CERTIFICATE

This is to certify that the project work done on **Customer Segmentation Using Machine Learning** submitted to Maharaja Surajmal Institute of Technology, Janakpuri, New Delhi by Kevin Singh Bagga, Manvendra Gupta, Shivam Kakkar, Shubham Kukreti in partial fulfilment of the requirement for the award of degree of Bachelor of Technology, is a Bonafede work carried out by him/her under my supervision and guidance. The project work is the original one and has not submitted anywhere else for any other degree.

Ms. Preeti Rathee — Dr Anupama Kaushik

(Project Guide) — (HOD,IT Dept)

ACKNOWLEDGEMENT

We express our sincere thanks and deep sense of gratitude to our project mentor **Ms. Preeti Rathee**, Department of Information Technology, Maharaja Surajmal Institute of Technology, for her vulnerable motivation and guidance, without which this report would not have been possible. We consider ourselves fortunate for having the opportunity to learn and work under her/his able supervision and guidance over the period of association. We have deep sense of admiration for her/his innate goodness**.**

Kevin Singh Bagga (01015003117)

Manvendra Gupta (01315003117)

Shivam Kakkar (02115003117)

Shubham Kukreti (02415003117)

Place:

Date:

ABSTRACT

We live in a world where large and vast amount of data is collected daily. Analysing such data is an important need. In the modern era of innovation, where there is a large competition to be better then everyone, the business strategy needs to be according to the modern conditions. The business done today runs on the basis of innovative ideas as there are large number of potential customers who are confounded to what to buy and what not to buy. The companies doing the business are also not able to diagnose the target potential customers. This is where the machine learning comes into picture, the various algorithms are applied to identify the hidden patterns in the data for better decision making. The concept of which customer segment to target is done using the customer segmentation process using the clustering technique. In this paper, the clustering algorithm used is K-means algorithm which is the partition in algorithm, to segment the customers according to the similar characteristics. To determine the optimal clusters, elbow method is used.

TABLE OF CONTENTS

List of Figures

List of Abbreviations

ML – Machine Learning

AI – Artificial Intelligence

ECML – European Conference on Machine Learning

PKDD – European Conference on Principles and Practice of Knowledge Discovery in Databases

KDD – Knowledge Discovery and Data Mining

SL – Supervised Learning

UL – Unsupervised Learning

SVM – Support Vector Machine

MDP – Markov Development Process

HAC – Hierarchical Agglomerative Clustering

RMSE – Root Mean Squared Error

MSE – Mean Squared Error

HVAC – Heating, Ventilation and Air Conditioning

List of Symbols

Regression:

x: input training data (univariate – one input variable(parameter))

y: labels to data (supervised learning)

θ_1: intercept

θ_2: coefficient of x

J: Cost Function

$h_\Theta(x)$: Hypothesis of sigmoid function

Chapter 1

Introduction

Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data. Data science is related to data mining, machine learning and big data.

Data science is a "concept to unify statistics, data analysis and their related methods" in order to "understand and analyse actual phenomena" with data. It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, domain knowledge and information science. Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.

1.1 Overview

The name machine learning was coined in 1959 by Arthur Samuel. Tom M. Mitchell provided a widely quoted, more formal definition of the algorithms studied in the machine learning field: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." This definition of the tasks in which machine learning is concerned offers a fundamentally operational definition rather than defining the field in cognitive terms. This follows Alan Turing's proposal in his paper "Computing Machinery and Intelligence", in which the question "Can machines think?" is replaced with the question "Can machines do what we (as thinking entities) can do?". In Turing's proposal the various characteristics that could be possessed by a thinking machine and the various implications in constructing one are exposed.

1.2 History and Relation with other fields

Arthur Samuel, an American pioneer in the field of computer gaming and artificial intelligence, coined the term "Machine Learning" in 1959 while at IBM.As a scientific endeavour, machine learning grew out of the quest for artificial intelligence. Already in the early days of AI as an academic discipline, some researchers were interested in having machines learn from data. They attempted to approach the problem with various symbolic methods, as well as what were then termed "neural networks"; these were mostly perceptron and other models that were later found to be reinventions of the generalized linear models of statistics. Probabilistic reasoning was also employed, especially in automated medical diagnosis.

However, an increasing emphasis on the logical, knowledge-based approach caused a rift between AI and machine learning. Probabilistic systems were plagued by theoretical and practical problems of data acquisition and representation. By 1980, expert systems had come to dominate AI, and statistics was out of favour. Work on symbolic/knowledge-based learning did continue within AI, leading to inductive logic programming, but the more statistical line of research was now outside the field of AI proper, in pattern recognition and information retrieval. Neural networks research had been abandoned by AI and computer science around the same time. This line, too, was continued outside the AI/CS field, as "connectionism", by researchers from other disciplines including Hopfield, Rumelhart and Hinton. Their main success came in the mid-1980s with the reinvention of backpropagation.

Machine learning, reorganized as a separate field, started to flourish in the 1990s. The field changed its goal from achieving artificial intelligence to tackling solvable problems of a practical nature. It shifted focus away from the symbolic approaches it had inherited from AI, and toward methods and models borrowed from statistics and probability theory. It also benefited from the increasing availability of digitized information, and the ability to distribute it via the Internet.

1.2.1 Data Mining

Machine learning and data mining often employ the same methods and overlap significantly, but while machine learning focuses on prediction, based on known properties learned from the training data, data mining focuses on the discovery of (previously) unknown properties in the data (this is the analysis step of knowledge discovery in databases). Data mining uses many machine learning methods, but with different goals; on the other hand, machine learning also employs data mining methods as "unsupervised learning" or as a pre-processing step to improve learner accuracy. Much of the confusion between these two research communities (which do often have separate conferences and separate journals, ECML PKDD being a major exception) comes from the basic assumptions they work with: in machine learning, performance is usually evaluated with respect to the ability to reproduce known knowledge, while in knowledge discovery and data mining (KDD) the key task is the discovery of previously unknown knowledge. Evaluated with respect to known knowledge, an uninformed (unsupervised) method will easily be outperformed by other supervised methods, while in a typical KDD task, supervised methods cannot be used due to the unavailability of training data.

1.2.2 Optimization

Machine learning also has intimate ties to optimization: many learning problems are formulated as minimization of some loss function on a training set of examples. Loss functions express the discrepancy between the predictions of the model being trained and the actual problem instances (for example, in classification, one wants to assign a label to instances, and models are trained to correctly predict the pre-assigned labels of a set of examples). The difference between the two fields arises from the goal of generalization: while optimization algorithms can minimize the loss on a training set, machine learning is concerned with minimizing the loss on unseen samples.

1.2.3 Statistics

Machine learning and statistics are closely related fields in terms of methods, but distinct in their principal goal: statistics draws population inferences from a sample, while machine learning finds generalizable predictive patterns. According to Michael

I. Jordan, the ideas of machine learning, from methodological principles to theoretical tools, have had a long pre-history in statistics. He also suggested the term data science as a placeholder to call the overall field.

Leo Breiman distinguished two statistical modelling paradigms: data model and algorithmic model, wherein "algorithmic model" means more or less the machine learning algorithms like Random forest.

Some statisticians have adopted methods from machine learning, leading to a combined field that they call *statistical learning*.

1.3 Why do we use Data Science?

The principal purpose of Data Science is to find patterns within data. It uses various statistical techniques to analyze and draw insights from the data. From data extraction, wrangling and pre-processing, a Data Scientist must scrutinize the data thoroughly. Then, he has the responsibility of making predictions from the data. The goal of a Data Scientist is to derive conclusions from the data. Through these conclusions, he is able to assist companies in making smarter business decisions.

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

Resurging interest in machine learning is due to the same factors that have made data mining and Bayesian analysis more popular than ever. Things like growing volumes and varieties of available data, computational processing that is cheaper and more powerful, and affordable data storage.

All of these things mean it's possible to quickly and automatically produce models that can analyse bigger, more complex data and deliver faster, more accurate results – even on a very large scale. And by building precise models, an organization has a better chance of identifying profitable opportunities – or avoiding unknown risks.

1.4 Where can we use Machine Learning?

ML is used when:

• Human expertise does not exist (navigating on Mars)

Fig 1.4.1 Rover on Mars

• Humans can’t explain their expertise (speech recognition)

Fig 1.4.2 Sample speech signals

• Models must be customized (personalized medicine)

• Models are based on huge amounts of data (genomics)

Chapter 2

Statistics

Statistics is used to process complex problems in the real world so that Data Scientists and Analysts can look for meaningful trends and changes in Data. In simple words, Statistics can be used to derive meaningful insights from data by performing mathematical computations on it.

2.1 Types of Statistics

When analysing data, such as the marks achieved by 100 students for a piece of coursework, it is possible to use both descriptive and inferential statistics in your analysis of their marks. Typically, in most research conducted on groups of people, you will use both descriptive and inferential statistics to analyse your results and draw conclusions. *So, what are descriptive and inferential statistics? And what are their differences?*

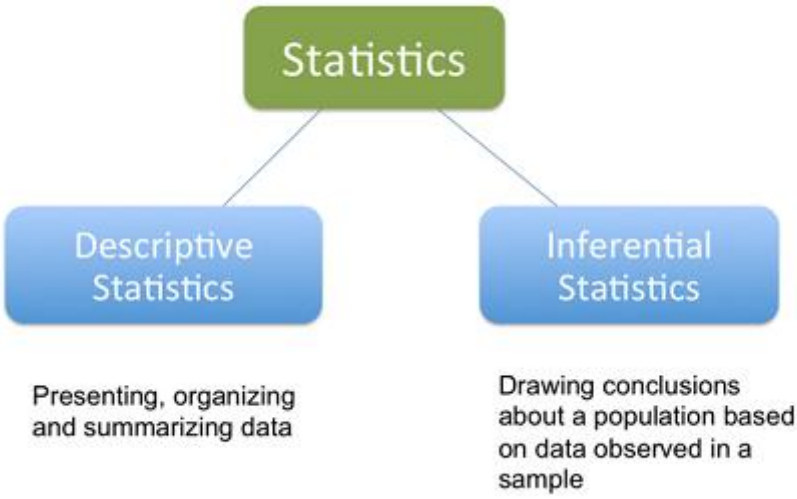


Fig 2.1 Types of Statistics

2.1.1 Descriptive Statistics

Descriptive statistics is a term given to the analysis of data that helps to describe, show and summarize data in a meaningful way. It is a simple way to describe our data. Descriptive statistics is very important to present our raw data ineffective/meaningful way using numerical calculations or graphs or tables. This type of statistics is applied on already known data.

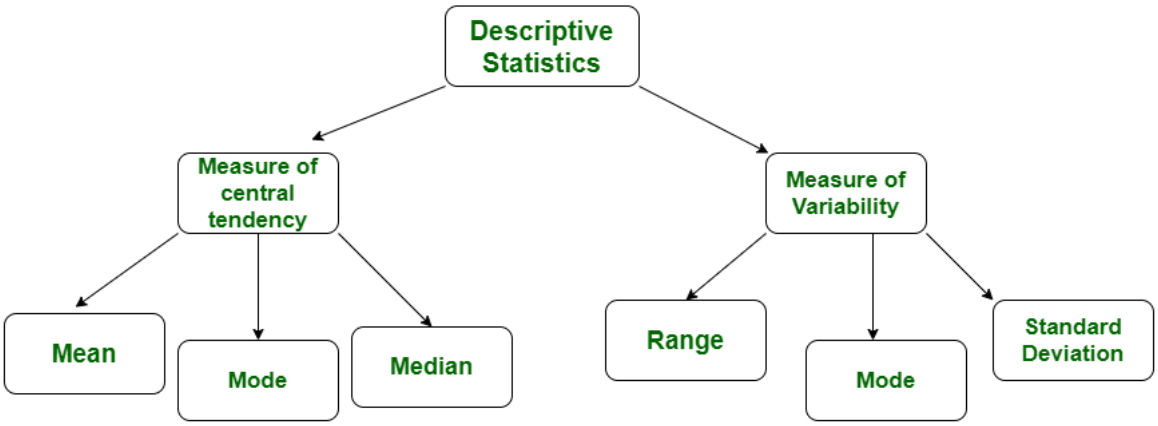


Fig 2.1.1 Flow Diagram Descriptive Statistics

2.1.2 Inferential Statistics

In inferential statistics predictions are made by taking any group of data in which you are interested. It can be defined as a random sample of data taken from a population to describe and make inference about the population. Any group of data which includes all the data you are interested is known as population. It basically allows you to make predictions by taking a small sample instead of working on whole population.

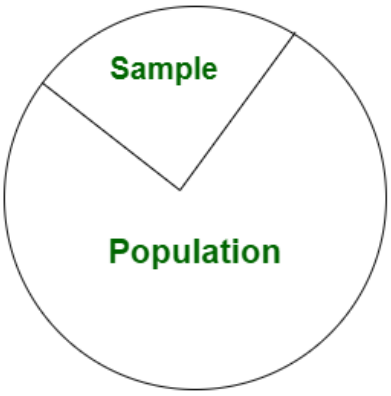


Fig 2.1.2 Flow Diagram Inferential Statistics

2.1.3 Descriptive Statistics v/s Inferential Statistics

S.NO.	DESCRIPTIVE STATISTICS	INFERENTIAL STATISTICS
1.	It gives information about raw data which describes the data in some manner.	It makes inference about population using data drawn from the population.
2.	It helps in organizing, analysing and to present data in a meaningful manner.	It allows us to compare data, make hypothesis and predictions.
3.	It is used to describe a situation.	It is used to explain the chance of occurrence of an event.
4.	It explain already known data and limited to a sample or population having small size.	It attempts to reach the conclusion about the population.
5.	It can be achieved with the help of charts, graphs, tables etc.	It can be achieved by probability.

Table 2.1.3 Difference in Statistics

2.2 Measures of Central Tendency

A measure of central tendency is a single value that attempts to describe a set of data by identifying the central position within that set of data. As such, measures of central tendency are sometimes called measures of central location. They are also classed as summary statistics.

2.2.1 Mean

The mean is equal to the sum of all the values in the data set divided by the number of values in the data set. So, if we have n values in a data set and they have values x1,x2, …,xn, the sample mean, usually denoted by $\bar{x}$ (pronounced "x bar"), is:

$$\frac{x_1 + x_2 + \cdots + x_n}{n}$$

Fig 2.2.1 Mean Formula

2.2.2 Median

The median is the middle value. It is the value that splits the dataset in half. To find the median, order your data from smallest to largest, and then find the data point that has an equal amount of values above it and below it.

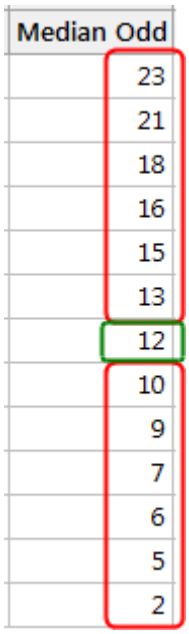


Fig 2.2.2-Median Formula (Odd)

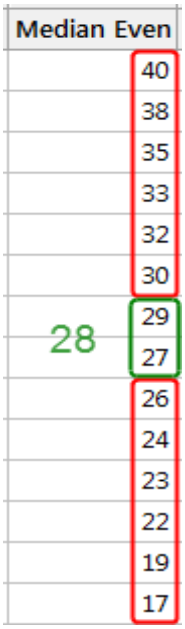
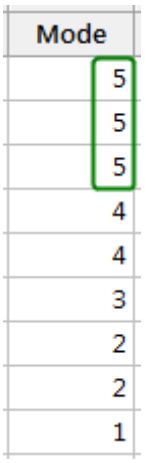


Fig 2.2.2-Median Formula (Even)

2.2.3 Mode

The mode is the value that occurs the most frequently in your data set. On a bar chart, the mode is the highest bar. If the data have multiple values that are tied for occurring the most frequently, you have a multimodal distribution. If no value repeats, the data do not have a mode.



Mode
5
5
5
4
4
3
2
2
1

Fig 2.2.3 Mode Formula

Chapter 3

Categories

The types of machine learning algorithms differ in their approach, the type of data they input and output, and the type of task or problem that they are intended to solve.

3.1 Supervised Learning

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and a desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task.

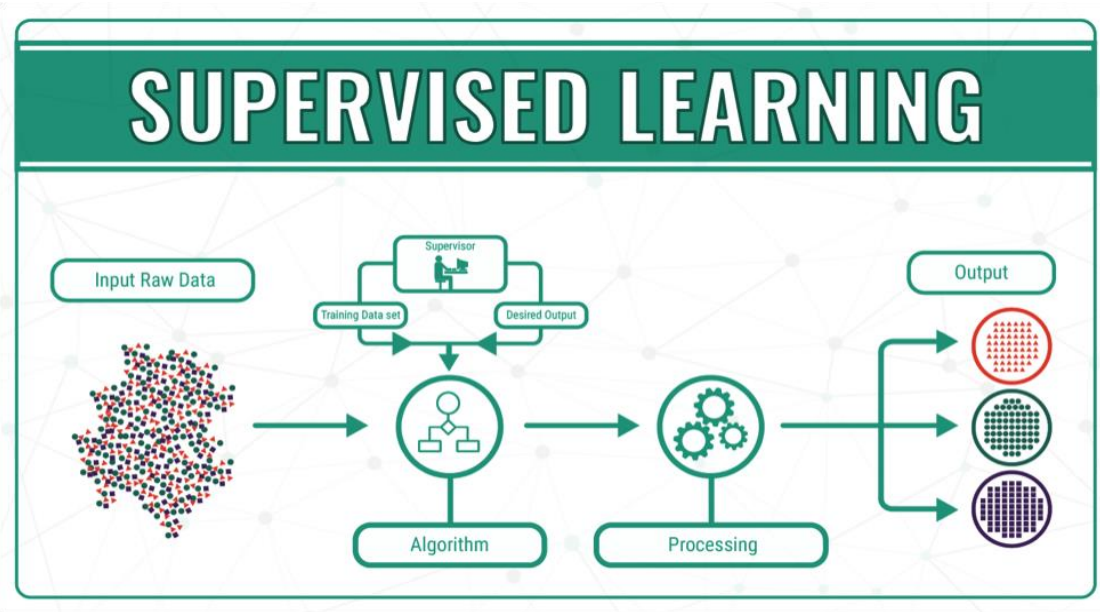


Fig 3.1 Flow Chart Depiction of Supervised Learning

Supervised learning algorithms include classification and regression. Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range. Similarity learning is an area of supervised machine learning closely related to regression and classification, but the goal is to learn from examples using a similarity function that measures how similar or related two objects are. It has applications in ranking, recommendation systems, visual identity tracking, face verification, and speaker verification.

3.2 Unsupervised Learning

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms therefore learn from test data that has not been labelled, classified or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, though unsupervised learning encompasses other domains involving summarizing and explaining data features.

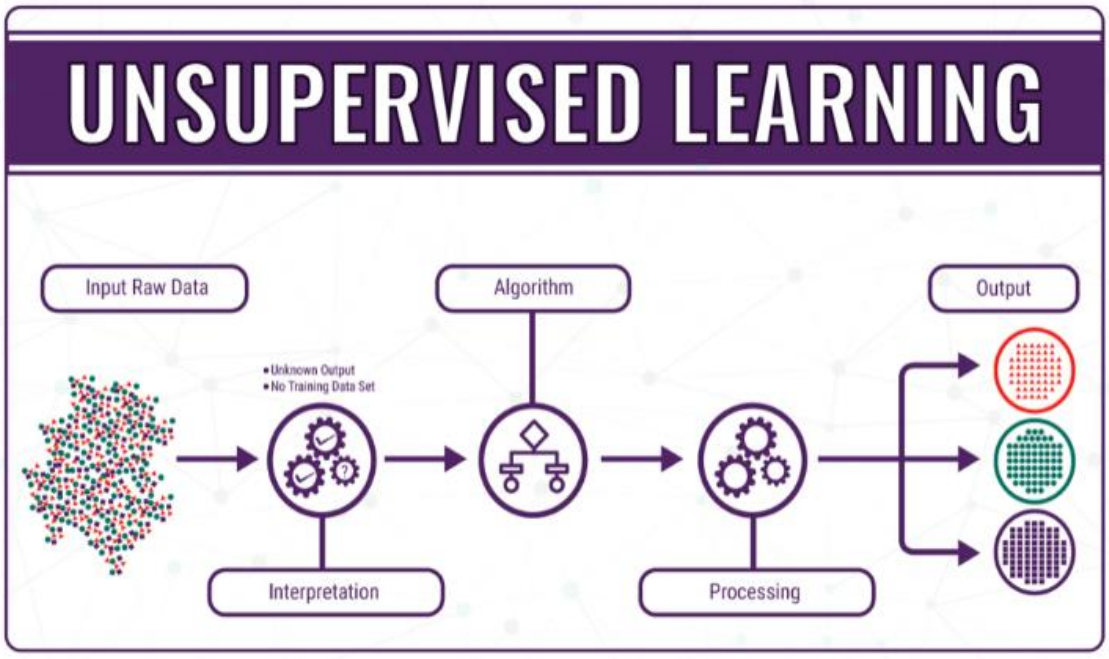


Fig 3.2 Flow chart depicting unsupervised learning

Cluster analysis is the assignment of a set of observations into subsets (called clusters) so that observations within the same cluster are similar according to one or more predesignated criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some similarity metric and evaluated, for example, by internal compactness, or the similarity between members of the same cluster, and separation, the difference between clusters. Other methods are based on estimated density and graph connectivity.

3.3 Semi-supervised Learning

Semi-supervised machine learning is a combination of supervised and unsupervised machine learning methods. In the case of semi-supervised learning algorithms, some of the training examples are missing training labels, but they can nevertheless be used to improve the quality of a model. In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets.

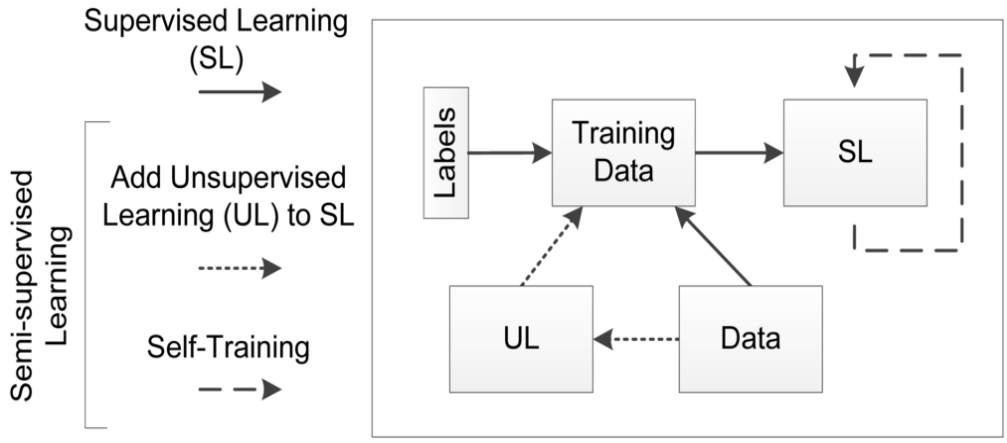


Fig 3.3 Semi-supervised Learning

With more common supervised machine learning methods, you train a machine learning algorithm on a "labelled" dataset in which each record includes the outcome information. This allows the algorithm to deduce patterns and identify relationships between your target variable and the rest of the dataset based on information it already has. In contrast, unsupervised machine learning algorithms learn from a dataset without the outcome variable. In semi-supervised learning, an algorithm learns from a dataset that includes both labelled and unlabelled data, usually mostly unlabelled.

3.4 Reinforcement Learning

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. In machine learning, the environment is typically represented as a Markov Decision Process (MDP). Many reinforcement learning algorithms use dynamic programming techniques. Reinforcement learning algorithms do not assume knowledge of an exact mathematical model of the MDP, and are used when exact models are infeasible. Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent.

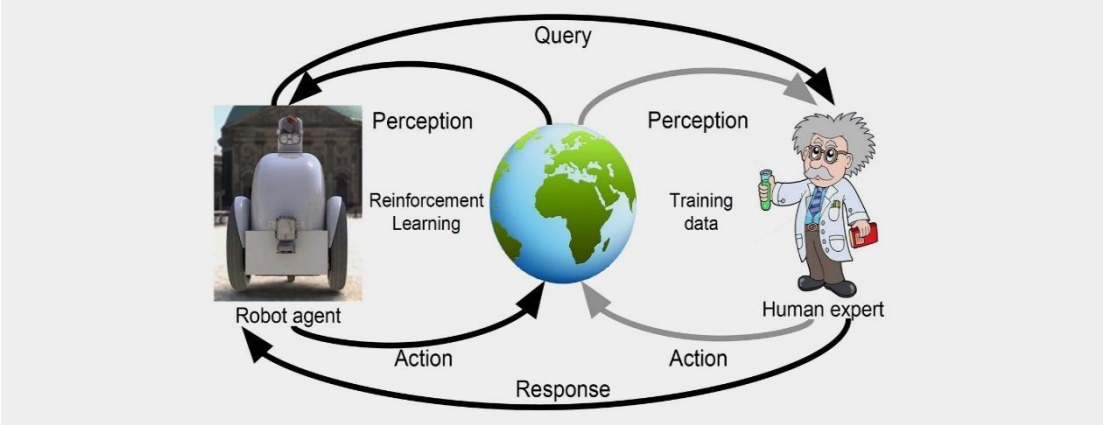


Fig 3.4 Reinforcement Learning

Chapter 4

Clustering

4.1 Introduction

Clustering is a Machine Learning technique that involves the grouping of data points. Given a set of data points, we can use a clustering algorithm to classify each data point into a specific group. In theory, data points that are in the same group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features. Clustering is a method of unsupervised learning and is a common technique for statistical data analysis used in many fields.

In Data Science, we can use clustering analysis to gain some valuable insights from our data by seeing what groups the data points fall into when we apply a clustering algorithm.

4.2 K-means Clustering

K-Means is probably the most well-known clustering algorithm. It's taught in a lot of introductory data science and machine learning classes. It's easy to understand and implement.

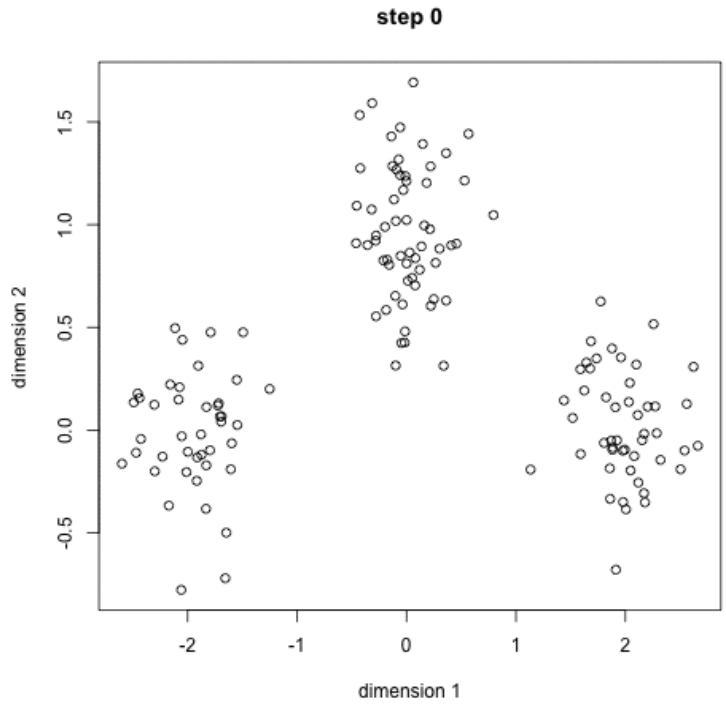


Fig 4.2 K-means Clustering

1. To begin, we first select a number of classes/groups to use and randomly initialize their respective centre points. To figure out the number of classes to use, it's good to take a quick look at the data and try to identify any distinct groupings. The centre points are vectors of the same length as each data point vector and are the "X's" in the graphic above.

2. Each data point is classified by computing the distance between that point and each group centre, and then classifying the point to be in the group whose centre is closest

3. Based on these classified points, we recompute the group centre by taking the mean of all the vectors in the group.

4. Repeat these steps for a set number of iterations or until the group centres don't change much between iterations. You can also opt to randomly initialize the group centres a few times, and then select the run that looks like it provided the best results.

K-Means has the advantage that it's pretty fast, as all we're really doing is computing the distances between points and group centres; very few computations! It thus has a linear complexity $O(n)$.

4.3 Hierarchical Clustering

Hierarchical clustering algorithms fall into 2 categories: top-down or bottom-up. Bottom-up algorithms treat each data point as a single cluster at the outset and then successively merge (or *agglomerate*) pairs of clusters until all clusters have been merged into a single cluster that contains all data points. Bottom-up hierarchical clustering is therefore called *hierarchical agglomerative clustering* or *HAC*. This hierarchy of clusters is represented as a tree (or dendrogram). The root of the tree is the unique cluster that gathers all the samples, the leaves being the clusters with only one sample.

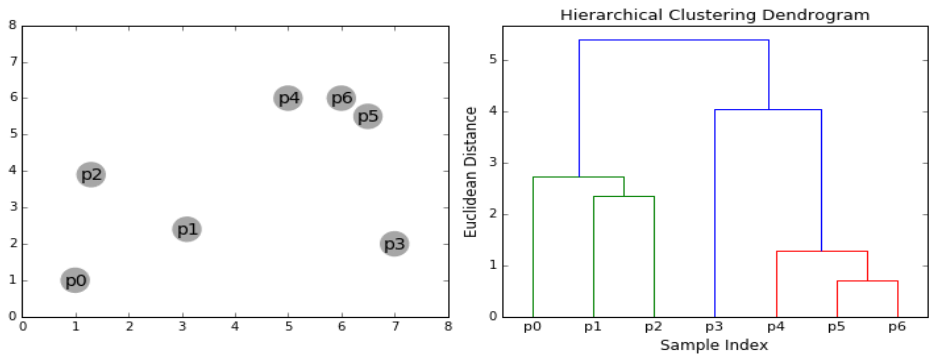


Fig 4.3 Hierarchical Clustering

Chapter 5

Project Work

5.1 Introduction

Over the years, the competition amongst businesses is increased and the large historical data that is available has resulted in the widespread use of data mining techniques in extracting the meaningful and strategic information from the database of the organisation. Data mining is the process where methods are applied to extract data patterns in order to present it in the human readable format which can be used for the purpose of decision support. According to, Clustering techniques consider data tuples as objects. They partition the data objects into groups or clusters, so that objects within a cluster are similar to one another and dissimilar to objects in other clusters.

Customer Segmentation is the process of division of customer base into several groups called as customer segments such that each customer segment consists of customers who have similar characteristics. The segmentation is based on the similarity in different ways that are relevant to marketing such as gender, age, interests, and miscellaneous spending habits.

The customer segmentation has the importance as it includes, the ability to modify the programs of market so that it is suitable to each of the customer segment, support in business decision; identification of products associated with each customer segment and to manage the demand and supply of that product; identifying and targeting the potential customer base, and predicting customer defection, providing directions in finding the solutions.

The thrust of this paper is to identify customer segments using the data mining approach, using the partitioning algorithm called as K-means clustering algorithm. The elbow method determines the optimal clusters.

5.2 Customer Segmentation

Over the years, as there is very strong competition in the business world, the organizations have to enhance their profits and business by satisfying the demands of their customers and attract new customers according to their needs. The identification

of customers and satisfying the demands of each customer is a very complex and tedious task. This is because customers may be different according to their demands, tastes, preferences and so on. Instead of "one-size-fits-all" approach, customer segmentation clusters the customers into groups sharing the same properties or behavioural characteristics. According to,[5] customer segmentation is a strategy of dividing the market into homogenous groups. The data used in customer segmentation technique that divides the customers into groups depends on various factors like, data geographical conditions, economic conditions, demographical conditions as well as behavioural patterns. The customer segmentation technique allows the business to make better use of their marketing budgets, gain a competitive edge over their rival companies, demonstrating the better knowledge of the needs of the customer. It also helps an organization in, increasing their marketing efficiency, determining new market opportunities, making better brand strategy, identifying customers retention.

5.3 Methodology

The data set used to implement clustering and K- means algorithm contains 10 attributes and has 850 tuples, representing the data of 850 customers. The attributes in the data set has CustomerId, age, Edu, Years Employed, Income(**K$**), Card Debt, Other Debt, Address, Debt-Income ratio.

5.4 Visualize age of customers

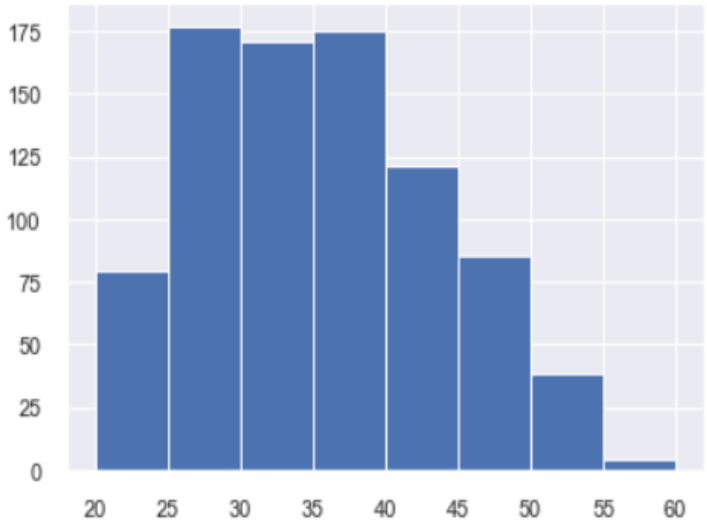


Fig 5.4 Customer Age – Bar representation

5.5 Methods Used and Results-

5.5.1 Elbow Method:

The elbow method is based on the observation that increasing the number of clusters can help to reduce the sum of within-cluster variance of each cluster. This is because having more clusters allows one to capture finer groups of data objects that are more similar to each other. To define the optimal clusters, Firstly, we use the clustering algorithm for various values of k. This is done by ranging k from 1 to 10 clusters. Then we calculate the total intra-cluster sum of square. Then, we proceed to plot intra-cluster sum of square based on the number of clusters. The plot denotes the approximate number of clusters required in our model. The optimum clusters can be found from the graph where there is a bend in the graph.

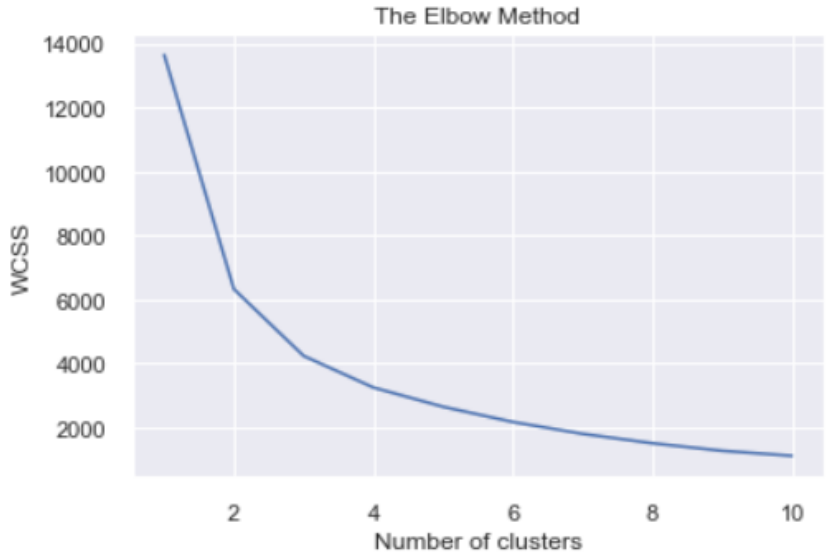


Fig 5.5.1.1 Elbow Curve to Select Optimal Number of Clusters

Here the optimal number of clusters is 3.

```
from sklearn.cluster import KMeans
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```

Fig 5.5.1.2 Code for Implementing Elbow Curve

Visualize the clusters

```
plt.bar(height = clus.value_counts(),x = ['1','2','3'])
plt.xlabel('Cluster', fontsize=13)
plt.ylabel('Number of Customer', fontsize=13)
```

Fig 5.5.1.3 Code for Representation of Count in Cluster by Elbow Method

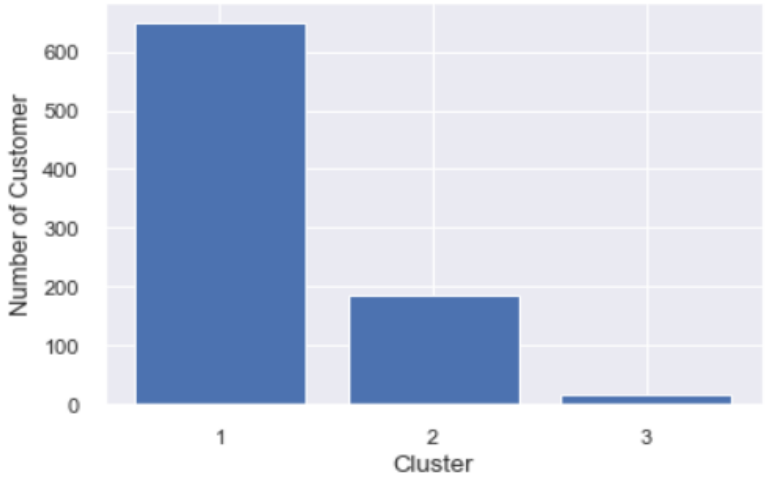


Fig 5.5.1.4 Customer Count in Cluster – Bar representation (Elbow Method)

```
kmeans = KMeans(n_clusters = 3, init = 'k-means++', random_state = 42)
y_kmeans = kmeans.fit_predict(X)
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 10, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 10, c = 'blue', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 10, c = 'green', label = 'Cluster 3')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 30, c = 'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.ylabel('Card Debt')
plt.xlabel('Other Debt')
plt.legend()
plt.show()
```

Fig 5.5.1.5 Code for Cluster's representation using Elbow Method

Fig 5.5.1.6 Cluster's representation using Elbow Method

5.5.2 Silhouette Coefficient Method

The Silhouette Coefficient for a point *i* is defined as follows:

$$S(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}$$

Fig 5.5.2.1 Formula for Calculating Silhouette Coefficient

where *b(i)* is the smallest average distance of point *i* to all points in any other cluster and *a(i)* is the average distance of *i* from all other points in its cluster. For example, if we have only 3 clusters A, B and C and i belongs to cluster C, then *b(i)* is calculated by measuring the average distance of *i* from every point in cluster A, the average distance of i from every point in cluster B and taking the smallest resulting value. The Silhouette Coefficient for the dataset is the average of the Silhouette Coefficient of individual points.

The Silhouette Coefficient tells us if individual points are correctly assigned to their clusters. We can use the following thumb rules while using Silhouette Coefficient:

S(i) close to 0 means that the point is between two clusters.

If it is closer to -1, then we would be better off assigning it to the other clusters.

If *S(i)* is close to 1, then the point belongs to the 'correct' cluster.

```
X=df.iloc[:, [4,8]].values
for n_cluster in range(2, 11):
    kmeans = KMeans(n_clusters=n_cluster).fit(X)
    label = kmeans.labels_
    sil_coeff = silhouette_score(X, label, metric='euclidean')
    print("For n_clusters={}, The Silhouette Coefficient is {}".format(n_cluster, sil_coeff))
```

Fig 5.5.2.2 Code for Silhouette Coefficients of up to 10 Clusters

```
For n_clusters=2, The Silhouette Coefficient is 0.7249334331557766
For n_clusters=3, The Silhouette Coefficient is 0.6246563472359685
For n_clusters=4, The Silhouette Coefficient is 0.5601389432778406
For n_clusters=5, The Silhouette Coefficient is 0.4704425335143334
For n_clusters=6, The Silhouette Coefficient is 0.46642275213653067
For n_clusters=7, The Silhouette Coefficient is 0.44821814431444257
For n_clusters=8, The Silhouette Coefficient is 0.4191154154047755
For n_clusters=9, The Silhouette Coefficient is 0.39474008589280185
For n_clusters=10, The Silhouette Coefficient is 0.38806400310609285
```

Fig 5.5.2.3 Silhouette Coefficients up to 10 Clusters

```
data = np.array(silh)
x, y = data.T
figure(figsize=(12, 6))
plt.scatter(x,y)
plt.plot(x,y)
plt.title('The Silhouette Method', fontsize=22)
plt.xlabel('Number of clusters', fontsize=18)
plt.ylabel('Silhouette Score', fontsize=18)
for x,y in zip(x,y):

    label = "{:.4f}".format(y)

    plt.annotate(label, # this is the text
                 (x,y), # this is the point to label
                 textcoords="offset points", # how to position the text
                 xytext=(0,10), # distance from text to points (x,y)
                 ha='center', fontsize=12) # horizontal alignment can be left, right or center
plt.xticks(np.arange(0,12,1))
plt.yticks(np.arange(0,0.7,0.1))
plt.show()
```

Fig 5.5.2.4 Code for Representation of Silhouette Score Graph

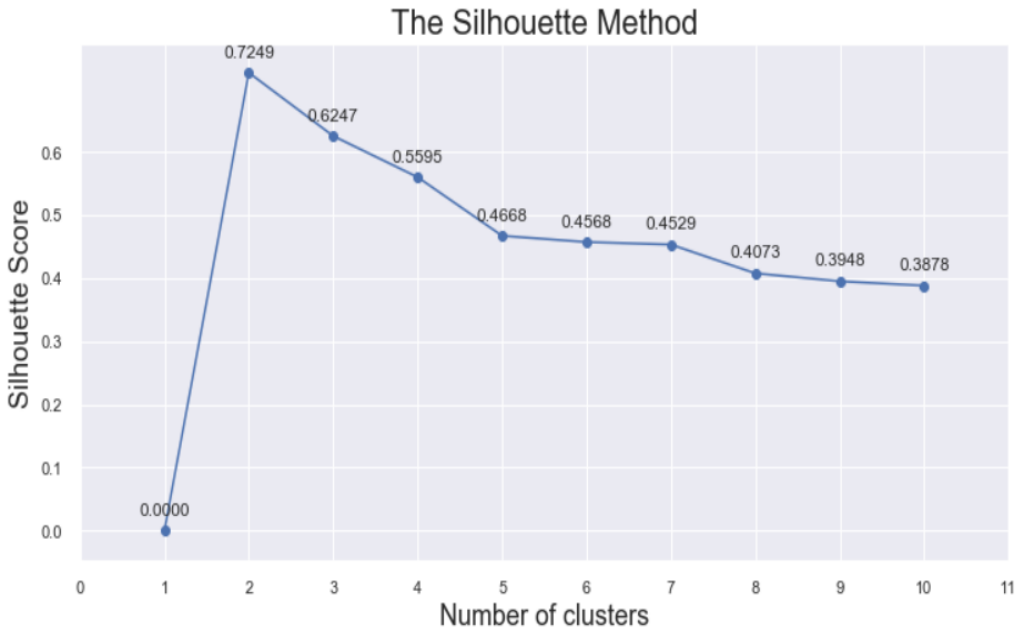


Fig 5.5.2.5 Graphical Representation of Silhouette Scores

Here the optimal number of clusters is 2.

Visualize the clusters

```
plt.bar(height = silh.value_counts(),x = ['1','2'])
plt.xlabel('Cluster', fontsize=13)
plt.ylabel('Number of Customer', fontsize=13)
```

Fig 5.5.2.6 Code for Representation of Count in Clusters by Silhouette Method

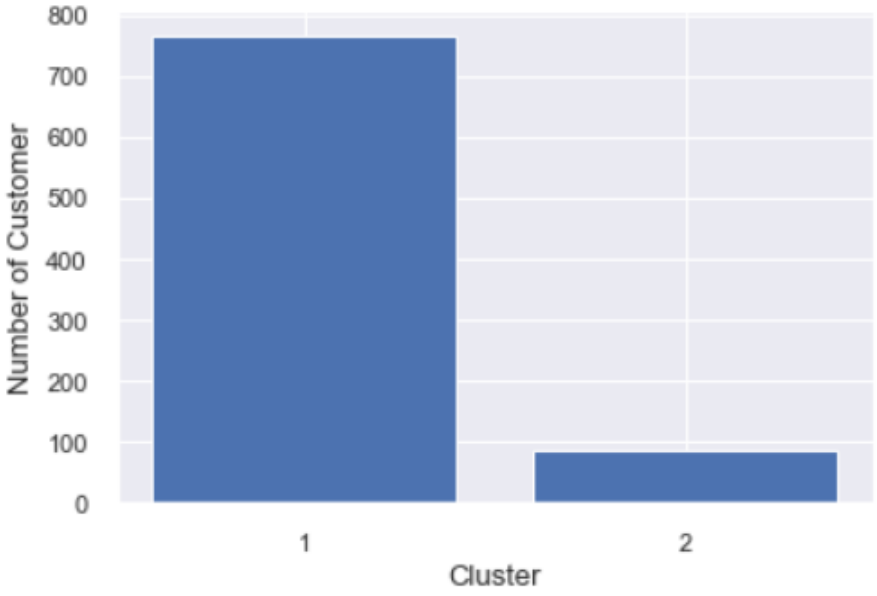


Fig 5.5.2.7 Bar Representation of Customer Count in Clusters (Silhouette Method)

```
kmeans = KMeans(n_clusters = 2, init = 'k-means++', random_state = 42)
y_kmeans = kmeans.fit_predict(X)
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 10, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 10, c = 'blue', label = 'Cluster 2')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 30, c = 'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Income')
plt.ylabel('DebtIncome Ratio')
plt.legend()
plt.show()
```

Fig 5.5.2.8 Code for Cluster's representation using Silhouette Score Method

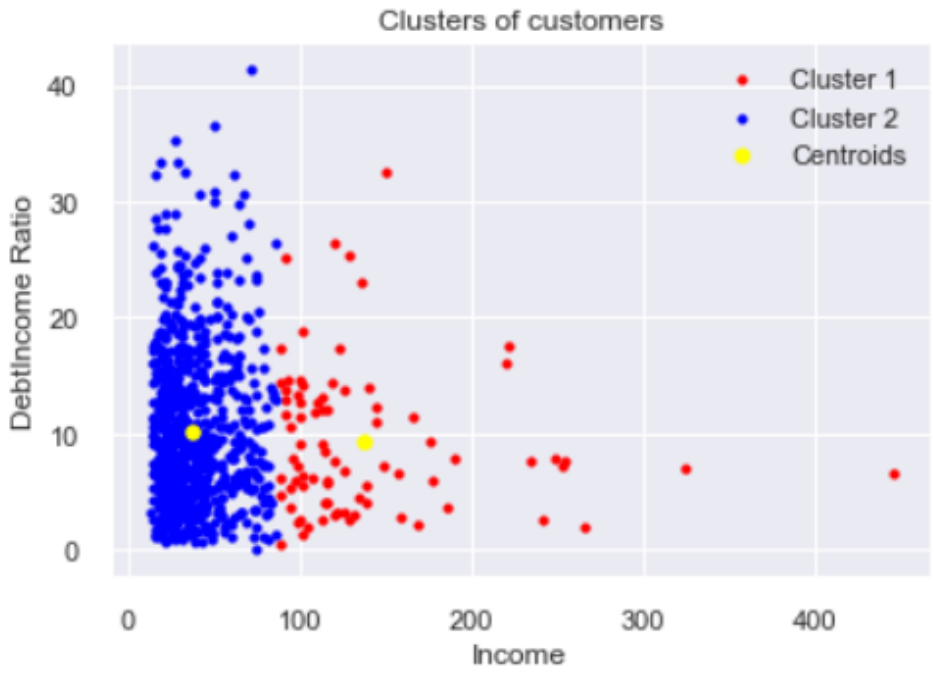


Fig 5.5.2.9 Cluster's representation using Silhouette Score Method

5.5.3 Dendrogram

This technique is specific to the agglomerative hierarchical method of clustering. The agglomerative hierarchical method of clustering starts by considering each point as a separate cluster and starts joining points to clusters in a hierarchical fashion based on their distances. In a separate blog, we will focus on the details of this method. To get the optimal number of clusters for hierarchical clustering, we make use a dendrogram which is tree-like chart that shows the sequences of merges or splits of clusters.

If two clusters are merged, the dendrogram will join them in a graph and the height of the join will be the distance between those clusters. We will plot the graph using the dendogram function from scipy library.

```
dend=sch.dendrogram(sch.linkage(X, method='ward'))
plt.title("Dendrogram")
plt.xlabel('Customer')
plt.ylabel('euclidean')
plt.show()
```

Fig 5.5.3.1 Code for Dendrogram Clustering

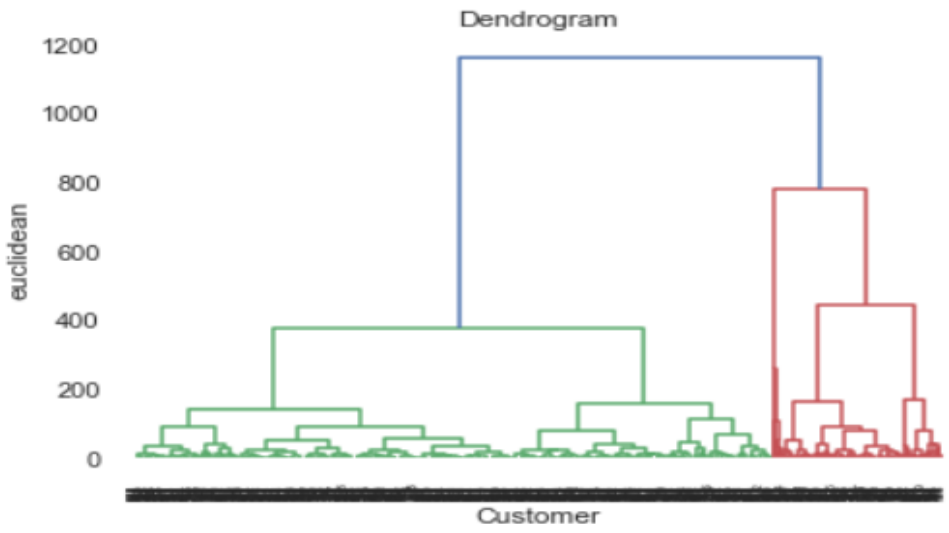


Fig 5.5.3.2 Dendrogram Clustering

Here the optimal number of clusters is 2.

Visualize the clusters

```
plt.bar(height = np.bincount(y_hc),x = ['1','2'])
plt.xlabel('Cluster', fontsize=13)
plt.ylabel('Number of Customer', fontsize=13)
```

Fig 5.5.3.3 Code for Representation of Count in Clusters by Dendrogram Method

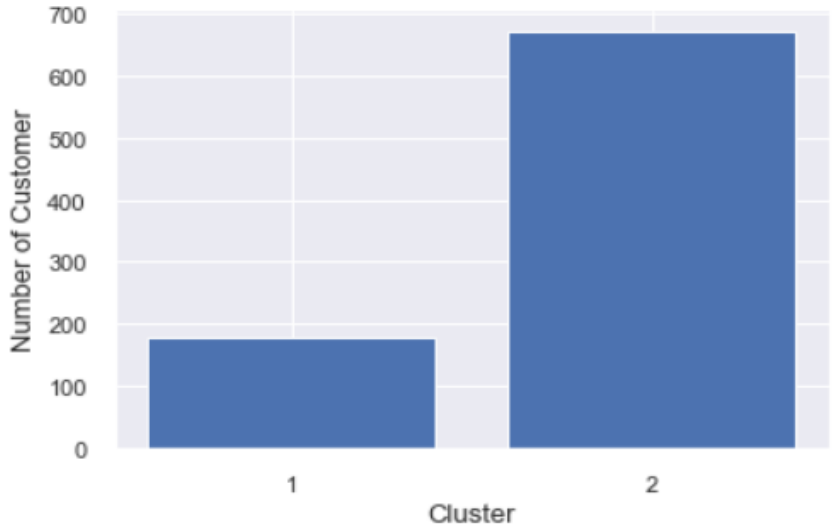


Fig 5.5.3.4 Bar Representation of Customers in Clusters by Dendrogram Method

```
plt.scatter(X[y_hc == 0, 0], X[y_hc == 0, 1], s = 10, c = 'blue', label = 'Cluster 1')
plt.scatter(X[y_hc == 1, 0], X[y_hc == 1, 1], s = 10, c = 'red', label = 'Cluster 2')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 30, c = 'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Income')
plt.ylabel('DebtIncome Ratio')
plt.legend()
plt.show()
```

Fig 5.5.3.5 Code for Cluster's representation using Dendrogram Method

Fig 5.5.3.6 Cluster's representation using Dendrogram Method

5.5.4 Calinski Harabasz Score Method

The Calinski-Harabasz Index is based on the idea that clusters that are (1) themselves very compact and (2) well-spaced from each other are good clusters. The index is calculated by dividing the variance of the sums of squares of the distances of individual objects to their cluster centre by the sum of squares of the distance between the cluster centres. Higher the Calinski-Harabasz Index value, better the clustering model. The formula for Calinski-Harabasz Index is defined as:

$$CH_k = \frac{BCSM}{k-1} \cdot \frac{n-k}{WCSM}$$

Fig 5.5.4.1 Formula for Calculating Calinski-Harabasz Coefficient

where k is the number of clusters, n is the number of records in data, BCSM (between cluster scatter matrix) calculates separation between clusters and WCSM (within cluster scatter matrix) calculates compactness within clusters.

KElbowVisualizer function is able to calculate Calinski-Harabasz Index as well:

```
model = KMeans()
visualizer = KElbowVisualizer(model, k=(2,30),metric='calinski_harabasz', timings= True)
z=visualizer.fit_predict(X)
visualizer.fit(X)        # Fit the data to the visualizer
visualizer.show()        # Finalize and render the figure
```

Fig 5.5.4.2 Code for Visualization of Calinski-Harabasz Scores up to 30 Clusters

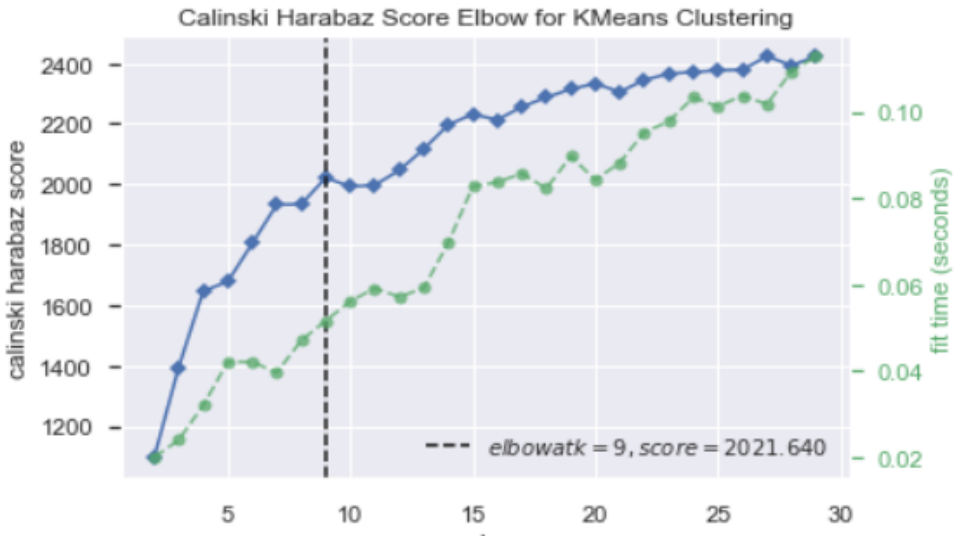


Fig 5.5.4.3 Visualization of Calinski-Harabasz Scores up to 30 Clusters

Here the optimal number of clusters is 8.

Visualize the clusters

```
plt.bar(height = ch.value_counts(),x = ['1','2','3','4','5','6','7','8'])
plt.xlabel('Cluster', fontsize=13)
plt.ylabel('Number of Customer', fontsize=13)
```

Fig 5.5.4.4 Code for bar Representation of Count by Calinski-Harabasz Method

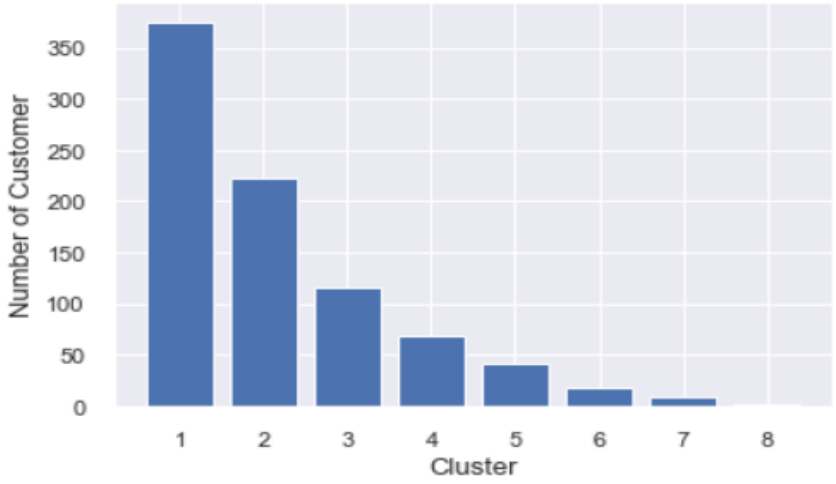


Fig 5.5.4.5 Bar Representation of Customer Count by Calinski-Harabasz Method

```
#kmeans = KMeans(n_clusters = 8, init = 'k-means++', random_state = 42)
plt.scatter(X[ch == 0, 0], X[ch == 0, 1], s = 10, c = 'blue', label = 'Cluster 1')
plt.scatter(X[ch == 1, 0], X[ch == 1, 1], s = 10, c = 'red', label = 'Cluster 2')
plt.scatter(X[ch == 2, 0], X[ch == 2, 1], s = 10, c='green', label = 'Cluster3')
plt.scatter(X[ch == 3, 0], X[ch == 3, 1], s = 10, c = 'cyan', label = 'Cluster 4')
plt.scatter(X[ch == 4, 0], X[ch == 4, 1], s = 10, c = 'magenta', label = 'Cluster 5')
plt.scatter(X[ch == 5, 0], X[ch == 5, 1], s = 10, c = 'orange', label = 'Cluster 6')
plt.scatter(X[ch == 6, 0], X[ch == 6, 1], s = 10, c = 'black', label = 'Cluster 7')
plt.scatter(X[ch == 7, 0], X[ch == 7, 1], s = 10, c = 'brown', label = 'Cluster 8')
plt.title('Clusters of customers')
plt.xlabel('Income')
plt.ylabel('DebtIncome Ratio')
plt.legend()
plt.show()
```

Fig 5.5.4.6 Code for Cluster's representation using Calinski-Harabasz Method

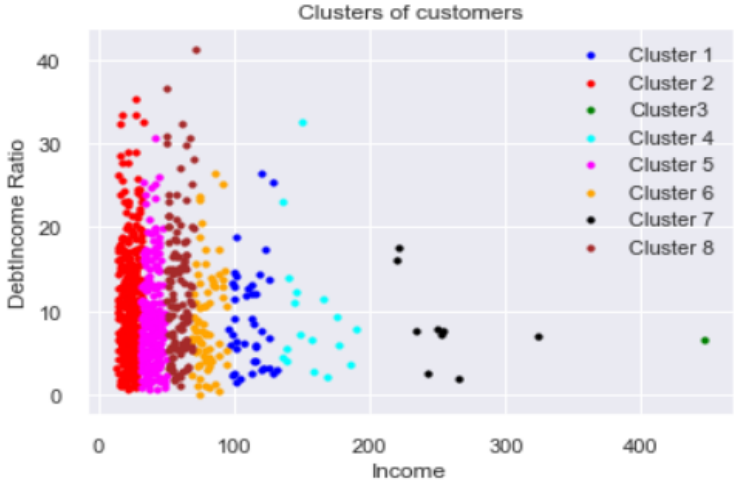


Fig 5.5.4.7 Cluster's representation using Calinski-Harabasz Method

CONCLUSION

In competitive market of e-commerce, the problem of identifying potential customer is gaining more and more attention. To address this problem timely, this paper proposes a study on comparison between various methods of finding the number of clusters in K-means clustering such as elbow method, average silhouette method, dendrogram etc. According to the dataset we used, we found that average silhouette method and dendrogram method are better for finding the appropriate number of clusters. After identification of targeted customers and their associative buying pattern, the business managers take the strategic profitable decisions accordingly. This integrated model could be directly brought into implementation for providing better profitable margins from sales.

REFERENCES

https://www.kaggle.com/somesh24/customer-segmentation

https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec

https://stanford.edu/~cpiech/cs221/handouts/kmeans.html

https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c

https://scikit-learn.org/stable/modules/clustering.html

https://pandas.pydata.org/docs/

https://numpy.org/doc/

https://matplotlib.org/tutorials/introductory/pyplot.html

https://www.scikit-yb.org/en/latest/api/cluster/silhouette.html

https://seaborn.pydata.org/tutorial.html