.37195909  0.33191707 -0.06897917
  -0.06140453  0.65509951  1.00128866  0.12812787 -0.19968518  0.32962651
   0.3750222  -0.36309504  0.27271444  0.50238599  0.42548915]
 [ 0.13272942  0.11367165  0.11285614  0.1190116   0.115676    0.11569867
   0.08130416  0.03890494  0.12812787  1.00128866  0.17952581  0.0269404
   0.10008351 -0.03197042 -0.04025955  0.11255393  0.00106226]
 [ 0.17896117  0.20124767  0.28129148 -0.09343665 -0.08091441  0.31760831
   0.32029384 -0.29947232 -0.19968518  0.17952581  1.00128866 -0.01094989
  -0.03065256  0.13652054 -0.2863366  -0.09801804 -0.26969106]
 [ 0.39120081  0.35621633  0.33189629  0.53251337  0.54656564  0.3187472
   0.14930637  0.38347594  0.32962651  0.0269404  -0.01094989  1.00128866
   0.85068186 -0.13069832  0.24932955  0.43331936  0.30543094]
 [ 0.36996762  0.3380184   0.30867133  0.49176793  0.52542506  0.30040557
   0.14208644  0.40850895  0.3750222   0.10008351 -0.03065256  0.85068186
   1.00128866 -0.16031027  0.26747453  0.43936469  0.28990033]
 [ 0.09575627  0.17645611  0.23757707 -0.38537048 -0.29500852  0.28006379
   0.23283016 -0.55553625 -0.36309504 -0.03197042  0.13652054 -0.13069832
  -0.16031027  1.00128866 -0.4034484  -0.5845844  -0.30710565]
 [-0.09034216 -0.16019604 -0.18102711  0.45607223  0.41840277 -0.22975792
  -0.28115421  0.56699214  0.27271444 -0.04025955 -0.2863366   0.24932955
   0.26747453 -0.4034484   1.00128866  0.41825001  0.49153016]
 [ 0.2599265   0.12487773  0.06425192  0.6617651   0.52812713  0.01867565
  -0.08367612  0.6736456   0.50238599  0.11255393 -0.09801804  0.43331936
   0.43936469 -0.5845844   0.41825001  1.00128866  0.39084571]
 [ 0.14694372  0.06739929 -0.02236983  0.49562711  0.47789622 -0.07887464
  -0.25733218  0.57202613  0.42548915  0.00106226 -0.26969106  0.30543094
   0.28990033 -0.30710565  0.49153016  0.39084571  1.00128866]]
```

Figure 2.3.1: Covariance Matrix

Output of Correlation

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Termin
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	0.3694
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	0.3375
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	0.3082
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	0.4911
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	0.5247
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	0.30001
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	0.1419
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	0.4079
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	0.3745
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	0.0999
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	-0.03061
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	0.8495
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	1.0000
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	-0.1601
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	0.2671
Expend	0.259592	0.124717	0.064169	0.660913	0.527447	0.018652	-0.083568	0.672779	0.501739	0.112409	-0.097892	0.432762	0.4387
Grad.Rate	0.146755	0.067313	-0.022341	0.494989	0.477281	-0.078773	-0.257001	0.571290	0.424942	0.001061	-0.269344	0.305038	0.2895

Figure 2.3.2: Correlation Matrix

Inference

As can be seen from above correlation matrix,

1. Application, accepantance, enrolled, full-time graduates are highly correlated.
2. High correlation visible in Top 10 and top 25 percentage.

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

Answer

The outliers before applying the scaling and after scaling remains the same. This shows scaling does not do any treatment to outliers. It only normalises the data to a standard scale for better understanding.

Outliers before scaling

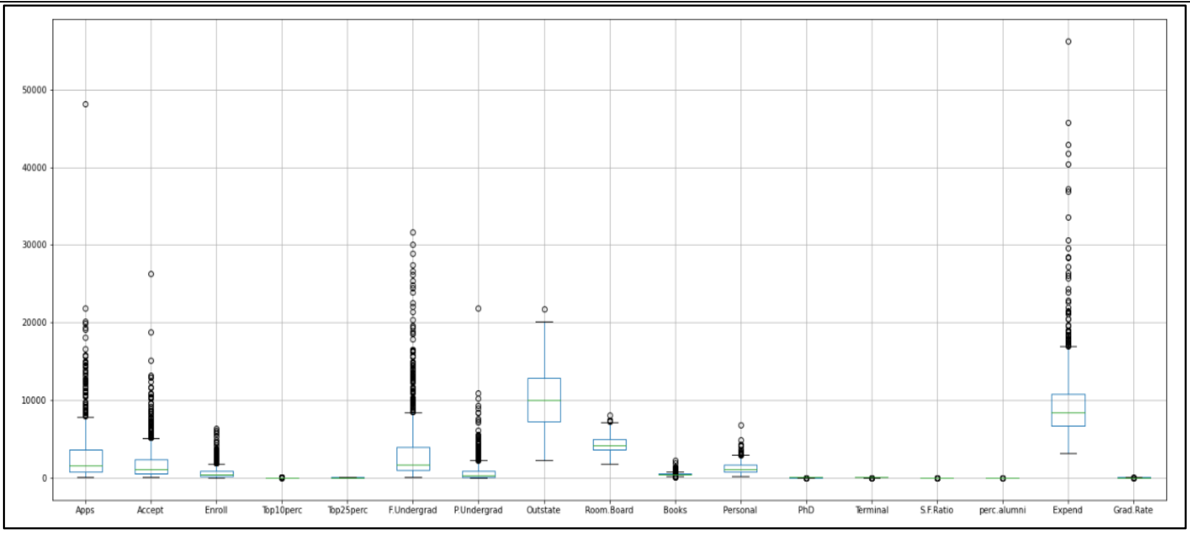


Figure 2.4.1: Outliers before Scaling

Outliers after scaling

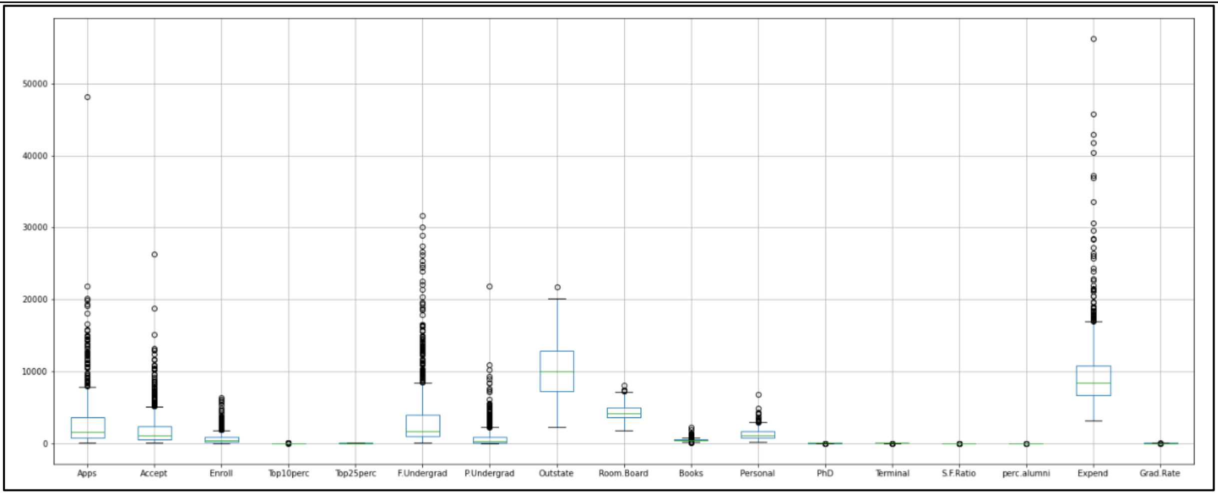


Figure 2.4.2: Outliers after Scaling

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

Answer	Using Sklearn PCA, eigen vales and eigen vectors are generated.

Output of eigenValues and eigenVectors

```
pca = PCA(n_components=17)
X_pca= pca.fit_transform(df_pca_scaled)
```

```
In [26]: pca.components_
Out[26]: array([[ 2.48765602e-01,  2.07601502e-01,  1.76303592e-01,
                  3.54273947e-01,  3.44001279e-01,  1.54640962e-01,
                  2.64425045e-02,  2.94736419e-01,  2.49030449e-01,
                  6.47575181e-02, -4.25285386e-02,  3.18312875e-01,
                  3.17056016e-01, -1.76957895e-01,  2.05082369e-01,
                  3.18908750e-01,  2.52315654e-01],
                [ 3.31598227e-01,  3.72116750e-01,  4.03724252e-01,
                 -8.24118211e-02, -4.47786551e-02,  4.17673774e-01,
                  3.15087830e-01, -2.49643522e-01, -1.37808883e-01,
                  5.63418434e-02,  2.19929218e-01,  5.83113174e-02,
                  4.64294477e-02,  2.46665277e-01, -2.46595274e-01,
                 -1.31689865e-01, -1.69240532e-01],
                [-6.30921033e-02, -1.01249056e-01, -8.29855709e-02,
                  3.50555339e-02, -2.41479376e-02, -6.13929764e-02,
                  1.39681716e-01,  4.65988731e-02,  1.48967389e-01,
                  6.77411649e-01,  4.99721120e-01, -1.27028371e-01,
                 -6.60375454e-02, -2.89848401e-01, -1.46989274e-01,
                  2.26743985e-01, -2.08064649e-01],
                [ 2.81310530e-01,  2.67817346e-01,  1.61826771e-01,
                 -5.15472524e-02, -1.09766541e-01,  1.00412335e-01,
                 -1.58558487e-01,  1.31291364e-01,  1.84995991e-01,
                  8.70892205e-02, -2.30710568e-01, -5.34724832e-01,
                 -5.19443019e-01, -1.61189487e-01,  1.73142230e-02,
                  7.92734946e-02,  2.69129066e-01],
                [ 5.74140964e-03,  5.57860920e-02, -5.56936353e-02,
                 -3.95434345e-01, -4.26533594e-01, -4.34543659e-02,
                  3.02385408e-01,  2.22532003e-01,  5.60919470e-01,
                 -1.27288825e-01, -2.22311021e-01,  1.40166326e-01,
                  2.04719730e-01, -7.93882496e-02, -2.16297411e-01,
                  7.59581203e-02, -1.09267913e-01],
                [-1.62374420e-02,  7.53468452e-03, -4.25579803e-02,
                 -5.26927980e-02,  3.30915896e-02, -4.34542349e-02,
                 -1.91198583e-01, -3.00003910e-02,  1.62755446e-01,
                  6.41054950e-01, -3.31398003e-01,  9.12555212e-02,
                  1.54927646e-01,  4.87045875e-01, -4.73400144e-02,
                 -2.98118619e-01,  2.16163313e-01],
                [-4.24863486e-02, -1.29497196e-02, -2.76928937e-02,
                 -1.61332069e-01, -1.18485556e-01, -2.50763629e-02,
                  6.10423460e-02,  1.08528966e-01,  2.09744235e-01,
                 -1.49692034e-01,  6.33790064e-01, -1.09641298e-03,
                 -2.84770105e-02,  2.19259358e-01,  2.43321156e-01,
                 -2.26584481e-01,  5.59943937e-01],
```

```
 [-1.03090398e-01, -5.62709623e-02,  5.86623552e-02,
  -1.22678028e-01, -1.02491967e-01,  7.88896442e-02,
   5.70783816e-01,  9.84599754e-03, -2.21453442e-01,
   2.13293009e-01, -2.32660840e-01, -7.70400002e-02,
  -1.21613297e-02, -8.36048735e-02,  6.78523654e-01,
  -5.41593771e-02, -5.33553891e-03],
 [-9.02270802e-02, -1.77864814e-01, -1.28560713e-01,
   3.41099863e-01,  4.03711989e-01, -5.94419181e-02,
   5.60672902e-01, -4.57332880e-03,  2.75022548e-01,
  -1.33663353e-01, -9.44688900e-02, -1.85181525e-01,
  -2.54938198e-01,  2.74544380e-01, -2.55334907e-01,
  -4.91388809e-02,  4.19043052e-02],
 [ 5.25098025e-02,  4.11400844e-02,  3.44879147e-02,
   6.40257785e-02,  1.45492289e-02,  2.08471834e-02,
  -2.23105808e-01,  1.86675363e-01,  2.98324237e-01,
  -8.20292186e-02,  1.36027616e-01, -1.23452200e-01,
  -8.85784627e-02,  4.72045249e-01,  4.22999706e-01,
   1.32286331e-01, -5.90271067e-01],
 [ 4.30462074e-02, -5.84055850e-02, -6.93988831e-02,
  -8.10481404e-03, -2.73128469e-01, -8.11578181e-02,
   1.00693324e-01,  1.43220673e-01, -3.59321731e-01,
   3.19400370e-02, -1.85784733e-02,  4.03723253e-02,
  -5.89734026e-02,  4.45000727e-01, -1.30727978e-01,
   6.92088870e-01,  2.19839000e-01],
 [ 2.40709086e-02, -1.45102446e-01,  1.11431545e-02,
   3.85543001e-02, -8.93515563e-02,  5.61767721e-02,
  -6.35360730e-02, -8.23443779e-01,  3.54559731e-01,
  -2.81593679e-02, -3.92640266e-02,  2.32224316e-02,
   1.64850420e-02, -1.10262122e-02,  1.82660654e-01,
   3.25982295e-01,  1.22106697e-01],
 [ 5.95830975e-01,  2.92642398e-01, -4.44638207e-01,
   1.02303616e-03,  2.18838802e-02, -5.23622267e-01,
   1.25997650e-01, -1.41856014e-01, -6.97485854e-02,
   1.14379958e-02,  3.94547417e-02,  1.27696382e-01,
  -5.83134662e-02, -1.77152700e-02,  1.04088088e-01,
  -9.37464497e-02, -6.91969778e-02],
 [ 8.06328039e-02,  3.34674281e-02, -8.56967180e-02,
  -1.07828189e-01,  1.51742110e-01, -5.63728817e-02,
   1.92857500e-02, -3.40115407e-02, -5.84289756e-02,
  -6.68494643e-02,  2.75286207e-02, -6.91126145e-01,
   6.71008607e-01,  4.13740967e-02, -2.71542091e-02,
   7.31225166e-02,  3.64767385e-02],
 [ 1.33405806e-01, -1.45497511e-01,  2.95896092e-02,
   6.97722522e-01, -6.17274818e-01,  9.91640992e-03,
   2.09515982e-02,  3.83544794e-02,  3.40197083e-03,
  -9.43887925e-03, -3.09001353e-03, -1.12055599e-01,
   1.58909651e-01, -2.08991284e-02, -8.41789410e-03,
  -2.27742017e-01, -3.39433604e-03],
 [ 4.59139498e-01, -5.18568789e-01, -4.04318439e-01,
  -1.48738723e-01,  5.18683400e-02,  5.60363054e-01,
  -5.27313042e-02,  1.01594830e-01, -2.59293381e-02,
   2.88282896e-03, -1.28904022e-02,  2.98075465e-02,
  -2.70759809e-02, -2.12476294e-02,  3.33406243e-03,
  -4.38803230e-02, -5.00844705e-03],
 [ 3.58970400e-01, -5.43427250e-01,  6.09651110e-01,
  -1.44986329e-01,  8.03478445e-02, -4.14705279e-01,
   9.01788964e-03,  5.08995918e-02,  1.14639620e-03,
   7.72631963e-04, -1.11433396e-03,  1.38133366e-02,
   6.20932749e-03, -2.22215182e-03, -1.91869743e-02,
  -3.53098218e-02, -1.30710024e-02]])
```

```
In [24]: pca.explained_variance_

Out[24]: array([5.45052162, 4.48360686, 1.17466761, 1.00820573, 0.93423123,
                0.84849117, 0.6057878 , 0.58787222, 0.53061262, 0.4043029 ,
                0.31344588, 0.22061096, 0.16779415, 0.1439785 , 0.08802464,
                0.03672545, 0.02302787])
```

Figure: 2.5.1: Eigen Vectors and Eigen values

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

Answer	1. Calculated the cumulative of all eigen vectors that summed to 100. 2. Checked for variance up to 90%. Rejected the features if the percentage gap is more than 5%. In this case PCA is 5. 3. Exported the data in dataframe as can be seen in below output.

Output

```
In [28]: tot = sum(eig_vals)
         var_exp = [( i /tot ) * 100 for i in sorted(eig_vals, reverse=True)]
         cum_var_exp = np.cumsum(var_exp)
         cum_var_exp

Out[28]: array([ 32.0206282 ,  58.36084263,  65.26175919,  71.18474841,
                 76.67315352,  81.65785448,  85.21672597,  88.67034731,
                 91.78758099,  94.16277251,  96.00419883,  97.30024023,
                 98.28599436,  99.13183669,  99.64896227,  99.86471628,
                100.        ])
```

```
In [36]: pca = PCA(n_components=5)
         X_pca= pca.fit_transform(df_pca_scaled)
```

```
In [30]: pca.components_

Out[30]: array([[ 0.2487656 ,  0.2076015 ,  0.17630359,  0.35427395,  0.34400128,
                  0.15464096,  0.0264425 ,  0.29473642,  0.24903045,  0.06475752,
                 -0.04252854,  0.31831287,  0.31705602, -0.17695789,  0.20508237,
                  0.31890875,  0.25231565],
                [ 0.33159823,  0.37211675,  0.40372425, -0.08241182, -0.04477866,
                  0.41767377,  0.31508783, -0.24964352, -0.13780888,  0.05634184,
                  0.21992922,  0.05831132,  0.04642945,  0.24666528, -0.24659527,
                 -0.13168987, -0.16924053],
                [-0.06309177, -0.10124947, -0.08298583,  0.03505557, -0.02414803,
                 -0.06139256,  0.13968168,  0.04659896,  0.14896738,  0.67741165,
                  0.49972111, -0.12702832, -0.06603759, -0.28984842, -0.14698927,
                  0.22674391, -0.20806465],
                [ 0.28131048,  0.26781741,  0.16182682, -0.05154726, -0.10976653,
                  0.10041227, -0.15855848,  0.13129135,  0.18499599,  0.08708922,
                 -0.23071057, -0.53472484, -0.51944301, -0.16118948,  0.01731422,
                  0.07927351,  0.26912907],
                [ 0.00574198,  0.05578538, -0.05569409, -0.39543428, -0.42653376,
                 -0.04345363,  0.30238534,  0.22253215,  0.56091945, -0.12728881,
                 -0.22231105,  0.14016642,  0.20471965, -0.07938829, -0.21629741,
                  0.07595799, -0.10926792]])
```

Figure: 2.6.1: PCA on dataset

```
In [37]: df_comp = pd.DataFrame(pca.components_,columns=list(df_pca_scaled))
         df_comp.T
```

Out[37]:

	0	1	2	3	4
Apps	0.248766	0.331598	-0.063092	0.281311	0.005741
Accept	0.207602	0.372117	-0.101249	0.267817	0.055786
Enroll	0.176304	0.403724	-0.082986	0.161827	-0.055694
Top10perc	0.354274	-0.082412	0.035056	-0.051547	-0.395434
Top25perc	0.344001	-0.044779	-0.024148	-0.109767	-0.426534
F.Undergrad	0.154641	0.417674	-0.061393	0.100412	-0.043454
P.Undergrad	0.026443	0.315088	0.139682	-0.158558	0.302385
Outstate	0.294736	-0.249644	0.046599	0.131291	0.222532
Room.Board	0.249030	-0.137809	0.148967	0.184996	0.560919
Books	0.064758	0.056342	0.677412	0.087089	-0.127289
Personal	-0.042529	0.219929	0.499721	-0.230711	-0.222311
PhD	0.318313	0.058311	-0.127028	-0.534725	0.140166
Terminal	0.317056	0.046429	-0.066038	-0.519443	0.204720
S.F.Ratio	-0.176958	0.246665	-0.289848	-0.161189	-0.079388
perc.alumni	0.205082	-0.246595	-0.146989	0.017314	-0.216297
Expend	0.318909	-0.131690	0.226744	0.079273	0.075958
Grad.Rate	0.252316	-0.169241	-0.208065	0.269129	-0.109268

Figure: 2.6.2: Data in dataframe after PCA

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]

Answer	With the help of linear equation, following is the explicit form of PCA for 1^{st} component:

	The Linear eq of 1 component: 0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad + 0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21 * perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate Similarly, for the other components the explicit form is generated, refer to output below.
Output of explicit form of PCA	(see code and output below)

```
In [27]: for j in range (1, 18):
             print('\nThe Linear eq of %s component:'%j)
             for i in range(0,17):
                 print('{} * {}'.format(np.round(pca.components_[0][i],2),df_pca_scaled.columns[i]),end=' + ')
             print('\n')
```

```
The Linear eq of 1 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 2 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 3 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 4 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 5 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 6 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 7 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 8 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +
```

```
The Linear eq of 9 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 10 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 11 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 12 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 13 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 14 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 15 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 16 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +


The Linear eq of 17 component:
0.25 * Apps + 0.21 * Accept + 0.18 * Enroll + 0.35 * Top10perc + 0.34 * Top25perc + 0.15 * F.Undergrad + 0.03 * P.Undergrad +
0.29 * Outstate + 0.25 * Room.Board + 0.06 * Books + -0.04 * Personal + 0.32 * PhD + 0.32 * Terminal + -0.18 * S.F.Ratio + 0.21
* perc.alumni + 0.32 * Expend + 0.25 * Grad.Rate +
```

Figure 2.7.1: Explicit form of PCA

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?

Answer	After adding the Eigen Values, we will get sum of 100. To decide the optimum number of principal components: 1. Check for cumulative variance up to 90% 2. The incremental value between the components should not be less than five percent. The Eigen vectors or PC for this case study is five, we can understand how much each variable contributes to the principal components. In other words, we can also say weights attached to each variable. With this Eigen vectors we can understand which variable has more weightage and influences the dataset in the principal components. The PCA reduces the multi collinearity and with this reduced collinearity we can runs models and improved efficiency scores. After PCA multicollinearity is reduced.

Output

```
In [28]: tot = sum(eig_vals)
         var_exp = [( i /tot ) * 100 for i in sorted(eig_vals, reverse=True)]
         cum_var_exp = np.cumsum(var_exp)
         cum_var_exp

Out[28]: array([ 32.0206282 ,  58.36084263,  65.26175919,  71.18474841,
                 76.67315352,  81.65785448,  85.21672597,  88.67034731,
                 91.78758099,  94.16277251,  96.00419883,  97.30024023,
                 98.28599436,  99.13183669,  99.64896227,  99.86471628,
                100.        ])

In [36]: pca = PCA(n_components=5)
         X_pca= pca.fit_transform(df_pca_scaled)

In [30]: pca.components_

Out[30]: array([[ 0.2487656 ,  0.2076015 ,  0.17630359,  0.35427395,  0.34400128,
                  0.15464096,  0.0264425 ,  0.29473642,  0.24903045,  0.06475752,
                 -0.04252854,  0.31831287,  0.31705602, -0.17695789,  0.20508237,
                  0.31890875,  0.25231565],
                [ 0.33159823,  0.37211675,  0.40372425, -0.08241182, -0.04477866,
                  0.41767377,  0.31508783, -0.24964352, -0.13780888,  0.05634184,
                  0.21992922,  0.05831132,  0.04642945,  0.24666528, -0.24659527,
                 -0.13168987, -0.16924053],
                [-0.06309177, -0.10124947, -0.08298583,  0.03505557, -0.02414803,
                 -0.06139256,  0.13968168,  0.04659896,  0.14896738,  0.67741165,
                  0.49972111, -0.12702832, -0.06603759, -0.28984842, -0.14698927,
                  0.22674391, -0.20806465],
                [ 0.28131048,  0.26781741,  0.16182682, -0.05154726, -0.10976653,
                  0.10041227, -0.15855848,  0.13129135,  0.18499599,  0.08708922,
                 -0.23071057, -0.53472484, -0.51944301, -0.16118948,  0.01731422,
                  0.07927351,  0.26912907],
                [ 0.00574198,  0.05578538, -0.05569409, -0.39543428, -0.42653376,
                 -0.04345363,  0.30238534,  0.22253215,  0.56091945, -0.12728881,
                 -0.22231105,  0.14016642,  0.20471965, -0.07938829, -0.21629741,
                  0.07595799, -0.10926792]])
```

Figure 2.8.1: Output of PCA

```
In [33]: df_comp = pd.DataFrame(pca.components_,columns=list(df_pca_scaled))
         df_comp
```

Out[33]:

	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.F.Rati
0	0.248766	0.207602	0.176304	0.354274	0.344001	0.154641	0.026443	0.294736	0.249030	0.064758	-0.042529	0.318313	0.317056	-0.17695
1	0.331598	0.372117	0.403724	-0.082412	-0.044779	0.417674	0.315088	-0.249644	-0.137809	0.056342	0.219929	0.058311	0.046429	0.24666
2	-0.063092	-0.101249	-0.082986	0.035056	-0.024148	-0.061393	0.139682	0.046599	0.148967	0.677412	0.499721	-0.127028	-0.066038	-0.28984
3	0.281310	0.267817	0.161827	-0.051547	-0.109767	0.100412	-0.158558	0.131291	0.184996	0.087089	-0.230711	-0.534725	-0.519443	-0.16118
4	0.005742	0.055785	-0.055694	-0.395434	-0.426534	-0.043454	0.302385	0.222532	0.560919	-0.127289	-0.222311	0.140166	0.204720	-0.07938

```
In [34]: plt.figure(figsize= (15,7))
         sns.heatmap(df_comp, cmap= 'rainbow', annot = True, fmt = '.2f', yticklabels= ['PC0', 'PC1', 'PC2', 'PC3','PC4'])
         plt.show()
```

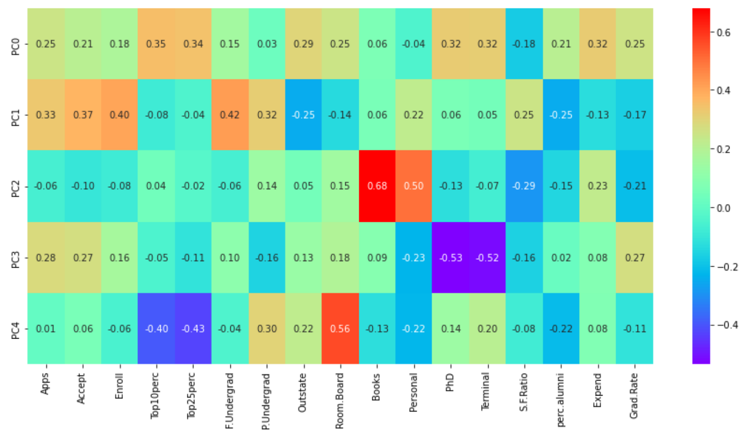


Figure 2.8.2: PCA Data in Data frame

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]

Answer	Based on observation on dataset and PCA test, following are business implications:

	1. The Education dataset which has details about the multiple colleges and universities helps to study the various aspects of colleges, applications, fees, type of enrolments, etc. 2. The univariate and multivariate analysis helps to understand the data pattern for each variable and the correlation between variables. 3. From multivariate analysis, we saw that many variables are highly correlated. 4. With the help of scaling, data is brought down to one standard scale which gives better understanding of the dataset. 5. The Principal Component Analysis helps to reduce multicollinearity in dataset, which further helps to perform more analysis and derive results. 6. The PCA for this case study is 5 which shows the maximum variance of dataset.