Enhancing Customer Satisfaction through Data Mining: Predictive Analytics for Improved Decision-Making

Supervisor: Pr. Anouar Riad Solh
Realized by: Maryam Chouki, Wijdan Elbakhouchi

February 25, 2025

Abstract

The primary objective of data mining is to extract valuable insights from large datasets and transform them into a comprehensible structure for further analysis. One of the key techniques in data mining is clustering, which is crucial for exploratory data analysis and customer segmentation. Clustering involves grouping similar objects together, ensuring that data points in the same cluster exhibit higher similarity compared to those in other clusters.

This project applies clustering techniques to analyze customer satisfaction data, enabling businesses to better understand their customers and optimize their strategies. Various clustering models exist, including Connectivity models, Distribution models, Centroid models, Density models, Subspace models, Group models, and Graph-based models. We utilize partitioning clustering (K-Means) and classification (K-Nearest Neighbors) to segment customers based on their feedback, purchasing behavior, and loyalty levels. Additionally, we implement association rule learning using the Apriori algorithm to uncover hidden relationships between customer attributes.

By integrating these methods, our project provides a comprehensive approach to predicting customer satisfaction and loyalty. The findings help businesses tailor their services, enhance customer experience, and drive strategic decision-making. The visualization of clustering results further supports intuitive understanding and actionable insights for business growth.

This study not only reviews clustering techniques but also demonstrates their practical application in customer satisfaction analysis, bridging the gap between theoretical data mining concepts and real-world business applications.

1 Introduction

Enhancing customer satisfaction is a vital component for business growth and sustainability. In today's data-driven world, organizations have access to vast amounts of customer data, which, if properly analyzed, can provide valuable insights into customer behavior, preferences, and expectations. Data mining [1][2] plays a pivotal role in this process by enabling the exploration and analysis of large datasets to uncover meaningful patterns and trends. The key objective is to harness the computational power of machines while leveraging human intuition to detect significant relationships that drive customer engagement and loyalty.

Data mining is an integral part of the broader process known as Knowledge Discovery from Databases (KDD) [1], which encompasses several essential steps, including data access, preparation, application of mining algorithms, result analysis, and the implementation of appropriate actions. This structured approach ensures that valuable insights can be extracted efficiently, allowing businesses to make data-driven decisions that enhance customer satisfaction.

This study presents a predictive analytics approach to improving customer satisfaction using data mining techniques such as clustering, classification, and association rule learning. By analyzing customer feedback, purchasing behavior, and loyalty levels, this research aims to provide businesses with actionable insights to refine their strategies, personalize customer interactions, and optimize decision-making processes. Through the integration of these advanced analytical methods, companies can enhance user experience, foster long-term customer relationships, and drive overall business success.

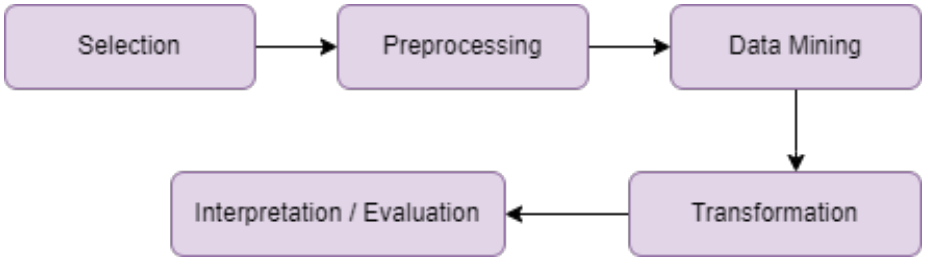

Figure 1: Process of Data Mining

2 Data Preparation and Exploration

We use a dataset containing various customer information:

- Demographic data (Age, Gender, Country, Income)
- Satisfaction factors (Product quality, Service quality, Purchase frequency)
- Loyalty level and satisfaction score

Before applying the algorithms, we perform data preprocessing, including:

- Encoding categorical variables (Gender, Country, Feedback, Loyalty)
- Normalizing numerical variables to standardize the scales.

3 Knowledge extraction using Association Rules

3.1 Understanding the Association Rules

The Apriori algorithm, introduced by Agrawal and Srikant (1994), is designed to identify frequent itemsets in large datasets by leveraging previously known frequent patterns. This algorithm follows an iterative (level-wise) approach, where k-itemsets (sets containing k items) are used to find (k+1)-itemsets.
A key principle of Apriori is the Apriori property, which states that:
"All nonempty subsets of a frequent itemset must also be frequent."
This means that if an itemset (I) is infrequent, then any superset containing (I) will also be infrequent. For example, if an itemset (I) does not meet the minimum support threshold, then adding another item (A) will not make it frequent[3].

3.2 Stages of the Apriori Algorithm

1 **Determine the minimum support :** This sets a threshold for filtering out infrequent patterns.

2 **First iteration :** Calculate the frequency of single items and keep only those meeting the minimum support (L1).

3 **Generate larger itemsets :** Use L1 to generate L2 (frequent 2-itemsets), then continue iterating to discover Lk (frequent k-itemsets) until no more frequent itemsets are found.

4 **Extract association rules :** Identify meaningful relationships between features based on confidence and lift scores.

3.3 Application to Customer Satisfaction Analysis

The Apriori algorithm is applied in this project to discover association rules between customer characteristics. By converting variables into a binary format, we extract valuable relationships, such as:

- (Product quality and satisfaction level)
- (Loyalty level and purchase frequency)

These insights allow businesses to detect hidden trends and correlations, helping them refine their marketing strategies and decision-making processes. By leveraging these association rules, companies can enhance customer experience, improve retention rates, and optimize service offerings.

4 Customer Segmentation with K-Means

4.1 Understanding K-means

The k-means[4]algorithm is a simple iterative clustering algorithm that partitions a given dataset into a user-specified number of clusters, k. The algorithm is simple to implement and run, relatively fast, easy to adapt, and common in practice. It is historically one of the most important algorithms in data mining. The k-means algorithm applies to objects that are represented by points in a d-dimensional vector space. Thus, it clusters a set of d-dimensional vectors, $D = \{x_i \mid i = 1, \ldots, N\}$, where $x_i \in R^d$ denotes the i-th object or "data point." As discussed we discussed before, k-means is a clustering algorithm that partitions D into k clusters of points. That is, the k-means algorithm clusters all of the data points in D such that each point x_i falls in one and only one of the k partitions. One can keep track of which point is in which cluster by assigning each point a cluster ID. Points with the same cluster ID are in the same cluster, while points with different cluster IDs are in different clusters. One can denote this with a cluster membership vector m of length N, where m_i is the cluster ID of x_i. The value of k is an input to the base algorithm. Typically, the value for k is based on criteria such as prior knowledge of how many clusters actually appear in D, how many clusters are desired for the current application, or the types of clusters found by exploring/experimenting with different values of k. How k is chosen is not necessary for understanding how k-means partitions the dataset D. In clustering algorithms, points are grouped by some notion of "closeness" or "similarity." In k-means, the default measure of closeness is the Euclidean distance. In particular, one can readily show that k-means attempts to minimize the following nonnegative cost function:

$$\text{Cost} = \sum_{i=1}^{N} \left(\arg\min_{j} \|x_i - c_j\|_2^2 \right) \tag{2.1}$$

In other words, k-means attempts to minimize the total squared Euclidean distance between each point x_i and its closest cluster representative c_j. Equation (2.1) is often referred to as the k-means objective function.

The k-means algorithm, depicted in Algorithm 2.1, clusters D in an iterative fashion, alternating between two steps: (1) reassigning the cluster ID of all points in D, and (2) updating the cluster representatives based on the data points in each cluster. The algorithm works as follows. First, the cluster representatives are initialized by picking k points in R^d. Techniques for selecting these initial seeds include sampling at random from the dataset, setting them as the solution of clustering a small subset of the data, or perturbing the global mean of the data k times. In Algorithm 2.1, we initialize by randomly picking k points. The algorithm then iterates between two steps until convergence.

Step 1: Data assignment Each data point is assigned to its closest centroid, with ties broken arbitrarily. This results in a partitioning of the data.

Step 2: Relocation of "means." Each cluster representative is relocated to the center (i.e., arithmetic mean) of all data points assigned to it. The rationale of this step is based on the observation that, given a set of points, the single best representative for this set (in the sense of minimizing the sum of the squared Euclidean distances between each point and the representative) is nothing but the mean of the data points. This is also why the cluster representative is often interchangeably referred to as the cluster mean or cluster centroid, and where the algorithm gets its name from.

The algorithm converges when the assignments (and hence the cj values) no longer change. One can show that the k-means objective function defined in Equation 2.1 will decrease whenever there is a change in the assignment or the relocation steps, and convergence is guaranteed in a finite number of iterations. Note that each iteration needs N $\times$ k comparisons, which determines the time complexity of one iteration. The number of iterations required for convergence varies and may depend on N, but as a first cut, k-means can be considered linear in the dataset size. Moreover, since the comparison operation is linear in d, the algorithm is also linear in the dimensionality of the data.

4.2 Application to Customer Satisfaction Analysis

The K-Means algorithm is used to group customers into clusters based on their characteristics. After testing different values of k, we choose 3 clusters, each corresponding to a distinct customer profile:

- Very satisfied customers
- Moderately satisfied customers
- Dissatisfied customers

In this context, the K-Means algorithm partitions the customer dataset into 3 clusters, where each cluster represents a group of customers with similar satisfaction levels. The value of k, which represents the number of clusters, is chosen

based on the goal of segmenting customers into meaningful groups that help businesses tailor their strategies. The algorithm minimizes the total squared Euclidean distance between each customer and the cluster's representative (centroid), helping to define the boundaries between the clusters. This segmentation enables businesses to personalize their offers and target customers more effectively.
The K-Means algorithm follows two primary steps, alternating iteratively:

1- reassigning each customer to the closest cluster

2- updating the centroids based on the customers' characteristics in each cluster

In the first step, each customer is assigned to the cluster whose centroid is closest, based on the Euclidean distance. In the second step, the centroids of the clusters are updated to the mean characteristics of the customers assigned to each cluster. These steps are repeated until the assignments no longer change, ensuring that the algorithm converges. The final result is a set of clusters that help the business understand the different types of customers, making it easier to customize strategies for each group.

5 Prediction of Satisfaction with K-NN

5.1 Understanding K-NN

The K-Nearest Neighbors (K-NN) algorithm is a classification algorithm that assigns a class to new data points based on their proximity to existing labeled data (training datasets). The algorithm determines the class of a new data point by analyzing the k nearest neighbors, where k is a predefined number of closest neighbors[5].
K-NN performs classification by projecting training data into a multi-dimensional space, where each data point is represented as a coordinate in that space. This space is divided into regions corresponding to different data classifications. When a new data point needs to be classified, it is also projected into the same multi-dimensional space. The algorithm then identifies the k nearest training data points and assigns the new data point to the most common class among its nearest neighbors. This approach ensures that classification is based on similarity to existing data, making K-NN a simple yet effective technique for pattern recognition and decision-making[6].

5.2 Choosing the Right k Value

Selecting the optimal value of k is crucial for K-NN's performance:

- if k **is too small**, the model becomes sensitive to noise (overfitting).
- if k **is too large**, classification becomes less precise and may blur decision boundaries.

- **Elbow Method:** A common approach to determine k is to plot accuracy vs.k and select the optimal balance point.
- **Cross-validation** can help optimize k based on the dataset.

‘

5.3 Application of K-NN to Customer Satisfaction Analysis

In this project, K-NN is used to predict customer satisfaction levels based on previous labeled data. The classification process follows these steps:

1 **Preprocessing customer data** (normalization, encoding categorical variables).

2 **Choosing k** (determined through cross-validation).

3 **Using Euclidean Distance to compute similarity** between new customers and existing ones.

4 **Classifying new customers** based on their nearest neighbors' satisfaction levels.

6 Conclusion

This analysis highlights the significance and effectiveness of Machine Learning algorithms in understanding and predicting customer satisfaction. By integrating the Apriori algorithm to uncover hidden relationships, K-Means for customer segmentation, and K-NN for satisfaction prediction, we gained valuable insights that enable businesses to implement more targeted and efficient strategies.

The findings of this study can be directly leveraged by companies to refine marketing approaches, personalize offerings, and enhance services, ultimately improving customer satisfaction and loyalty. By identifying customer segments with similar behaviors and preferences, businesses can optimize promotional campaigns and tailor user experiences, strengthening their competitive edge in the market.

References:

[1] Oded Maimon, Lior Rokach, "Data Mining AND Knowlwdge Discovery Handbook", Springer Science+Business Media.Inc,pp.321-352, 2005.
[2] Arun K Pujari " Data Mining Techniques" pg. 42-67 and pg. 114- 149,2006.
[3] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in Proc. 20th Int. Conf. Very Large Data Bases (VLDB), Santiago, Chile, Sep. 1994, pp. 487–499.
[4] Wu X., Kumar V. (eds.) The Top Ten Algorithms in Data Mining (CRC, 2009)(ISBN 1420089641)
[5] M.J.ZakiandW.Meira,"FundamentalconceptsandAlgorithms,"2014
[6] Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor(K-NN) Algorithm to Test the Accuracy of Types of Breast Cancer