College Data Mining

# Python Packages Used in this exercise

Below are the python packages I have used:

1. NumPy
2. Pandas
3. Matplotlib
4. Seaborn
5. Sklearn

# Data Load and Cleansing

First, the csv file was loaded into a dataframe. Then I used describe() method to get the five number summary for all numeric variables. There I found one record had graduation rate more than 100%. So, I dropped that record from the dataframe. Then I checked for null values. The dataframe did not have any null values. Then I converted the “Private” column to category data type.

Below is the data structure after cleansing:

Data columns (total 19 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 college\_name 776 non-null object

1 Private 776 non-null category

2 Apps 776 non-null int64

3 Accept 776 non-null int64

4 Enroll 776 non-null int64

5 Top10perc 776 non-null int64

6 Top25perc 776 non-null int64

7 F.Undergrad 776 non-null int64

8 P.Undergrad 776 non-null int64

9 Outstate 776 non-null int64

10 Room.Board 776 non-null int64

11 Books 776 non-null int64

12 Personal 776 non-null int64

13 PhD 776 non-null int64

14 Terminal 776 non-null int64

15 S.F.Ratio 776 non-null float64

16 perc.alumni 776 non-null int64

17 Expend 776 non-null int64

18 Grad.Rate 776 non-null int64

dtypes: float64(1), int64(16), object(2)

memory usage: 121.2+ KB

# Exploratory Data Analysis

First, I tried to look at the distribution of values for all numerical variable using histogram. I saw some form of skewness in all the numerical variables.

A picture containing window, building

Description automatically generated

Then I tried to find outliers by using boxplots on all the numerical variables.

Chart

Description automatically generated with low confidence

Almost all the numerical variables have outliers. The outliers were not dropped for our modelling purposes.

Below is the record count distribution for the categorical variable

dataset["Private"].value\_counts()

Yes 564  
No 212  
Name: Private, dtype: int64

Another numerical categorical column called “Cluster” was created to represent the string categorical values in the “Private” column (1 for “Yes” and 0 for “No”). That would later be used for accuracy of our clustering.

Then I looked at the correlation matrix for the dataset to identify important features.

Chart, treemap chart

Description automatically generated

I used this matrix to identify the features those are strongly correlated with “Cluster” (and as a result also with “Private”). I noticed that “Enroll”, “F.Undergrad”, “Outstate” have strong correlation with “Cluster”. From these 3, I selected “Enroll” as my first variable.

Now for proper visualization of k means clustering, we need scatterplot with data as much scattered as possible. Which means the variables used in the scatterplot should not have strong correlation and that would place the data points far from a concentrated regression line and help us identify clusters.

So, I used pair plot to find the interactions between all these variables to select the second variable for our plots.

A picture containing indoor, tiled

Description automatically generated

Now looking at the pair plots and correlation matrix above and keeping in mind that “Enroll” was selected as one of our variables. The graph for “Enroll vs. Grad.Rate” was picked for visualizing our clusters for this exercise.

# K means Cluster modelling

Now I have created four k means clustering model with the same data for comparison and better understanding:

1. Model using all the numerical variables and elbow method
2. Model using all the numerical variables and 2 clusters
3. Model using 2 numerical variables and elbow method
4. Model using 2 numerical variables and 2 clusters

All the models were created via unsupervised learning. In each case, I created an array from the features of the dataframe and scaled them using standard scaler.

## Model using all the numerical variables and elbow method

The elbow method produced the following graph:

Chart, line chart

Description automatically generated

Looking at the lowest elbow, number of clusters were determined to be 5.

Below is the final output of the clustering:

Chart, scatter chart

Description automatically generated

It is hard to segregate the clusters as there are so many overlaps. Also, it is hard to identify the centroids for each cluster.

## Model using all the numerical variables and 2 clusters

Here I have used 2 clusters for modelling. Below is the final output:

## Chart, scatter chart Description automatically generated

It is again hard to segregate the clusters as there are so many overlaps. But there seems to be pattern here as the top right data points seems to be mostly red and the bottom left data points seems to be mostly blue in the scatterplot.

Now we can measure the accuracy of our clustering against our binary coded categorical column.

|  |  |  |  |
| --- | --- | --- | --- |
| Private | Cluster | Kmean\_cluster (model output) | count |
| No | 0 | 0 | 66 |
| No | 0 | 1 | 146 |
| Yes | 1 | 0 | 225 |
| Yes | 1 | 1 | 339 |

Below is our classification report:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1-score | support |
| 0 | 0.23 | 0.31 | 0.26 | 212 |
| 1 | 0.7 | 0.6 | 0.65 | 564 |
| accuracy |  |  | 0.52 | 776 |
| macro avg | 0.46 | 0.46 | 0.45 | 776 |
| weighted avg | 0.57 | 0.52 | 0.54 | 776 |

From the report and the counts, we can say that our model has 52% accuracy, and it predicts a good number of both false positives and false negatives which is not good enough.

## Model using 2 numerical variables and elbow method

For this model, I used only “Enroll” and “Grad.Rate” for features.

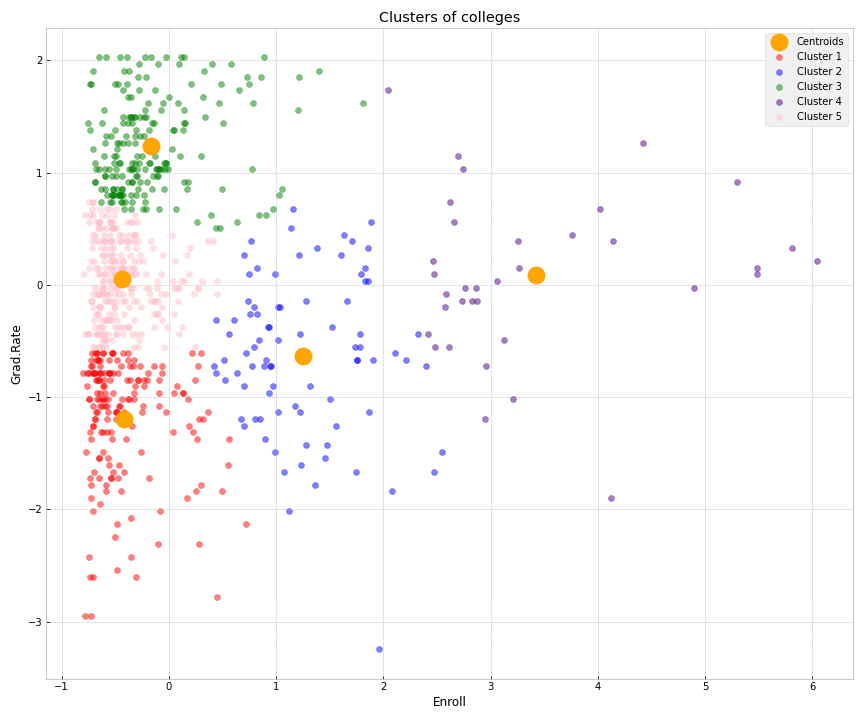
The elbow method produced the following graph:

Chart, line chart

Description automatically generated

Looking at the lowest elbow, number of clusters were determined to be 5.

Below is the final output of the clustering:



This time it is easy to segregate the clusters. All the data points are clustered around their respective centroids. Overall, this model is good.

## Model using 2 numerical variables and 2 clusters

For this model, I used only “Enroll” and “Grad.Rate” for features and I have used 2 clusters for modelling. Below is the final output:

Chart, scatter chart

Description automatically generated

Again, this time it is easy to segregate the clusters. All the data points are clustered around their respective centroids.

Now we can measure the accuracy of our clustering against our binary coded categorical column.

|  |  |  |  |
| --- | --- | --- | --- |
| Private | Cluster | Kmean\_cluster (model\_output) | count |
| No | 0 | 0 | 98 |
| No | 0 | 1 | 114 |
| Yes | 1 | 0 | 12 |
| Yes | 1 | 1 | 552 |

Below is our classification report:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1-score | support |
| 0 | 0.89 | 0.46 | 0.61 | 212 |
| 1 | 0.83 | 0.98 | 0.9 | 564 |
| accuracy |  |  | 0.84 | 776 |
| macro avg | 0.86 | 0.72 | 0.75 | 776 |
| weighted avg | 0.85 | 0.84 | 0.82 | 776 |

From the report and the counts, we can say that our model has 84% accuracy, but still, it predicts a good number of false positives. Overall, this model is really good.