

Czech University of Life Sciences Prague

Faculty of Economics and Management

System Engineering and Informatics



Master Thesis

**CUSTOMER DATA MANAGEMENT USING DATA
CLUSTERING**

SRIMATHI RAVISANKAR

© 2022 CULS Prague

CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

DIPLOMA THESIS ASSIGNMENT

SRIMATHI RAVISANKAR

Informatics

Thesis title

Customer data management using data clustering

Objectives of thesis

In many current industries, customers loyalty management becomes a significant part of making a business successful. The diploma thesis objective is to design an algorithm for the replication of purchases within a defined brand. The primary purpose of algorithm design is to reduce the loss of customer loyalty.

Methodology

The theoretical part of literary research is based on the study and analysis of professional literary sources. The acquired knowledge will be used in the design part.

In the application part, customer segments will be analysed using the selected algorithm regarding the probability of switching to another brand. For customers with a high probability of transition, a strategy will be designed to stimulate their current loyalty to the current brand. The effectiveness of the proposed strategy of stimulating brand loyalty will be verified on a selected sample of data.

The proposed extent of the thesis

60-80p.

Keywords

Customer loyalty, Probability of transition, Customer relationship management.

Recommended information sources

BURGER, R. *The mathematical theory of selection, recombination, and mutation*. Chichester; New York:

Wiley, 2000. ISBN 0471986534

FARR, J L. – TIPPINS, N T. *Handbook of employee selection*. New York, London: Taylor & Francis Group,

2010. ISBN 978-0-8058-6437-3.

ROBINSON, S. -ETHERINGTON, L. *Customer loyalty: a guide for time travellers*. Basingstoke [England];

New York: Palgrave Macmillan, 2006. ISBN 9781403997630

selection

SZWARC, P. -EBRARY, INC. *Researching customer satisfaction & loyalty: e-book: how to find out what*

people really think. London: Kogan Page, 2005. ISBN 978-0-7494-4621-5.

Expected date of thesis defence

2021/22 SS – FEM

The Diploma Thesis Supervisor

doc. Ing. Tomáš Macák, Ph.D.

Supervising department

Department of Management

Electronically approved: 27. 10. 2021

prof. Ing. Ivana Tichá, Ph.D.

Head of department

Electronically approved: 23. 11. 2021

Ing. Martin Pelikán, Ph.D.

Dean

Declaration of Honour:

I declare that I have worked on my diploma thesis titled "Customer Data Management using data clustering" by myself and I have used only the sources mentioned at the end of the thesis. As the author of the diploma thesis, I declare that the thesis does not break copyrights of any their person.

In Prague on 25.03.2022

ACKNOWLEDGEMENT

First and foremost, I thank the almighty for showering grace and blessings on me for making this project a great success.

I would like to extend my heartfelt gratitude to Professor **doc. Ing. Tomáš Macák, Ph.D.** for his valuable guidance and timely encouragements. He put all his valuable experience and expertise in directing, suggesting and supporting me throughout the project to bring out of the best.

ABSTRACT AND KEY WORDS

Customer data loyalty management is becoming a more essential source of advantage for businesses as the market becomes more competitive. Existing churn prediction algorithms, however, cannot operate well when dealing with huge data in the sector. Furthermore, decision-makers are constantly confronted with sloppy operational management.

To address these issues, the semantic-driven subtractive clustering method (SDSCM) is presented as a novel clustering methodology.

The Mean shift clustering algorithm is used for further optimizing the clustering of data points. The Marketing strategy is proposed by using the customer relationship management tool (CRM).

The primary purpose of algorithm design is to reduce the loss of customer loyalty. The CRM is used to provide marketing strategy solutions to the prospect who has high risk of leaving the company.

Keywords: Customer loyalty, Probability of transition, Customer relationship management.

ABSTRACT AND KEY WORDS

Řízení loajality zákaznických dat se stává stále důležitějším zdrojem výhod pro podniky, protože trh se stává konkurenceschopnějším. Stávající algoritmy pro predikci odchodu však nemohou dobře fungovat při práci s obrovskými daty v tomto sektoru. Kromě toho jsou osoby s rozhodovací pravomocí neustále konfrontovány s nedbalým provozním řízením.

K vyřešení těchto problémů je prezentována sémanticky řízená metoda subtraktivního shlukování (SDSCM) jako nová metodologie shlukování.

Algoritmus shlukování středního posunu se používá pro další optimalizaci shlukování datových bodů. Marketingová strategie je navržena pomocí nástroje pro řízení vztahů se zákazníky (CRM).

Primárním účelem návrhu algoritmu je snížit ztrátu loajality zákazníků. CRM se používá k poskytování řešení marketingové strategie potenciálním zákazníkům, kteří mají vysoké riziko odchodu ze společnosti.

Keywords: Zákaznická loajalita, Pravděpodobnost přechodu, Řízení vztahů se zákazníky.

Table of content

CHAPTER	TITLE	PAGE
	ABSTRACT AND KEY WORDS	6
	LIST OF SYMBOLS	10
1	INTRODUCTION	11
2	OBJECTIVES AND METHODOLOGY OF THESIS	12
3	LITERATURE REVIEW	13
	3.1 Overview	13
	3.2 Fuzzy clustering analysis based on AFS Theory	13
	3.3 Churn prediction model in Telkom industry	14
	3.4 Map Reduce: Simplified Data processing on large clusters	14
	3.5 Subtractive clustering-based segmentation	15
	3.6 The map reduce model on Hadoop	15
4	PRACTICAL PART	17
	4.1 SYSTEM STUDY	17
	4.1.1 Overview	17
	4.1.2 Purpose	17
	4.1.3 Hardware Requirements	17
	4.1.4 Software Requirements	17
	4.2 SYSTEM DESIGN	30
	4.2.1 Axiomatic Fuzzy Set	30
	4.2.2 Subtractive Clustering Method	31
	4.2.3 Mean shift Clustering Algorithm	34
	4.3 IMPLEMENTATION	35
	4.3.1 Algorithm	35
	4.3.1.1 Subtractive Clustering Method	35
	4.3.1.2 SDSCM	36
	4.3.1.3 Mean Shift Algorithm	37
	4.3.2 CODE	39
5	DISCUSSION OF RESULTS AND RECOMMENDATIONS	50
	5.1 Results	50

	5.2 Marketing Strategy	52
6	CONCLUSION	60
7	REFERENCES	61
	LIST OF FIGURES	63

LIST OF SYMBOLS

Notation	Description
η	Fuzzy concept
x_i	$x_i \in X$ and x_1^F is the first centroid in the parameter determination. x_1^* is the first centroid in SCM.
$\mu_\eta(x_i)$	Membership of x_i
p_i	Sum of absolute difference of $\mu_\eta(x_i)$, and p_1^F is the minimum.
d_i	Euclidean distance between first cluster centroid and other data points.
τ_1	Neighbor radius
τ_2	Weight coefficient
l	Cycle index
M_i	Mountain function and M_l^* is the maximum in the l cycle.
ϵ	Termination Condition

1. INTRODUCTION

To avoid customer churn, these businesses must focus more on existing consumers due to the strong competition. Customer churn is defined as the loss of customers who switch from one company to another within a specified timeframe. Customer churn can result in significant financial losses and can even harm a company's reputation, according to industry experience. Customer churn can be calculated using existing models. They are, nevertheless, appropriate for little structured data, such as account data and call details, which all have fewer than ten thousand items. Companies have acquired unprecedented volumes of data sources because of the widespread adoption of smart phones and the expansion of mobile internet. The massive volume of data has the characteristics of big data, and hence is referred to as "telco big data," which includes call detailed records, Internet traffic logs, user profiles, location updates, social networking information, and so on. The sample of telecommunication data set is studied in this project.

2. OBJECTIVES AND METHODOLOGY OF THESIS

2.1 Objectives:

The work aims to design an algorithm to analyze customer loyalty in the telecommunication sector and to provide a strategy plan to avoid the customer leaving the company's service.

2.2 Methodology:

Cluster analysis is a common data mining technique for finding intriguing patterns in data, such as consumer groupings based on their behavior. There are numerous clustering algorithms to choose from, and there is no single ideal clustering technique for every situation. When applying a clustering algorithm to our data, we can utilize clustering analysis to acquire some key insights from our data by seeing what categories the data points fit into.

The Semantic Driven Subtractive Clustering Method technique will be implemented as part of my project on Telkom data set. SDSCM outperforms subtractive clustering method (SCM) and fuzzy c-means in terms of semantic clustering strength (FCM). The map reduce framework in the Hadoop environment is used to implement this technique. The cluster centroids are obtained in this manner. The centroids are then fed into the Mean shift clustering algorithm, which produces clusters. Then, marketing strategy is provided to high-risk customer by using CRM (Customer Relationship Management) tool in order to avoid the prospect to leave the company.

3. LITERATURE REVIEW

3.1 Overview:

Data management refers to a set of procedures for handling data that a corporation collects or creates in order to make educated business decisions. The essential concept behind the entire procedure is to treat data like a valuable asset.

Data management, when precisely managed, reduces data transfer, aids in the discovery of performance flaws and allows users to have all relevant information at their fingertips. With data management in place, a company may minimize unneeded duplications, and staff will not have to repeat the same research or activities. The ability of an organization to make the appropriate decisions swiftly in the face of change is critical to its success. The business is likely to lose money and miss chances if it takes too long to react to market movements or rival activities. Decision-makers can get crucial information faster and respond correctly with organized data. The more high-quality data you have, the clearer the picture becomes and the better judgments you can make. Lack of knowledge or flaws in available data, on the other hand, might lead to deadly business mistakes.

The existing AFS and SCM algorithms have been discussed in this section. Many contemporary clustering systems rely solely on data points and the fuzzy idea. The data points were discovered by intuition in traditional algorithms. Assumptions are used to implement the fuzzy idea. For clustering, Mean shift clustering algorithm is utilized.

3.2 Fuzzy clustering analysis based on AFS Theory: [10]

The membership functions in current fuzzy theories are frequently given by personal intuition, and the logic operations are performed using a type of triangle norms, or t-norm, which is determined ahead of time and is independent of the original data and facts [1]. In real-world applications, large-scale intelligent systems are typically very large and complex, containing such a large number of concepts that defining membership functions by personal intuition and selecting a suitable triangular norm from an infinite number of triangular norms to implement fuzzy logic systems is impossible. In this study, I have implemented a new algorithmic framework based on AFS theory for constructing fuzzy sets (membership functions) and associated logic operations, which are derived impersonally and automatically by a consistent algorithm based on the original data and facts. The examples will assist us in identifying and emphasizing the key benefits of the new algorithmic framework. AFS structures and AFS algebra determine the membership functions and related logic operations in AFS

theory. An AFS structure is a triple mathematical object, which is a special family of combinatory objects, in which X is the universe of discourse, M is a set of some simple concepts or attributes on X , and t is a mathematical abstract of the complex relations containing in the original data and facts, such as databases, sub preference relations, and even descriptions of human intuition. The AFS algebra is a set of molecular lattices (totally distributive lattices) produced from sets like X , M . The suggested approach in this research may obtain a large number of complicated fuzzy ideas on X and associated logic operations using AFS algebra and AFS structure. The membership functions and associated logic operations are more accurate and impersonal reflections of the original data and facts than other fuzzy theories' human intuitions, and they maintain the information contained in the original data and facts to a large extent.

3.3 Churn prediction model in telecom industry:

In modern tele-communication CRM systems, customer churn prediction is a key component [2]. Unlike most studies, this work attempts to divide customers into clusters based on the weight assigned by the boosting algorithm. As a result, a client cluster with a higher risk has been discovered. In this study, the basic learner is logistic regression and a churn prediction model is created on each cluster separately. A single logistic regression model is used to compare the results. Boosting also gives a good separation of churn data, according to experimental evaluation; hence, boosting is recommended for churn prediction analysis.

3.4 Map Reduce: Simplified Data processing on large clusters

MapReduce is a programming methodology for processing and creating huge data collections, as well as an implementation. A map function processes a key/value pair to yield a set of intermediate key/value pairs, while a reduce function merges all intermediate values associated with the same intermediate key. As illustrated in the research, this model can convey a wide range of real-world tasks. This functional programming language automatically parallelizes and executes programs on a huge cluster of commodity devices. The run-time system is in charge of splitting the input data, scheduling the program's execution over a group of machines, dealing with machine failures, and coordinating the required inter-machine

communication. This enables programmers with little or no experience with parallel and distributed systems to take advantage of a big distributed system's resources. Our MapReduce implementation is very scalable and runs on a huge cluster of commodity servers. A typical MapReduce calculation processes many terabytes of data over thousands of machines. Hundreds of MapReduce programs have been created, and upwards of one thousand MapReduce jobs are executed every day on Google's clusters.

3.5 Subtractive clustering-based segmentation

The subtractive clustering method is a refinement of the mountain clustering approach. Mountain clustering divides the data space into small sections using a grid function; at each grid point, a potential function known as the mountain function is generated, and grid points with greater values of this potential function are accepted as cluster centres [4].

Generally dense locations in the data space can be considered as cluster centres, if a grid point has the highest mountain function value, it signifies that there are more data points around it than the other grid points. The grid function's spatial resolution has an impact on the precision with which the exact cluster centres are defined. However, depending on the scale of the data, increasing the spatial resolution to generate finer grids will increase the cost of processing dramatically.

3.6 The map reduce model on Hadoop

MapReduce is a parallel programming framework for handling big data sets across multiple machines [5]. In a nutshell, a MapReduce job comprises three stages: map, shuffle, and reduce. The shuffle phase of a MapReduce operation typically has a high-performance overhead and is often a bottleneck in practice. Sequential and parallel computations are combined in the MapReduce model. The reduce tasks cannot begin until all map tasks have been completed, despite the fact that tasks in the map and reduce phases are running concurrently. In addition, because all map jobs are independent of one another and have no communication, tasks should be reduced.

Multiple MapReduce cycles are necessary to implement several algorithms due to their iterative nature. Data is spread among all machines in each cycle, and each machine processes its own input independently.

These machines' output is either the final result or another round's input. Iterative programs aren't supported by MapReduce-based systems. Because each additional task incurs significant running time cost due to synchronization, communication, and congestion difficulties every round, the number of MapReduce rounds is an important measure to be improved in MapReduce.

4. PRACTICAL PART

4.1 SYSTEM STUDY:

4.1.1 Overview:

The system study comprises a view of the project's objective and business context, as well as the project's major functional needs and system requirements.

4.1.2 Purpose:

The entire flow of our suggested system is discussed in this section. The proposed system explains how to use the SDSCM algorithm to cluster data points. Original data and facts, rather than intuition, determine the membership functions and associated logic operations. Furthermore, AFS algebra and structure can be used to represent many complicated notions utilizing a few simple concepts or attributes.

4.1.3 Hardware Requirements:

- Operating System : Mac
- RAM : 8 GB

4.1.4 Software Requirements

- Framework : Hadoop-3.2.1 , Python and Matplotlib.
- Software: Eclipse, Jupyter Notebook, Customer Relationship Management (CRM) tool.

A. Hadoop:

Hadoop is a distributed data processing system that allows you to process enormous amounts of data. It's made to help in scaling. It has a lot of processing power and storage. It has a large library for detecting exceptions and failures in the application layer. Hadoop can be deployed as a single node cluster or as a multimode cluster. In my project, I employed a single node cluster. The following modules comprise a Hadoop project:

- **Hadoop Common:** The Hadoop modules' common utilities.
- **HDFS (Hadoop Distributed File System):** HDFS is a distributed file system that allows users to access application data quickly.
- **Hadoop YARN:** It is a framework for managing cluster resources and scheduling jobs.
- **Hadoop Map Reduce:** It is a YARN-based technology that allows massive data sets to be processed in parallel.
 - For distributed storage and computing, Hadoop employs a master-slave architecture. Hadoop runs the entire process as a map-reduce program. Job is the Hadoop term for the execution of a task. The user must provide the following information before the Hadoop framework conducts the task:
 - The input and output files locations in the distributed file system
 - The map and reduce functions are in these classes.

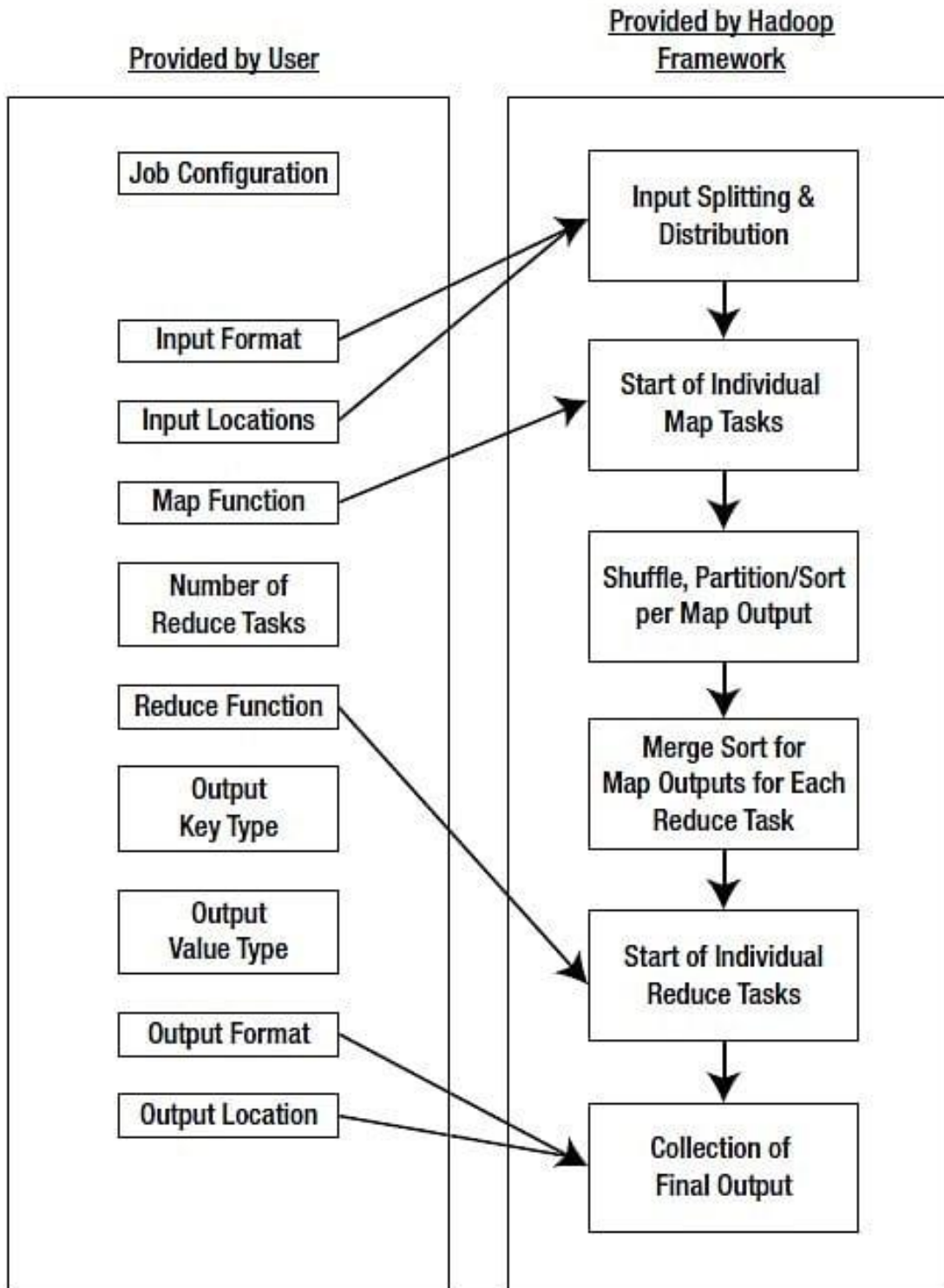


Figure 1: Different Task in Map Reduce program [20]

The diagram above depicts the parameters and modules that can be set up during a MapReduce operation.

The following are the numerous tasks that are required to run a map reduction program:

- The job configuration is specified by the user by adjusting several work-specific parameters.
- The number of reducer tasks, as well as the reduce function, are also specified by the user.
- In addition, the user must specify the input format as well as the input locations.
- This information is used by the Hadoop framework to divide the input into many components.
- Each piece of data is supplied into a map function that the user has created.
- The map tasks process and emit intermediate data based on the supplied data.
- The map phase's output is sorted, and the intermediate data can be partitioned with a default or custom partitioning.
- The reduction function merges intermediate values or executes a user-specified function after processing the data in each partition.
- The map and reduce functions need the user to provide the types of output key and output value.
- The Hadoop framework collects the output of the reduce function and saves it to disk in output files.

Steps of Commands used for running Map reduce program:

- ssh localhost
- ssh-keygen -t rsa -P ""
- cat /Users/hadoop/.ssh/id_rsa.pub >> /Users/hadoop/.ssh/authorized_keys
- hadoop\$tar -xzvf hadoop-*
- bin/hdfs namenode -format //remove namenode

- `rm -r /tmp/hadoop-hadoop/dfs/data/current` // to remove the data node
- *Upload input file:* In localhost:9870
- *To add folder:* `bin/hdfs dfs -mkdir /user`
- `bin/hadoop jar WordCount.jar WordCount /user /op2`
- `bin/hdfs dfs -cat /op2/part-r-00000`
- *Inside sbin folder:* `./start-all.sh ./stop-all.sh`

B. Python:

Python is a high-level programming language that may be used for a variety of applications. Its design philosophy places a strong emphasis on code readability, with a lot of indentation. Its language elements and object-oriented approach are designed to assist programmers in writing clear, logical code for both small and big projects.

Python is garbage-collected and typed dynamically. It supports a variety of programming paradigms, including procedural, object-oriented, and functional programming. Because of its extensive standard library, it's often called a "batteries included" language.

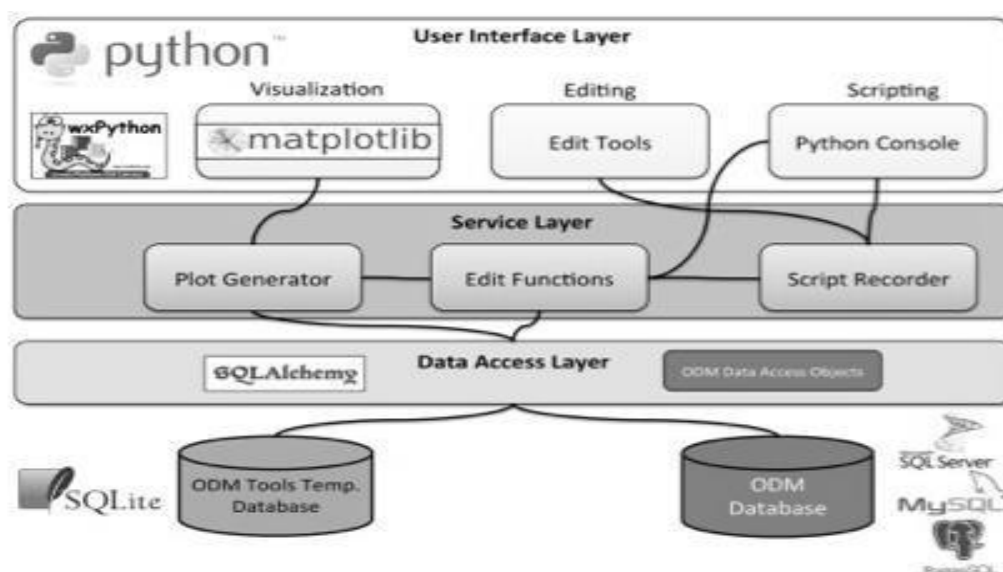


Figure 2: Python Architecture [15]

The diagram above depicts different components in python architecture.

C. Matplotlib: Visualization with Python:

- Matplotlib is a Python toolkit that allows you to create static, animated, and interactive graphs. Matplotlib enables both easy and difficult tasks.
- It is a graphing library for Python and its NumPy numerical math extension. It provides an object-oriented API for integrating plots into programs written in general-purpose GUI toolkits. A procedural "pylab" interface based on a state machine is also available. Matplotlib is utilized by SciPy.

Architecture:

Matplotlib's architecture is also tiered, having three layers. The scripting layer, artist layer, and backend layer are the three layers.

- *Scripting Layer:*

The Scripting Layer serves as an interface, allowing users to use matplotlib via the Python plotting API. The `plot()`, `title()`, `savefig()`, `draw()`, `figure()`, `switch backend()` and other functions are exposed.

- *Artist Layer*

Lines, Rectangles, Polygons, Axis, and other rendered objects are managed by the Artist Layer. It does not, however, render them; instead, it provides helpful abstractions for rendered objects.

- *Backend Layer:*

The actual rendering of the objects is done by the Backend Layer. The renderer object mentioned before is in charge of rendering primitive forms with its own set of functions like `draw path()`, `draw image()`, and so on.

D. Jupyter Notebook:

The Jupyter Notebook is an open-source web tool that lets data scientists create and share documents that include code, equations, computational output, visualizations, and other multimedia elements, as well as explanatory text.

It can be used for a wide range of data science tasks, including data cleansing and transformation, numerical simulation, exploratory data analysis, data visualization etc.

It is an easy-to-use, interactive data science environment that may be used as an IDE, as well as a presentation or educational tool. Jupyter is a virtual "notebook" that allows you to work with Python. It's becoming increasingly popular among data scientists due to its versatility.

It allows you to integrate code, photos, charts, comments, and other elements in accordance with each step of the "data science process." It is also a type of interactive computing, in which users run code, observe the results, adjust, and repeat in an iterative interaction between the programmers and the data.

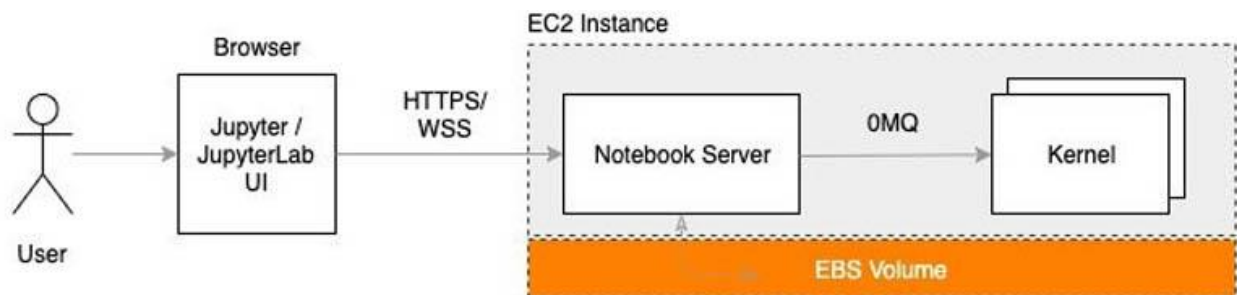


Figure 3: Notebook Architecture [12]

The diagram depicts how the user can access the jupyter notebook that is set up in amazon EC2 cloud.

E. CRM:

CRM (Customer Relationship Management) is a business technique for managing contacts with customers and future customers. CRM aids businesses in streamlining procedures, establishing customer relationships, increasing sales, improving customer service, and boosting profits.

The purpose is to strengthen customer service relationships, increase client retention, and increase sales. CRM systems collect consumer information through a variety of channels, or points of contact, between the customer and the firm, such as the company's website, phone, live chat, direct mail, marketing materials, and social media. CRM systems can also provide detailed information on a customer's personal information, purchase history, purchasing preferences, and issues to customer-facing employees.

CRM systems may assist businesses of all sizes, from tiny firms to multinational conglomerates, by:

- Customer information such as previous purchases and interaction history should be readily available to help customer care agents provide better and faster service.
- Through reporting and visualization options, businesses may use customer data to find trends and insights about their customers.
- Sales funnel and customer support procedures that are tedious but vital can be automated.

Customer acquisition and retention are the ultimate goals of CRM. This will be the foundation upon which your customer strategy will be built. Providing experiences that keep your consumers coming back is one way to improve customer

acquisition and retention. CRM is the cornerstone of such experiences, both as a strategy and as a tool.

Customer data can also be used to populate commission modelling, sales forecasting, territory segmentation, campaign design, and product innovation, as well as other sales, marketing, and service operations, all of which can help improve customer acquisition, retention, and revenue generation.

Customer management software and solutions assist you in streamlining the customer interaction process, developing strong customer connections, increasing customer loyalty, and ultimately increasing sales and profitability.

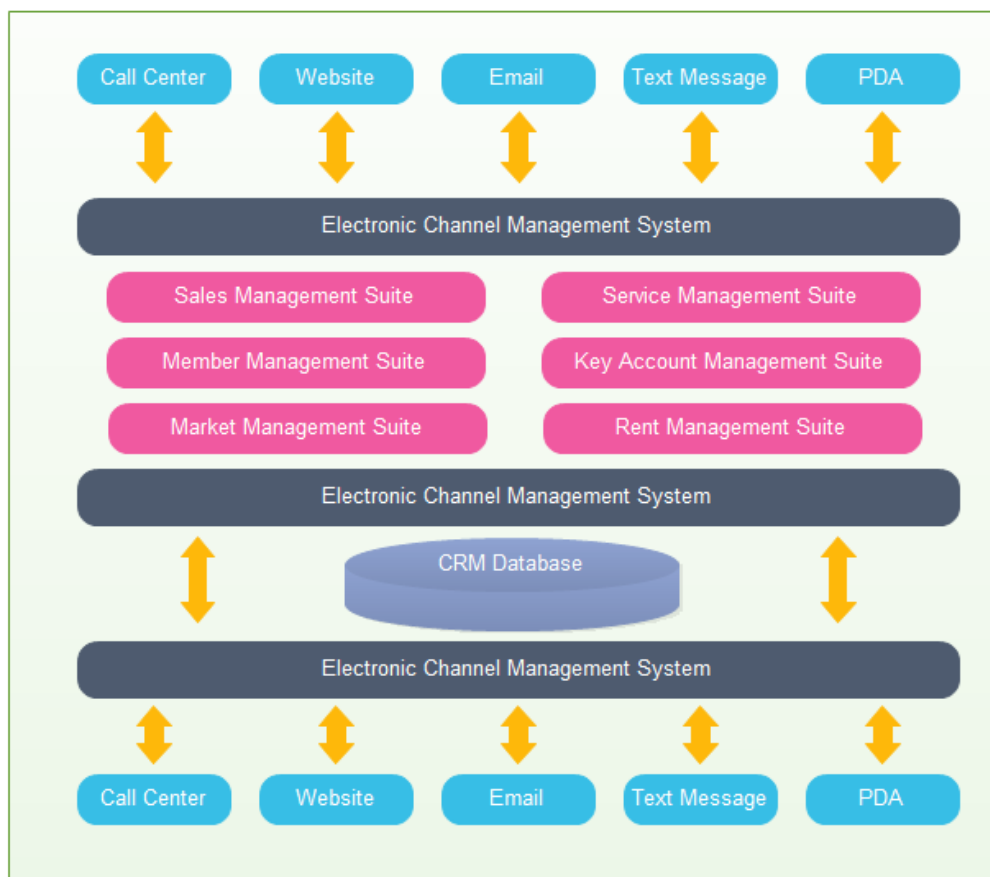


Figure 4: CRM Architecture [13]

The above diagram depicts the various components of CRM architecture.

Consider the difficulties that a CRM tries to answer when determining if your firm will benefit from one:

- All of your consumer data should be kept in one place. You can utilize a CRM platform to provide a single, up-to-date point of truth that everyone can access instead of transmitting information to various divisions.
- It's simple to maintain track of earlier encounters with a CRM if customers often contact with multiple personnel from your firm.
- Your sales teams' productivity can be tracked using a CRM. It can also aid in the creation of a workflow or process.

i. Contacts:

The term contact is used in most CRM systems to refer to a person who has purchased your product or service or a company representative who is in charge of purchasing. The difference between a contact and a lead is that a lead is a potential client, but a contact is usually a current one. Please keep in mind that the terms "contact" and "lead" are used differently in telemarketing than they are in standard CRM systems. The term "contact" refers to a cold lead, whereas "leads" refers to people who have expressed an interest in trying or purchasing the product (warm leads).

Contacts can be imported into CRM, manually entered, or converted from leads who have decided to buy the product. Contacts are frequently linked to businesses (accounts) and transactions (opportunities). A contact is an individual, while a customer is the party who makes the purchase. A customer is always generated, and most of the time, contact is created as well.

In a Business - to - business transaction, the "purchasing party" is always a corporate name, and the contacts are the company's workers. In a Business to consumer transaction, purchasing party could be a single person, a family, or another

form of business. Your primary contact and billing information should always be saved in the customers app in both circumstances. The contacts app allows you to save information about a person, such as their phone number, home location, and personal hobbies and interests.

ii. Leads:

A lead is a person or company that has the potential to become a client or customer. Advertising, direct marketing, networking, outbound calls, website enquiries, email marketing, and social media marketing are just a few of the strategies used by businesses to create leads. When a salesperson qualifies and enters lead information into a company's sales pipeline, the sales process begins. He then communicates with the lead via email, phone, or physical visit to learn more about his needs, informs the potential customer about his product or service, and finally persuades him to purchase the product or service. Depending on numerous elements such as the lead's decision-making process, buying requirement and urgency, and so on, the lead-to-customer conversion process may occur quickly or take days, weeks, or even months.

Lead management encompasses the complete process of identifying and creating leads, as well as pursuing and completing a purchase. A lead is the initial step in the sales process. It denotes a customer who has expressed an interest in purchasing a service or a product.

The following order is followed in a sales cycle:

- Lead
- Opportunity
- Quote
- Order

A lead can be generated through a variety of marketing lead generation procedures such as trade shows, advertising, direct marketing, or other personal sales activities such as telemarketing or email. The marketing staff can sort leads into three categories: cold, warm, and hot. If a lead appears to be promising, the sales department can turn it into an opportunity.

In the sense that it is a precursor to an opportunity, a lead is distinct from an opportunity. For more complicated and longer sales project cycles, opportunities are used. The sales reps take advantage of opportunities to manage the sales project and increase their chances of obtaining customers while reducing sales time.

iii. Customer:

A customer is a person or company who are the leads (who are interested) and already member of organization who has got the service.

iv. Opportunity:

Opportunities in some CRM systems are a technique of managing a business unit rather than a person or a firm. It's the deal you think you'll make, and it's there to keep track of the specifics of the potential sale. And this is where your team's sales abilities will really shine.

A sales prospect, a requested service or product, a sales volume, and a sales likelihood are all terms used to describe an opportunity. A bid invitation, a sales transaction, or a trade show can all lead to the prospect of selling a service or product. One of the key features of the CRM module is opportunity management, which helps you to manage the sales process.

In the following situations, sales opportunity management should be used:

- When a company's sales cycle is longer.
- When a firm employs a large number of sales reps.
- When a significant number of sales orders are received and distributed.

- *Categories of opportunity:*

a. Sources of Opportunity:

In a sales cycle, this allows you to define the source of opportunities. You can highlight predicted sales volume from a variety of sources, such as a trade show, a bid invitation, and so on.

b. Opportunity Group:

You can now execute opportunity grouping. You can create separate groups for new and current clients.

c. Priority:

If the opportunity has a chance of converting to a lead, you may also assign it a priority.

v. Sales:

Throughout the sales process, a sales CRM is a tool for managing all touchpoints with prospects and customers. Any encounter between sales agents and their leads, whether direct or indirect, is a touchpoint. The program keeps track of prospect communication, organizes customer data automatically, provides follow-up reminders, and much more.

Sales CRM software packages essentially make the lead nurturing process easier. Administrative tasks and data management are both automated, allowing you to spend less time entering data and more time creating meaningful interactions with prospects.

Salespeople spend their days balancing everything from prospecting and cold calls to deal management and field sales. Managers are also responsible for evaluating the success of their sales reps, performing ride-along and coaching their team. Things can rapidly become dysfunctional in your department without a centralized framework for managing these day-to-day processes.

Sales CRMs are intended to be one-stop shops for managing day-to-day tasks. They can connect to your existing tools and communication channels, allowing you to manage all of your responsibilities from a single, centralized platform. This data centralization results in a single source of truth for your entire company.

Sales staff can quickly access the most precise, up-to-date data they need to appropriately follow up with leads or close sales. They can also communicate with prospects via email, phone, or the website's chat box—all without switching between apps or browser tabs.

4.2 SYSTEM DESIGN:

4.2.1 Axiomatic Fuzzy Set: [10]

AFS is an good method to express the fuzzy concept. Rather than intuition, original data and facts are used to determine the membership functions and logic operations. Furthermore, utilizing AFS algebra and AFS structure, a few simple concepts or attributes can describe a variety of complicated concepts.

The EI algebra and EI 2 algebra are linked by the AFS structure. A completely new system of fuzzy sets and systems has been built by utilizing these ancient mathematical objects. For traditional mathematical viewpoints, it is more acceptable. Many traditional mathematical results are immediately applicable. The system can be controlled by computers, which is the most crucial aspect. The fact that any element in the EI algebra over a set M corresponds to a unique fuzzy set from a universe X to the perfectly distributive molecular lattice EI 3 algebra over X, M, and M will be demonstrated in the following section. Any element of the EI algebra over a set M corresponds to a single fuzzy set from universe X to $w \times 0, 1$.

The membership of items in a set is evaluated in binary terms according to a bivalent condition in classical set theory – an element either belongs to the set or does not belong to the set. Fuzzy set theory, on the other hand, allows for a gradual assessment of the membership of elements in a set, which is characterized using a membership function having a value in the real unit interval $[0, 1]$. Because the indicator functions of classical sets are special instances of the membership functions of fuzzy sets, if the latter only accept values 0 or 1, fuzzy sets generalize classical sets. Crisp sets are commonly used in fuzzy set theory to refer to classical bivalent sets. The fuzzy set theory can be used to a wide range of fields, including bioinformatics, when information is incomplete or inaccurate.

4.2.2 Subtractive Clustering Method: [10]

A. Mountain Method:

In the original mountain method, proposed by Yager and Filev (1994), a grid is created in the data space, and then a potential function, referred to as the mountain function, is calculated on each grid point. The grid points with higher mountain values are selected as the cluster centroids.

Let us consider an unlabelled data set $X = \{x_1, \dots, x_n\}$ in the p -dimensional space R^p . Let x_{jk} be the k -th coordinate of the j -th data point for $1 \leq j \leq n$ and $1 \leq k \leq p$. The p -dimensional space R^p is restricted to a p -dimensional hypercube $I_1 \times I_2 \times \dots \times I_p$ where the intervals I_k , $1 \leq k \leq p$ are defined by the ranges of the coordinates x_{jk} . Obviously, the hypercube contains the data set X . Then the intervals I_k are subdivided into r_k equidistant points. This discretization forms a p -dimensional grid in the hypercube with grid points v_i for $1 \leq i \leq N$ where $N (= r_1 \times \dots \times r_p)$ is the number of grid points.

Let $d(v_i, x_j)^2 = \|v_i - x_j\|^2$ be the square of distance between a grid point v_i and a data point x_j . Of the distance measures proposed to date, the Euclidean distance is the most widely used (Bezdek *et al.*, 1999; Yager and Filev, 1994). The mountain function at a grid point v_i is defined as

$$M(v_i) = \sum_{j=1}^n e^{-\alpha \|v_i - x_j\|^2} \quad (1) \text{ Source: [10]}$$

where α is a positive constant. A higher value of the mountain function indicates that v_i has more data points x_j in its vicinity. Thus, it is reasonable to select a v_i with a high value of the mountain function $M(v_i)$ as a cluster centroid.

After calculating the mountain function for each grid point, the cluster centroids are selected by destroying the mountains. Let M_1^* be the maximum value of the mountain function:

$$M_1^* = \text{Max}_i [M(v_i)] \quad (2) \text{ Source: [10]}$$

and let v_1^* be the grid point whose mountain value is M_1^* . Then v_1^* is selected as the first cluster centroid. To find other cluster centroids, we must first eliminate the effects of

the cluster centroids that have already been identified. To achieve this, a value inversely proportional to the distance of the grid point from the found centroids is subtracted from the previous mountain function; this process is carried out using the equation:

$$\widehat{M}^j(v_i) = \widehat{M}^{j-1}(v_i) - M_{j-1}^* \sum_{j=1}^n e^{-\beta \|v_i - v_{j-1}^*\|^2} \quad (3) \text{ Source: [10]}$$

where \widehat{M}^j is the new mountain function, \widehat{M}^{j-1} is the old mountain function, M_{j-1}^* is the maximum value of \widehat{M}^{j-1} , v_{j-1}^* is the newly found centroid, and β is a positive constant. We see that the mountain values of grid points closer to the newly found centroid are decreased to a much greater extent than those further away. Thus, the procedure to approximate the cluster centroids is as follows:

B. Subtractive Method

The clustering performance of the mountain method strongly depends on the grid resolution, with finer grids giving better performance. As the grid resolution is increased, however, the method becomes computationally expensive. Moreover, the mountain method becomes computationally inefficient when applied to high dimensional data because the number of grid points required increases exponentially with the dimension of data.

Chiu suggested an improved version of the mountain method, referred to as the subtractive method, in which each data point is considered as a potential cluster centroid (Chiu, 1995). Under this method, the mountain function is calculated on data points rather than grid points. The computational load of this method presumably still increases with increasing dimension of data, just not at the same rate for the original mountain method.

The mountain function at a data point x_i is defined as

$$M(x_i) = \sum_{j=1}^n e^{-\alpha \|x_i - x_j\|^2} \quad (4) \text{ Source: [10]}$$

where α is a positive constant and $\|x_i - x_j\|^2$ is the square of distance between x_i and x_j . Using this mountain function, cluster centroids are selected in a manner similar to that used in the original mountain method. Let M_1^* be the maximum value of the mountain function

$$M_1^* = \text{Max}_i [M(x_i)] \quad (5) \text{ Source: [10]}$$

and let x_i^* be the data point whose mountain value is M_1^* ; this data point is selected as the first cluster centroid. The modified mountain function used to find subsequent cluster centroids is defined as

$$\widehat{M}^j(x_i) = \widehat{M}^{j-1}(x_i) - M_{j-1}^* \sum_{j=1}^n e^{-\beta \|x_i - x_{j-1}^*\|^2} \quad (6) \text{ Source: [10]}$$

where x_{j-1}^* is the newly found centroid and β is a positive constant. The procedure for the subtractive method is similar to that of the mountain method except for the interval and the grid being eliminated. The mountain and subtractive methods can both be used either as (1) stand-alone clustering methods by assigning each data point to a specific cluster based on the distances between the data point and the centroids or (2) supporting tools to estimate the initial cluster centroids for other clustering methods.

In SCM, each data point is considered as a potential cluster centroid and the potential is computed as

$$M_l(x_i) = \sum_{j=1}^n \exp\left(-\frac{\|x_i - x_j\|^2}{(\tau_1/2)^2}\right) \quad (7) \text{ Source: [10]}$$

where τ_1 is the neighbour radius, which influences the scope of a cluster centroid. The larger the τ_1 is, the greater its impact will be. Thus, the data point with maximum mountain function is the first centroid. Then update the mountain function of each data according to the following equation:

$$M_l(x_i) = M_l(x_i) - M_{l-1}^* \exp\left(-\frac{\|x_i - x_l^*\|^2}{(\tau_2/2)^2}\right) \quad (8) \text{ Source: [10]}$$

where τ_2 is the influencing weight of the last cluster centroid. Data points near the first cluster centroid will have greatly reduced potential and thus unlikely to be the next cluster centroid.

4.2.3 Mean Shift Algorithm:

The mean-shift algorithm assigns datapoints to clusters iteratively by shifting points to the cluster centroid, which has the largest density of datapoints. The difference between the K-Means and Mean-Shift algorithms is that with the latter, the number of clusters is decided by the algorithm based on the data.

The Mean-Shift clustering algorithm has the following benefits:

- It is not necessary to make any model assumptions, as is the case with K-means or Gaussian mixture.
- It can also model nonconvex complex clusters.
- It simply requires one parameter, bandwidth, to determine the number of clusters automatically.
- As opposed to K-means, there are no local minima.

4.3 IMPLEMENTATION

4.3.1 Algorithm:

4.3.1.1 Subtractive Clustering Method: [10]

Step 1:

Initialize the parameters α, β and the intervals $I_k, 1 \leq k \leq p$.

Step 2:

Quantize the intervals and determine the grid.

Step 3:

Compute the mountain functions $M(v_i)$ for each $v_i, 1 \leq i \leq n$.

Step 4:

Choose the grid point v_i for which $M(v_i)$ is highest as a cluster centroid.

Step 5:

Destroy and recompute the mountain function.

Step 6:

If the number of centroids found is equal to the pre-specified number of clusters, then stop; otherwise go to Step 4.

4.3.1.2 Semantic Driven Subtractive Clustering method (SDSCM): [10]

Step 1:

According to the fuzzy concept η given by the user, compute the membership of x_i .

$$\mu_\eta(x) = \text{SUP}(m(\eta) / m(X)) \quad \forall x \in X \quad (9) \text{ Source: [10]}$$

Step 2:

Compute the sum of absolute differences of $\mu_\eta(x_i)$.

$$p_i = \sum_k^n |\mu_\eta(x_i) - \mu_\eta(x_k)| \quad (10) \text{ Source: [10]}$$

Step 3:

Select the minimum value of p_i as the first cluster centroid

$$p_i^F = \min\{p_i\}$$

$$x_i^F = x_i$$

Step 4:

1. Compute the Euclidean distance between the first cluster centroid and other data points.

$$d_i = |x_i - x_1^F|^2 \quad (11) \text{ Source: [10]}$$

2. Average of d_i , $\bar{d} = \frac{1}{n} \sum_{i=1}^n d_i$

3. Calculate the neighbour radius τ_1

$$\tau_1 = \frac{1}{n} \sum_{i=1}^n (d_i - \bar{d}) \quad (12) \text{ Source: [10]}$$

Step 5:

Set the weight coefficient τ_2 .

$$\tau_2 = 1.5\tau_1 \quad (13) \text{ Source: [10]}$$

Step 6:

Let $l=1$ and compute the mountain function of x_i .

$$M_l(x_i) = \sum_{j=1}^n \exp\left(-\frac{\|x_i - x_j\|^2}{(\tau_1/2)^2}\right) \quad (14) \text{ Source: [10]}$$

Step 7:

Select the maximum mountain function

$$M_l^* = \text{Max}_i[M_l(x_i)] \quad (15) \text{ Source: [10]}$$

Make x_i the first cluster centroid

$$x_1^* = x_i$$

Step 8:

Let $l=l+1$, and update the mountain function of each data vector

$$M_l(x_i) = M_l(x_i) - M_{l-1}^* \exp\left(-\frac{\|x_i - x_l^*\|^2}{(\tau_2/2)^2}\right) \quad (16) \text{ Source: [10]}$$

Step 9:

Select the data associated with larger $M_l(x_i)$ to be the second centroid and execute *Step 6* repeatedly until

$$M_l^* < \epsilon M_1^* \quad (17) \text{ Source: [10]}$$

is satisfied. ϵ is a positive constant less than 1. When the ratio is smaller than ϵ , the iteration stops.

- It can fully express semantic signification of fuzzy concept.
- It can automatically determine the neighbour radius and the weight coefficient of SCM.
- It sets a termination condition based on an earlier study reasonably.

4.3.1.3 Mean Shift Algorithm:

- i. Create centroids for all data points.
- ii. Take the average of all feature sets inside the radius of the centroid and use that as the new centroid.
- iii. Step ii should be repeated until convergence is achieved.
- iv. Start by iterating through each centroid, looking for all feature sets that are within range.
- v. Take the average and use it to determine the "new centroid."
- vi. Make a unique variable that keeps track of a sorted list of all known centroids.
- vii. Measure movement by comparing the old centroids with the new ones. Set the centroids attribute to the final centroids if full convergence and optimization is satisfied.

4.3.2 CODING

Step 1:

```
import java.io.IOException;
import java.util.*;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
public class Mu1 {
    public static class TokenizerMapper extends Mapper<Object, Text, Text, FloatWritable>{
        private static int count=0;
        /*
input: dataset
*/
        public void map(Object key, Text value, Context context) throws IOException,
InterruptedException {
            String line = value.toString();
            String[] w=line.split(" ");
            int a=0,b=0,c=0;
            int s=0;
            Integer a1,b1,c1;
            a1=Integer.valueOf(w[0]);
            b1=Integer.valueOf(w[2]);
            c1=Integer.valueOf(w[4]);
            a=a1.intValue();
            b=b1.intValue();
            c=c1.intValue();
            s=a+b+c;
            Integer s1=new Integer(s);
            float sf=s1.floatValue();
            count++;
            context.write(new Text(String.valueOf(sf+count)), new FloatWritable((float) count));
        }
    }
}
/*
output:
key: sum+ number
value: number
*/
public static class IntSumReducer extends Reducer<Text,FloatWritable,Text,FloatWritable>
{
    public static int c=24;
    public static float c1=0;
```

```

    public void reduce(Text key, Iterable<FloatWritable> values,Context context) throws
IOException, InterruptedException {
        c--;
        c1++;
        FloatWritable cf=new FloatWritable((float)(c/24.0f));
        // to calculate the m(n) lowest value
        String kw=key.toString();
        String[] w=kw.split(" ");
        float k1=Float.valueOf(w[0]);
        k1=k1-c1;
        context.write(new Text(String.valueOf(k1)),cf);
    }
}
/*
output:
key: xsum
value: mu
*/

```

```

public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "Mu");
    job.setJarByClass(Mu1.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(FloatWritable.class);
    FileInputFormat.addInputPath(job,new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}

```

Step 2 to Step 5:

```

import java.io.IOException;
import java.util.*;
import java.lang.*;
import java.util.*;
import java.io.RandomAccessFile;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
public class Tou1 {

```



```

public static class TokenizerMapper extends Mapper<LongWritable,Text, IntWritable,
FloatWritable>{
    public static float di=0.0f;
    public static int index=0;
/*
input:
key: xsum
value:mu
*/
    public void map(LongWritable key, Text value, Context context) throws
IOException, InterruptedException {
        String a,b;
        float min=10000.0f;
        float s1=0.0f;
        int cluster=0;

        RandomAccessFile areader=new
RandomAccessFile("/home/hadoop/hadoop/project/tou/mu.txt","r");
        RandomAccessFile breader=new
RandomAccessFile("/home/hadoop/hadoop/project/tou/mu.txt","r");
        while(!(a=areader.readLine()).equals(null))
        {
            float m=Float.valueOf(a);
            while(!(b=breader.readLine()).equals(null))
            {
                float mn=Float.valueOf(b);
                float dd=Math.abs(m-mn);
                s1=s1+dd;//find the value of p
                index++;//cluster centroid id
                if(min>s1)
                {
                    min=s1; // (pi(F))
                    cluster=index;// (xi(F)) first cluster centroid
                }
            }
        }
        breader.seek(0);
    }
    }//areader
    breader.close();
    areader.close();
    float min1;
    // to calculate di
    RandomAccessFile areader1=new
RandomAccessFile("/home/hadoop/hadoop/project/tou/xsum.txt","r");
    areader1.seek((int)min);
    min1=Float.valueOf(areader1.readLine());
    areader1.seek(0);
    while(!((a=areader1.readLine()).equals(null)))
    {
        float mn=(Float.valueOf(a));
        di=(mn-min1)*(mn-min1);
    }
}

```

```

        context.write(new IntWritable(1), new FloatWritable(di));
    }
    areader1.close();
} //map
} //mapper

/*
mapper output:
key: added sum of all columns
value: di
*/

public static class IntSumReducer extends
Reducer<IntWritable,FloatWritable,IntWritable,FloatWritable> {
    public static float dbar=0.0f;
    public static int count=24;
    public static float tou1=0.0f;
    public static float tou2=0.0f;
    LinkedList db = new LinkedList();
    public void reduce(IntWritable key,Iterable<FloatWritable> values,Context context) throws
IOException, InterruptedException {
        float r=0.0f;
        for (FloatWritable value : values)
        {
            float s=value.get();
            dbar+=s;//dbar
            db.add(s);
        }
        dbar/=count;
        float dd=0.0f;
        Iterator<Float> itr=db.iterator();
        while(itr.hasNext())
        {
            dd=itr.next();
            tou1+=(dd)-dbar;
        }
        tou1/=count;
        tou2=tou1*1.5f;
        context.write(new IntWritable((int)Math.abs(tou1)) ,new
FloatWritable(Math.abs(tou2)));
    } //reduce
} //reducer

/*
output of reduce
key: tou1
value: tou2
*/

public static void main(String[] args) throws Exception {

```

```

Configuration conf = new Configuration();
Job job = Job.getInstance(conf, "Tou1");
job.setJarByClass(Tou1.class);
job.setMapperClass(TokenizerMapper.class);
job.setCombinerClass(IntSumReducer.class);
job.setReducerClass(IntSumReducer.class);
job.setOutputKeyClass(IntWritable.class);
job.setOutputValueClass(FloatWritable.class);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}

```

Step 6 to Step 9:

```

import java.io.IOException;
import java.io.RandomAccessFile;
import java.util.*;
import java.lang.*;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
public class Mountain {
public static class TokenizerMapper extends Mapper<Object, Text,
IntWritable,FloatWritable> {
IntWritable one=new IntWritable(1);
int test=0;
static float maxM=0.0f;

/*
Input of map
key: tou1
value: tou2
*/
public void map(LongWritable key, Text value, Context context) throws IOException,
InterruptedException {
float temp=0.0f;
try{
float xone=0.0f;
String a;
RandomAccessFile x=new
RandomAccessFile("/home/hadoop/hadoop/project/tau.txt","r");//file with tau values
a=x.readLine();
float tau1=Float.valueOf(a);
a=x.readLine();
float tau2=Float.valueOf(a);

```

```

float inter1,inter2=0.0f;
float inter3=0.0f;
float inter4=0.0f;
float diff=0.0f;
float tau12=(tau1/2)*(tau1/2);//deno 1
float tau22=(tau2/2)*(tau2/2);//deno 1-1
LinkedList mi = new LinkedList();
LinkedList mil = new LinkedList();//centroid value
float xs=0.0f;//difference value
float xx=1.0f;//condition
String b;
int d=0;
while(xx!=0.0f)
{
    RandomAccessFile areader=new
    RandomAccessFile("/home/hadoop/hadoop/project/xsum.txt","r");//dataset x
    value
    RandomAccessFile breader=new
    RandomAccessFile("/home/hadoop/hadoop/project/xsum.txt","r");
    while(!(a=areader.readLine()).equals(null))
    {
        float xi=Float.valueOf(a);
        while(!(b=breader.readLine()).equals(null))
        {
            float xj=Float.valueOf(b);
            diff=(xi-xj)*(xi-xj);//numo
            inter1=(float)Math.exp(-(diff/tau12));//exp value
            inter2+=inter1;
            if(maxM<inter2)
            {
                {
                    if(temp!=xone)
                    {
                        test=1;
                    }
                }
                temp=xone;
                maxM=inter2;
                xone=xi;
            }
        }
        breader.seek(0);
        mi.add(inter2);//Mi values in linked list 1
        inter2=0.0f;
    }
}
breader.close();
areader.close();
//update mountain
float mI=0.0f;
RandomAccessFile areader1=new
RandomAccessFile("/home/hadoop/hadoop/project/xsum.txt","r");//dataset x value

```

```

RandomAccessFile breader1=new
RandomAccessFile("/home/hadoop/hadoop/project/xsum.txt","r");
while(!((a=areader1.readLine()).equals(null)))
{
    float xi=Float.valueOf(a);
    while(!((b=breader1.readLine()).equals(null)))
    {
        float xj=Float.valueOf(b);
        diff=(xi-xj)*(xi-xj);//numo
        inter3=(float)Math.exp(-(diff/tau12));//exp value
        Iterator<Float> itr=mi.iterator();
        while(itr.hasNext()){
            mI=(itr.next());
            inter4=mI*inter3;//left side value
            mil.add(inter4);//Mi value Linked list l+1
        }
    }
    breader1.seek(0);
} //while
breader1.close();
areader1.close();
float mm=0.0f;
float mm1=0.0f;
Iterator<Float> itr1=mi.iterator();
Iterator<Float> itr2=mil.iterator();
while(itr1.hasNext()){
    mm=(itr1.next());
    mm1=(itr2.next());
    xs=mm-mm1;
    if(xs>(0.6*mm))
        xx=0.0f;
}
}
if(test==1)
{
    context.write(new IntWritable(one),new FloatWritable(xone));
    test=0;
}
}
catch(Exception e)
{
    System.out.println(e);
}
} //map
} //mapper
/*
Output of reduce:
Key: No: of centroid
Value: Centroid
*/

```

```

public static class IntSumReducer extends
Reducer<IntWritable,FloatWritable,IntWritable,FloatWritable> {
    public static int cen=0;
    public void reduce(IntWritable key, Iterable<FloatWritable> values,Context context)
throws IOException, InterruptedException {
        for (FloatWritable value : values)
        {
            cen++;
            context.write(new IntWritable(cen),value);
        }
    }//reduce
} //reducer

```

```

public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "Mountain");
    job.setJarByClass(Mountain.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(IntWritable.class);
    job.setOutputValueClass(FloatWritable.class);
    FileInputFormat.addInputPath(job,new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}

```

Mean Shift Algorithm: [23]

```
import matplotlib.pyplot as plt
from matplotlib import style
style.use('ggplot')
import numpy as np
import pandas as pd

X = pd.read_csv('data.txt', delimiter = "\t") #datapoints in file

colors = 10*["g","r","c","b","k"]

class Mean_Shift:
    def __init__(self, radius=4):
        self.radius = radius

    def fit(self, data):
        centroids = {}

        for i in range(0,int(len(data)/4)):
            centroids[i] = [820,8] #Assigning determined centroids as a initial centroids
        for i in range(int((len(data)/4)+1),int(len(data)/2)):
            centroids[i] = [800,18]
        for i in range(int((len(data)/2)+1),int((len(data)*3)/4)):
            centroids[i] = [328,17]
        for i in range(int((len(data)*3)/4), int(len(data))):
            centroids[i] = [300,3]

        while True:
            new_centroids = []
            for i in centroids:
                in_bandwidth = []
                centroid = centroids[i]
```

```

for featureset in data:
    if np.linalg.norm(featureset-centroid) < self.radius:
        in_bandwidth.append(featureset)

new_centroid = np.average(in_bandwidth,axis=0)
new_centroids.append(tuple(new_centroid))

uniques = sorted(list(set(new_centroids)))

prev_centroids = dict(centroids)

centroids = { }
for i in range(len(uniques)):
    centroids[i] = np.array(uniques[i])

optimized = True

for i in centroids:
    if not np.array_equal(centroids[i], prev_centroids[i]):
        optimized = False
    if not optimized:
        break

if optimized:
    break

self.centroids = centroids

```

```

clf = Mean_Shift()
clf.fit(X)
centroids = clf.centroids

```



```
plt.scatter(X[:,0], X[:,1], s=150)
```

```
for c in centroids:
```

```
    plt.scatter(centroids[c][0], centroids[c][1], color='k', marker='*', s=150)
```

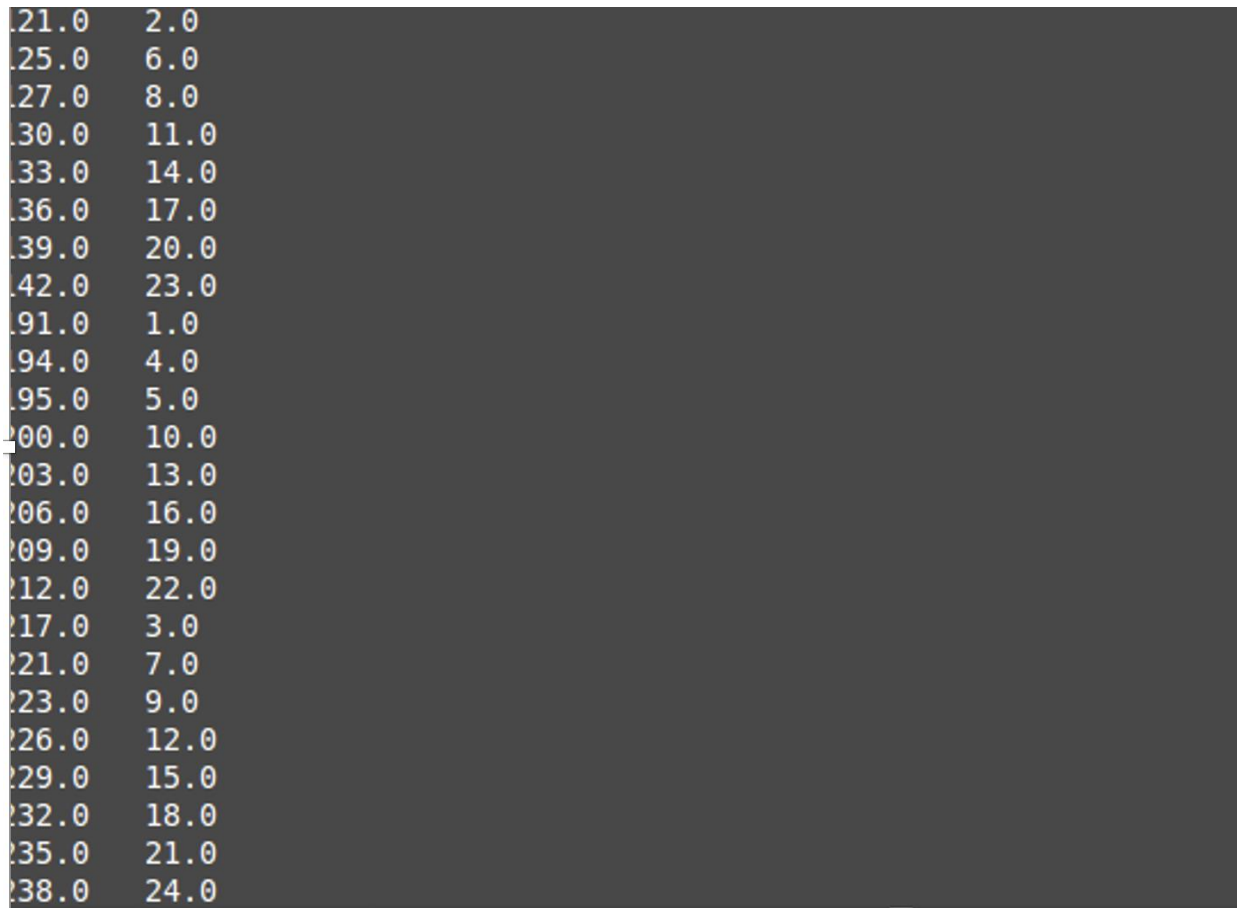
```
plt.show()
```

5. DISCUSSION OF RESULTS AND RECOMMENDATIONS

5.1 Results:

1. Phone call details from the dataset and calculated the membership values from it.

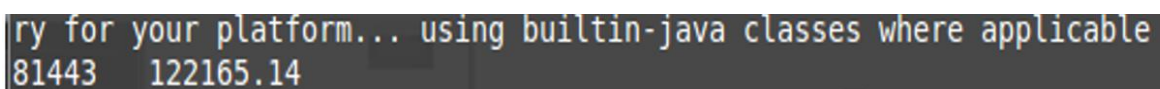
[High Membership values -> greater the loyalty]



21.0	2.0
25.0	6.0
27.0	8.0
30.0	11.0
33.0	14.0
36.0	17.0
39.0	20.0
42.0	23.0
91.0	1.0
94.0	4.0
95.0	5.0
100.0	10.0
103.0	13.0
106.0	16.0
109.0	19.0
112.0	22.0
117.0	3.0
121.0	7.0
123.0	9.0
126.0	12.0
129.0	15.0
132.0	18.0
135.0	21.0
138.0	24.0

Figure 5: Output of membership values

2. Neighbour radius and weight coefficient from the membership values.



```
ry for your platform... using builtin-java classes where applicable
81443 122165.14
```

Figure 6: Output of neighbor radius and weight coefficient

- Cluster centroids and the number of centroids with the help of neighbour radius and weight coefficient. Based on the cluster centroid's membership values, we can determine that 801 has lower membership value and hence the particular cluster is considered to have high risk customers.

```
hadoop@inspiron-Inspiron-5420 ~/hadoop/project $  
1 300.0  
2 328.0  
3 800.0  
4 801.0  
hadoop@inspiron-Inspiron-5420 ~/hadoop/project $
```

Figure 7: Output of Semantic Driven Subtractive Clustering Method

- Mean shift algorithm:

Data points present in each cluster by using Mean shift algorithm. Cluster centroids are given as input. (In Jupyter Notebook- using Python)

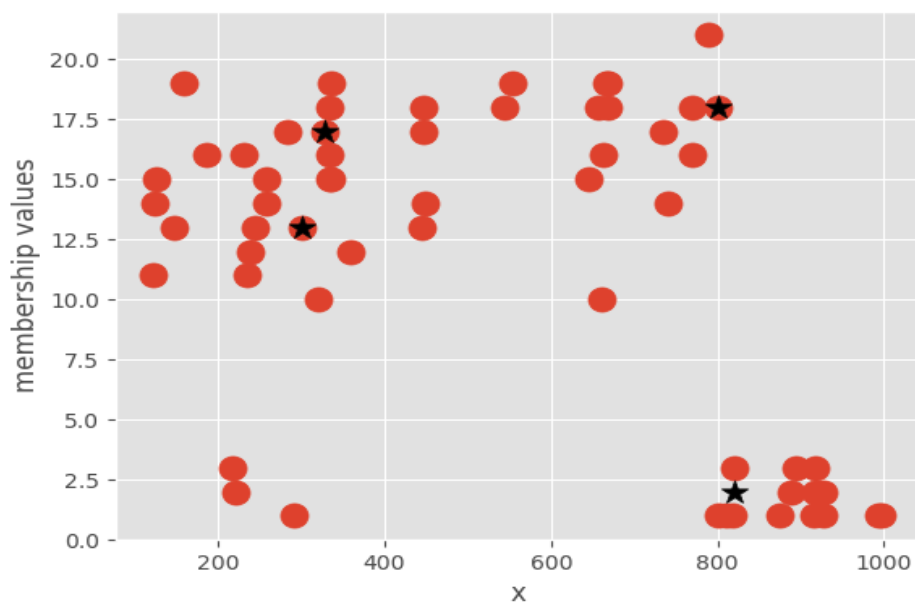


Figure 8: Output of Mean shift

5.2 Marketing Strategy:

To avoid these high-risk consumers from leaving the organization, a marketing strategy must be devised. Three techniques have been devised to avoid them from leaving the organization.

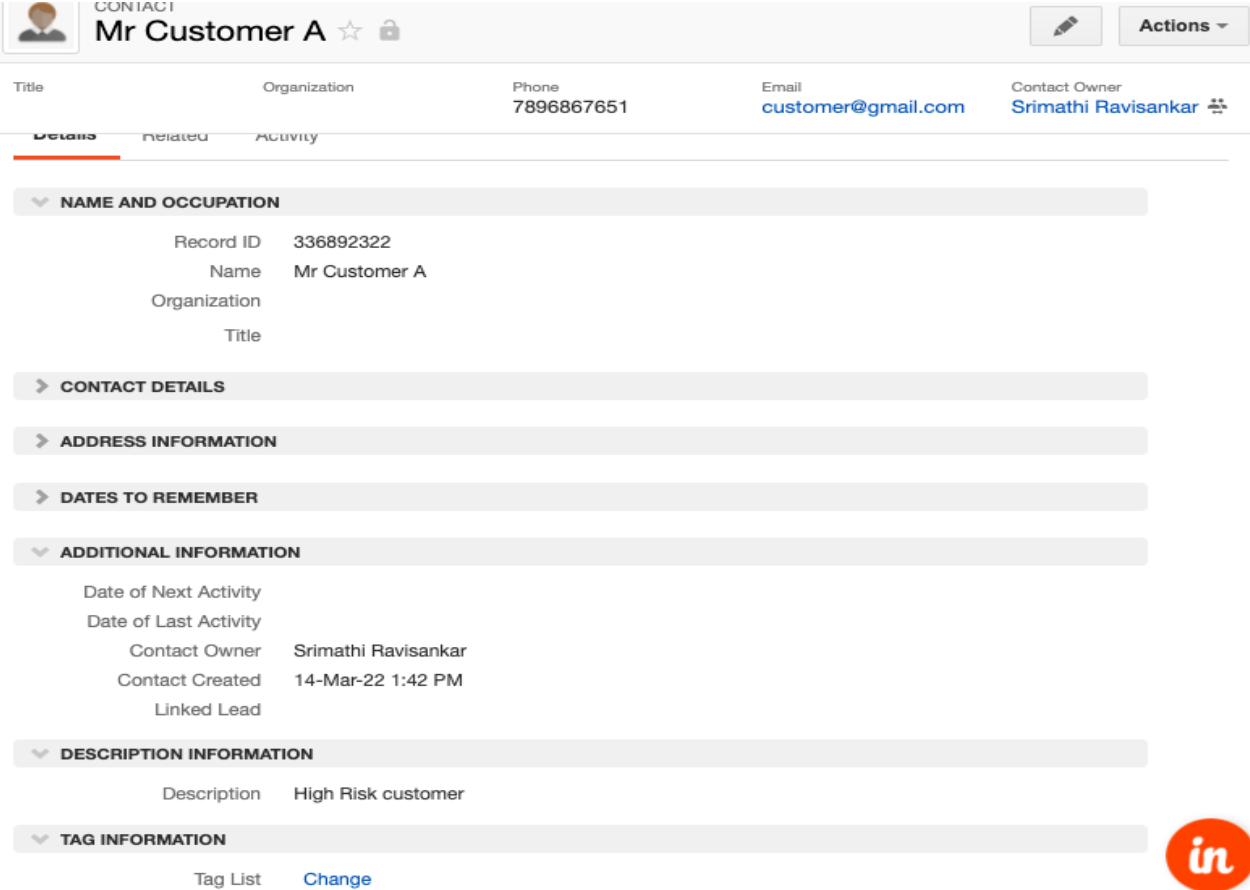
- Customers are offered discounts and are urged to stay with the company's services. For example: Unlimited calls for a defined amount of time is given to high-risk customer for a cheap rate.
- They may be eligible for further rewards, if they refer others. For example: If a client refers another individual, calls for two members are free for a limited time.
- Long-term subscription benefits could be offered to them. For example: A yearlong subscription is less expensive than a single month (nearly half the price). This could persuade them to sign up for an annual subscription.

5.2.1 CRM:

We can provide offers to the customer who is about to leave the company. The following are done in customer relationship management tool.

A. Strategy 1:

- **Contact:** High risk customer A, who is determined to leave the company, is targeted to receive the offer by strategy 1.



The screenshot displays a CRM contact record for 'Mr Customer A'. At the top, there is a header bar with a profile icon, the name 'Mr Customer A', and an 'Actions' dropdown menu. Below this, a metadata row shows fields for Title, Organization, Phone (7896867651), Email (customer@gmail.com), and Contact Owner (Srimathi Ravisankar). The main content area is divided into several expandable sections: 'NAME AND OCCUPATION' (showing Record ID 336892322, Name Mr Customer A, Organization, and Title), 'CONTACT DETAILS', 'ADDRESS INFORMATION', 'DATES TO REMEMBER', 'ADDITIONAL INFORMATION' (showing Date of Next Activity, Date of Last Activity, Contact Owner Srimathi Ravisankar, Contact Created 14-Mar-22 1:42 PM, and Linked Lead), 'DESCRIPTION INFORMATION' (showing Description High Risk customer), and 'TAG INFORMATION' (showing Tag List and a Change link). A red circular logo with the letters 'in' is positioned in the bottom right corner of the interface.

Figure 9: Contact of Customer A for strategy 1

- **Opportunity:**

The high-risk customer is created with an opportunity initially. According to strategy 1, the high-risk customer is targeted to have an offer for daytime calls for one month. He is targeted to have a strategy 1 offer, since he had very low calls in previous days.

OPPORTUNITY
Blockbuster offer for your Day time calls ☆ 🔒

Organization: TeleCom Co... Forecast Close Date: 11-Apr-22 Opportunity Value: Probability Of Winning: 60% Time in Stage: Closed Won Opportunity Owner: Srimathi Ravis...

PIPELINE: Opportunity Pipeline Opportunity State: Closed Won Change Pipeline Stage

Details Related Activity

DETAILS

Record ID: 33962320
 Opportunity Name: Blockbuster offer for your Day time calls
 Organization: TeleCom Company
 Current State: Won Change
 Reason: Product Fit

ADDITIONAL INFORMATION

Category:
 Probability Of Winning: 60%
 Opportunity Created: 12-Mar-22 2:59 PM
 Forecast Close Date: 11-Apr-22
 Actual Close Date: 12-Mar-22
 User Responsible: Srimathi Ravisankar
 Linked Email Address: srimaravi96-O33962320-E4BZL01@mailbox.insight.ly
 Opportunity Value:
 Date of Next Activity:
 Date of Last Activity:

DESCRIPTION INFORMATION

Description: Blockbuster offer for your Day time calls. Customer can have unlimited day

TAG INFORMATION

Tag List: Change

Figure 10: Opportunity created for Customer A



- **Sales:**

The customer is contacted, especially through email, with special offers to avoid him going away. Due to high risk, he is asked to schedule a time for a call to explain the current offer in order to avoid probability of transition from company's service.

Email Body Dear Customer,
You are lucky! You are given a special offer for this month with unlimited day calls for 50CZK/month. If you speak more calls, you will be given more opportunity by our company as a bonus.

If you suggest a couple of times that work for you, I'll get something on the calendar.

Thank you very much!

Figure 11: Email according to Strategy 1

B. Strategy 2:

- **Contact:** High risk customer B, who is determined to leave the company, is targeted to receive the offer by strategy 2.

The screenshot shows a CRM contact record for 'Mrs Customer B'. The header includes a profile icon, the name 'Mrs Customer B', and a star icon. Below the header is a table with columns for Title, Organization, Phone, Email, and Contact Owner (Srimathi Ravisankar). The main content area is divided into sections: 'NAME AND OCCUPATION' (Record ID: 336905354, Name: Mrs Customer B, Organization, Title), 'CONTACT DETAILS', 'ADDRESS INFORMATION', 'DATES TO REMEMBER', 'ADDITIONAL INFORMATION' (Date of Next Activity, Date of Last Activity, Contact Owner: Srimathi Ravisankar, Contact Created: 15-Mar-22 5:53 AM, Linked Lead), 'DESCRIPTION INFORMATION' (Description: High Risk Customer - Require offer for referrals.), and 'TAG INFORMATION' (Tag List: Change).

Figure 12: Contact of Customer B for strategy 2

- **Opportunity:**

The high-risk customer B is created with an opportunity initially. According to strategy 2, the high-risk customer is targeted to have an offer for referral. He is targeted to have a strategy 2 offer, since he had very few calls for past months and also those calls were to one particular person.

The screenshot shows a CRM interface for an opportunity. At the top, the title is "Refer a friend and earn Free calls for one month" with a star and lock icon. Below the title, key metrics are displayed: Organization (TeleCom Co...), Forecast Close Date (15-Mar-22), Opportunity Value, Probability Of Winning (10%), Time in Stage (0 Days), and Opportunity Owner (Srimathi Ravis...). A pipeline progress bar shows stages: Proposal (current), Negotiation, and Closed. The "Details" tab is active, showing fields for Record ID (33975109), Opportunity Name, Organization (TeleCom Company), and Current State (OPEN). Additional information includes Category, Probability Of Winning (10%), Opportunity Created (15-Mar-22 5:59 AM), Forecast Close Date (15-Mar-22), Actual Close Date (14-Apr-22), User Responsible (Srimathi Ravisankar), and Linked Email Address (srimaravi96-O33975109-E4BZL01@mailbox.insight.ly). Description information shows the opportunity name and a partial description. Tag information includes a Tag List and a Change link. A red "in" logo is visible in the bottom right corner.

Figure 13: Opportunity created for Customer B

- **Sales:** The customer is contacted, especially through email, with special offers to avoid him going away. He is asked to watch certain offers of company. This can reduce the probability of transition from company's service.

Email Body Customer, good day!

Congratulations! We're thrilled to let you know that you've been given a special referral discount. You will receive a one-month free call discount for the person you refer if you use the link below to suggest someone. You can only refer one person this month.

[Click here for referral](#)

If you could spare a few moments to watch our discount tutorial video, that would be amazing.

[Click here for tutorial](#)

Thank you

Figure 14: Email according to Strategy 2

C. Strategy 3:

- **Contact:** High risk customer C, who is determined to leave the company, is targeted to receive the offer by strategy 3.

The screenshot shows a CRM interface for a contact record. At the top, there is a header bar with a profile icon, the text 'CONTACT Ms Customer C', and a star icon. To the right of the header are a pencil icon and an 'Actions' dropdown menu. Below the header is a table with columns for 'Title', 'Organization', 'Phone', 'Email', and 'Contact Owner'. The 'Contact Owner' column contains the name 'Srimathi Ravisankar' and a small icon. Below the table is a navigation bar with 'Details', 'Related', and 'Activity' tabs. The 'Details' tab is selected and underlined. The main content area is divided into several sections, each with a dropdown arrow on the left: 'NAME AND OCCUPATION' (containing Record ID 336905375, Name Ms Customer C, Organization, and Title), 'CONTACT DETAILS', 'ADDRESS INFORMATION', 'DATES TO REMEMBER', 'ADDITIONAL INFORMATION' (containing Date of Next Activity, Date of Last Activity, Contact Owner Srimathi Ravisankar, Contact Created 15-Mar-22 5:56 AM, and Linked Lead), 'DESCRIPTION INFORMATION' (containing Description High Risk Customer - Requires long term discounts), and 'TAG INFORMATION' (containing Tag List and a blue 'Change' link).

Figure 15: Contact of Customer C for strategy 3

- **Opportunity:**

The high-risk customer C is created with an opportunity initially. According to strategy 3, the high-risk customer is targeted to have an offer for annual subscription. He is targeted to have a strategy 3 offer, since he had very few calls and he is expected to leave the company's service soon as he doesn't

have any calls for past few days. To maintain long term contact of customer, this offer is provided to him.

OPPORTUNITY
Best offer ever: Get Annual subscription and ...

Organization: Telecom Co... Forecast Close Date: 15-Mar-22 Opportunity Value: Probability Of Winning: 90% Time in Stage: 0 Days Opportunity Owner: Srimathi Ravis...

PIPELINE: Opportunity Pipeline Stage 4 of 6 : Proposal 0 Days in Stage Change Pipeline Stage

Details Related Activity

DETAILS

Record ID: 33975149
 Opportunity Name: Best offer ever: Get Annual subscription and have half price in your savings.
 Organization: Telecom Company
 Current State: **OPEN** Change

ADDITIONAL INFORMATION

Category:
 Probability Of Winning: 90%
 Opportunity Created: 15-Mar-22 6:05 AM
 Forecast Close Date: 15-Mar-22
 Actual Close Date: 14-Apr-22
 User Responsible: Srimathi Ravisankar
 Linked Email Address: srimaravi96-O33975149-E4BZL01@mailbox.insight.ly
 Opportunity Value:
 Date of Next Activity:
 Date of Last Activity:

DESCRIPTION INFORMATION

Description: Best offer ever: Get Annual subscription and have half price in your savings. less

TAG INFORMATION

Tag List: Change

TeleCom Company

Figure 16: Opportunity created for customer C

▪ **Sales:**

The customer is contacted, especially through email, with special offers to avoid him going away. He is asked to use the coupon within this month, this can urge him to not to leave the company’s service immediately and continue for one long year.

Email Body Hello Customer!

Thank you for subscribing to our service for past few months. We're happy to be able to provide you with a great discount on our Annual subscription. In comparison to a monthly membership, you can save half the cost. You can also disregard the monthly fee payment. To activate your discount, please apply the following coupon,

[Coupon for a savings](#)

Until 30-04-2022, the discount offer is active.

Good luck with your day!

Figure 17: Email according to strategy 3

6. CONCLUSION

As a result, the high-risk consumers were determined based on the clustering results and strategic solutions were supplied to such customers in order to avoid the probability of transition from company's service.

The following were successfully implemented,

- Clustering centroids were determined using SDSCM.
- Clusters were found using Mean Shift clustering algorithm.
- High risk customers were identified.
- Strategy solutions were given in order to avoid customer's probability of transition.

Thus, these methodologies can be used to anticipate customer attrition in service businesses with terabytes of data. As a result, the business can devise particular marketing techniques to boost profits. The accuracy of SCM and Mean shift clustering is improved by SDSCM. Additionally, employing AFS reduces the possibility of inaccuracy in operations management. SDSCM has stronger clustering, according to the results of the experiments. We transform the serial SDSCM into a big data SDSCM and use the MapReduce framework to implement it. Using big data SDSCM and big data Mean shift algorithms, we tackle the customer turnover problem in Telecommunication companies. In the context of big data, the process of resolving the customer churn problem in Telecom has provided new insights for managers to improve customer loyalty management.

7. REFERENCES:

1. Hadden, A. Tiwari, R. Roy, and D. Ruta, "Computer assisted customer churn management: State-of-the-art and future trends," *Comput. Oper. Res.*, vol. 34, no. 10, pp. 2902–2917, Oct. 2007. N. Lu, H. Lin, J. Lu, and G. Zhang, "A customer churn prediction model in telecom industry using boosting," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1659–1665, May 2014.
2. B. Q. Huang, T. K. Mohand, and B. Brian, "Customer churn prediction in telecommunications," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 1414–1425, Jan. 2012.
3. E. G. Castro and M.S.G. Tsuzuki, "Churn prediction in online games using players' login records: A frequency analysis approach," *IEEE Trans. Comput. Intell. AI Games*, vol. 7, no. 3, pp. 255–265, Sep. 2015.
4. W. H. Au, K. C. C. Chan, and Y. Xin, "A novel evolutionary data mining algorithm with applications to churn prediction," *IEEE Trans. Evol. Comput.*, vol. 7, no. 6, pp. 532–545, Dec. 2003.
5. K. J. Kohlhoff, S. P. Vijay, and B. A. Russ, "K-means for parallel architectures using all-prefix-sum sorting and updating steps," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 8, pp. 1602–1612, Aug. 2013.
6. C. Boutsidis and M. I. Malik, "Deterministic feature selection for kmeansclustering," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 6099–6110, Sep. 2013.
7. F. Afsari, M. Eftekhari, E. Eslami, and P. Y. Woo, "Interpretability-based fuzzy decision tree classifier a hybrid of the subtractive clustering and the multi-objective evolutionary algorithm," *Soft Comput.*, vol. 17, no. 9, pp. 1673–1686, Jan. 2013.
8. A. Garcia-Piquer, A. Fornells, J. Bavardit, and A. Orriols-Puig, "Largescale experimental evaluation of cluster representations for multiobjective evolutionary clustering," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 36–53, Feb. 2014.
9. A. Soualhi, C. Guy, and R. Hubert, "Detection and diagnosis of faults in induction motor using an improved artificial ant clustering technique," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 4053–4062, Sep. 2013. [22] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.
10. Wenjie, "A Big Data Clustering Algorithm for Mitigating the Risk of Customer Churn," *A Big Data Clustering Algorithm for Mitigating the Risk of Customer Churn*, June 2016.
11. J.LI, "Jamesxl." [Online]. Available: <http://jamesxli.blogspot.com/2012/03/on-mean-shift-and-k-means-clustering.html>.
12. G. A. P. & V. Pande, "aws.amazon.com" [Online]. Available: <https://aws.amazon.com/blogs/machine-learning/dive-deep-into-amazon-sagemaker-studio-notebook-architecture/>.

13. J. Freeman, "edrawsoft" May 2021. [Online]. Available: <https://www.edrawsoft.com/crm-application-architecture-example.html>.
14. M. Tandon, "medium," [Online]. Available: <https://medium.com/gsoc-2k19-with-mozilla/project-setup-matplotlib-internals-and-canvas-apis-fb5a497f9f16>.
15. B. Lainjo, "researchgate," [Online]. Available: https://www.researchgate.net/figure/Architecture-of-Python-programming-language-11_fig5_335600941.
16. J. Bean, "Zendesk," [Online]. Available: <https://www.zendesk.com/blog/sales-crm/>.
17. [Online]. Available: <https://www.bitrix24.com/glossary/what-is-contact-definition-crm.php>.
18. "Oracle," [Online]. Available: <https://www.oracle.com/cz/cx/what-is-crm/>.
19. W. Chai, "techtarget," [Online]. Available: <https://www.techtarget.com/searchcustomerexperience/definition/CRM-customer-relationship-management>.
20. [Online]. Available: <https://www.projectpro.io/hadoop-tutorial/hadoop-mapreduce-tutorial->
21. J. Brownlee, "Machine Learning Mastery," [Online]. Available: <https://machinelearningmastery.com/clustering-algorithms-with-python/>.
22. [Online]. <https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68>
23. [Online]. <https://pythonprogramming.net/mean-shift-from-scratch-python-machine-learning-tutorial/>
24. [Online]. <https://machinelearningmastery.com/clustering-algorithms-with-python/>

LIST OF FIGURES

<i>Figure 1: Different Task in Map Reduce program</i>	18
<i>Figure 2: Python Architecture</i>	20
<i>Figure 3: Notebook Architecture</i>	22
<i>Figure 4: CRM Architecture</i>	24
<i>Figure 5: Output of membership values</i>	49
<i>Figure 6: Output of neighbor radius and weight coefficient</i>	49
<i>Figure 7: Output of Semantic Driven Subtractive Clustering Method</i>	50
<i>Figure 8: Output of Mean shift</i>	50
<i>Figure 9: Contact of Customer A for strategy 1</i>	52
<i>Figure 10: Opportunity created for Customer A</i>	53
<i>Figure 11: Email according to Strategy 1</i>	54
<i>Figure 12: Contact of Customer B for strategy 2</i>	54
<i>Figure 13: Opportunity created for Customer B</i>	55
<i>Figure 14: Email according to Strategy 2</i>	56
<i>Figure 15: Contact of Customer C for strategy 3</i>	57
<i>Figure 16: Opportunity created for customer C</i>	58
<i>Figure 17: Email according to strategy 3</i>	58