CUSTOMER SEGEMENTATION USING CLUSTERING ANALYSIS

The Project

submitted in partial fulfillment for the award of degree of

MASTER OF COMPUTER APPLICATIONS

Submitted by

Mr. PASAGADUGULA VEERESH

(Regd.No: 20L31F0060)

Batch 2020-2022

Under the esteemed guidance of

Ms. K.G.PRASANTHI

Assistant Professor

Department of MCA

VIGNAN's INSTITUTE OF INFORMATION TECHNOLOGY
(AUTONOMOUS)
(Approved by AICTE - New Delhi & Affiliated to JNTUK, Kakinada)
Beside VSEZ, Duvvada, Vadlapudi Post, Gajuwaka, Visakhapatnam - 530 049.

VIGNAN's INSTITUTE OF INFORMATION TECHNOLOGY
(AUTONOMOUS)
(Approved by AICTE - New Delhi & Affiliated to JNTUK, Kakinada)
Beside VSEZ, Duvvada, Vadlapudi Post, Gajuwaka, Visakhapatnam - 530 049.

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project work entitled **"Customer Segmentation Using Clustering Analysis"** is a bonafide record of project work carried out under my supervision by **P.Veeresh** bearing Regd.No: **20L31F0060** in partial fulfillment of the degree of **Master of Computer Applications** of Vignan's Institute of Information Technology(A) affiliated to Jawaharlal Nehru Technology University, Kakinada, during the academic year 2020-2022.

Project guide
Ms. K.G.Prasanthi
Department of MCA

Head of Department
Dr. G.Rajendra Kumar
Department of MCA

EXTERNAL EXAMINER

DECLARATION

I hereby declare that this project report entitled **"Customer Segmentation Using Clustering Analysis"** has undertaken by me for the fulfillment of **Master of Computer Applications**. I completed this project work under the guidance of **Ms. K.G.Prasanthi,** Assistant Professor, Department of MCA. I declare that this project report has not been submitted anywhere in the part of fulfillment for any degree of any other University.

Place: Visakhapatnam **P.Veeresh**

Date: **(20L31F0060)**

ACKNOWLEDGEMENT

An endeavor over a long period can be successfully with the advice and support of many well-wishers. I take this opportunity to express our gratitude and appreciation to all of them.

I express my sincere gratitude to my internal guide **Ms. K.G.Prasanthi,** for encouragement and cooperation in completion of my project. I am very fortunate in getting the generous help and constant encouragement from him/her.

I would be very grateful to our project coordinator **Mrs. A.Sirisha,** for the continuous monitoring of my project work. I truly appreciate for her time and effort spent.

I would like to thank our Head of the Department **Dr. G.Rajendra Kumar,** and all other teaching and non-teaching staff of the department for their cooperation and guidance during my project.

I sincerely thank to **Dr. B.Arundhati,** Principal of VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY (A) for her inspiration to undergo this project.

I wanted to convey my sincere gratitude to **Dr. V. Madhusudhan Rao**, Rector of VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY (A) for allocating the required resources and for the knowledge sharing during my project work.

I extended my grateful thanks to our honorable Chairman **Dr. L. Rathaiah,** for giving me an opportunity to study in his esteemed institution.

P. Veeresh

(20L31F0060)

VIGNAN's INSTITUTE OF INFORMATION TECHNOLOGY
(AUTONOMOUS)
(Approved by AICTE - New Delhi & Affiliated to JNTUK, Kakinada)
Beside VSEZ, Duvvada, Vadlapudi Post, Gajuwaka, Visakhapatnam - 530 049.

MASTER OF COMPUTER APPLICATIONS

VISION

- We aim to generate groomed, technical competent and skilled intellectual professionals.
- We serve as a valuable resource for modern industry and current society.

MISSION

- Providing strong theoretical and practical knowledge in computer science discipline with an emphasis on software development.
- To provide need-based quality training in the field of information technology.
- Impart quality education to meet global standards and achieve excellence in teaching-learning and research.
- To provide students with the tools to become productive, participating global citizens and life-long learners.

VIGNAN's INSTITUTE OF INFORMATION TECHNOLOGY
(AUTONOMOUS)
(Approved by AICTE - New Delhi & Affiliated to JNTUK, Kakinada)
Beside VSEZ, Duvvada, Vadlapudi Post, Gajuwaka, Visakhapatnam - 530 049.

MASTER OF COMPUTER APPLICATIONS

PROGRAMME OUTCOMES

At the end of the programme the student shall be able to

- Application of Engineering Knowledge
- problem analysis
- Design development of solutions
- conduct investigation of complex problems
- Modern tool usage
- The engineer and society
- Environment and sustainability
- Ethics
- Individual team work
- communication
- Project management and finance
- Lifelong learning

CUSTOMER SEGEMENTATION USING CLUSTERING ANALYSIS

ABSTRACT

It is essential to analyze the huge amounts of data that our environment often produces. In the present technological world, where everyone is competing to the corporate strategy is supposed to be superior to everyone else adapted to the circumstances by hand. By Considering all the potential customers today struggled with knowing what to buy and what to avoid. New ideas are very essential to business companies are unable to assess their primary customers by themselves. Where the Machine Learning occurs, and different techniques are used to discovered the hidden patterns in data to help in decision-making. The machine learning method called clustering involves comparing data from several groups such as market research, image processing, pattern recognition, search engines optimization and medical data processing and others.

Customer market research involves segmentation, which is the topic of our endeavor. The definition of customer segmentation is the grouping of consumers based on their similar characteristics. In the present environment, it's essential for businesses should categorize their customers according to their age, location, gender, and additional characteristics. This enables businesses should concentrate on particular customers who are most likely to buy their products. The use of machine learning comes to provide an advantage over their business competitors. We can successfully to improve their business strategies, the primary intention of this project using the K-means algorithm to divide customers groups based on their attributes. Finally, by taking the mean value as be the major indication, the data from the various clusters inform us which groups the new customers belongs to.

INDEX

LIST OF FIGURES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

CHAPTER - 1
INTRODUCTION

1.1 Brief Information about the project:

Customer segmentation is the process of dividing individuals who have characteristics relevant to marketing, such as age, gender, interests, and spending habits, into groups. Customer segmentation is a method used by companies to target particular, smaller groups of consumers with relevant messages that would encourage them to make a purchase. This technique is based on the concept that each and every customer is unique. In order to more effectively focus their marketing efforts to each segment, business also want to have a deeper understanding of their customers' preferences and needs.

In order to divide customers into specific targeting groups, it is necessary to identify significant differentiators that separate them. When determining customer segmentation techniques, factors also with a customer's demographics (age, race, religion, gender, family size, ethnicity, income, and level of education), geography (where they live and work), psychographics (social class, lifestyle, and personal traits), and affective (spending, consumption, usage, and desired benefits) tendencies are taken into account. The ability to adjust marketing strategies so that they are suitable for each customer category and to support business goals is called customer segmentation. It helps in identifying the items related to each client segment, managing supply and demand for those products, identifying and focused on a potential customer base, and predicting customer problems.

By target specific consumer groups with a customer segmentation strategy, business owners might use its marketing resources more efficiently and increase their chances of cross-selling. When companies give customized messages to a set of clients as part of a marketing mix suited to their needs, it is simpler for them to identify innovative offers to motivate them to spend more.

1.2 Motivation of project:

Segmentation is not a label concept. In 1956, Wendel Smith published the first article on segmentation. In fact, segmentation is an important element of many marketing techniques. A more diverse population's homogeneous groups might be developed, which

significantly facilitated the manufacture and marketing of products and services. These divisions were usually classified based on age, gender, income, ethnicity, or other factors. Marketing teams quickly adopted behavioral segmentation.

1.3 Objective of the project:

The process of segmenting a company's customers into groups that reflect similarity among customers in each group is called as customer segmentation. In order to enhance each customer's value to the company, it is crucial to choose how to connect with each type of customer. Marketing professionals may be able to contact each customer in the most effective method with the help of customer segmentation. A customer segmentation analysis enables marketers to identify different groups of customers with a high level of accuracy based on demographic, behavioral, and other factors using the enormous amount of data on customers (and potential customers) that is available.

1.4 Organization of the project:

1.4.1 Literature Review: This chapter contains an introduction, a comparison, the project's origins, and possible alternatives.

1.4.2 System Analysis: The description of the existing system, the proposed system, and the requirements specifications comprise the majority of this chapter.

1.4.3 System Design: This chapter includes an overview of the modules and their algorithms as well as example use scenarios with class, sequence, collaboration, and activity diagrams.

1.4.4 Technology Description: The majority of this chapter is focused to describing the project's technical foundation.

1.4.5 Sample Code: The sample code in this chapter is for a few modules.

1.4.6 Testing: The majority of this chapter's attention is on testing procedures and module test cases.

1.4.7 Screenshots: The most of this chapter's output screens are from the project.

1.4.8 Conclusion: The project's main finding is that customers must be separated into groups based on characteristics.

CHAPTER 2

LITERATURE SURVEY

CHAPTER – 2
LITERATURE SURVEY

2.1 Customer segmentation in services based on characteristics:

Existing businesses must embrace marketing tactics to stay competitive as new enterprises open their doors every day. In today's society, the key marketing rule is "change or perish." Businesses are finding it more difficult to meet the demands of each and every one of their consumers as the number of customers grows [1]. In this case, data mining can assist in identifying hidden tendencies in a company's database. Client segmentation is a data mining approach that splits a customer base into multiple groups based on characteristics such as gender, age, hobbies, and other buying habits [2]. Customers are split into groups based on shared qualities, in other words.

Segmentation can influence marketing strategy directly or indirectly because it opens up many new paths to discover, such as which segment the product will be good for, customizing marketing plans for each segment, providing discounts for a specific segment, and deciphering the customer and object relationship, which was previously unknown to the company [3]. A customer segmentation strategy allows firms to target particular groups of consumers, resulting in more efficient marketing resource allocation and greater potential for cross and up-selling. It's easier for firms to create unique offers to entice customers to spend more when they deliver customized communications to a group of customers as part of a marketing mix tailored to their requirements.

Consumer segmentation may help with customer loyalty and retention by improving customer service. Because of their individualized character, marketing materials that employ customer segmentation are more valued and appreciated by the consumer who gets them than impersonal brand communications that ignore purchase history or any type of customer relationship [4]. Customer segmentation has been demonstrated to benefit from clustering. Clustering is a sort of unsupervised learning that allows us to locate clusters in unlabeled datasets. Clustering techniques include K-means, hierarchical clustering, and others [5]. The major purpose of this work is to apply a data mining strategy to find consumer groups using the K-means clustering algorithm to partition data. The silhouette method yields the most clusters.

2.2 Company Needs to Maintain the Requirement's for Customers:

Segmentation, according to Sandstrom (2003), is based on the idea that customers who use a company's goods and services are not equally valuable. Customers hold different levels of significance for business, and in order to be competitive, they must shift their attention from reduced consumers to high-profit consumers in an uneven approach. A company needs to pay close attention to the customers who use its services or products more frequently or in larger quantities in order to create efficient customer groups (Sandstrom, 2003).

A corporation will be able to more easily segment its client base into groups when it has the necessary knowledge of what its customers need. Additionally, the business may more easily discover what delights and even surprises its clients. This information can be utilized to enhance their services or goods in the future. Customer service is becoming just as important to consumers as the actual product or service, thus it's crucial that businesses have this component in place. Finally, consumer segmentation can make deciding how much and what the business should emphasize when it comes to the level of services that the various groups should receive simpler (Buttle, 2009).

Market segmentation is crucial in both emerging production industries and service industries, according to Lambert (1990). To best meet client demand and raise revenue for the business, all types of businesses must discover a system that divides the market into appropriate categories. When an organization begins the process of segmenting, it is advised to see the customers from a need-based perspective, with the customer with the highest demand being prioritized.

2.3 Joint Optimization of Customer Segmentation and Marketing Policy:

A business offers a service or a product to the market. As a result, it is crucial for a business to achieve the customer service components in order to satisfy its clients. Established service providers have the necessary expertise and understanding to satisfy their clients' requests, expectations, and needs (Mattson, 2004). Customer service can be defined as the actions taken by a business to include buyers, sellers, and other organizations that can promote its good or service. Successful customer segmentation in the services sector benefits the business by improving relationships with buyers and sellers, which also boosts

competitiveness (Pauline, 2009). There are many different types of service-providing businesses with a wide variety of clients out there in the market. Service businesses use a variety of tactics to target and segment their clients in order to be competitive and best satisfy customer demand.

Businesses that want to know what their customers expect from them both now and in the future may consider identifying client groups and tracking how they change over time. This is crucial for companies that operate in dynamic markets with consumers whose needs and attitudes are always changing due to new technologies and competing products. Customers are often segmented by using some type of cluster analysis to create a list of segments to which new customers will be assigned in the future.

CHAPTER 3
SYSTEM ANALYSIS

CHAPTER - 3

SYSTEM ANALYSIS

3.1 Existing System:

Customer data is currently mainly stored through documentation and computer software, which is growing daily. At the end of the day, they will examine their data to determine how many products were sold, the number of real customers, etc. They identified who was beneficial to their business and improved their sales by analyzing data that was collected. More paperwork and time are needed. Moreover, it is not a very effective method of locating the desired customer data.

Disadvantages:

- Slow in processing customer's information
- Data inconsistency/ Redundancy
- Inadequate of accuracy in customer's records
- It is inefficient and time-consuming

3.2 Proposed System:

This fresh approach will be essential in conquering the existing process, that focuses on paperwork and computer digital data. Everyday data collection involves a growing quantity of documentation, which takes a lot of time. In today's world, new technologies were emerging. Machine learning is a powerful innovation that employs a variety of algorithms to predict the outcome. Consequently, to answer our problem statement, we'll use K-Means Clustering, which classifies the data into groups based on similar characteristics. The data will subsequently be presented.

Advantages:

- Improve Product Development
- Improve services
- Better Understandable

3.3 Feasibility Study:

The feasibility of the project and the possibility that the system will benefit the organization are both examined in the preliminary investigation. The major goal of the feasibility study is to determine if it would be technically, operationally, and economically feasible to add new modules and fix existing systems. If there are infinite resources and time, any system is feasible. Aspects in the feasibility research portion of the initial examination include the following:

3.3.1 ECONOMIC FEASIBILITY

3.3.2 OPERATIONAL FEASIBILITY

3.3.3 TECHNICAL FEASIBILITY

3.3.1 Economic Feasibility:

This study is being done to see what kind of financial impact the system will have on the company. The corporation has a finite amount of money to invest in the system's research and development. The costs must be supported by evidence. As a result, the developed system came in under budget, which was made possible by the fact that most of the technologies were free to use. Only the specialized goods needed to be bought.

3.3.2 Operational Feasibility:

Only if a project can be transformed into an information system are they valuable. This will satisfy the organization's operational needs. Project execution must consider the operational feasibility considerations as a key component. To test a project's operational viability, some crucial problems are raised, such as the following:

- Is there sufficient support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible applications?

The design of this system is intended to address the aforementioned problems. Prior to beginning, management concerns and user requirements were taken into account. Therefore,

there is no concern about user opposition undermining the potential application benefits. The best possible use of the computer resources would be made possible by the well-planned architecture, which would also contribute to an improvement in performance.

3.3.3 Technical Feasibility:

This study is being done to evaluate the system's technical requirements, or technical feasibility. Any system created must not place a heavy burden on the technical resources at hand. The number of technological resources available will be heavily strained as a result. As a result, the client will face high expectations. The created system must have a low demand because its implementation merely necessitates little or no adjustments.

CHAPTER 4
SYSTEM ARCHITECTURE

CHAPTER - 4

SYSTEM SPECIFICATIONS

4.1 Functional Requirements:

The functional needs of a software framework or a component thereof are described by an element or function in software engineering. An arrangement of information sources, inputs, outputs, conduct, and yields is referred to as a function. These functional requirements could include counts, specific places of interest, the control and handling of data and information, and other stated usefulness that defines what a system should accomplish. Every instance in which the framework makes use of the functional requirements is represented by behavioral requirements.

- Divide them based on Characteristics
- Collect customers data from the Kaggle
- Train the customer's data

4.2 Non-Functional Requirements:

EFFICIENCY:

It is a measurable concept and can often be expressed as a percentage of result that could ideally be expected**.**

MAINTAINABILITY:

The ability of a developed system to update itself over time, rectify flaws and issues that surfaced after distribution, and adapt to changing user needs is known as maintainability. Since the programme was created using Python, it is simple to identify and fix any errors depending on user requirements by simply adding the necessary files or APIs to the already installed programme.

SCALABILITY:

This will provide information on how the system responds or how the system provides its throughput when the input data load is altered. Any quantity of manually entered data will not affect how the system functions.

PORTABILITY:

One of the key characteristics of non-functional requirements is portability, which will indicate whether the software needs to be completely rewritten when it is transferred from one device to another. Python is utilized in the project, making it simple to install and use on any other platform.

4.3 Requirements Specification

4.3.1 Hardware Requirements:

Processor	:	Intel Core i5
Speed	:	2.2 GHz
RAM	:	8 GB
Hard Disk	:	500 GB

4.3.2 Software Requirements:

Operating System	:	Windows 11
Technology	:	Python 3.9
IDE	:	Jupyter Notebook

CHAPTER 5
SYSTEM DESIGN

CHAPTER - 5
SYSTEM DESIGN

Introduction:

The design phase's goal is to conceptualize a solution to the issue that the requirement document has identified. The initial stage in transitioning from the problem domain to the solution domain is this phase. In other words, design guides us toward how to meet needs by beginning with what is needed. The quality of the programme is most likely affected by the system's architecture, which also has a significant impact on later phases, particularly testing and maintenance. This document, which serves as the solution's blueprint, is utilized for implementation, testing, and maintenance.

System Design and Detailed Design are two phases that are frequently separated to complete the design process. The goal of system design, also known as top-level design, is to specify each module that should be included in the system as well as how those modules work together to create the intended outcomes. At the conclusion of the system design, all significant data structures, file formats, output formats, and significant system modules are selected, along with their requirements. The internal logic of each module defined in the system architecture is chosen during detailed design. In this phase, a high-level design description language is typically used to specify the specifics of a module's data. It is independent of the language that will ultimately be used to create the product. The identification of the modules is the main focus of system design, whereas the logic design for each module is the main focus of detailed design.

In other works, system design focuses on the necessary components, whereas detailed design focuses on how the components can be implemented in software. Identification of software components and the stipulation of their interactions are concerns of design. creating a plan for the document phase and specifying the software structure. One of the desirable characteristics of big systems is modularity. It suggests that the system is split up into various components. Developers fill the gap between the requirements specification created during requirements elicitation and analysis and the system that is supplied to the user during the system design activities. Create the environment that promotes a quality in development. The process of translating requirements into a software representation is called software design.

5.1 System Architecture:

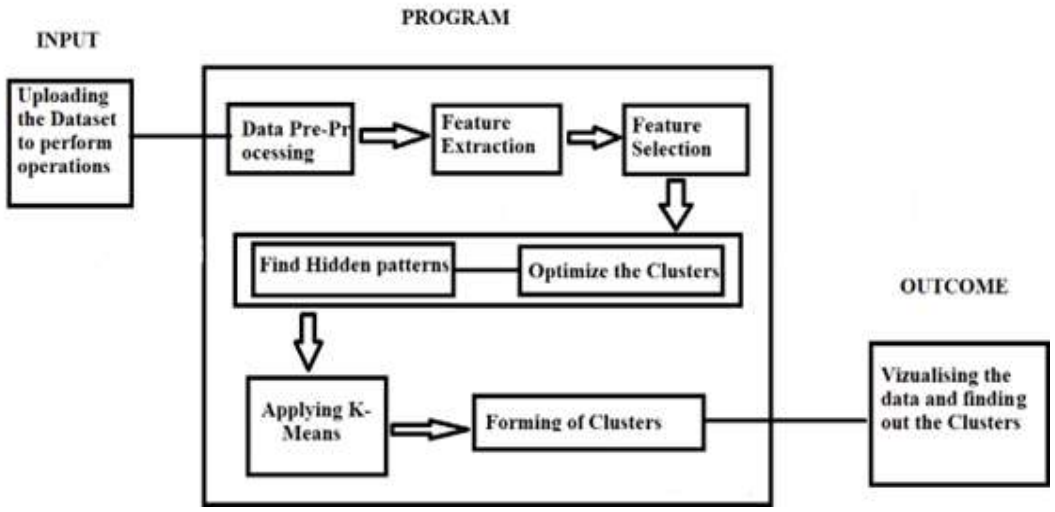


Fig 5.1: System Architecture

we will analyze the data through data visualization. Finally, we will get the outcome.

Architecture Description:

In order to scope with missing data, duplicate values, and null values, we first examine the dataset. Next, we conduct exploratory data analysis. Finally, we implement the k-means clustering algorithm, which is an unsupervised learning technique in machine learning.

In order to locate potential clients, we must have access to the data of that specific company, and we must perform some operations using their data sets. We will first view the dataset before performing exploratory data analysis. it addresses the null values, duplicate values, and missing data. Then we will use our machine learning technique, k-means clustering, which is an example of unsupervised learning.

Utilizing various techniques, we will choose the best clusters, given values as K, and discover the clusters in the data's hidden patterns. We'll use data visualization to analyze the data. We will eventually receive the results.

5.2 Data Flow Diagram:

Data flow diagram called as "bubble chart" it is another name for a data flow diagram. It is a pictorial or graphical representation that can be used to depict the data that is input into a system, the various operations that are performed on the data, and the output that the system.

A graphical tool used to describe and analyze the knowledge gained instantly through a manual or automatic system, together with the process, knowledge stores, and delays inside the system. The process by which knowledge is transformed from input to output is also logically and specifically defined in terms of the system's physical components. The DFD is often referred to as a bubble chart or a data flow graph. The fundamental notation is

DFD Level 0 :

It is also called a Context Diagram.

It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities.

DFD Level 1 :

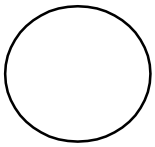
It provides a more detailed breakout of pieces of the Context Level Diagram. You will highlight the main functions carried out by the system, as you break down the high-level process of Context Diagram into its subprocesses.

DFD Level 2 :

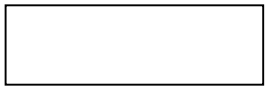
In this goes one step deeper into parts of Level1.It may require more text necessary level detail about the system's functioning but going beyond Level 3 is uncommon. Doing so can create complexity that makes it difficult to communicate, compare or model effectively. The flow diagrams we certainly used to represent the data flow diagrams and in this only we can How to draw the data flow diagrams in various activities like class, sequential etc. and we can show them below those are the ones who certainly used to draw the flow diagrams and used often in the UML diagrams and connections. The main notations we widely used for the Data flow diagrams are mentioned below and these are and various elements can be drawn by using of those.

- **Data flow**

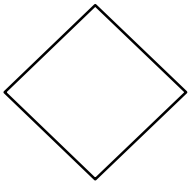
- **Process:**



- **Source:**



- **Rhombus**:



- **Data store:**

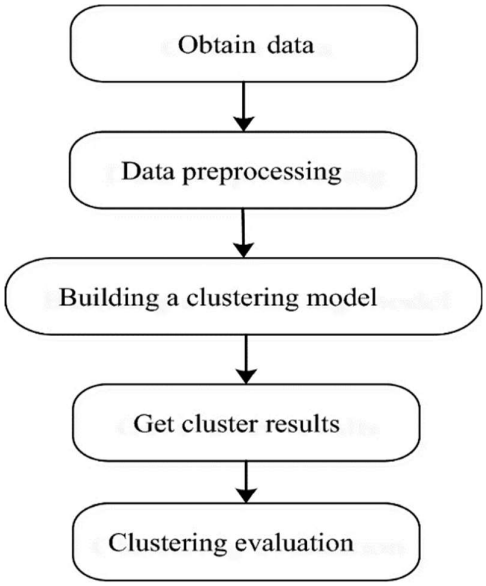


Fig 5.2: Data Flow Diagram

The above data flow diagram demonstrates the evaluation and data collection using the dataset that the user has provided. After that, the data is pre-processed depending on the user's provided data set and is then built into a clustering model using supervised and unsupervised algorithms. We employ an unsupervised method for clustering, such as K-means, Agglomerative, etc., which produces clusters as a result. The clusters will then be visualized as the outcome of the evaluation of the cluster.

K-Means was the first clustering operation carried out in the dataflow. K-Means was run using k's from 1 to 25 to provide a sample range for enough clustering to occur, as K-Means requires a pre-determined k to cluster. In connection with this, the top 51 clustering were chosen from each of 100 clustering iterations using randomly initialized centroids.

5.3 UML Diagrams:

The software engineer can use the modelling notation that is governed by a set of syntactic, semantic, and pragmatic norms to express an analytical model using the unified modelling language. A UML system is represented using five separate views, each of which offers a radically different viewpoint on the system. In particular, UML is built using two distinct domains. The system's user model and structural model views are the emphasis of UML analysis modelling. The behavioral modelling, implementation modelling, and environmental model views are the focus of UML design modelling. The following types are used to categorize them.

5.3.1 Use case diagram

5.3.2 Class diagram

5.3.3 Sequence diagram

5.3.4 Collaboration diagram

5.3.5 Activity diagram

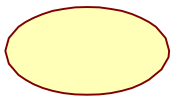
5.3.1 Use Case Diagram:

Use case diagrams show the system's functionality (utilize cases), the users who use it (actors), and the relationship between the users and the functionality. Software development's analysis phase use cases to express the system's high-level needs. Use Case Diagrams' main objectives are as follows:

Graphical Notation: The basic components of Use Case diagrams are the

Actor: An Actor, as mentioned, is a user of the system, and is depicted using a stick figure. The role of the user is written beneath the icon. Actors are not limited to humans. If a system communicates with another application, and expects input or delivers output, then that application can also be considered an actor.

Use Case: A Use Case is functionality provided by the system; Use Cases are depicted with anellipse. The name of the use case is written within the ellipse



Association: These Associations are used to link Actors with Use Cases, and indicate that anActor participates in the Use Case in some form.

The below Figure represents the use case diagram for the project at various cases and having the association with use cases and how the process will get starts it shows like as data uploading, data processing, Feature Extraction, Feature Selection an getting optimal Clusters Finally we willVisualize the clusters using any plot methods and having the use case templates also shown:

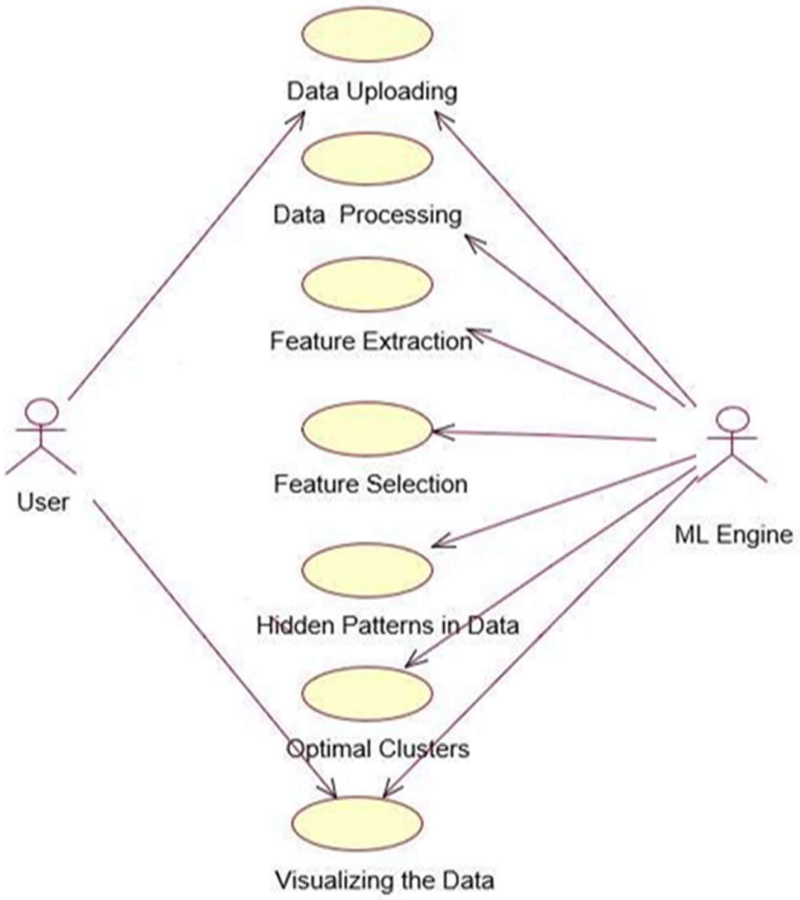


Fig 5.3.1: Use case diagram for System

Use case Templates:

Use case name	Data Uploading
Participating actors	User
Flow of events	User should login to the system. User can browse and upload the dataset.
Entry condition	It searches for the data where it is located and load.
Exit condition	Dataset is successfully uploaded.

Table 5.3.1.1: Use case template for Data uploading

Use case name	Data processing
Participating actors	User
Flow of events	It is a subpart in data preprocessing. Under this some operations are applied to handle unnecessary data like duplicates and missing values.
Entry condition	The file consists of features.
Exit condition	Noticing how much percentage of values are missing in all features.

Table 5.3.1.2: Use case template for Data processing

Use case name	Feature selection
Participating actors	User, ML engine
Flow of events	It is process of reduces the number of input values in the model.
Entry condition	The numerical input has been taken from user.
Exit condition	It will give successful results.

Table 5.3.1.3: Use case template for Feature selection

Use case name	Feature extraction
Participating actors	User, ML engine
Flow of events	Describes large set of data where it reduces number of resources while performing.
Entry condition	We will extract the features from the dataset.
Exit condition	It will give successful results .

Table 5.3.1.4: Use case template for Feature extraction

Use case name	Optimal Clusters
Participating actors	User, ML engine
Flow of events	As there should be finding out the optimal clusters to visualize the data and it's necessary to find out the clusters
Entry condition	We will find the optimal clusters based on pattern in data
Exit condition	It will find the clusters

Table 5.3.1.5: Use case template for Optimal cluster

5.3.2 Class Diagram :

The basic edifice of object-oriented modelling is the class diagram. It is used for both precise modelling that converts the models into programming code and for general conceptual modelling of the structure of the application. Data modelling can also employ class diagrams. The major objects, interactions, and classes that need to be programmed are all represented by the classes in a class diagram. A class with three sections is depicted in the diagram by a box that has three parts:

Graphical Notation: The elements on a Class diagram are classes and the relationships between them.

Class: Classes are the building blocks in object-oriented programming. A Class is depicted using a rectangle divided into three sections. The top section is the name of the Class. The middle section defines the properties of the class. The bottom section lists the methods of the class.

Association: An Association is a generic relationship between two classes, and is modeled by a line connecting the two classes. This line can be qualified with the type of relationship, and can also feature multiplicity rule (e.g., one-to- one, one-to-many, many-to-many) for the relationship.

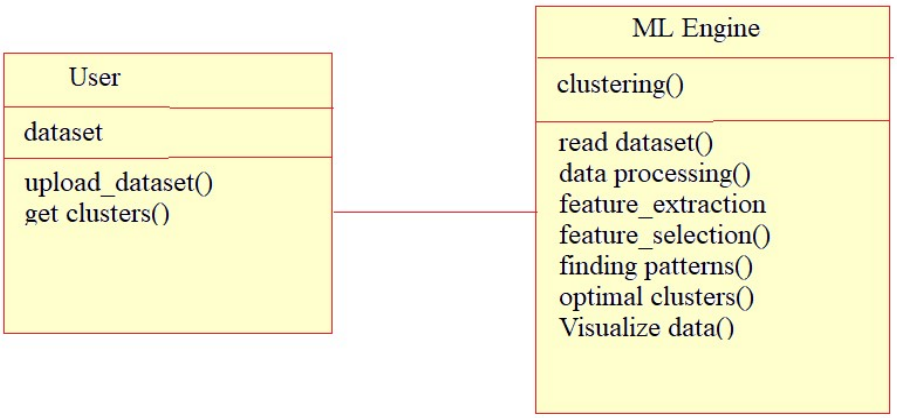


Fig 5.3.2 Class Diagram for system

5.3.3 Sequence Diagram:

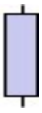
Sequence diagrams show the interactions that take place between classes to accomplish a goal, such a use case. These interactions between classes are known as messages since UML is made for object-oriented programming. The Sequence diagram models the evolution of these communications through time by listing objects horizontally and time vertically.

Graphical Notation: In a Sequence diagram, classes and actors are listed as columns, with vertical lifelines indicating the lifetime of the object over time.

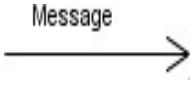
Object: Objects are instances of classes, and are arranged horizontally. The pictorial representation for an Object is a class (a rectangle) with the name prefixed by the object name (optional) and a semi-colon.

Lifeline: The Lifeline identifies the existence of the object over time. The notation 2for a Lifeline is a vertical dotted line extending from an object.

Activation: Activations, modelled as rectangular boxes onthe lifeline, indicate when the object is performing an action.



Message: Messages, modeled as horizontal arrows betweenActivations, indicate the communications between objects.



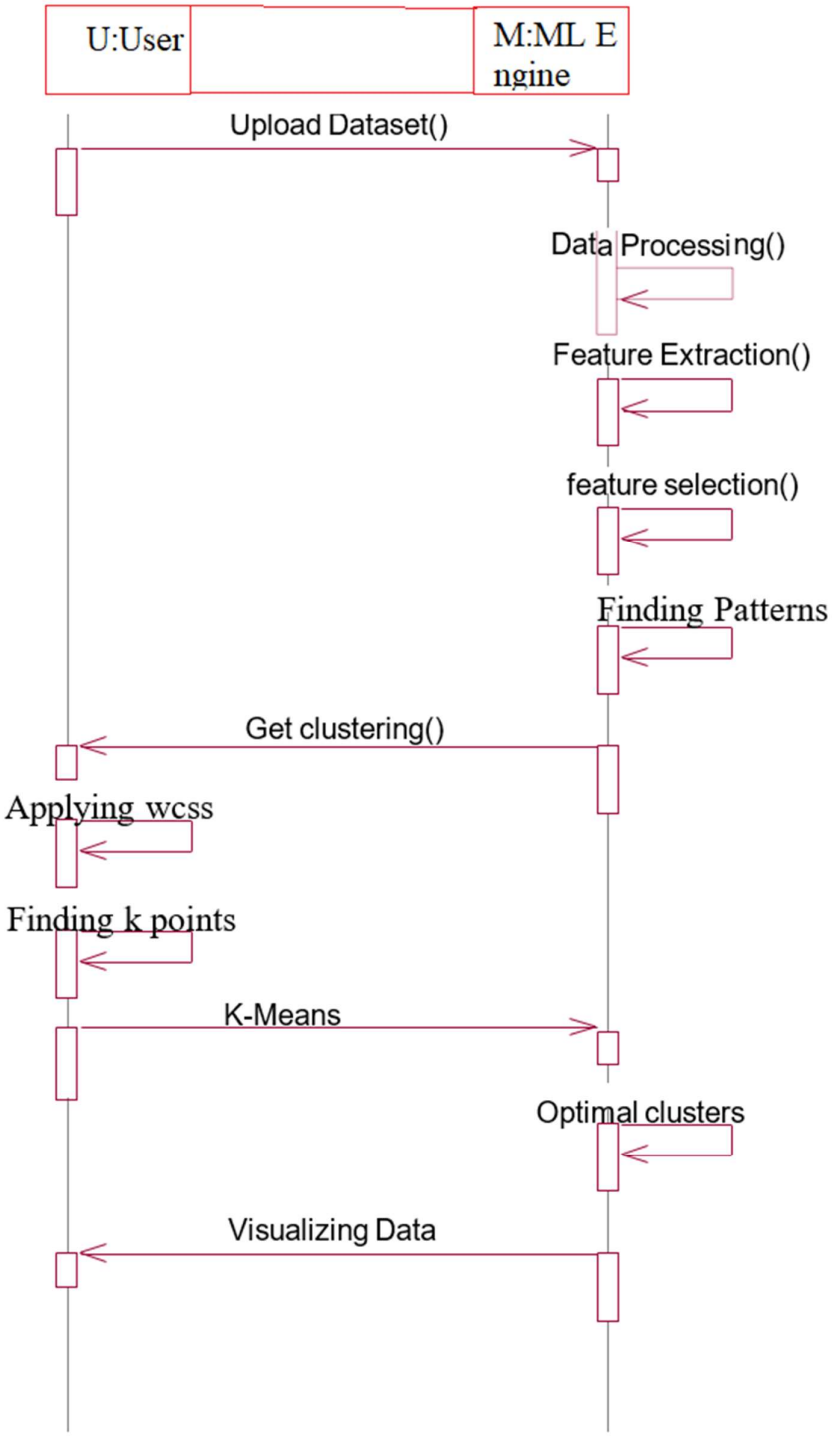
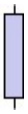


Fig 5.3.3 Sequence diagram for system

5.3.4 Collaboration Diagram:

Collaboration diagrams depict how things interact with one another, just like the other Behavioral diagrams do. This kind of diagram combines elements of object and sequence diagrams. The Collaboration diagram uses a free-form arrangement of objects similar to that seen in an Object diagram, in contrast to the Sequence diagram, which portrays the interaction in a column and row type structure. This facilitates seeing all interactions involving a specific object.

Object	Objects are instances of classes, and are arranged horizontally. The pictorial for an Object is a class (a rectangle) with the name prefixed by the object name (optional) and a semi-colon	
Lifeline	The Lifeline identifies the existence of the object over time. The notation 2for a Lifeline is a vertical dotted line extending from an object.	
Activation	Activations, modelled as rectangular boxes on the lifeline, indicate when the object is performing an action.	
Message	Messages, modeled as horizontal arrows Between Activation Indicates that the communications between objects.	

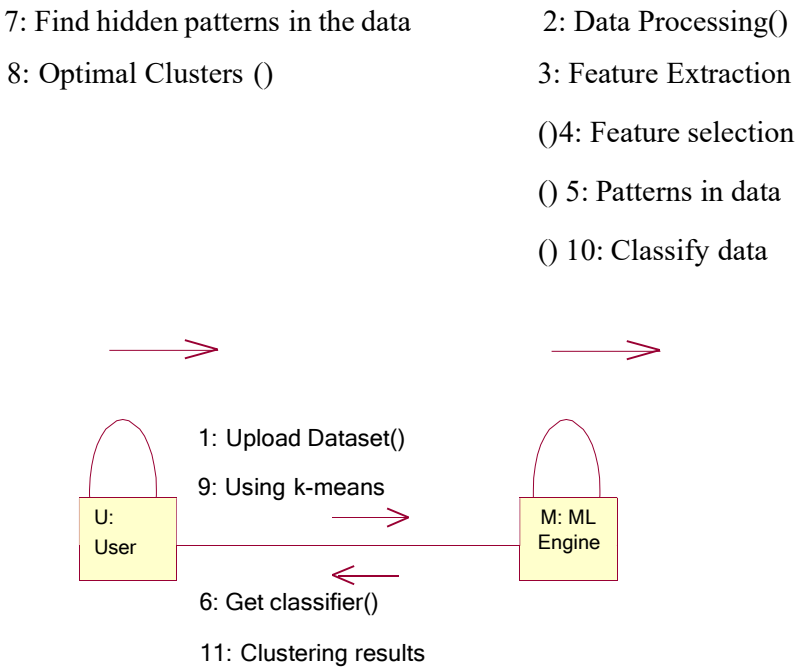


Fig 5.3.4: Collaboration diagram for System

5.3.5 Activity Diagram:

This depicts how events go through the system. The actions that take place within a use case or a behavior of an item normally happen in order. A simplified view of what happens throughout an operation or a process is intended by an activity diagram. Each activity is represented by a rounded rectangle, and after processing within it has been compiled, the system moves on automatically to the next activity. The transition from one action to the next is symbolized by an arrow. A system's activities are described in an activity diagram. Activities are the state that depicts a series of procedures being carried out. These are comparable to dataflow diagrams and flowcharts.

Initial state: which state is starting the process?

Action State: An action state represents the execution of an atomic action, typically the invocation of an operation. An action state is a simple state with an entry action whose only exit transition is triggered by the implicit event of completing the execution of the entry action.

Transition:

A transition is a directed relationship between a source state vertex and a target state vertex. It may be part of a compound transition, which representing the complete response of the static machine to a particular event instance.

Final state:

A final state represents the last or "final" state of the enclosing composite state. There may be more than one final state at any level signifying that the composite state can end in different ways or conditions.

Decision:

A state diagram (and by derivation an activity diagram) expresses decision that depend on Boolean conditions are used to indicate different possible transitions that depend on Boolean conditions of the owning object.

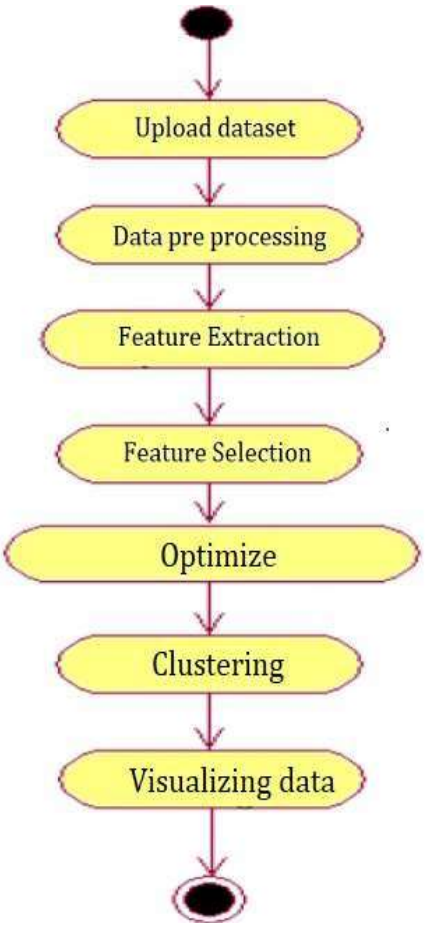


Fig 5.3.5 Activity Diagram for System

CHAPTER 6
SYSTEM IMPLEMENTATION

CHAPTER - 6
SYSTEM IMPLEMENTATION

6.1 Modules

6.1.1 Data Collecting:

It is a process of collecting the relevant data to your based criteria and this will be taken from the UCI Machine Learning Repository that is used for various data collections and using this module we can perform further actions like data extraction, selection etc.,

- The information gathered might not be relevant to the problem statement.
- Incomplete data or sub-data might not be present. For a certain class of prediction, that can manifest as missing images or empty values in columns.
- The model may propagate ingrained biases regarding gender, age, or area, depending on how the data, individuals, and labels themselves are selected.

6.1.2 Data Pre-Processing:

It is consisting of some basic steps like importing dataset, libraries, and finding missing values. This module provides the basic operations on data and provides effective output. Based on the analysis of data the outcome appears in this project we use of seaborn package to visualize the data efficiently and effectively.

- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Splitting dataset into training and test set
- Feature scaling

6.1.3 Data Cleaning

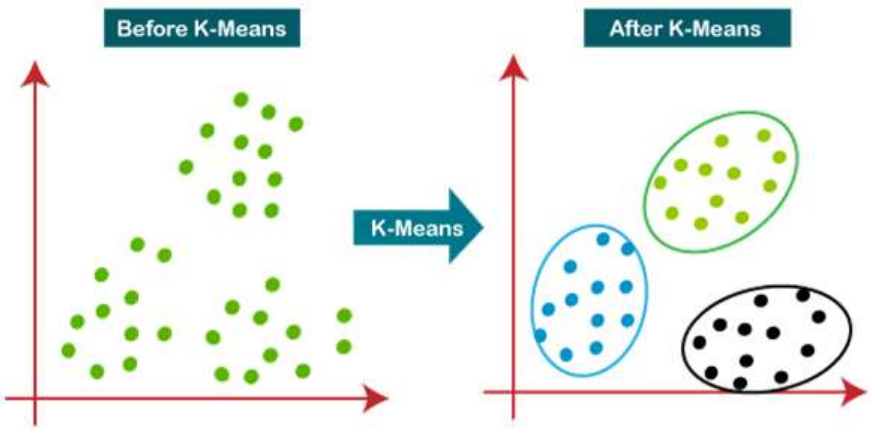
One of the crucial components of machine learning is data cleaning. It is crucial to the process of creating a model. However, effective data cleaning determines a project's success or failure. If the dataset is thoroughly cleaned, there is a potential that we can get decent results using straightforward techniques as well. This can be quite helpful at times, especially when it comes to computing when the dataset size is enormous.

- Removal of unwanted observations
- Fixing Structural errors
- Handling missing data

6.2 Methodology (Algorithm)

K-Means Clustering Algorithm:

- K-Means Machine learning clustering issues are resolved using clustering, an unsupervised learning approach. It is an iterative technique that splits the unlabeled dataset into k distinct clusters, each of which contains just the datasets that belong to that group and share comparable attributes.
- Here, K specifies how many pre-defined clusters must be produced as part of the process for example, if K=2, there will be two clusters, if K=3, there will be three clusters, and so on.
- It gives us the ability to divide the data into various groups and provides a practical method for automatically identifying the groups in the unlabeled dataset without the need for any training. The algorithm is centroid-based, with each cluster having a unique centroid.
- This algorithm's primary goal is to reduce the total distances between each data point and its corresponding clusters. The algorithm starts with an unlabeled dataset as its input, separates it into k clusters, and then continues the procedure until it runs out of clusters to use. In this algorithm, the value of k should be predetermined.



How does the K-Means Algorithm Work?

The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

The Elbow Method

- The Elbow approach applies k-means clustering on the dataset for a range of k values, such as from 1 to 10, and computes the average score for each value of k. By default, the distortion scores the sum of the square distances between each point and its designated center is calculated.
- The optimal value for k can be seen graphically by plotting these overall characteristics for each model. The "elbow," or point of inflexion on the curve, is the optimal value of k if the line chart resembles an arm. The "arm" can point upward or downward, but if there is a sharp inflection point, it is likely that the underlying model works best at that point.
- To determine the ideal number of clusters, we employ the Elbow Method, which compares the Within Cluster Sum of Squares (WCSS) to the number of clusters (K Value). The WCSS calculates the sum of observations' distances from their cluster centroids using the formula shown in equation 16 below.

$$WCSS = \sum_{i \in n} (X_i - Y_i)^2$$

Here, Yi is centroid for observation Xi. The main goal is to maximize number of clusters and in limiting case each data point becomes its own cluster centroid.

Technologies Description:

Introduction to Python

Python is a high-level object-oriented programming language that was created by Guido van Rossum. It is also called general-purpose programming language as it is used in almost every domain we can think of as mentioned below:

- Web Development
- Software Development
- Game Development
- AI & ML
- Data Analytics

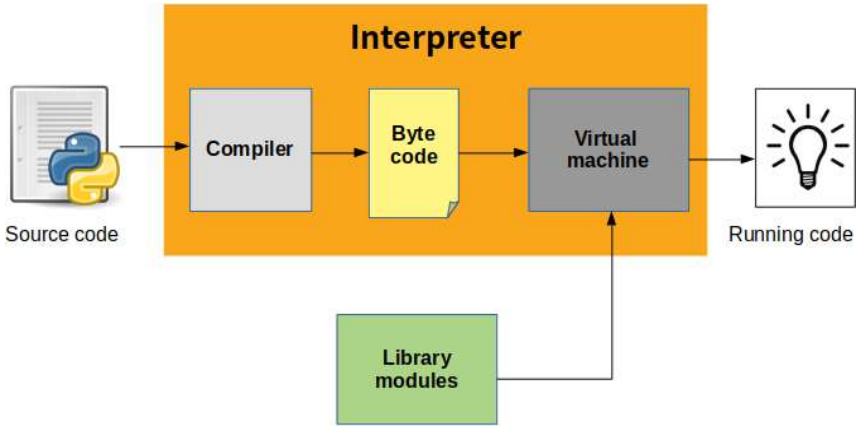
Every programming language has a function or use case that is specific to a given domain. JavaScript, for instance, is the most popular language used by web developers because it allows the developer the ability to manage applications using various frameworks, such as react, Vue, and angular, which are used to create attractive User Interfaces. They also simultaneously have advantages and disadvantages. So, if we take python as an example, it is general-purpose, which means it is widely used in every domain. Now that you understand why learning Python is not a prerequisite, you can see why Python is popular among developers as well. Python's syntax is comparable to that of the English language, making it easier for programmers to write programs with fewer lines of code. Because it is open-source, there are numerous libraries available that simplify the work of developers and ultimately lead to great productivity. Python is frequently used for creating websites and applications, automating repetitive tasks, and analyzing and displaying data.

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code
- building large applications.
- It provides very high-level dynamic data types and supports dynamic checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

History of Python:

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3,C, C++, Algol-68, Smalltalk, Unix shell, and other scripting languages. At the time when he began implementing Python, Guido van Rossum was also reading the published scripts from "Monty Python's Flying Circus" (a BBC comedy series from the seventies, in the unlikely case you didn't know). It occurred to him that he needed a name that was short, unique, and slightly mysterious, so he decided to call the language Python.



Python Features:

- **Easy-to-learn:** Python has few keywords, simple structure, and a clearly defined syntax.
- **Easy-to-read:** Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain:** Python's source code is fairly easy-to-maintain.
- **A broad standard library:** Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode:** Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable:** Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

INTRODUCTION TO MACHINE LEARNING:

A subfield of artificial intelligence (AI) and computer science called machine learning focuses on using data and algorithms to simulate how humans learn, gradually increasing the accuracy of the system. IBM has a long history with artificial intelligence. One of its own, Arthur Samuel, is credited with creating the term "machine learning" with his research on the game of checkers (PDF, 481 KB) (link lives outside IBM). In 1962, Robert Nealey, a self-

described checkers master, competed against an IBM 7094 computer, but he was defeated. The rapidly expanding discipline of data science includes machine learning as a key element.

Algorithms are trained to generate classifications or predictions using statistical techniques, revealing important insights in data mining operations. The decisions made as a result of these insights influence key growth indicators in applications and enterprises, ideally. Data scientists will be more in demand as big data develops and grows because they will be needed to help identify the most important business issues and then the data to answer them. A machine learning algorithm's learning system is divided into three primary components.

A Decision Process:

In general, predictions or classifications are made using machine learning algorithms. Your algorithm will generate an estimate of a pattern in the input data based on certain input data, which can be labelled or unlabeled.

An Error Function:

An error function is used to assess how well the model predicts. If there are known examples, an error function can compare them to gauge the model's correctness.

A Model Optimization Process:

Weights are modified to lessen the difference between the known example and the model estimate if the model can match the data points in the training set more accurately. The programme will repeat this evaluation and optimize the process, updating weights autonomously till threshold of accuracy has been met.

Machine learning classifiers fall into three primary categories.

Supervised machine learning

The term "supervised learning," which is also used to refer to supervised machine learning, refers to the process of teaching algorithms to correctly classify data or predict outcomes using labelled datasets. The model modifies its weights as input data is fed into it until the model is properly fitted. This happens as part of the cross-validation process to make sure the model doesn't fit too well or too poorly. Supervised learning assists firms in finding scalable solutions to a range of real-world issues, such as classifying spam in a different folder from your email. Neural networks, naive bayes, linear regression, logistic regression, random forests, support

vector machines (SVM), and other techniques are used in supervised learning.

Unsupervised machine learning

Unsupervised learning, commonly referred to as unsupervised machine learning, analyses and groups unlabeled datasets using machine learning algorithms. These algorithms identify hidden patterns or data clusters without the assistance of a human. It is the appropriate solution for exploratory data analysis, cross-selling tactics, consumer segmentation, picture and pattern recognition because of its capacity to find similarities and differences in information.

Semi-supervised learning

A satisfying middle ground between supervised and unsupervised learning is provided by semi-supervised learning. It guides categorization and feature extraction from a larger, unlabeled data set during training using a smaller, labelled data set. If you don't have enough labelled data or can't pay to label enough data to train a supervised learning system, semi-supervised learning can help.

6.3 SOURCE CODE

1.Program: Importing the Essential Libraries and Uploading dataset

```
#Importing the necessary libraries
import numpy as np
import pandas as pd
#Importing the matplotlib for visualize the data
import matplotlib.pyplot as plt
import seaborn as sns
from mpl_toolkits.mplot3d
import Axes3D get_ipython().run_line_magic('matplotlib', 'inline')
#In[5]:
#Reading the excel file
data=pd.read_csv("Mall_Customers.csv")
# In[6]:
#Number of customers we have
print("Number of customers we have data for-" , len(data))
# In[7]:
data.head()
# In[8]:
len(data)
# In[9]:
data.isnull()
# In[10]:
data.describe()
# In[11]:
data.corr()
# In[12]:
import warnings
warnings.filterwarnings('ignore')
```

2.Program : Distribution of Data about Annual Income and Age

```
# # Distribution of data
#In[13]:
#Distribution of Annnual Income
plt.figure(figsize=(10, 6))
sns.set(style = 'whitegrid')
sns.histplot(data['Annual Income (k$)'])
plt.title('Distribution of Annual Income (k$)', fontsize=20)
plt.xlabel('Range of Annual Income (k$)')
plt.ylabel('Count')

# In[14]:
#Distribution of age
plt.figure(figsize=(10, 6))
sns.set(style ='whitegrid')
```

```
sns.histplot(data['Age'])
plt.title('Distribution of Age', fontsize = 20)
plt.xlabel('Range of Age')
plt.ylabel('Count')
```

3. Program : Plotting the Spending score & annual income with respect to gender

```
plt.figure(figsize=(12, 8))
sns.set(style = 'whitegrid')
sns.histplot(data['Spending Score (1-100)'])
plt.title('Distribution of Spending Score (1-100)', fontsize = 20)
plt.xlabel('Range of Spending Score (1-100)')
plt.ylabel('Count')

 # Gender Analysis
# In[16]:
genders = data.Gender.value_counts()
sns.set_style("darkgrid")
plt.figure(figsize=(10,4))
sns.barplot(x=genders.index, y=genders.values)
plt.show()

# In[17]:
plt.figure(figsize=(10,8))
sns.scatterplot(x = 'Age' , y = 'Annual Income (k$)' , hue="Gender",data = data ,s = 60 )
plt.xlabel('Age')
plt.ylabel('Annual Income (k$)')
plt.title('Age vs Annual Income w.r.t Gender')
plt.legend()
plt.show()

# In[18]:
plt.figure(figsize=(10,9))
sns.scatterplot(x = 'Annual Income (k$)', y = 'Spending Score (1-100)', hue="Gender",data =
data,s=60 )
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100) vs Annual Income (k$)')
plt.legend()
plt.show()
```

4. Program : Spending Score Buckets to represent the number of customers based on Spending Score

```
age18_25 = data.Age[(data.Age <= 25) & (data.Age >= 18)]
age26_35 = data.Age[(data.Age <= 35) & (data.Age >= 26)]
age36_45 = data.Age[(data.Age <= 45) & (data.Age >= 36)]
age46_55 = data.Age[(data.Age <= 55) & (data.Age >= 46)]
age55above = data.Age[(data.Age >= 56)]
x = ["18-25","26-35","36-45","46-55","55+"]
```

```
y = [len(age18_25.values), len(age26_35.values), len(age36_45.values), len(age46_55.values),
len(age55 above.values)]
plt.figure(figsize=(10,6))
sns.barplot(x=x, y=y)
plt.title("Customer and Ages Barplot")
plt.xlabel("Age")
plt.ylabel("Number of Customers")
plt.show()

# # Spending Score Buckets
# In[20]:

ss1_20 = data["Spending Score (1-100)"][(data["Spending Score (1-100)"] >= 1) &
(data["SpendingScore (1-100)"] <= 20)]
ss21_40 = data["Spending Score (1-100)"][(data["Spending Score (1-100)"] >= 21) &
(data["Spending Score (1-100)"] <= 40)]
ss41_60 = data["Spending Score (1-100)"][(data["Spending Score (1-100)"] >= 41) &
(data["Spending Score (1-100)"] <= 60)]
ss61_80 = data["Spending Score (1-100)"][(data["Spending Score (1-100)"] >= 61) &
(data["Spending Score (1-100)"] <= 80)]
ss81_100 = data["Spending Score (1-100)"][(data["Spending Score (1-100)"] >= 81) &
(data["Spending Score (1-100)"] <= 100)]

score_x = ["1-20", "21-40", "41-60", "61-80", "81-100"]
score_y = [len(ss1_20.values), len(ss21_40.values), len(ss41_60.values), len(ss61_80.values),
len(ss81_100.values)]
plt.figure(figsize=(10,6))
sns.barplot(x=score_x, y=score_y,palette="Set2")
plt.title("Spending Scores")
plt.xlabel("Score")
plt.ylabel("Number of Customer Having the Spending Score In That Range")
plt.show()

# # Annual Income (1000 USD)
```

5.Program : Plotting the graph on Annual Income to find the number of Customers

```
ai0_30 = data["Annual Income (k$)"][(data["Annual Income (k$)"] >= 0) & (data["Annual Income
(k$)"] <= 30)]
ai31_60 = data["Annual Income (k$)"][(data["Annual Income (k$)"] >= 31) & (data["Annual
Income(k$)"] <= 60)]
ai61_90 = data["Annual Income (k$)"][(data["Annual Income (k$)"] >= 61) & (data["Annual
Income(k$)"] <= 90)]
ai91_120 = data["Annual Income (k$)"][(data["Annual Income (k$)"] >= 91) & (data["Annual
Income (k$)"] <= 120)]
ai121_150 = data["Annual Income (k$)"][(data["Annual Income (k$)"] >= 121) & (data["Annual
Income (k$)"] <= 150)]
income_x = ["$ 0 - 30,000", "$ 30,001 - 60,000", "$ 60,001 - 90,000", "$ 90,001 - 120,000", "$
120,001 - 150,000"]
income_y = [len(ai0_30.values), len(ai31_60.values), len(ai61_90.values), len(ai91_120.values),
```

```
len(ai121_150.values)]
plt.figure(figsize=(15,6))
sns.barplot(x=income_x, y=income_y,
palette="nipy_spectral_r")plt.title("Annual Incomes")
plt.xlabel("Income")
plt.ylabel("Number of Customer")
plt.show()

# In[22]:
#Taking another look at the data

# In[23]:
data
```

6.Program : Clustering based on two features as Spending score and Annual Income

```
df1=data[["CustomerID","Gender","Age","Annual Income (k$)","Spending Score (1-100)"]]
X=df1[["Annual Income (k$)","Spending Score (1-100)"]]

# In[25]:
#The input data
X.head()

# In[26]:
#Scatterplot of the input data

plt.figure(figsize=(10,6))
sns.scatterplot(x = 'Annual Income (k$)',y = 'Spending Score (1-100)', data = X ,s = 60 )
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100) vs Annual Income (k$)')
plt.show()

# There seems to be some patterns in the data.
# KMeans clustering.
# In[27]:
#Importing KMeans from sklearn
```

7.Program : Importing the K-means alogorithm to find the partitioning of clusters using of WCSS

```
from sklearn.cluster import KMeans
# Now we calculate the Within Cluster Sum of Squared Errors (WSS) for different values of k.
# Next, we choose the k for which WSS first starts to diminish.

# In[28]:
wcss=[]
for i in range(1,11):
km=KMeans(n_clusters=i)
km.fit(X)
wcss.append(km.inertia_)
```

```
# In[29]:
#The elbow curve
plt.figure(figsize=(12,6))
plt.plot(range(1,11),wcss)
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker ="8")
plt.xlabel("K Value")
plt.xticks(np.arange(1,11,1))
plt.ylabel("WCSS")
plt.show()

# In[30]:
#this is known as the elbow graph, the x axis being the number of
clusters
#the number of clusters is taken at the elbow joint point
#this point is the point where making clusters is most relevant
#the numbers of clusters is kept at maximum

# In[31]:
#Taking 5 clusters
```

8. Program : Now finding the Clusters for two features and taking k=5 clusters as maximum and finding out the clusters and represent them with labels

```
km1=KMeans(n_clusters=5)

# In[32]:
#Fitting the input data
km1.fit(X)

# In[33]:
#predicting the labels of the input data
y=km1.predict(X)

# In[34]:
#adding the labels to a column named label
df1["label"] = y

# In[35]:
#The new dataframe with the clustering done
df1.head()

# In[36]:
#Scatterplot of the clusters
plt.figure(figsize=(12,9))
sns.scatterplot(x = 'Annual Income (k$)',y = 'Spending Score (1-100)',hue="label",
palette=['green','orange','brown','dodgerblue','red'], legend='full',data = df1  ,s = 70 )
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Spending Score (1-100) vs Annual Income (k$)')
plt.show()
```

 # In[37]:

9. Program : Now Providing the Customer ID according to the Formed clusters or groups

```
cust1=df1[df1["label"]==1]
print('Number of customer in 1st group=', len(cust1))
print('They are -', cust1["CustomerID"].values)
print("----------------------------------------------------")
```

 # In[38]:

```
cust2=df1[df1["label"]==2]
print('Number of customer in 2nd group=', len(cust2))
print('They are -',  cust2["CustomerID"].values)
print("----------------------------------------------------")
```

 # In[39]:

```
cust3=df1[df1["label"]==0]
print('Number of customer in 3rd group=', len(cust3))
print('They are -', cust3["CustomerID"].values)
print("----------------------------------------------------")
```

 # In[40]:

```
cust4=df1[df1["label"]==3]
print('Number of customer in 4th group=', len(cust4))
print('They are -', cust4["CustomerID"].values)
print("----------------------------------------------------")
```

 # In[41]:

```
cust5=df1[df1["label"]==4]
print('Number of customer in 5th group=', len(cust5))
print('They are -', cust5["CustomerID"].values)
print("----------------------------------------------------")
```

 # # Clustering on the basis of 3D data

 # In[42]:

```
#Now we shall take 3 input features
df2=data[["CustomerID","Gender","Age","Annual Income (k$)","Spending Score (1-100)"]]
```

 # In[43]:

```
df2.head()
```

 # In[44]:

```
#Taking the features
X2=df2[["Age","Annual Income (k$)","Spending Score (1-100)"]]
```

10. Program : Now Calculating the WCSS for different values of k that used to diminish the k values

```
wcss = []
for k in range(1,11):
kmeans = KMeans(n_clusters=k, init="k-means++")
kmeans.fit(X2)
wcss.append(kmeans.inertia_)


# In[46]:
plt.figure(figsize=(12,11))
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker ="8")
plt.xlabel("K Value")
plt.xticks(np.arange(1,11,1))
plt.ylabel("WCSS")
plt.show()

# In[47]:
#We choose the k for which WSS starts to diminish

km2 = KMeans(n_clusters=5)
y2 = km.fit_predict(X2)
df2["label"] = y2

# In[48]:
#The new data
df2.head()
```

11. Program : Finally, we are plotting the 3D data on basis of three features andproviding them customer id's for each labels to find out the clusters

```
fig = plt.figure(figsize=(20,10))
ax = fig.add_subplot(111, projection='3d')
ax.scatter(df2.Age[df2.label == 0], df2["Annual Income (k$)"][df2.label == 0],
df2["Spending Score(1-100)"][df2.label == 0], c='purple', s=70)
ax.scatter(df2.Age[df2.label == 1], df2["Annual Income (k$)"][df2.label == 1],
df2["Spending Score(1-100)"][df2.label == 1], c='red', s=70)
ax.scatter(df2.Age[df2.label == 2], df2["Annual Income (k$)"][df2.label == 2],
df2["Spending Score(1-100)"][df2.label == 2], c='blue', s=70)
ax.scatter(df2.Age[df2.label == 3], df2["Annual Income (k$)"][df2.label == 3],
df2["Spending Score(1-100)"][df2.label == 3], c='green', s=70)
ax.scatter(df2.Age[df2.label == 4], df2["Annual Income (k$)"][df2.label == 4],
df2["Spending Score(1-100)"][df2.label == 4], c='yellow', s=70)

ax.view_init(35, 185)
plt.xlabel("Age")
plt.ylabel("Annual Income (k$)")
ax.set_zlabel('Spending Score (1-100)')
plt.show()
```

```
# Now printing the Customer ID according to the groups.
# In[50]:
cust1=df2[df2["label"]==1]
print('Number of customer in 1st group=', len(cust1))
print('They are -', cust1["CustomerID"].values)
print("-----------------------------------------------------")

# In[51]:
cust2=df2[df2["label"]==2]
print('Number of customer in 2nd group=', len(cust2))
print('They are -', cust2["CustomerID"].values)
print("-----------------------------------------------------")

# In[52]:
cust3=df2[df2["label"]==3]
print('Number of customer in 3rd group=', len(cust3))
print('They are -', cust3["CustomerID"].values)
print("-----------------------------------------------------")

# In[53]:
cust4=df2[df2["label"]==4]
print('Number of customer in 4th group=', len(cust4))
print('They are -', cust4["CustomerID"].values)
print("-----------------------------------------------------")

# In[54]:
cust4=df2[df2["label"]==5
print('Number of customer in 5th group=', len(cust5))
print('They are -', cust4["CustomerID"].values)
print("-----------------------------------------------------")
```

CHAPTER 7
SYSTEM TESTING

CHAPTER - 7
SYSTEM TESTING

7.1 Introduction:

In general, software engineers distinguish software faults from software failures. In caseof a failure, the software does not do what the user expects. A fault is a programming error that may or may not actually manifest as a failure. A fault can also be described as an error in the correctness of the semantic of a computer program. A fault will become a failure if the exact computation conditions are met, one of them being that the faulty portion of computer software executes on the CPU.A fault can also turn into a failure when the software is ported to a different hardware platform or a different compiler, or when the software gets extended. Software testing isthe technical investigation of the product under test to provide stakeholders with quality related information.

System Testing and Implementation:

The purpose is to exercise the different parts of the module code to detect coding errors. After this the modules are gradually integrated into subsystems, which are then integrated themselves too eventually forming the entire system. During integration of module integration testing is performed. The goal of this is to detect designing errors, while focusing the interconnection between modules. After the system was put together, system testing is performed. Here the system is tested against the system requirements to see if all requirements were met and the system performs as specified by the requirements. Finally accepting testing is performed to demonstrate to the client for the operation of the system.

For the testing to be successful, proper selection of the test case is essential. There are two different approaches for selecting test case. The software or the module to be tested is treated as ablack box, and the test cases are decided based on the specifications of the system or module. Forthis reason, this form of testing is also called “black box testing”.

Testing is an extremely critical and time-consuming activity. It requires proper planning ofthe overall testing process. Frequently the testing process starts with the test plan. This plan identifies all testing related activities that must be performed and specifies the schedule, allocates the resources, and specifies guidelines for testing. The test plan specifies conditions that should be tested; different units to be tested, and the manner in which the module will be

integrated together. Then for different test unit, a test case specification document is produced, which lists all the different test cases, together with the expected outputs, that will be used for testing. During the testing of the unit the specified test cases are executed and the actual results are compared with the expected outputs. The final output of the testing phase is the testing report and the error report, or a set of such reports. Each test report contains a set of test cases and the result of executing the code with the test cases. The error report describes the errors encountered and the action taken to remove the error.

Testing Techniques

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of conditions known as test cases and the output is evaluated to determine whether the program is performing as expected. In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

Black Box Testing

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in data structure or external database access
- Performance errors
- Initialization and termination errors.

White Box Testing

In this testing, the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been uses to generate the test cases in the following cases:

- Guarantee that all independent paths have been executed.
- Execute all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational.
- Execute internal data structures to ensure their validity.

Testing Strategies

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

This System consists of 3 modules. Those are Reputation module, route discovery module, audit module. Each module is taken as unit and tested. Identified errors are corrected and executable unitare obtained.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

System Testing:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

Functional Testing:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Procedures : interfacing systems or procedures must be invoked.

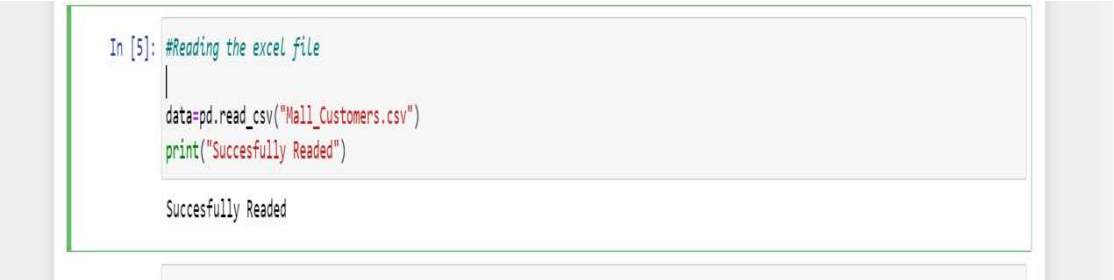
7.2 Sample Test Cases Specification:

Test case id	Test case name	Input	Expected Output	Observed Output	Result
T1	Upload Data Set	Enter valid Path of dataset	Dataset should load successful.	Dataset loaded successfully.	Pass
T2	Upload Data Set	Enter invalid file path or null.	Application should show an error	An Error Occurred with message "invalid path"	Fail

Table 7.2.1: Sample Test Cases Specification

TESET CASE 1:

TEST SCREEN FOR T1:



DESCRIPTION: The dataset was uploaded Successfully

TEST CASE 2:

TEST SCREEN FOR T2:

DESCRIPTION: The above figure shows that there is an error with uploading of thedataset

CHAPTER 8
EXPERIMENTAL RESULTS

CHAPTER - 8
EXPERIMENTAL RESULTS

```
#Importing the necessary libraries

In [1]: import numpy as np

In [2]: import pandas as pd

In [3]: #importing the matplotlib for visualize the data

        import matplotlib.pyplot as plt

In [4]: import seaborn as sns
        from mpl_toolkits.mplot3d import Axes3D
        %matplotlib inline

In [5]: #Reading the excel file

        data=pd.read_csv("Mall_Customers.csv")
        print("Succesfully uploaded")

        Succesfully uploaded

In [6]: #Number of customers we have

        print("Number of customers we have data for-" , len(data))

        Number of customers we have data for- 200

In [7]: data.head()
```

Fig 8.1 : Importing And data preprocessing

Description: Importing the necessary packages and uploading the dataset for data preprocessing to findout the number of customers in the data

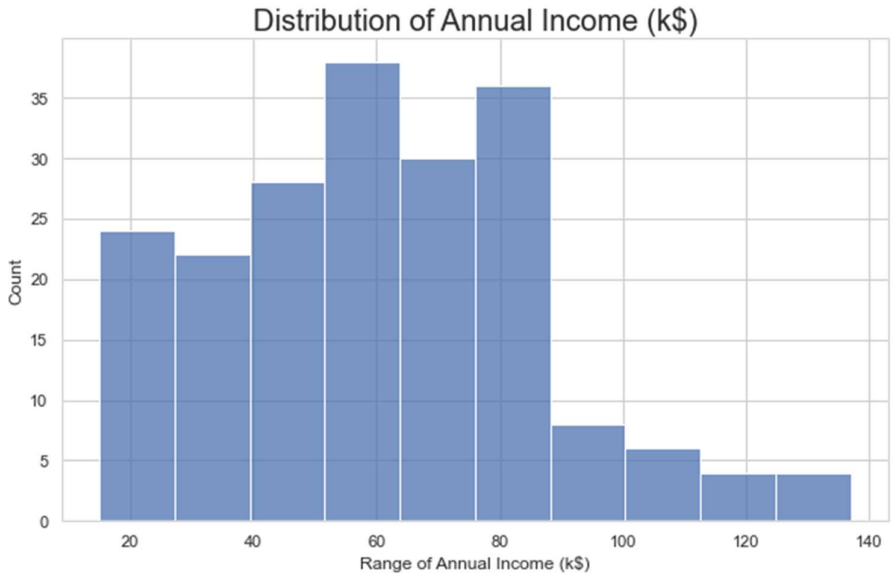


Fig 8.2: Annual Income Distribution

Description: The above graph is representing the annual income and consisting of range andcount to identify the customers

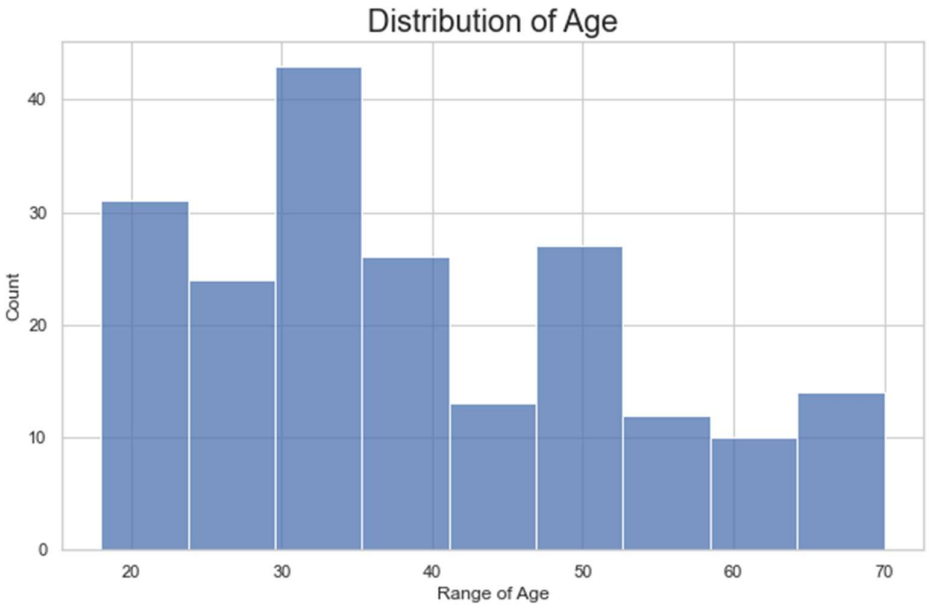


Fig 8.3 : Distribution of Age

Description: The above graph is representing the annual income and consisting of range and count to identify the customers

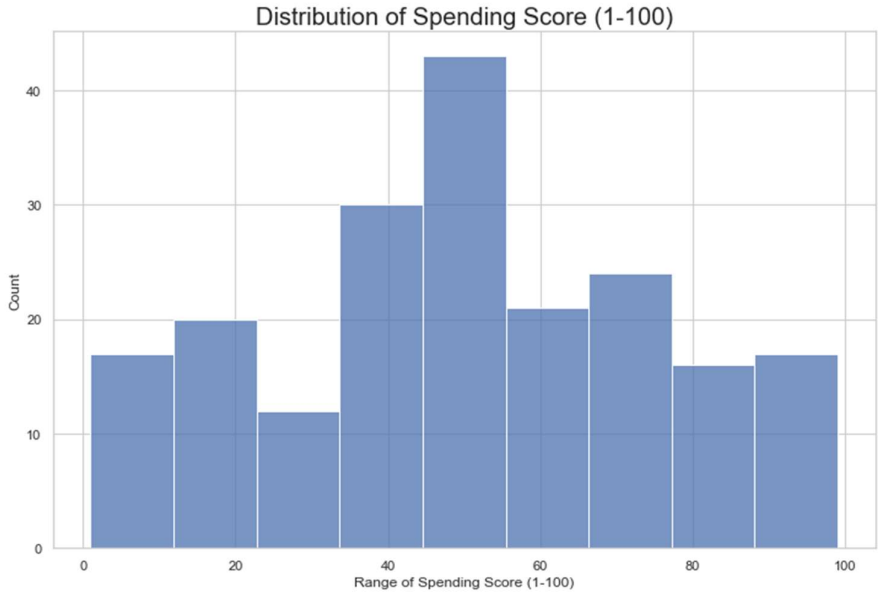


Fig 8.4 : Distribution of Spending Scores

Description: Plotting the Spending scores of Distributions to know the Range and count of the cores

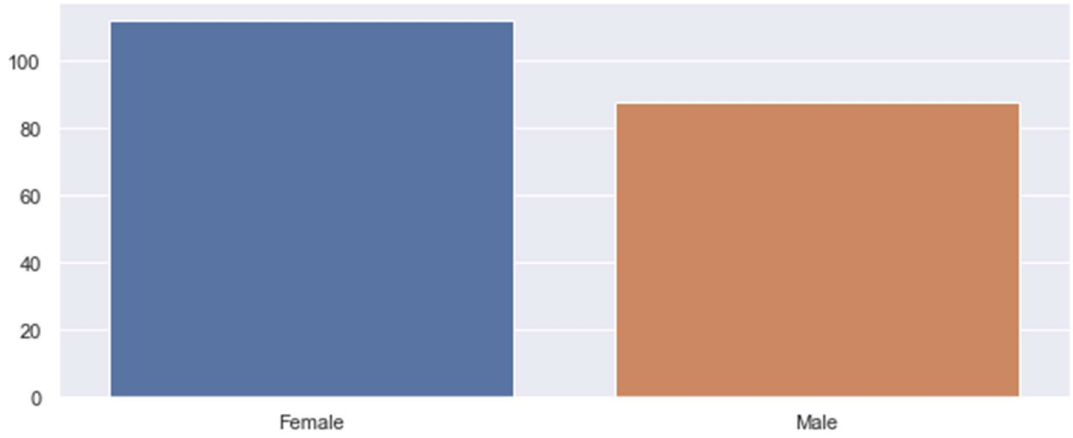


Fig 8.5 : Analysis of Gender

Description: Comparing the Gender based on male and females

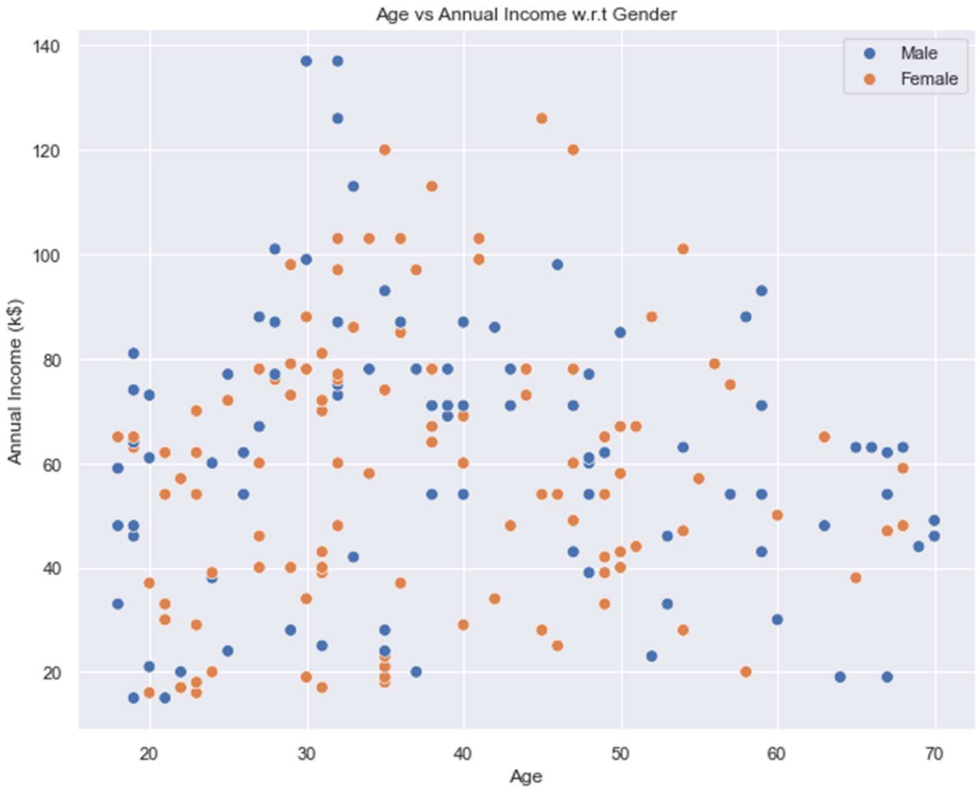


Fig 8.6 : Graph of Two Features

Description: the graph based on Age and Annual Income with respect to Gender to find out the Targeted customer based on gender whether male or female are the high valuable customers

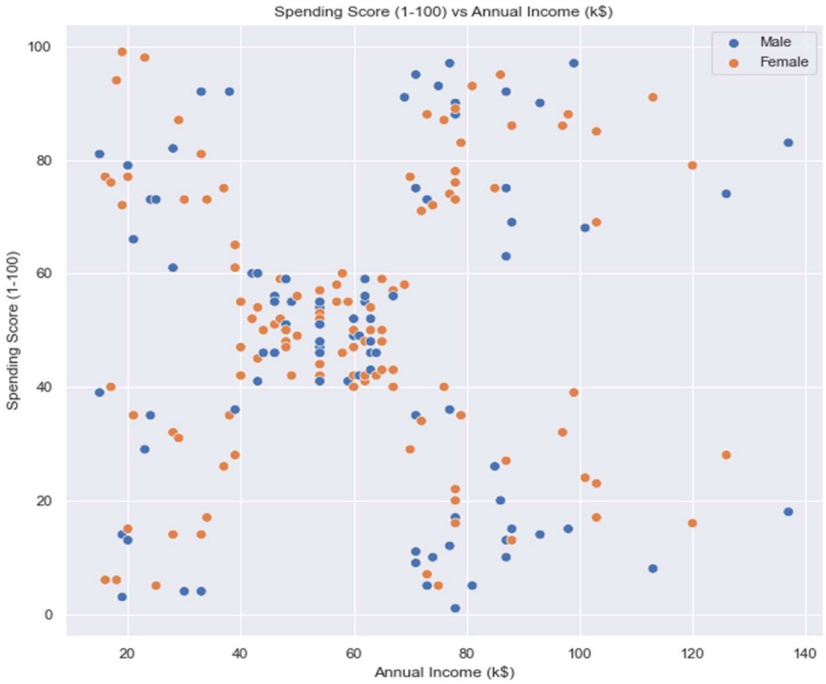


Fig 8.7 : Identifying the Patterns

Description: The above graph shows pattern in data of Spending score and Annual incometo describe the similarities in those two features

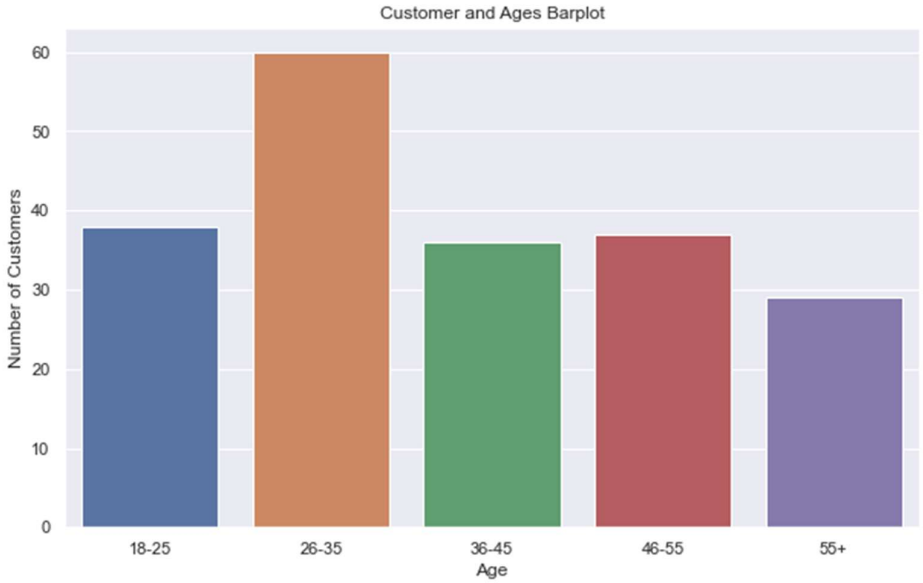


Fig 8.8 : Age of the Customers

Description: In the graph it shows the relationship between the customers related to their ages & represents that based on the Age criteria also we can relate the Number of customers

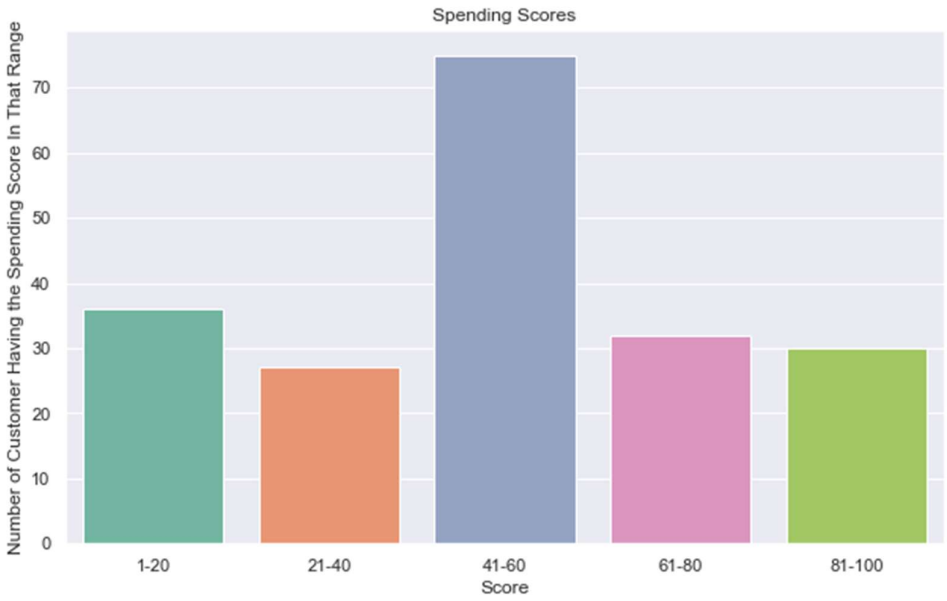


Fig 8.9 : Spending Score Plot

Description: The above figure shows that the plotting of spending scores by the customers and having the range and score to identify spending scores

Fig 8.10 : Annual Income Plot

Description: The plotting of Annual Income shows that the Income of the customers and how many customers are having high income

Fig 8.11 : Scatter Plot of Input Data

Description: The above graph represents the hidden patterns in the data related with theSpending scores and annual income

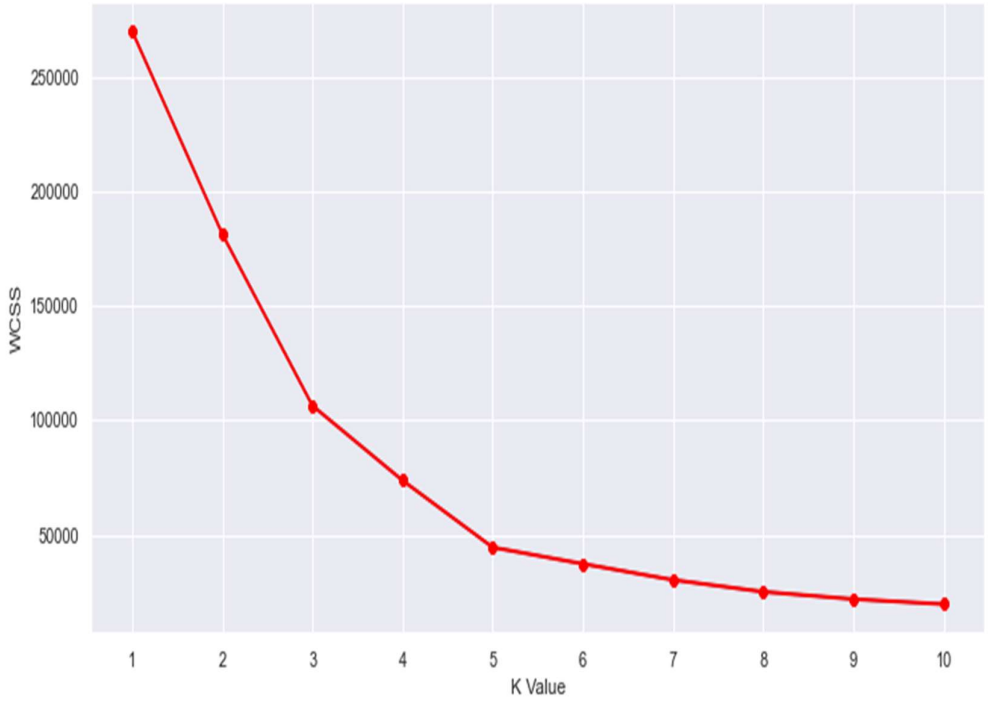


Fig 8.12 : Finding K value

Description: The above graph represents the K value and it helps to form group of clusters by using elbow method

Fig 8.13 : Distribution Plot

Description: The above graph shows the clusters based on the two features of Spending scores and Annual Income and providing them with a label and the Label 4 has the less spending score and less Annual income.

```
In [50]: cust1=df2[df2["label"]==0]
         print('Number of customer in 1st group=', len(cust1))
         print('They are -', cust1["CustomerID"].values)
         print("----------------------------------------------")

Number of customer in 1st group= 10
They are - [182 184 186 188 190 192 194 196 198 200]
----------------------------------------------

In [51]: cust2=df2[df2["label"]==1]
         print('Number of customer in 2nd group=', len(cust2))
         print('They are -', cust2["CustomerID"].values)
         print("----------------------------------------------")

Number of customer in 2nd group= 29
They are - [ 44  52  53  59  62  66  69  70  76  78  79  82  85  88  89  92  95  96
  98 100 101 104 106 112 114 115 116 121 123]
----------------------------------------------

In [52]: cust3=df2[df2["label"]==2]
         print('Number of customer in 3rd group=', len(cust3))
         print('They are -', cust3["CustomerID"].values)
         print("----------------------------------------------")

Number of customer in 3rd group= 22
They are - [129 131 135 137 139 141 145 149 151 153 155 157 159 163 165 167 169 171
 173 175 177 179]
----------------------------------------------

In [53]: cust4=df2[df2["label"]==3]
         print('Number of customer in 4th group=', len(cust4))
         print('They are -', cust4["CustomerID"].values)
         print("----------------------------------------------")

Number of customer in 4th group= 22
They are - [ 2  4  6  8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 46]
----------------------------------------------

In [54]: cust5=df2[df2["label"]==4]
         print('Number of customer in 5th group=', len(cust5))
         print('They are -', cust5["CustomerID"].values)
         print("----------------------------------------------")

Number of customer in 5th group= 41
They are - [ 41  47  51  54  55  57  58  60  61  63  64  65  67  68  71  72  73  74
  75  77  80  81  83  84  86  87  90  91  93  97 102 103 105 107 108 109
 110 111 117 118 120]
```

Fig 8.14 : ID for groups

Description: The above figure shows that the number of groups having different types of customer ID's and in label 2 it shows that there is a high cluster having high cores and annual income and treated as Target Customers

CHAPTER 9
CONCLUSION & FUTURE SCOPE

CHAPTER - 9
CONCLUSION & FUTURE SCOPE

If you properly manage the best current customer segmentation process, however, the impact it can have on every part of your organization sales, marketing, product development, customer service, etc. is immense. Your business will possess stronger customer focus and market clarity, allowing it to scale in a far more predictable and efficient manner.

Ultimately, that means no longer needing to take on every customer that is willing to pay for your product or service, which will allow you to instead hone in on a specific subset of customers that present the most profitable opportunities and efficient use of resources. That is critical for every business, of course, but at the expansion stage, it can often be the difference between incredible success and certain failure.

CHAPTER 10
BIBLIOGRAPHY

CHAPTER - 10
BIBLIOGRAPHY

TEXTBOOKS:

1. Introduction to Machine Learning with Python: A Guide for Data Scientists,Andreas, Muller & Sarah Guido, Orielly Publications,2019.
2. Machine Learning, Tom. Mitchell, Mc Graw-Hill publication,2017.
3. Programming and problem solving with python, Ashok Namdev Kamthane, AmitAshok TMH,2019.

WEB SITES:

1. https://neptune.ai/blog
2. https://ijcrt.org/IJCRT_196650
3. https://www.researchgate.net/publication

REFFERENCES:

[1] Jiawei Han, Micheline Kamber, Jian Pei “Data Mining Concepts and Techniques”, Third Edition.
[2] D. Aloise, A. Deshpande, P. Hansen, and P. Popat, “The Basis of Market Segmentation” Euclidean sum-of-squares clustering,” Machine Learning, vol. 75, pp. 245-249, 2009.
[3] S. Dasgupta and Y. Freund, “Random Trees for Vector Quantization,” IEEE Trans. on Information Theory, vol. 55, pp. 3229-3242, 2009.
[4] “Customer Segmentation Using K Means Clustering,” Towards Data Science, Apr. 2019.
[5] Ruhul Reddy, “Who’s who: Understanding your business with customer segmentation,” INTERCOM.
[6] Kristen Baker, “The Ultimate Guide to Customer Segmentation: How to Organize Your Customers to Grow Better,” Hunspot.
[7] Tim Ehrens, “customer segmentation,” TechTarget.
[8] V.Vijilesh, “CUSTOMER SEGMENTATION USING MACHINE LEARNING,” International Research Journal of Engineering and Technology (IRJET), vol. 08, no. 05, May 2021.
[9] Expert Systems with Applications, vol. 100, Feb. 2018, "Retail Business Analytics: Customer Visit Segmentation Using Market Basket Data."
[10] "CUSTOMER SEGMENTATION USING MACHINE LEARNING," IJCRT, AMAN BANDUNI and ILAVENDHAN A, vol. 05, 2018.
[11] Tushar Kansal; Suraj Bahuguna; Vishal Singh; Tanupriya Choudhury, “Customer Segmentation using K-means Clustering,” IEEE, Jul. 2019.

Plagiarism Checker X Originality Report

Similarity Found: 10%

Date: Thursday, September 01, 2022
Statistics: 510 words Plagiarized / 5282 Total words
Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

Abstract We lived on Earth, where a large volume of data is collected daily and stored in computers. Monitoring of that vast volume of data is an important factor. The business strategy needs to be appropriate for a particular environment in the modern era of innovation, where there is strong competition to exceed everyone. Due to the large number of potential clients who are confused about what to buy and what not to buy, today's business is focused on innovative ideas. The companies operating also possess the capacity to identify the target potential customers.

Using the clustering technique, the customer segmentation process determines which customer segment to target. We lived on Earth, where a large volume of data is collected daily and stored in computers. Monitoring of that vast volume of data is an important factor. The business strategy needs to be appropriate for a particular environment. In the modern era of innovation, where there is strong competition to exceed everyone. Due to the large number of potential clients who are confused about what to buy and what not to buy, today's business is focused on innovative ideas. The companies operating also possess the capacity to identify the target potential customers. Using the clustering technique, the customer segmentation process determines which customer segment to target.

Introduction Customer segmentation is the process of dividing individuals who have characteristics relevant to marketing, such as age, gender, interests, and spending habits, into groups. Customer segmentation is a method used by companies to target particular, smaller groups of consumers with relevant messages that would encourage them to make a purchase. This technique is based on the concept that each and every customer is unique.

In order to more effectively focus their marketing efforts to each segment, business also want to have a deeper understanding of their customers' preferences and needs. In order to divide customers into specific targeting groups, it is necessary to identify significant differentiators that separate them. When determining customer segmentation techniques, factors also with a customer's demographics (age, race, religion, gender, family size, ethnicity, income, and level of education), geography (where they live and work), psychographics (social class, lifestyle, and personal traits), and affective (spending, consumption, usage, and desired benefits) tendencies are taken into account.

The ability to adjust marketing strategies so that they are suitable for each customer category and to support business goals is called customer segmentation. It helps in identifying the items related to each client segment, managing supply and demand for those products, identifying and focused on a

potential customer base, and predicting customer problems. By target specific consumer groups with a customer segmentation strategy, business owners might use its marketing resources more efficiently and increase their chances of cross-selling.

When companies give customized messages to a set of clients as part of a marketing mix suited to their needs, it is simpler for them to identify innovative offers to motivate them to spend more. Motivation of project Segmentation is not a label concept. In 1956, Wendel Smith published the first article on segmentation. In fact, segmentation is an important element of many marketing techniques. A more diverse population's homogeneous groups might be developed, which significantly facilitated the manufacture and marketing of products and services. These divisions were usually classified based on age, gender, income, ethnicity, or other factors. Marketing teams quickly adopted behavioral segmentation. Objective of project The process of segmenting a company's customers into groups that reflect similarity among customers in each group is called as customer segmentation. In order to enhance each customer's value to the company, it is crucial to choose how to connect with each type of customer. Marketing professionals may be able to contact each customer in the most effective method with the help of customer segmentation. A customer segmentation analysis enables marketers to identify different groups of customers with a high level of accuracy based on demographic, behavioral, and other factors using the enormous amount of data on customers (and potential customers) that is available.

INTERNET SOURCES:

<1% - https://www.mysciencework.com/patent/show/large-volume-data-transfer-US9160820B 2
<1% - https://innovationmanagement.se/2013/07/03/5-key-points-to-consider-when-developi ng-an-innovation-strategy/
<1% - https://www.indicative.com/resource/volume-of-data/
1% - https://www.techtarget.com/searchcustomerexperience/definition/customer-segmentati on
<1% - https://fourweekmba.com/customer-segmentation/
<1% - https://marketing-insider.eu/market-segmentation/
<1% - https://www.hult.edu/blog/benefits-challenges-cultural-diversity-workplace/
<1% - https://gocardless.com/guides/posts/what-is-customer-segmentation-analysis/
<1% - https://www.bartleby.com/essay/Proposed-System-Technical-Requirements-Specificatio ns-FKZYXFWKFV85
<1% - https://open.library.okstate.edu/technicalandprofessionalwriting/chapter/chapter-5/ <1% - https://www.acuity.com/the-focus/manufacturer/why-you-need-to-pay-close-attentionto-your-supply-chain
<1% - https://blog.hubspot.com/service/customer-segmentation