CLUSTERING OF COLLEGE DATA.

A PROJECT REPORT

Submitted by

RAHUL.P	**113216106102**
NAVEEN BALAJI.T.J	**113216106080**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATIONS ENGINEERING

VELAMMAL ENGINEERING COLLEGE

OCTOBER 2018

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**CLUSTERING OF COLLEGE DATA**" is the bonafide work of "**RAHUL. P"**(113216106102) and "**NAVEEN BALAJI.T.J**"(113216106080) who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.S.MARYJOANS	**MAGESH.V**
Head of the Department	Value added course incharge
Department of Electronics and Communication Engineering.	Department of Electronics and Communication Engineering.
Velammal Engineering College, Chennai - 66.	Velammal Engineering College, Chennai - 66.

External Supervisor

CERTIFICATE OF EVALUATION

College Name : VELAMMAL ENGINEERING COLLEGE

Department : ELECTRONICS & COMMUNICATION ENGG.

Semester : 5th

Title of the project	Name of the Student
CLUSTERING OF COLLEGE DATA	**RAHUL .P (113216106102)** **NAVEEN BALAJI .T.J (113216106080)**

The Project report submitted by the above students in partial fulfillment for the award of "Bachelor of Engineering" Degree in "Electronics and Communication Engineering" of "Anna University" and confirmed to be the report done by the above students and then evaluated.

ACKNOWLEDGEMENT

At the outset, we would like to express our gratitude to our beloved and respected **Chairman, Thiru. M. V. Muthuramalingam** for this constant guidance and support.

We would like to express our thanks to our **Chief Executive Officer, Thiru.M.V.M.Velmurugan** for his encouragement and blessing.

We thank our **Principal, Dr.N.Duraipandian** for his support during the course of the project.

We wish to express our sincere thanks and gratitude to **Dr.S.MARY JOANS, Professor & Head, Department of Electronics and Communication Engineering** who has been a guiding force and constant source of inspiration to us.

Our thanks to all faculty and non-teaching staff members of our department for their constant support to complete.

TABLE OF CONTENTS

ABSTRACT

This project is to analyse the given college data-set. It contains a large number of numerical variables. The given dataset can perform analysis using **numpy, pandas, sklearn** modules etc.. And it can be visualized using **numpy, matplotlib , seaborn , SciPy** libraries. The data is gathered, detected and analysed together using python language and also represented graphically. The goal is to cluster the given 777 colleges into their meaningful classes, This can be done using the learning technique in ML called Unsupervised Learning. There is a most popular, and most widely used Unsupervised Learning algorithm called as K-Means Clustering. I am going to take advantage of that learning algorithm.

DATASET REVIEW

The Data-set given is called College.csv contains statistical data of 777 colleges in the US. It had been extracted by US News and World Report in the year 1995. The format of the data in the data-set is given below.

The data given below represents the features of each and every college.

Format:

A data frame with 777 observations on the following 18 variables :

Private - A factor with levels No and Yes indicating private or public university
Apps - Number of applications received
Accept - Number of applications accepted
Enroll - Number of new students enrolled
Top10perc - Pct. new students from top 10% of H.S. class
Top25perc - Pct. new students from top 25% of H.S. class
F.Undergrad - Number of full-time undergraduates
P.Undergrad - Number of part-time undergraduates
Out-state - Out-of-state tuition
Room.Board - Room and board costs
Books - Estimated book costs
Personal - Estimated personal spending
PhD - Pct. of faculty with Ph.D.'s
Terminal - Pct. of faculty with terminal degree
S.F.Ratio - Student/faculty ratio
perc.alumni - Pct. alumni who donate
Expend - Instructional expenditure per student
Grad.Rate - Graduation rate

From the data-set we come to know that there are only features and no targets, And also the problem description also doesn’t specify a target. So this problem can be considered as an unsupervised learning problem. So We can use the Clustering Algorithm to find the mapping in the data and better understand it. So we can do both visualization and Clustering.

INTRODUCTION

Machine Learning:

Machine learning is an application of artificial **intelligence** (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. **Machine learning** focuses on the development of computer programs that can access data and use it learn for themselves.

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. **The primary aim is to allow computers learn automatically** without human intervention or assistance and adjust actions accordingly.

Types of Machine Learning:

Machine learning algorithms are often categorized as supervised or unsupervised.

- **Supervised machine learning algorithms** can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training data-set, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.
- In contrast, **unsupervised machine learning algorithms** are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from data-sets to describe hidden structures from unlabeled data.
- **Reinforcement machine learning algorithms** is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.

Explosion of Data(Information):

The information **explosion** is the rapid increase in the amount of published information or **data** and the effects of this abundance. As the amount of available **data** grows, the problem of managing the information becomes more difficult, which can lead to information overload

- The world's technological capacity to store information grew from 2.6 (optimally compressed) exabytes in 1986 to 15.8 in 1993, over 54.5 in 2000, and to 295 (optimally compressed) exabytes in 2007. This is equivalent to less than one 730-MB CD-ROM per person in 1986 (539 MB per person), roughly 4 CD-ROM per person of 1993, 12 CD-ROM per person in the year 2000, and almost 61 CD-ROM per person in 2007. Piling up the imagined 404 billion CD-ROM from 2007 would create a stack from the Earth to the Moon and a quarter of this distance beyond (with 1.2 mm thickness per CD).
- The world's technological capacity to receive information through one-way broadcast networks was 432 exabytes of (optimally compressed) information in 1986, 715 (optimally compressed) exabytes in 1993, 1,200 (optimally compressed) exabytes in 2000, and 1,900 in 2007.
- The world's effective capacity to exchange information through two-way telecommunication networks was 0.281 exabytes of (optimally compressed) information in 1986, 0.471 in 1993, 2.2 in 2000, and 65 (optimally compressed) exabytes in 2007.

According to GOOGLE, there are three trends in data gathering today:

Type 1: Expansion of the number of fields being collected, known as the "collect more" trend.

Type 2: Replace an existing aggregate data collection with a person-specific one, known as the "collect specifically" trend.

Type 3: Gather information by starting a new person-specific data collection, known as the "collect it if you can" trend.

Machine Learning in my project:

I Described Earlier that I am using Machine Learning to cluster the colleges into relevant classes. So, I used a ML technique called Unsupervised Learning. In that technique there is an algorithm called as KMeans Clustering, which is the most widely used and most popular Clustering Algorithm. It can be used to cluster the data into different categories.

The number of clusters can be found by a method, called elbow method, which I will describe later in this report.

By using Clustering and suitable visualization techniques we can find the mapping in the data, and can classify the data better.I have a lot of plots to explain later. And, the dataset seems interesting. The Clustering technique is a bunch of math formulas, which are written in python in a library called **SKLearn ,** which we can import and use easily, rather than writing a bunch of math formulas in python.

So, I used **SKLearn, NumPy, Pandas, MatPlotLib, SeaBorn** and **SciPy** for my Project.

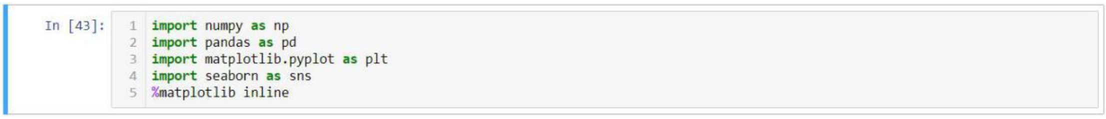
DATA PRE - PROCESSING

Understanding The DATA:

The Data-set given contains 777 observations of colleges each observation has 18 features, they are as follows;

Private - A factor with levels No and Yes indicating private or public university
Apps - Number of applications received
Accept - Number of applications accepted
Enroll - Number of new students enrolled
Top10perc - Pct. new students from top 10% of H.S. class
Top25perc - Pct. new students from top 25% of H.S. class
F.Undergrad - Number of full-time undergraduates
P.Undergrad - Number of part-time undergraduates
Out-state - Out-of-state tuition
Room.Board - Room and board costs
Books - Estimated book costs
Personal - Estimated personal spending
PhD - Pct. of faculty with Ph.D.'s
Terminal - Pct. of faculty with terminal degree
S.F.Ratio - Student/faculty ratio
perc.alumni - Pct. alumni who donate
Expend - Instructional expenditure per student
Grad.Rate - Graduation rate

Step 1 - Importing Libraries:

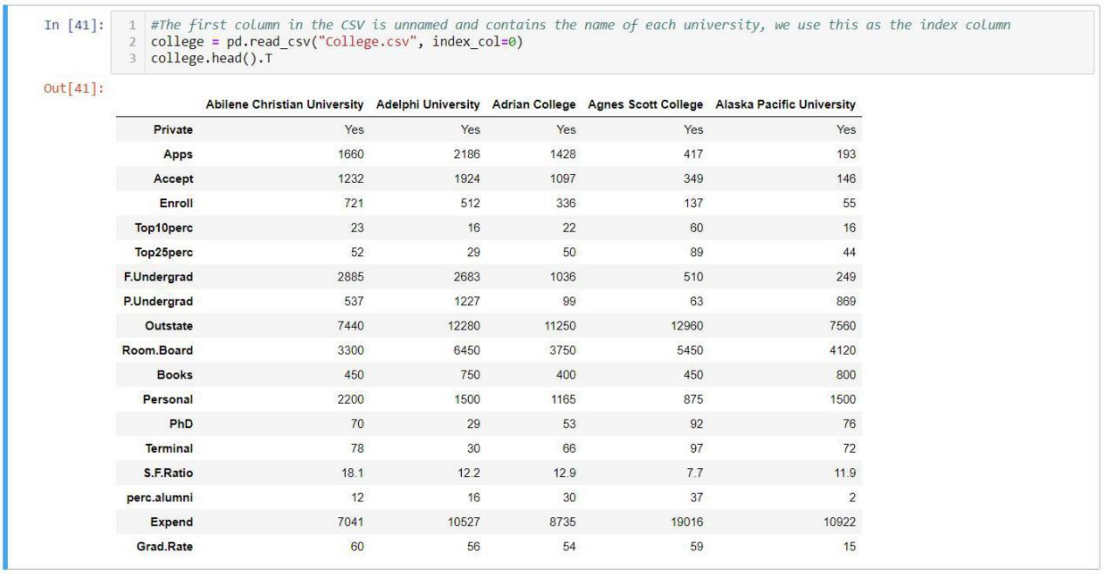


In [43]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

This is how we import libraries in python, Libraries are set of python files, which some other developer has written, that can be used by us for our problems, We can also contribute to the developer community by writing our own libraries

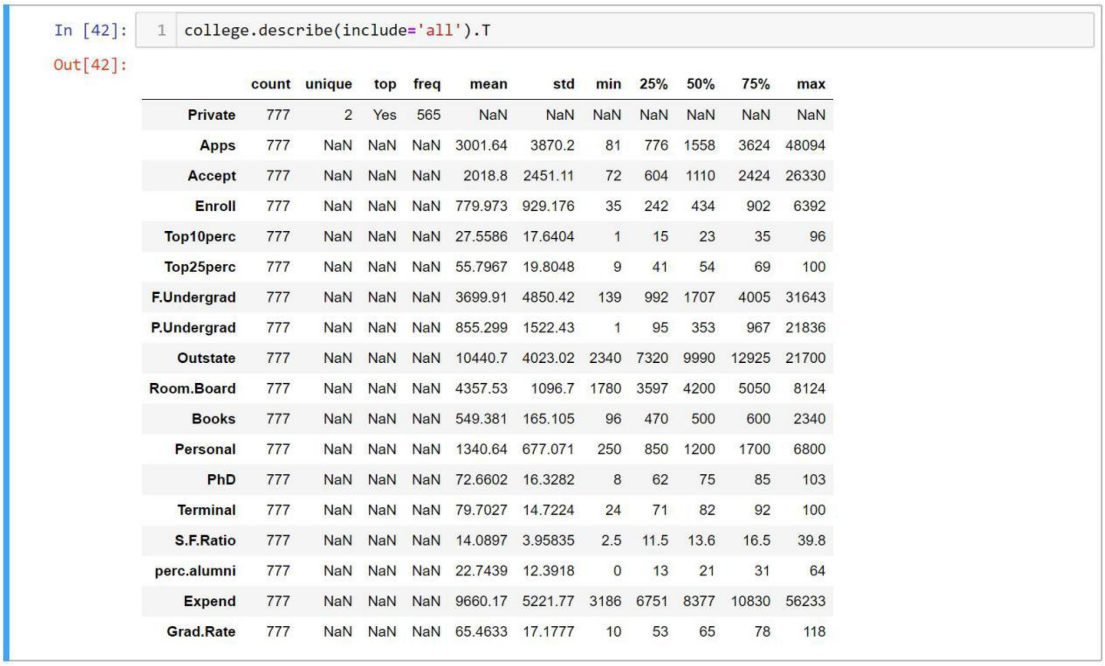
Step 2 - Importing the Dataset:



```
#The first column in the CSV is unnamed and contains the name of each university, we use this as the index column
college = pd.read_csv("College.csv", index_col=0)
college.head().T
```

	Abilene Christian University	Adelphi University	Adrian College	Agnes Scott College	Alaska Pacific University
Private	Yes	Yes	Yes	Yes	Yes
Apps	1660	2186	1428	417	193
Accept	1232	1924	1097	349	146
Enroll	721	512	336	137	55
Top10perc	23	16	22	60	16
Top25perc	52	29	50	89	44
F.Undergrad	2885	2683	1036	510	249
P.Undergrad	537	1227	99	63	869
Outstate	7440	12280	11250	12960	7560
Room.Board	3300	6450	3750	5450	4120
Books	450	750	400	450	800
Personal	2200	1500	1165	875	1500
PhD	70	29	53	92	76
Terminal	78	30	66	97	72
S.F.Ratio	18.1	12.2	12.9	7.7	11.9
perc.alumni	12	16	30	37	2
Expend	7041	10527	8735	19016	10922
Grad.Rate	60	56	54	59	15

The Data-set can be imported by using pandas library as shown in the figure. The Output of the figure shows the features of the first 5 colleges in the Data-set, this is just to see how the data looks like.



```
In [42]:  college.describe(include='all').T
Out[42]:
```

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Private	777	2	Yes	565	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Apps	777	NaN	NaN	NaN	3001.64	3870.2	81	776	1558	3624	48094
Accept	777	NaN	NaN	NaN	2018.8	2451.11	72	604	1110	2424	26330
Enroll	777	NaN	NaN	NaN	779.973	929.176	35	242	434	902	6392
Top10perc	777	NaN	NaN	NaN	27.5586	17.6404	1	15	23	35	96
Top25perc	777	NaN	NaN	NaN	55.7967	19.8048	9	41	54	69	100
F.Undergrad	777	NaN	NaN	NaN	3699.91	4850.42	139	992	1707	4005	31643
P.Undergrad	777	NaN	NaN	NaN	855.299	1522.43	1	95	353	967	21836
Outstate	777	NaN	NaN	NaN	10440.7	4023.02	2340	7320	9990	12925	21700
Room.Board	777	NaN	NaN	NaN	4357.53	1096.7	1780	3597	4200	5050	8124
Books	777	NaN	NaN	NaN	549.381	165.105	96	470	500	600	2340
Personal	777	NaN	NaN	NaN	1340.64	677.071	250	850	1200	1700	6800
PhD	777	NaN	NaN	NaN	72.6602	16.3282	8	62	75	85	103
Terminal	777	NaN	NaN	NaN	79.7027	14.7224	24	71	82	92	100
S.F.Ratio	777	NaN	NaN	NaN	14.0897	3.95835	2.5	11.5	13.6	16.5	39.8
perc.alumni	777	NaN	NaN	NaN	22.7439	12.3918	0	13	21	31	64
Expend	777	NaN	NaN	NaN	9660.17	5221.77	3186	6751	8377	10830	56233
Grad.Rate	777	NaN	NaN	NaN	65.4633	17.1777	10	53	65	78	118

This Code above is used to describe the data-set, From this code we can understand the data-set well by recognizing the mean, std, min, max, freq, etc.. for all the 18 features.From the above output, we come to know that the Private feature is a binary variable(Yes / No).

The Data-set be got is already clean and feasible with the algorithm, so the step of **"Data Cleaning"** can be skipped.

But there is an other crucial steps we have to do, that is called **"Feature Extraction"**, which means obtain the features that matter the most, which can highly increase the efficiency of our algorithm.

The optimum features can be found by using, p-value of the features, If the p-value is above 0.5, then that feature can be ignored, This can be done by using stats-models module in python.

DATA VISUALISATION

Data visualization is a general term that describes any effort to help people understand the significance of data by placing it in a visual context. Patterns, trends and correlations that might go undetected in text-based data can be exposed and recognized easier with data visualization software.

Today's data visualization tools go beyond the standard charts and graphs used in Microsoft Excel spreadsheets, displaying data in more sophisticated ways such as **info-graphics dials and gauges, geographic maps, spark lines, heat maps, and detailed bar, pie and fever charts.** The images may include interactive capabilities, enabling users to manipulate them or drill into the data for querying and analysis. Indicators designed to alert users when data has been updated or predefined conditions occur can also be included.

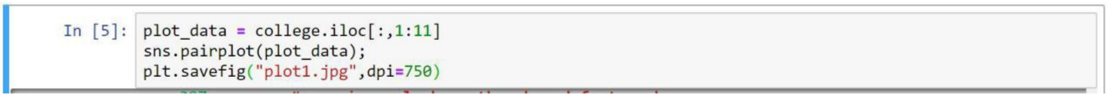
Info-graphics predate writing as a means of disseminating information -- cave drawings are probably the earliest known example. People were also creating and using maps before the advent of written language.

The process of creating info-graphics is sometimes referred to as Data Visualization.

In my project I had used a plotting library known as SeaBorn, which is built on top of Matplotlib which is python’s own Plotting Library. I used Seaborn, because it is easier and more efficient than matplotlib, and it can also group plot automatically.

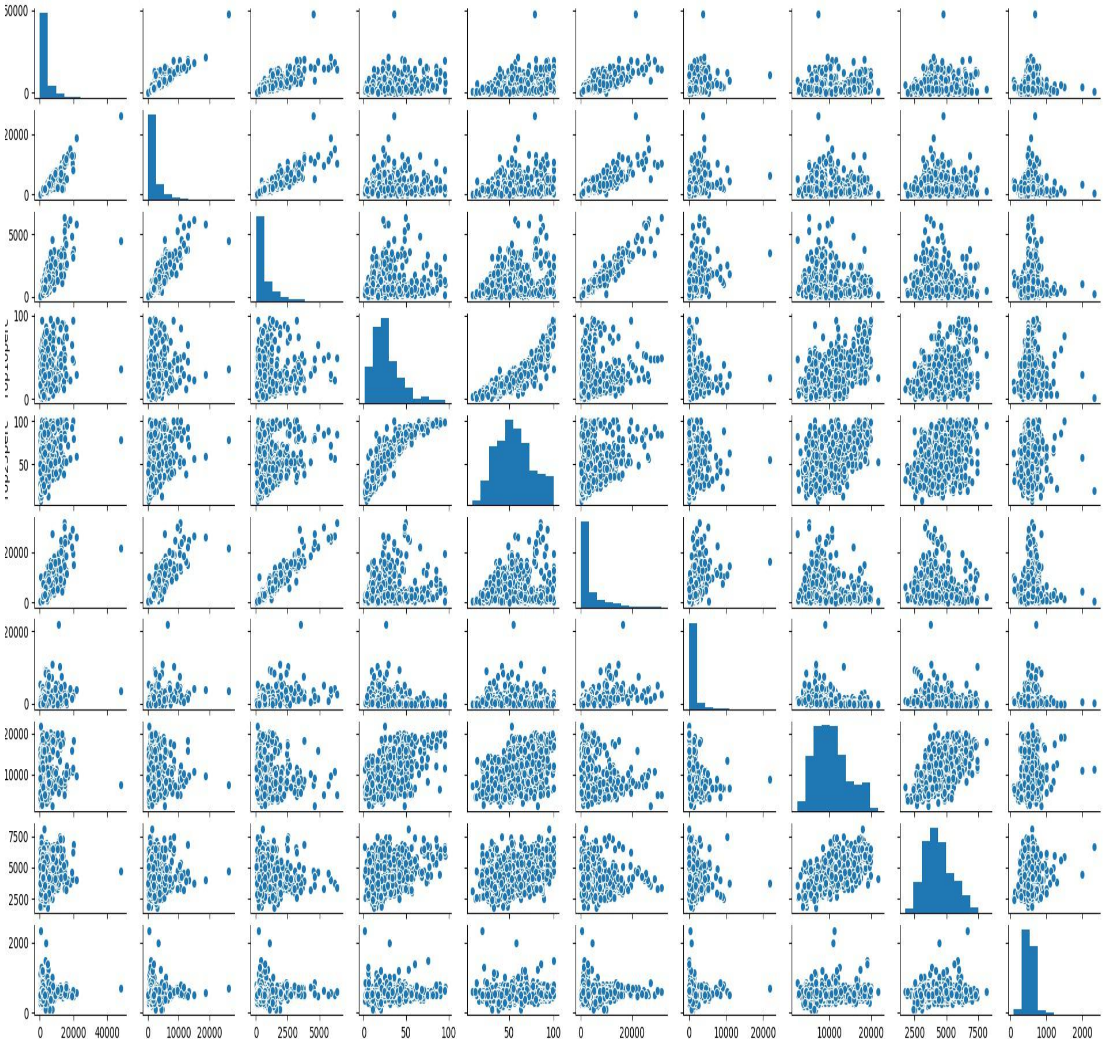
Since, it is a clustering problem, we must recognize the Relationship between each an every feature to every other feature, with seaborn, it Comes so handy, Seaborn’s pairplot function can do it for us. The code to find the plot and relationship between features is shown below.

Step 3 - Visualizing the relationship:



```
In [5]: plot_data = college.iloc[:,1:11]
        sns.pairplot(plot_data);
        plt.savefig("plot1.jpg",dpi=750)
```

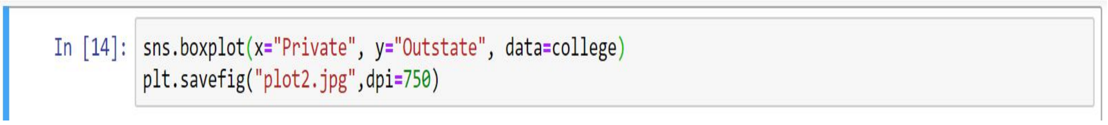
Results:



The following graph shows the relationship between first 10 features and plots all the combinations, in (10 * 10) pairs, by this we can find the relationship between them.

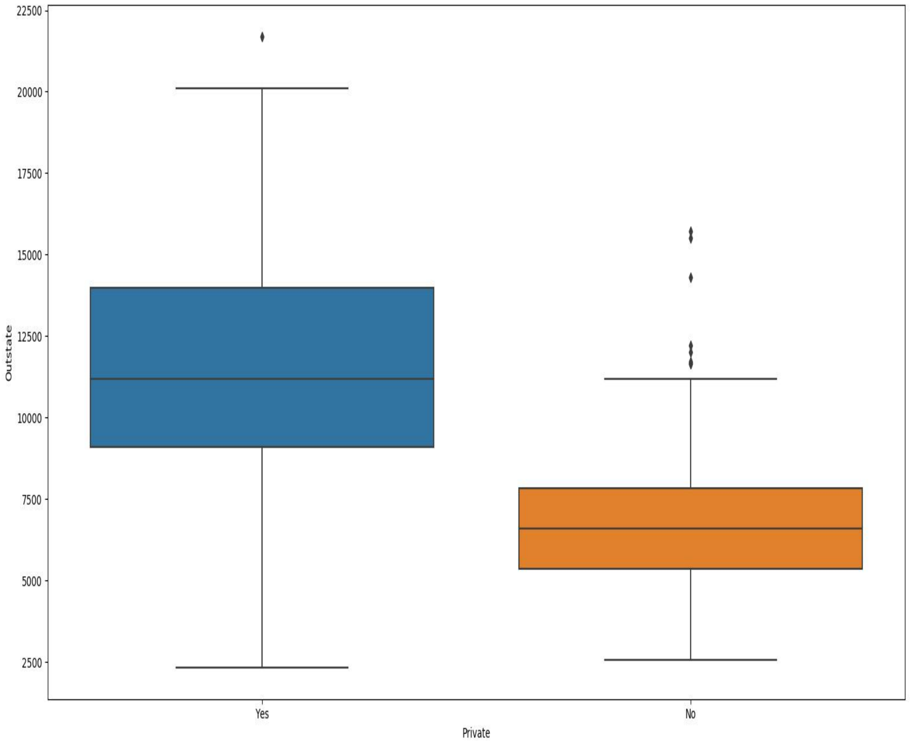
Step 4 - Visualizing the Clusters:

i) Now if we plot the Cluster of Private colleges which are out of state, we can find the relationship.

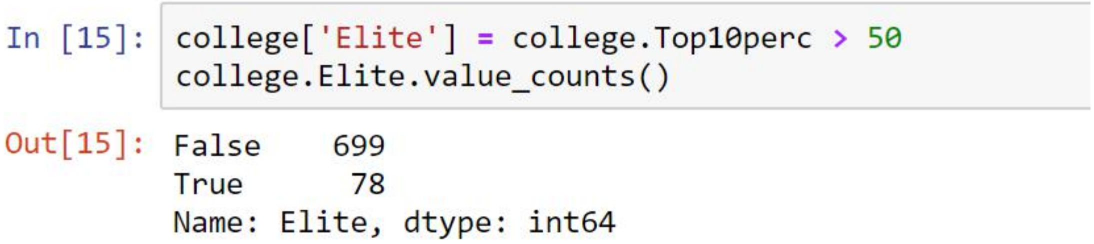


```
In [14]: sns.boxplot(x="Private", y="Outstate", data=college)
         plt.savefig("plot2.jpg",dpi=750)
```

Results:

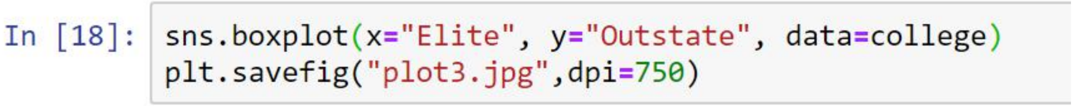


We can find the Top 10 percent colleges which are best in all aspects.They are called “Elite” Colleges. We found that using the code below,



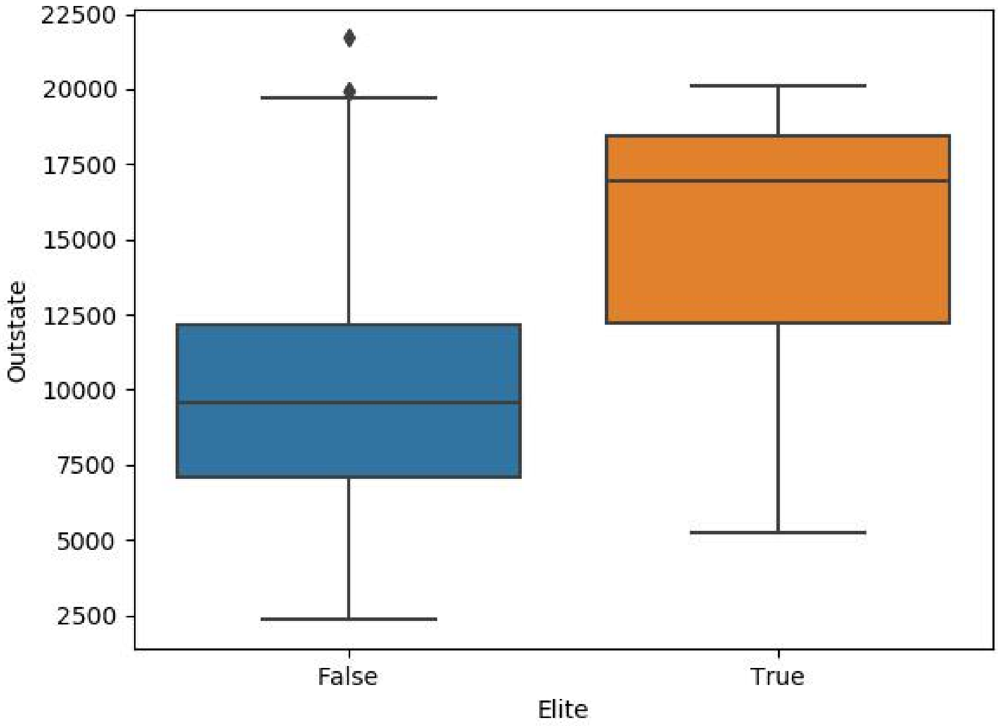
```
In [15]: college['Elite'] = college.Top10perc > 50
         college.Elite.value_counts()

Out[15]: False    699
         True      78
         Name: Elite, dtype: int64
```

We Can Cluster the “**Elite**” Colleges which are Out-Of-State by plotting a box plot as shown below:



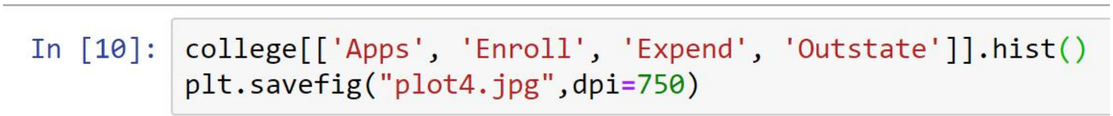
```
In [18]: sns.boxplot(x="Elite", y="Outstate", data=college)
         plt.savefig("plot3.jpg",dpi=750)
```

Results:



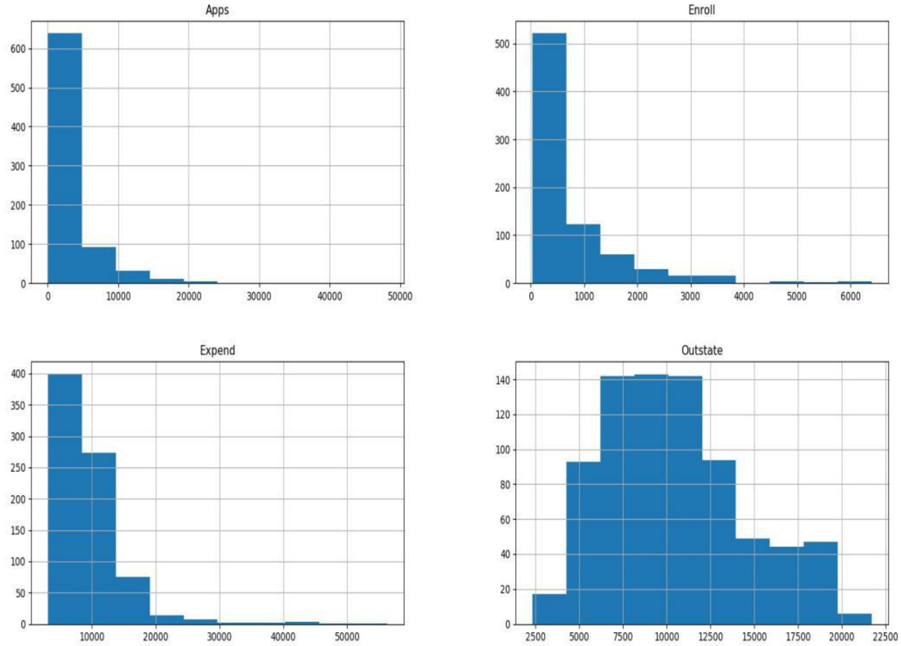
Step 5 - Visualizing with Histogram:

We Can also make use of Histograms to get a better Intuition from the data. The hist() function in seaborn comes handy to use, We can Take advantage of that function.



```
In [10]: college[['Apps', 'Enroll', 'Expend', 'Outstate']].hist()
         plt.savefig("plot4.jpg",dpi=750)
```

Results:



With this we can stop visualizing and start analysing.

DATA ANALYSIS

Data analysis is a **process of inspecting, cleansing, transforming, and modeling data** with the **goal of discovering useful information, informing conclusions, and supporting decision-making**. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, while being used in different business, science, and social science domains.

Data mining is a particular data analysis technique that focuses on modeling and knowledge discovery for predictive rather than purely descriptive purposes, while **business intelligence** covers data analysis that relies heavily on aggregation, focusing mainly on business information.In statistical applications, data analysis can be divided into **descriptive statistics, exploratory data analysis** (EDA), and **confirmatory data analysis** (CDA). EDA focuses on discovering new features in the data while CDA focuses on confirming or falsifying existing **hypotheses. Predictive analytics** focuses on application of statistical models for predictive forecasting or classification, while **text analytics** applies statistical, linguistic, and structural techniques to extract and classify information from textual sources, a species of **unstructured data.** All of the above are varieties of data analysis.

Data integration is a precursor to data analysis, and data analysis is closely linked[how?] to **data visualization** and data dissemination. The term data analysis is sometimes used as a synonym for data modeling.

Analysis refers to breaking a whole into its separate components for individual examination. Data analysis is a process for obtaining raw data and converting it into information useful for decision-making by users. Data is collected and analyzed to answer questions, test hypotheses or disprove theories.

Analysis Using SNS Plots:

Study and analysis of the data is the very first step of any data science work. You need to get the general information about the nature and distribution of the data to plan your work-flow accordingly. This is where visualization comes in as we say “a picture says thousand words”. With informative plots, it is easier to gain insights from the data and also to convey the insights to others.

Analysis:

As we plotted the data on a number of different Seaborn plots. We can infer from those plots the necessary information like Private colleges, Elite Colleges, No. Of students from top 10% of the H.S class, Top 25% from the H. S. class, F.Undergrad, P.Undergrad, Out-State tuition, Room-Board, Cost of the books, Personal, PhD, Faculty with Terminal Degree etc...

From these plots we can better understand the data.

Unsupervised Learning:

Machine learning technique for finding hidden patterns or intrinsic structures in data.

Unsupervised learning is a type of machine learning algorithm used to draw inferences from datas-ets consisting of input data without labeled responses.

The most common unsupervised learning method is **cluster analysis**, which is used for exploratory data analysis to find hidden patterns or grouping in data. The clusters are modeled using a measure of similarity which is defined upon metrics such as Euclidean or probabilistic distance.

Common clustering algorithms include:

- **Hierarchical clustering**: builds a multilevel hierarchy of clusters by creating a cluster tree
- **k-Means clustering**: partitions data into k distinct clusters based on distance to the centroid of a cluster
- **Gaussian mixture models**: models clusters as a mixture of multivariate normal density components.

Hierarchical Clustering

Hierarchical clustering groups data over a variety of scales by creating a cluster tree or dendrogram. The tree is not a single set of clusters, but rather a multilevel hierarchy, where clusters at one level are joined as clusters at the next level. This allows you to decide the level or scale of clustering that is most appropriate for your application. The Statistics and Machine Learning Toolbox™ function clusterdata supports agglomerative clustering and performs all of the necessary steps for you. It incorporates the pdist, linkage, and cluster functions, which you can use separately for more detailed analysis. The dendrogramfunction plots the cluster tree.

K - Means Clustering

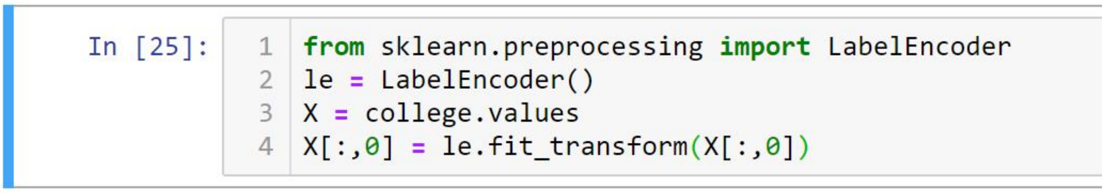
k-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells.

Using Clustering Algorithms on my Data-set:

Using K-Means algorithm:

In order to implement K-Means we must find the number of clusters required. For that we are going to use a powerful statistical method called Elbow method.

But the first column of our data-set must be Encoded using LabelEncoder as shown below,

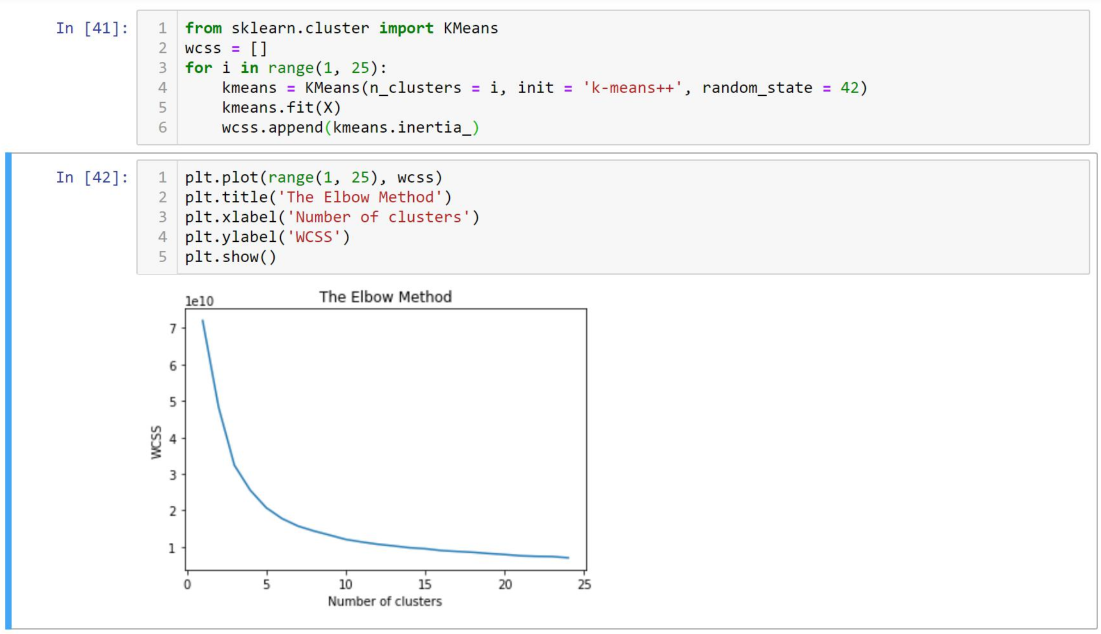


In [25]:

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
X = college.values
X[:,0] = le.fit_transform(X[:,0])
```

This Code Converts the first Column which contains (‘Yes’ / ‘No’) to computer understandable binary values(1 / 0). Now, we can implement our Elbow method to find the number of clusters required.

The Elbow(L) method:



In [41]:

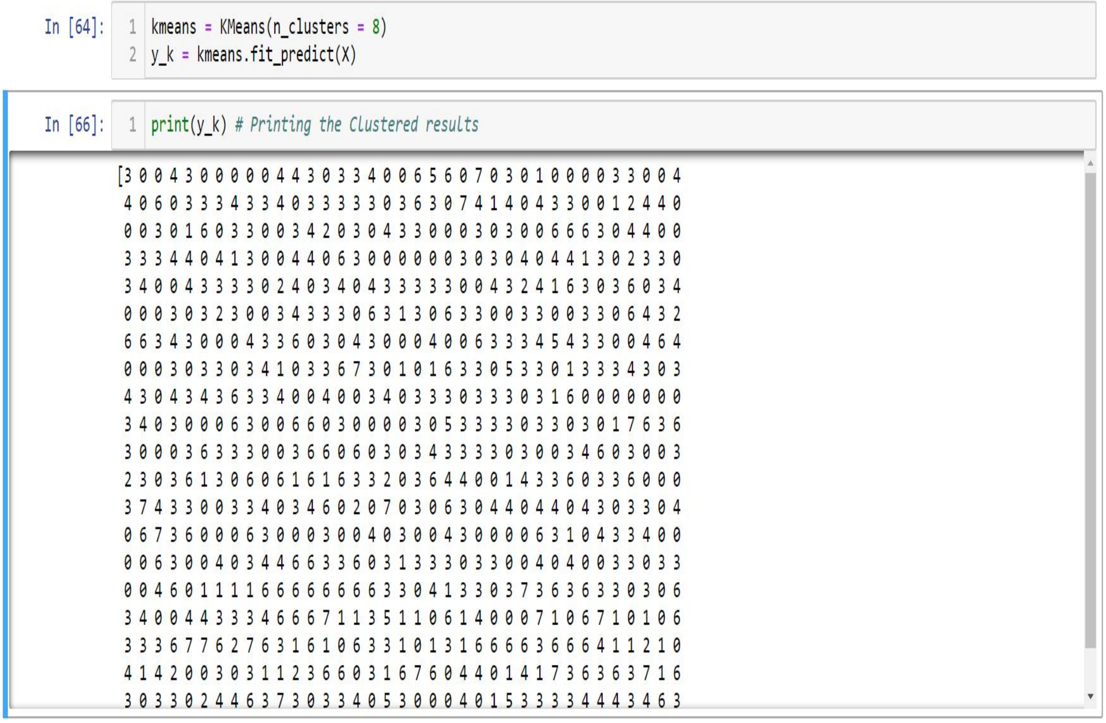
```
from sklearn.cluster import KMeans
wcss = []
for i in range(1, 25):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
```

In [42]:

```
plt.plot(range(1, 25), wcss)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```

From the Elbow method by looking at the graph, we can find that, the slope of the L - curve, get less and less when the number of Clusters increase.But the slope gets more lesser when the number of Clusters is equal to 8. So, we will get better results when number of clusters is equal to 8.

The K-Means Algorithm:



In [64]:
```
kmeans = KMeans(n_clusters = 8)
y_k = kmeans.fit_predict(X)
```

In [66]:
```
print(y_k) # Printing the Clustered results
```

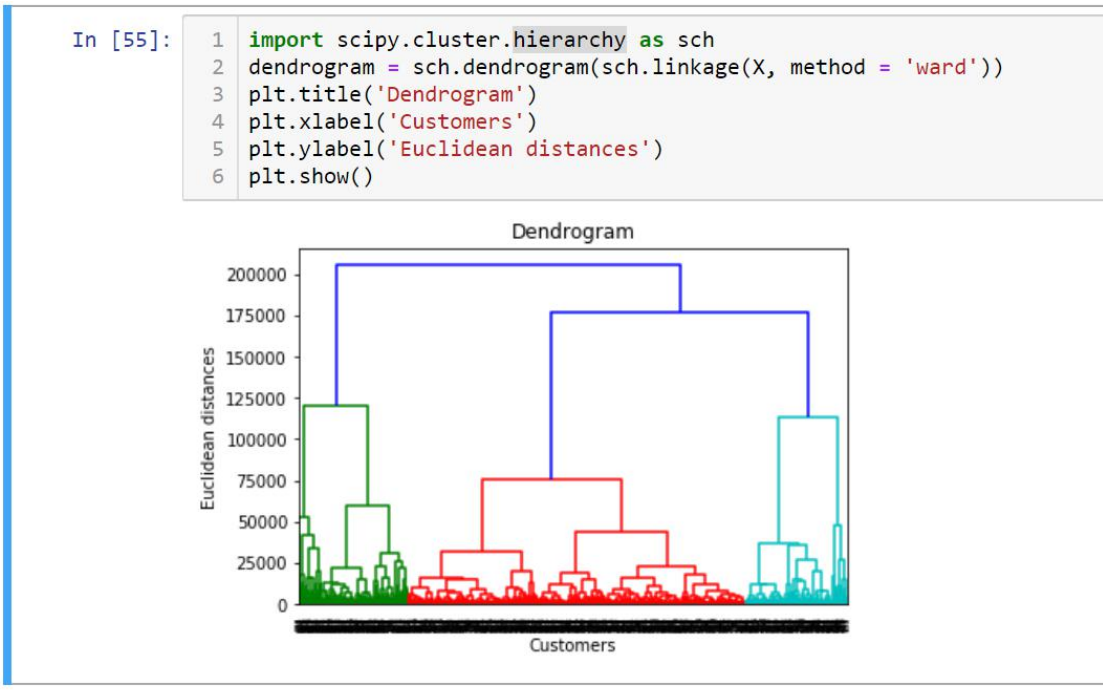
```
[3 0 0 4 3 0 0 0 0 0 4 4 3 0 3 3 4 0 0 6 5 6 0 7 0 3 0 1 0 0 0 0 3 3 0 0 4
 4 0 6 0 3 3 3 4 3 3 4 0 3 3 3 3 3 0 3 6 3 0 7 4 1 4 0 4 3 3 0 0 1 2 4 4 0
 0 0 3 0 1 6 0 3 3 0 0 3 4 2 0 3 0 4 3 3 0 0 0 3 0 3 0 0 6 6 6 3 0 4 4 0 0
 3 3 3 4 4 0 4 1 3 0 0 4 4 0 6 3 0 0 0 0 0 0 3 0 3 0 4 0 4 4 1 3 0 2 3 3 0
 3 4 0 0 4 3 3 3 3 0 2 4 0 3 4 0 4 3 3 3 3 3 0 0 4 3 2 4 1 6 3 0 3 6 0 3 4
 0 0 0 3 0 3 2 3 0 0 3 4 3 3 3 0 6 3 1 3 0 6 3 3 0 0 3 3 0 0 3 3 0 6 4 3 2
 6 6 3 4 3 0 0 0 4 3 3 6 0 3 0 4 3 0 0 0 4 0 0 6 3 3 3 4 5 4 3 3 0 0 4 6 4
 0 0 0 3 0 3 3 0 3 4 1 0 3 3 6 7 3 0 1 0 1 6 3 3 0 5 3 3 0 1 3 3 3 4 3 0 3
 4 3 0 4 3 4 3 6 3 3 4 0 0 4 0 0 3 4 0 3 3 3 0 3 3 3 0 3 1 6 0 0 0 0 0 0 0
 3 4 0 3 0 0 0 6 3 0 0 6 6 0 3 0 0 0 0 3 0 5 3 3 3 3 0 3 3 0 3 0 1 7 6 3 6
 3 0 0 0 3 6 3 3 3 0 0 3 6 6 0 6 0 3 0 3 4 3 3 3 3 0 3 0 0 3 4 6 0 3 0 0 3
 2 3 0 3 6 1 3 0 6 0 6 1 6 1 6 3 3 2 0 3 6 4 4 0 0 1 4 3 3 6 0 3 3 6 0 0 0
 3 7 4 3 3 0 0 3 3 4 0 3 4 6 0 2 0 7 0 3 0 6 3 0 4 4 0 4 4 0 4 3 0 3 3 0 4
 0 6 7 3 6 0 0 0 6 3 0 0 0 3 0 0 4 0 3 0 0 4 3 0 0 0 0 6 3 1 0 4 3 3 4 0 0
 0 0 6 3 0 0 4 0 3 4 4 6 6 3 3 6 0 3 1 3 3 3 0 3 3 0 0 4 0 4 0 0 3 3 0 3 3
 0 0 4 6 0 1 1 1 1 6 6 6 6 6 6 6 6 3 3 0 4 1 3 3 0 3 7 3 6 3 6 3 3 0 3 0 6
 3 4 0 0 4 4 3 3 3 4 6 6 6 7 1 1 3 5 1 1 0 6 1 4 0 0 0 7 1 0 6 7 1 0 1 0 6
 3 3 3 6 7 7 6 2 7 6 3 1 6 1 0 6 3 3 1 0 1 3 1 6 6 6 6 3 6 6 6 4 1 1 2 1 0
 4 1 4 2 0 0 3 0 3 1 1 2 3 6 6 0 3 1 6 7 6 0 4 4 0 1 4 1 7 3 6 3 6 3 7 1 6
 3 0 3 3 0 2 4 4 6 3 7 3 0 3 3 4 0 5 3 0 0 0 4 0 1 5 3 3 3 3 4 4 4 3 4 6 3
```

The output we got is a list of clusters, which are clustered by using their Euclidean distances from the Centroid. So, We have found the mapping of types of Colleges present in the US using K-Means Algorithm.

Using Hierarchical Clustering Algorithm:

The Hierarchical Clustering Algorithm Clusters the Data using the a special mathematical model called as a dendrogram. The Dendrogram is used to find the structure of the data-set to find the clusters in them.

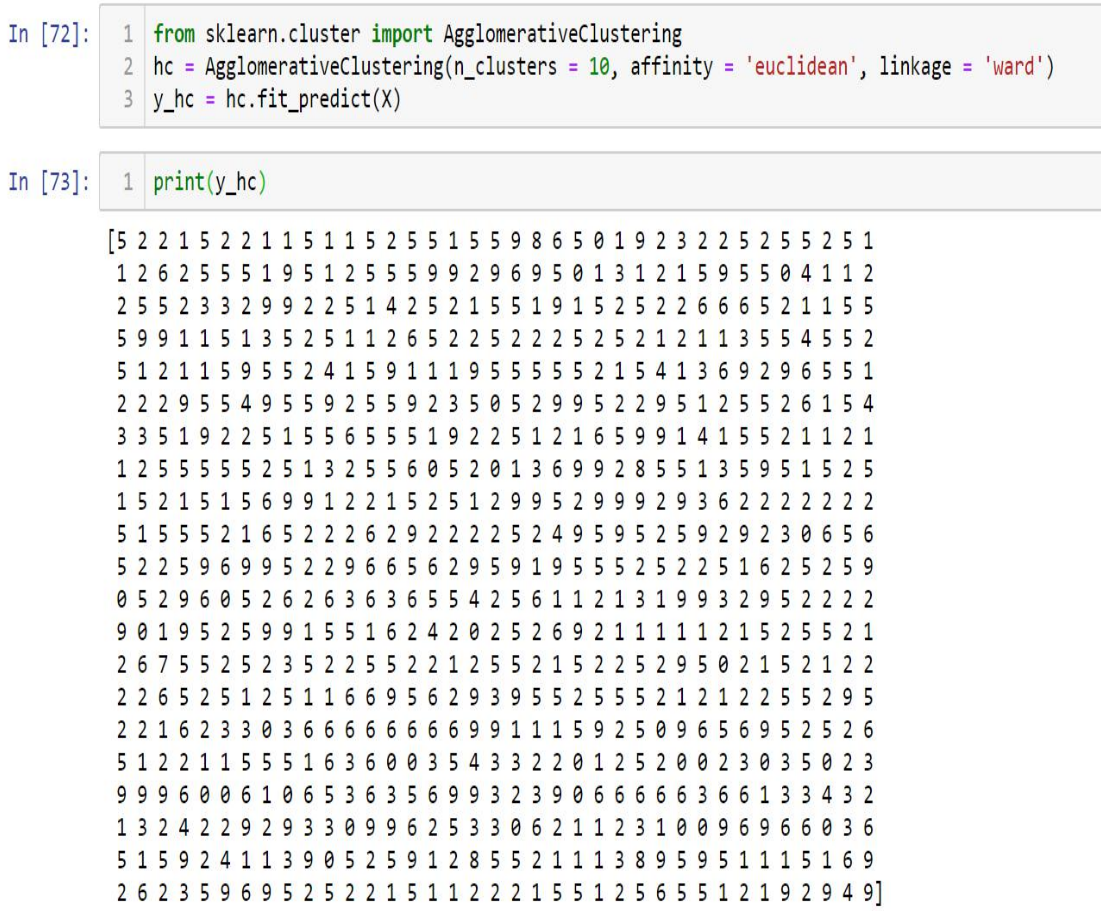
The Dendrogram Model:



```
import scipy.cluster.hierarchy as sch
dendrogram = sch.dendrogram(sch.linkage(X, method = 'ward'))
plt.title('Dendrogram')
plt.xlabel('Customers')
plt.ylabel('Euclidean distances')
plt.show()
```

The Number of Clusters in the data-set can be found by drawing a straight line through the most visible dendrogram lines. By counting the straight lines above the straight line, we can find the number of clusters, that our data-set should be clustered into.. By Doing that we found that there can be 10 categories that our Data-set can be clustered into.

The Code for the Hierarchical Clustering Algorithm is as shown in the figure below,



```
from sklearn.cluster import AgglomerativeClustering
hc = AgglomerativeClustering(n_clusters = 10, affinity = 'euclidean', linkage = 'ward')
y_hc = hc.fit_predict(X)
```

```
print(y_hc)
```

```
[5 2 2 1 5 2 2 1 1 5 1 1 5 2 5 5 1 5 5 9 8 6 5 0 1 9 2 3 2 2 5 2 5 5 2 5 1
 1 2 6 2 5 5 5 1 9 5 1 2 5 5 5 9 9 2 9 6 9 5 0 1 3 1 2 1 5 9 5 5 0 4 1 1 2
 2 5 5 2 3 3 2 9 9 2 2 5 1 4 2 5 2 1 5 5 1 9 1 5 2 5 2 2 6 6 6 5 2 1 1 5 5
 5 9 9 1 1 5 1 3 5 2 5 1 1 2 6 5 2 2 5 2 2 2 5 2 5 2 1 2 1 1 3 5 5 4 5 5 2
 5 1 2 1 1 5 9 5 5 2 4 1 5 9 1 1 1 9 5 5 5 5 5 2 1 5 4 1 3 6 9 2 9 6 5 5 1
 2 2 2 9 5 5 4 9 5 5 9 2 5 5 9 2 3 5 0 5 2 9 9 5 2 2 9 5 1 2 5 5 2 6 1 5 4
 3 3 5 1 9 2 2 5 1 5 5 6 5 5 5 1 9 2 2 5 1 2 1 6 5 9 9 1 4 1 5 5 2 1 1 2 1
 1 2 5 5 5 5 5 2 5 1 3 2 5 5 6 0 5 2 0 1 3 6 9 9 2 8 5 5 1 3 5 9 5 1 5 2 5
 1 5 2 1 5 1 5 6 9 9 1 2 2 1 5 2 5 1 2 9 9 5 2 9 9 9 2 9 3 6 2 2 2 2 2 2 2
 5 1 5 5 5 2 1 6 5 2 2 2 6 2 9 2 2 2 2 5 2 4 9 5 9 5 2 5 9 2 9 2 3 0 6 5 6
 5 2 2 5 9 6 9 9 5 2 2 9 6 6 5 6 2 9 5 9 1 9 5 5 5 2 5 2 2 5 1 6 2 5 2 5 9
 0 5 2 9 6 0 5 2 6 2 6 3 6 3 6 5 5 4 2 5 6 1 1 2 1 3 1 9 9 3 2 9 5 2 2 2 2
 9 0 1 9 5 2 5 9 9 1 5 5 1 6 2 4 2 0 2 5 2 6 9 2 1 1 1 1 1 2 1 5 2 5 5 2 1
 2 6 7 5 5 2 5 2 3 5 2 2 5 5 2 2 1 2 5 5 2 1 5 2 2 5 2 9 5 0 2 1 5 2 1 2 2
 2 2 6 5 2 5 1 2 5 1 1 6 6 9 5 6 2 9 3 9 5 5 2 5 5 5 2 1 2 1 2 2 5 5 2 9 5
 2 2 1 6 2 3 3 0 3 6 6 6 6 6 6 6 6 9 9 1 1 1 5 9 2 5 0 9 6 5 6 9 5 2 5 2 6
 5 1 2 2 1 1 5 5 5 1 6 3 6 0 0 3 5 4 3 3 2 2 0 1 2 5 2 0 0 2 3 0 3 5 0 2 3
 9 9 9 6 0 0 6 1 0 6 5 3 6 3 5 6 9 9 3 2 3 9 0 6 6 6 6 6 3 6 6 1 3 3 4 3 2
 1 3 2 4 2 2 9 2 9 3 3 0 9 9 6 2 5 3 3 0 6 2 1 1 2 3 1 0 0 9 6 9 6 6 0 3 6
 5 1 5 9 2 4 1 1 3 9 0 5 2 5 9 1 2 8 5 5 2 1 1 1 3 8 9 5 9 5 1 1 1 5 1 6 9
 2 6 2 3 5 9 6 9 5 2 5 2 2 1 5 1 1 2 2 2 1 5 5 1 2 5 6 5 5 1 2 1 9 2 9 4 9]
```

So from this array, we can obtain the features that are belonging to 10 categories, so if we know the features we can find which cluster the dendrogram belongs to.

Disadvantage of Hierarchical Clustering: Since this technique is done using Dendrograms, the dendrograms tend to shrink, for larger data-sets, so it is very difficult to find optimum number of clusters from the given data-set. It works best with the smaller data-set, but the efficiency decreases with increasing Data samples.

DEEP LEARNING

Deep Learning And Neural Networks:

Neural networks are a set of algorithms, modeled loosely after the human brain, that are designed to recognize patterns. They interpret sensory data through a kind of machine perception, labeling or clustering raw input. The patterns they recognize are numerical, contained in vectors, into which all real-world data, be it images, sound, text or time series, must be translated.

Neural networks help us cluster and classify. You can think of them as a clustering and classification layer on top of the data you store and manage. They help to group unlabeled data according to similarities among the example inputs, and they classify data when they have a labeled data-set to train on. (Neural networks can also extract features that are fed to other algorithms for clustering and classification; so you can think of deep neural networks as components of larger machine-learning applications involving algorithms for reinforcement learning, classification and regression.)

What kind of problems does deep learning solve, and more importantly, can it solve yours? To know the answer, you need to ask questions:

- What outcomes do I care about? Those outcomes are labels that could be applied to data: for example, spam or not_spam in an email filter, good_guy or bad_guy in fraud detection, angry_customer or happy_customer in customer relationship management.
- Do I have the data to accompany those labels? That is, can I find labeled data, or can I create a labeled data-set (with a service like AWS Mechanical Turk or Figure Eight or Mighty.ai) where spam has been labeled as spam, in order to teach an algorithm the correlation between labels and inputs?

Clustering Using Neural Networks:

Clustering or grouping is the detection of similarities. Deep learning does not require labels to detect similarities. Learning without labels is called unsupervised learning. Unlabeled data is the majority of data in the world. One law of machine learning is: the more data an algorithm can train on, the more accurate it will be. Therefore, unsupervised learning has the potential to produce highly accurate models.

- Search: Comparing documents, images or sounds to surface similar items.
- Anomaly detection: The flip-side of detecting similarities is detecting anomalies, or unusual behavior. In many cases, unusual behavior correlates highly with things you want to detect and prevent, such as fraud.

Artificial Neural Network Architecture:

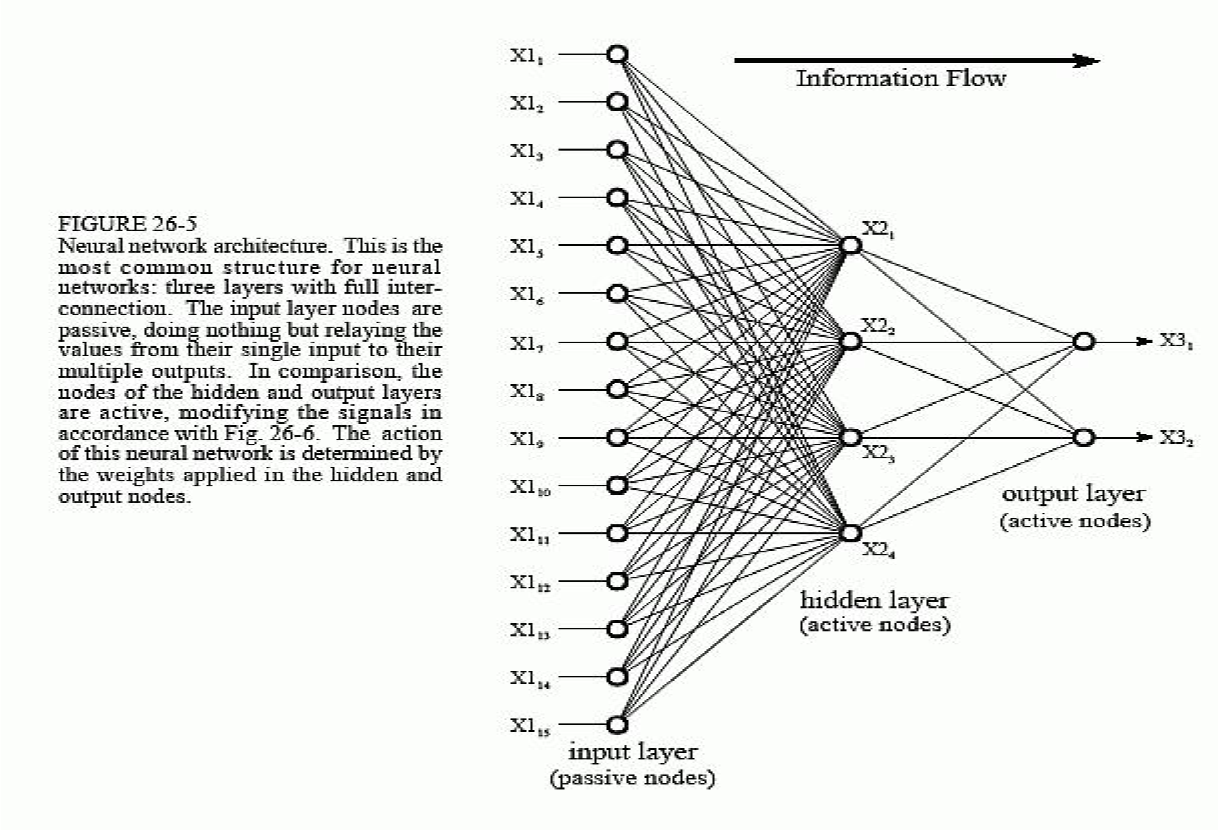


FIGURE 26-5
Neural network architecture. This is the most common structure for neural networks: three layers with full interconnection. The input layer nodes are passive, doing nothing but relaying the values from their single input to their multiple outputs. In comparison, the nodes of the hidden and output layers are active, modifying the signals in accordance with Fig. 26-6. The action of this neural network is determined by the weights applied in the hidden and output nodes.

A Typical Artificial Neuron:

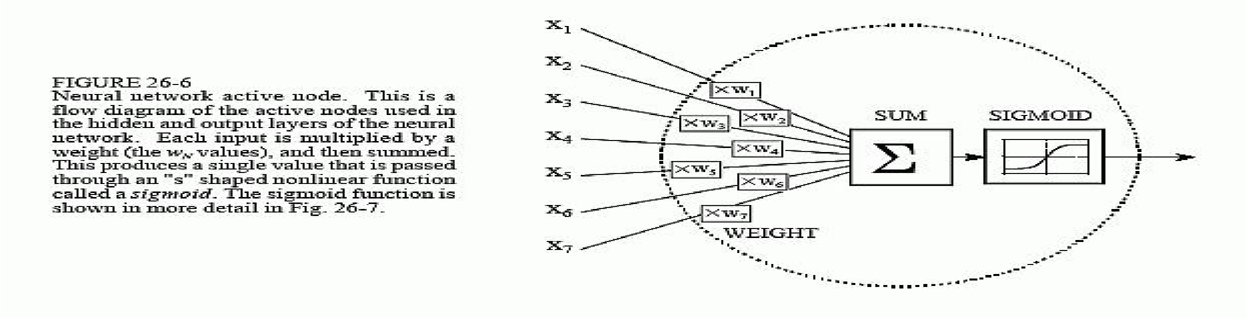
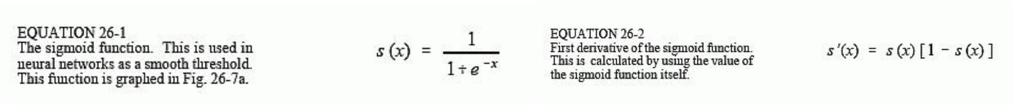


FIGURE 26-6
Neural network active node. This is a flow diagram of the active nodes used in the hidden and output layers of the neural network. Each input is multiplied by a weight (the w_N values), and then summed. This produces a single value that is passed through an "s" shaped nonlinear function called a *sigmoid*. The sigmoid function is shown in more detail in Fig. 26-7.

Activation and Loss Function in a Neuron:



EQUATION 26-1
The sigmoid function. This is used in neural networks as a smooth threshold. This function is graphed in Fig. 26-7a.

$$s(x) = \frac{1}{1+e^{-x}}$$

EQUATION 26-2
First derivative of the sigmoid function. This is calculated by using the value of the sigmoid function itself.

$$s'(x) = s(x)\,[1 - s(x)]$$

Deep Learning for Unsupervised Learning:

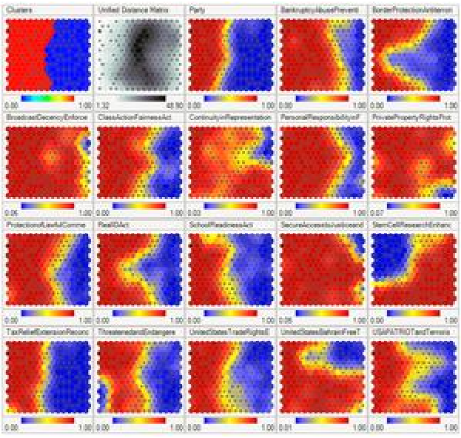
Neural networks are widely used in unsupervised learning in order to learn better representations of the input data. For example, given a set of text documents, NN can learn a mapping from document to real-valued vector in such a way that resulting vectors are similar for documents with similar content, i.e. distance preserving. This can be achieved using, for example, auto-encoders - a model that is trained to reconstruct the original vector from a smaller representation (hidden layer activations) with reconstruction error (distance from the ID function) as cost function. This process doesn't give you clusters, but it creates meaningful representations that can be used for clustering. You could, for instance, run a clustering algorithm on the hidden layer's activations.

Clustering: There are a number of different NN architectures specifically designed for clustering. The most widely known is probably self organizing maps. A SOM is a NN that has a set of neurons connected to form a topological grid (usually rectangular). When some pattern is presented to an SOM, the neuron with closest weight vector is considered a winner and its weights are adapted to the pattern, as well as the weights of its neighbourhood. In this way an SOM naturally finds data clusters. A somewhat related algorithm is growing neural gas (it is not limited to predefined number of neurons).

Another approach is Adaptive Resonance Theory where we have two layers: "comparison field" and "recognition field". Recognition field also determines the best match (neuron) to the vector transferred from the comparison field and also have lateral inhibitory connections. Implementation details and exact equations can readily found by googling the names of these models, so I won't put them here.

Self - Organizing Maps:

A self-organizing map (SOM) or self-organizing feature map (SOFM) is a type of artificial neural network (ANN) that is trained using unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map, and is therefore a method to do dimensionality reduction. Self-organizing maps differ from other artificial neural networks as they apply competitive learning as opposed to error-correction learning (such as backpropagation with gradient descent), and in the sense that they use a neighborhood function to preserve the topological properties of the input space.



A self-organizing map showing U.S. Congressvoting patterns. The input data was a table with a row for each member of Congress, and columns for certain votes containing each member's yes/no/abstain vote. The SOM algorithm arranged these members in a two-dimensional grid placing similar members closer together. The first plot shows the grouping when the data are split into two clusters. The second plot shows average distance to neighbors: larger distances are darker.

The third plot predicts Republican (red) or Democratic (blue) party membership. The other plots each overlay the resulting map with predicted values on an input dimension: red means a predicted 'yes' vote on that bill, blue means a 'no' vote. The plot was created in Synapse.

This makes SOMs useful for visualization by creating low-dimensional views of high-dimensional data, akin to multidimensional scaling.

In our project we can use SOMs to Cluster our Data well than the conventional Algorithms like K-Means, Hierarchical Clustering etc…

FUTURE OF MACHINE LEARNING

Forecasts About the Future of Machine Learning

1. Improved unsupervised algorithms:

In machine learning, unsupervised algorithms are employed to make predictions from data-sets when only input data is available without corresponding output variables.

Whereas in supervised learning the output of the algorithm is already known, its unsupervised counterpart is closely associated with true artificial intelligence—the concept that a machine can learn to identify complicated processes and patterns without any direct human intervention.

In the coming years, we are likely to see improvements in unsupervised machine learning algorithms. The advancements in developing better algorithms will result in faster and more accurate machine learning predictions.

2. Improved cognitive services:

Cognitive services consist of a set of machine learning SDKs, APIs, and services, which allow developers to include intelligent capabilities into their applications.

With such services, developers can empower their applications to carry out various duties, such as vision recognition, speech detection, and speech understanding.

Therefore, developers will be able to build more engaging and discoverable applications that can effectively interpret users’ needs based on natural communication techniques.

3. Rise of Robots:

As machine learning is becoming more sophisticated, we’ll see increased usage of robots. Robotization depends on machine learning for accomplishing various purposes, including robot vision, self-supervised learning, and multi-agent learning.

Soon, we expect robots to become more intelligent at accomplishing tasks. Drones, robots in manufacturing places, and other types of robots are likely to be used increasingly to make our lives easier.

CONCLUSION

Therefore the Data-set given has been manipulated, clustered and Analysis has been performed by using more powerful state-of-the art Machine Learning Algorithms like K-Means Clustering, Hierarchical Clustering, Self-Organizing Maps (SOMs).

Machine learning is not only used in this problem, It can solve many problems that humans cannot dream. The growing power of computing and growing data, can be used to perform different types of analysis on that data.