Customer’s Purchase Prediction
Using Customer Segmentation Approach
for Clustering of Categorical Data

Juhi Singh¹, Mandeep Mittal²

¹ Department of Computer Science, Amity School of Engineering and Technology, Delhi, India
² Department of Mathematics, Amity Institute of Applied Sciences, Amity University Uttar Pradesh, Noida, India

Received: 27 November 2019
Accepted: 07 March 2021

Abstract
Traditional clustering algorithms which use distance between a pair of data points to calculate their similarity are not suitable for clustering of boolean and categorical attributes. In this paper, a modified clustering algorithm for categorical attributes is used for segmentation of customers. Each segment is then mined using frequent pattern mining algorithm in order to infer rules that helps in predicting customer’s next purchase. Generally, purchases of items are related to each other, for example, grocery items are frequently purchased together while electronic items are purchased together. Therefore, if the knowledge of purchase dependencies is available, then those items can be grouped together and attractive offers can be made for the customers which, in turn, increases overall profit of the organization. This work focuses on grouping of such items. Various experiments on real time database are implemented to evaluate the performance of proposed approach.

Keywords
categorical data, clustering algorithm, frequent pattern mining, association rules, customer relationship management.

Introduction

With the intent of improving business relationship with customers and helping in their retention, Customer Relationship Management (CRM) uses various strategies and technologies to analyze customer’s behavior, their purchasing habits and their inclination towards particular products. CRM systems are designed so that information on customers such as their personal information, transaction history, purchasing preferences and concerns can be compiled. The information obtained can then be analyzed to categorize customers into various groups based on certain criteria and the individual groups can then be targeted and customized offers can be made for customers in a more personalized format. In our previous work, we have used K-means algorithm to categorize data. However, the major drawback of K-means algorithm is that it works only on numeric data but usually in the real-world data set the objects are defined over categorical domain or domain with mixed numeric and categorical values. A modified approach is hence used in this paper for clustering of boolean and categorical attributes. Moreover, the generated categories were further analyzed and correlations between different data items were calculated with the help of association rule mining algorithm. Association rules relate two or more data items that are frequently purchased together. If we have sufficient number of association rules, customer’s purchase can be predicted. The approach is explained with the help of a simple example. The obtained results proved that the proposed approach provides better understanding of customer’s purchase pattern and thus the purchase can be predicted which in turn increases overall profit.

This paper presents a method to analyze customer’s behavior by mining the historical transactional database. This knowledge can then be exploited by decision makers to predict the future purchase and strategic decisions can be made to improve the customer relationship.
Literature review

The emergence of business-to-customer (B2C) marketing has opened new ways of improving customer retention and, in turn, profit enhancement. Various technologies such as data mining along with data warehousing and software such as campaign management software etc. can be used for better understanding of customer's purchase pattern. Managers can analyze their customers by evaluating their behavior, profile, loyalty, profitability and customer segregation. By extracting the hidden information present in the large databases managers can identify their valuable customers and predict their future behaviors and purchases well in advance, thus enabling firms to make appropriate knowledge driven decisions. The investigation of systems is, however, very difficult due to the presence of internal and external disturbances and also because of the limitation of our understanding. The available datasets tend to contain noises such as missing or wrong information and various kinds of uncertainty. As the new technologies for mining information have evolved, our understanding of uncertainties of systems has also been gradually deepened and intense research is being carried out for studying uncertain systems. The focus of grey systems theory is to study the problems that contain small samples of datasets having poor information. The grey system theory deals with uncertain systems having partially known information by using techniques to generate and extract useful information from what is available so that systems’ operational behaviors can be effectively monitored (Liu et al., 2011). Effect of multiple conditions on level of inventory is studied in (Jaggi, Khanna and Mittal, 2011; Mittal, Khanna and Jaggi, 2017) and (Jayaswal et al., 2019). In the real world, the dataset containing customer’s transactional information can be noisy having poor or missing information and therefore need to be analyzed thoroughly for identifying customer’s buying patterns. With the help of data mining clustering algorithms, customers can be categorized into various segments. Clustering has many application areas such as information retrieval, machine learning, image processing, bioinformatics, web data analysis, voice mining, text mining, scientific data exploration, pattern recognition, customer relationship management (CRM) etc. The main objective of clustering is to partition data objects into various subsets so that the objects which belong to the same clusters are close to each other and the data objects which belong to the different clusters are dissimilar from each other (Han, Pei and Kamber, 2011). For achieving this objective, a number of clustering algorithms have been proposed in the literature. A detailed review, covering various clustering algorithms and their comparative analysis is provided in (Jain et al. 1999; Xu and Wunsch, 2005) and (Jain, 2010). Real time datasets are generally high dimensional. Clustering algorithms for such datasets are presented in (Kriegel et al., 2000) and (Bai et al., 2011).

For an optimal solution, the main goal of clustering is to maximize both the homogeneity within a cluster and the heterogeneity among different clusters which is studied in (Hancer and Karaboga, 2017). Several research papers (Chen et al., 2002; Feizi-Derakhshi and Zafarani, 2012; Ayed, 2014; Xu and Wunsch (2005) have extensively covered the comparison among popular and known algorithm such as K-means, DBSCAN, DENCLUE, K-NN, fuzzy K-means, and SOM etc to choose an appropriate clustering algorithm for a given dataset. Singhal et al., (2013) provided a survey of clustering algorithms based on different criteria such as their score (merits), application domains, applicability factors, the number of clusters, size of datasets, stability, time complexity and so on. Sajana et al. (2016) and Fahad et al. (2014) linked big data challenges to clustering algorithms. Cai et al. (2016) and Zhao et al. (2017) compared respectively DBSCAN and K-means clustering algorithms for financial datasets and agglomerative hierarchical clustering and SOM for packaging modularization datasets. Gao et al. (2016) provides an overview of ant colony optimization with clustering to solve routing problem. Shen and Duan (2020) has applied K-means clustering algorithm to analyze teaching satisfaction from a database of students and campuses. The application of clustering algorithm in assessing the quality of underground water is provided by Vo-Van et al. (2020).

Data set can also be of temporal in nature. Liao (2005) has reviewed various clustering algorithms that can be used for such time series data sets. Clustering can also be applied on data streams. Such clustering algorithms are studied in detail in (Aggarwal et al., 2003) and (Cao et al., 2010). Hunt and Jorgensen (2011) have provided an overview of the various approaches for clustering of mixed datasets. Real time dataset contains both numerical attributes as well as categorical attributes such as transaction records of customers, characteristics of a user, browsing history of users, set of attributes for forecasting, web documents etc. Applying clustering algorithms to categorical datasets is difficult because the domain of categorical attributes is unordered. Several clustering algorithms for clustering of categorical or mixed type (contains both categorical and numeric values) of data are proposed in (Chen, 2009; Cao, 2010; Aggarwal and Yu, 2010; Barbará and Couto, 2002) and (Chen and
Liu, 2009). One of the most popular and well known widely used clustering algorithms is $K$-means algorithm. However, it works only on numeric datasets. An extension to $k$ means algorithm, known as $k$-prototype algorithm is proposed in (Huang, 1998). In our work, $k$-prototype algorithm for segmentation of customers is used. The created clusters can further be analyzed for extracting hidden information using various Association Rule Mining algorithms as done in our previous work (Singh et al., 2016). Association Rules are of the form that if the customer buys a certain product, then what is the possibility of purchasing another related product i.e. ARM provides a method to infer certain rules among given two or more purchases of the customer. By applying ARM [infer rules] a pattern can be recognized among a group of customers having similar purchase behavior. Effect of ARM on classification of inventory is provided by (Reshu and Mittal, 2019). Author’s contribution is shown below in tabular form in Table 1.

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Clustering of Numeric Data</th>
<th>Clustering of Categorical data</th>
<th>Customer Relationship Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang, 1998</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Han and Kamber, 2001</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Barbará and Couto, 2002</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Chen, 2009</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Cao et al., 2010</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Hunt and Jorgensen, 2011</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Singh et al., 2016</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Shen and Duan, 2020</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>This Paper</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Proposed work

The main objective of this research is to divide customers into number of categories on the basis of their past transactions so that their behavior can be predicted using various mathematical and statistical techniques. Various clustering algorithms are proposed in literature for this purpose. The major categories of clustering algorithms are:

- **Partitioning methods:** Such type of algorithms partition the given input dataset into disjoint clusters where each cluster represents a prototype.
- **Hierarchical methods:** In these methods similar type of clusters are grouped into larger cluster.
- **Density based methods:** In these methods we chose some data points first and start clustering procedure clustering starts and then other neighboring points are included if the neighborhood is sufficiently dense.
- **Grid Based methods:** In these methods, firstly, the given space of instances, which are to be grouped, is divided into a grid type structure and then clustering techniques are applied which uses cells of the grid as basic units.

The most famous and efficient partitioning clustering algorithm which is generally being used in scientific and industrial applications is $K$-means algorithm. In $K$-means algorithm, clusters are fully dependent on the choice of initial centroid. The working of $K$-means is very simple. Initially, $k$ data elements out of given $n$ data elements are selected as centers. For placing a data element in appropriate cluster its distance from each center is calculated using Euclidean’s distance formulae. The element is then assigned to that cluster the distance from whose center is the minimum. The center is recalculated, and the process keeps repeating until no more changes occur in clusters. The main objective of this algorithm is to minimize an objective function which is known as squared error function and is given by:

$$J(v) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} (\text{Euclidean} \ x_i, v_j)^2,$$

where:

- $\text{Euclidean} \ x_i, v_j$ is the Euclidean distance between $x_i$ and $v_j$,
- $c_i$ is the number of data points in $i$-th cluster, $c$ is the number of cluster centers.

The pseudo code of $K$-means algorithm as given in (Singh, 2016) is:

#### Input:
- $D = \{d_1, d_2, \ldots, d_n\}$ // Set of data items
- $k$ // Number of desired clusters

#### Output:
- $K$ // Set of clusters

Randomly select $k$ values as initial clusters
For each data item in $D$ repeat
J. Singh, M. Mittal: Customer’s Purchase Prediction Using Customer Segmentation Approach...

Calculate the distance between data item $d_i$ and all $k$ clusters
Assign the item $d_i$ to the clusters which has the minimum distance
Recalculate the new centre for each cluster

Until no change occurs in centre of cluster

$K$-means algorithm works in different iterations. It can be depicted more clearly with the help of Fig. 1. Firstly, all items belong to one cluster and 3 centres are selected. During each iteration, items are placed in their respective clusters. Final clusters are obtained at the end of 10-th iteration when no changes in clusters are observed.

$K$-means algorithm gives computationally fast results; however, it has one major drawback that it works only on numeric data. The variables here are measured on ratio scale as it minimises a cost function and then $K$-means can be applied. However such approach does not necessarily give meaningful results as the categorical data is generally not ordered. An extension to $K$-means algorithm known as $k$-prototype algorithm is proposed by Huang (1998). In $k$-prototype algorithm both numeric and categorical attributes are considered. If the domain is numeric then it is represented by continuous values and if the domain is categorical then it contains single values. The objects that are from same domain are represented by same set of attributes $A_1, A_2, A_3, \ldots, A_m$. $\text{DOM}(A_i)$ describes domain of attribute $A$. An object $X$, as given by Huang (1998), is described as conjunction of attribute-value pairs as follows:

$[A_1 = x_1] \land [A_2 = x_2] \land [A_3 = x_3] \land \ldots \land [A_n = x_n],$

where $x_j \in \text{DOM}(A_j)$ for $1 \leq j \leq m$.

Hence an object $X$ can be represented as

$[x_1^r, x_2^r, \ldots, x_p^r, x_{p+1}^c, \ldots, x_m^c],$

where the first $p$ elements of the object are numeric in nature while the other elements are of categorical type. Suppose $S^r$ is the factor that calculates dissimilarity measure for numeric attributes, and it is given by squared Euclidean distance and $S^c$ is the factor that calculates dissimilarity measure for categorical attributes and is defined by the number of mismatches occur in categories between two objects. The total dissimilarity factor between two objects for all the attributes is then given by $S^r + \alpha S^c$ where $\alpha$ is the weighing factor for two types of attributes.

If the objects are numeric in nature, their dissimilarity can simply be calculated by calculating Euclidean distance between them. However, if the objects are categorical in nature, their dissimilarity is calculated by counting the number of mismatches. The small the number of mismatches is, the more similar are the two objects. As given by Kaufman and Rousseuw (1990), the dissimilarity measure between any two categorical objects $X$ and $Y$ is calculated as:

$d_1(X, Y) = \sum_{j=1}^{m} \delta(x_j, y_j),$

Fig. 1. $K$-means algorithm
where $\delta(x_j, y_j) = 0$ if $x_j = y_j$
and $\delta(x_j, y_j) = 1$ if $x_j \neq y_j$.

Using the above stated formulae, if X and Y are two
mixed type objects containing total of $m$ attributes,
out of which $p$ are numeric attributes and remaining
are categorical attributes, then the dissimilarity be-
tween two objects is calculated as:

$$d(X, Y) = \sum_{j=1}^{p} (x_j - y_j)^2 + \alpha \sum_{j=p+1}^{m} \delta(x_j, y_j),$$  \hspace{1cm} (2)

where the first term of the formulae is the squared
Euclidean distance between numeric attributes and
second term is number of mismatches that occurred
between categorical attributes.

In this paper, credit approval dataset suggested by
Quinlan (1993) is used. This dataset contains 690 in-
stances. Each instance is described by 16 attributes
defined by 16 attributes out of which 6 are numeric
attributes and 9 are categorical attributes. For example,
following are the two instances. Each instance is described by 16 attributes

| Instance | 1: a, 58.67, 4.46, u, g, q, h, 3.04, t, t, 06, f, g, 00043, 560, + |
| Instance | 2: a, 24.50, 0.5, u, g, q, h, 1.5, t, f, 0, f, g, 00280, 824, + |

then the distance between them is calculated as follows:

$$d(1, 2) = 0 + 34.17 + 3.96 + 0 + 0 + 0 + 1.54$$
$$= 547.67.$$  

Suppose we have $n$ instances. Firstly, a number $k$ is
selected which signifies how many clusters we want
have. After selecting number of clusters, $k$ instances
are randomly taken as initial center. Now to place an
instance, its distance from each center is calculated
and it is placed in the cluster which has shortest dis-
tance i.e., the cluster which is closest. This process
repeats itself until there are no more changes in the
cluster. The pseudo code for the algorithm can be
written as:

**Input:**
$D = \{d_1, d_2, \ldots, d_n\}$ \hspace{1cm} // Set of instance
$K$ \hspace{1cm} // Number of desired clusters

**Output:**
$K$ \hspace{1cm} // Set of clusters

**K-Mode algorithm:**
Select $k$ values as initial clusters randomly

**For each instance repeat**
Calculate the distance between instance and all $k$
clusters by using the formulae given in (2) assign the
instance $d_i$ to the clusters which has the minimum
distance
Recalculate new center for each cluster;
until no change occurs in center of cluster;

By applying the above algorithm data instances can be
categorized into clusters. The obtained clusters are
further analyzed using association rule mining algo-
rithm in order to find out hidden patterns and cor-
relation between different data items were calculated.
Association rules relate two or more data items that
are frequently purchased together. Suppose a trans-
actional database is given which contains $n$ trans-
cactions and containing $i$ items $\{I_1, I_2, \ldots, I_i\}$. In order
to find association between items, we first have to cal-
tulate two factors support and confidence. Support of
item $I_i$ is defined as the frequency of its occurrences
in total transactions and is given by:

Support($I_i$) = Frequency($I_i$)/Total number
of transactions.

The relationships between items are expressed in
terms of confidence. Confidence is defined as condi-
tional probability conf ($I_1 \rightarrow I_2$) which refers to the
probability of purchasing $I_2$ when $I_1$ has already been
purchased and is given by

Conf($I_1 \rightarrow I_2$) = support($I_1 \cup I_2$)/support($I_1$).

For calculating support and confidence, A priori al-
gorithm has been used in this paper. It is the most
commonly known algorithm which was first proposed
in (Agrawal, Imielinski and Swami, 1993). The aim of
A priori algorithm is to find the frequently occurring
item-sets by calculating minimum support and gen-
erate association rules based on threshold confidence.
This algorithm runs in multiple passes. Initially, the
algorithm simply counts occurrences of each item to
determine the frequent 1-itemsets. Any other pass,
say pass $k$, consists of two phases. In the first phase,
candidate itemsets, $C_k$, is generated using the Apri-
ori candidate generation function (apriori-gen) which
uses the frequent itemsets $L_{k-1}$ found in the ($k-1$)-th
pass are used to generate the. Then the database is
scanned again to calculate the support of candidates
in $C_k$. The A priori algorithm as explained by Jain
(2010) is as follows

$L_1 = \{$frequent 1-itemsets$\}$
for ($k = 2$; $L_{k-1} \neq \emptyset$; $k + +$) generate
$C_k = \text{apriori-gen}(L_{k-1})$; // generate new candidates
for all transactions $t$ do begin
$C_t = \text{subset}(C_k, t)$; // Candidates contained in $t$
for all candidates $c \in C_t$ do
$c$.count++;
end
$L_k = \{c \in C_k | c.\text{count} \geq \text{minsup}\}$
end
Frequent Itemsets = $C_k \cup L_k$;

The $\text{apriori-gen}$ function takes as input, $L_{k-1}$, the set of all frequent ($k-1$) item sets and gives output as a superset of the set of all frequent $k$-item sets. These frequent item sets are then treated as candidate sets and support is counted only for these candidate sets. By using this algorithm, association rules can be evaluated. From large number of association rules, customer’s purchase can be predicted well in advance and the obtained information can be exploited for making better offers for the customers. A numerical example is presented in the next section to elaborate the approach.

**Numerical example**

To explain the above approach, an artificial dataset is taken as given in Table 2. The dataset contains 9 transactions $T$ (set of transactions) = \{T$_1$, T$_2$, T$_3$, T$_4$, T$_5$, T$_6$, T$_7$, T$_8$, T$_9$\} each of which is described by 4 items(attributes) $A$(set of attributes) = $A^c \cup A^v$, where $A^c$ is set of categorical attributes and $A^v$ is set of numeric attributes. Here, item 1 and item 2 are categorical and item 3 and item 4 are numeric in nature.

<table>
<thead>
<tr>
<th>Transactions</th>
<th>Item 1</th>
<th>Item 2</th>
<th>Item 3</th>
<th>Item 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>T$_1$</td>
<td>Bread</td>
<td>Butter</td>
<td>29</td>
<td>41</td>
</tr>
<tr>
<td>T$_2$</td>
<td>Lotion</td>
<td>Conditioner</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>T$_3$</td>
<td>Mobile Charger</td>
<td>Mobile Cover</td>
<td>95</td>
<td>96</td>
</tr>
<tr>
<td>T$_4$</td>
<td>Shampoo</td>
<td>Conditioner</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>T$_5$</td>
<td>Bread</td>
<td>Butter</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td>T$_6$</td>
<td>Earphone</td>
<td>Mobile Charger</td>
<td>91</td>
<td>98</td>
</tr>
<tr>
<td>T$_7$</td>
<td>Earphone</td>
<td>Mobile Cover</td>
<td>96</td>
<td>98</td>
</tr>
<tr>
<td>T$_8$</td>
<td>Rice</td>
<td>Butter</td>
<td>29</td>
<td>37</td>
</tr>
<tr>
<td>T$_9$</td>
<td>Shampoo</td>
<td>Hair Oil</td>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>

Let us take the number of clusters as three and initial centre as T$_1$, T$_4$ and T$_7$. The dissimilarity between each transaction of $T$ = \{T$_1$, T$_2$, T$_3$, T$_4$, T$_5$, T$_6$, T$_7$, T$_8$, T$_9$\} and initial centre is shown in Table 3.

<table>
<thead>
<tr>
<th>Distance among transactions</th>
</tr>
</thead>
<tbody>
<tr>
<td>T$_1$</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>65</td>
</tr>
<tr>
<td>126</td>
</tr>
</tbody>
</table>

By using the above stated algorithm, three clusters were obtained i.e. C$_1$ = \{T$_1$, T$_5$, T$_8$\}, C$_2$ = \{T$_2$, T$_4$, T$_9$\} and C$_3$ = \{T$_3$, T$_6$, T$_7$\} and their corresponding centre are $Z_1$ = \{Bread, Butter, 28, 38, 33\}, $Z_2$ = \{Shampoo, Conditioner, 2.33, 5.33\} and $Z_3$ = \{Earphone, Mobile Cover, 94, 97.33\}.

By applying clustering algorithm, we have divided our transactions into 3 groups where similar types of transaction are placed into one group. Each of these groups is then further analyzed using association rule mining algorithm to infer rule. For example, in cluster C$_1$ it can be easily seen that 100% of the times when bread was purchased, butter was also purchased wherein there is no correlation between the purchase of bread and rice. Similarly, other rules from other categories can also be obtained by calculating confidence factor between the two items. Some of the rules are as follows:

- Conf(Butter $\rightarrow$ 29) = 66%
- Conf(Shampoo $\rightarrow$ Conditioner) = 50%
- Conf(3 $\rightarrow$ 4) = 50%
- Conf(Mobile Charger $\rightarrow$ Mobile Cover) = 50%
- Conf(Earphone $\rightarrow$ 98) = 100%

Similarly, all rules can be obtained by fixing the minimum threshold for confidence factor. Once the knowledge about frequently purchased items is obtained, it can be used for making attractive deals for consumers. For example, as clear from the above rule that mobile charger and mobile cover are frequently purchased together with a probability of 50% so managers can make some discount offers for them collectively. The knowledge can be exploited in order to target individual customers and effective offers can be made to increase the overall revenue of the organization.

**Discussion**

With the vast amount of transactional data available, it becomes very difficult to analyze the correlation among items and to find association rules. A more practical approach is to categorize the transactions first and then find correlation among items. As clear from the above example, the transactional database...
is categorized into three categories and then apriori algorithm was applied to each of these categories and various association rules were obtained.

Obtained rules establish the purchase dependency among items with some probability and relate items such as shampoo and conditioner, mobile charger and mobile cover etc. This knowledge can then be exploited to predict customer’s future purchase.

Managerial implications

In this era of high competition, it is very important to understand customer’s behaviour so that personalized offers can be made to them. Data mining provides an insight into customer’s purchasing patterns by establishing correlation among items. If managers have knowledge of past transactional data then future transactions can be predicted which will help in retaining the customers and in turn, increasing the profit of a firm.

Conclusions

In this paper, a novel approach is proposed which combines clustering and association rule mining to predict the behavior of the customers. The customer’s transactional data was first classified into clusters. The obtained clusters were then mined for calculating various rules of the form, “if item X is purchased then what is the probability of purchasing Y”. Such rules are helpful in establishing relation between similar types of purchase and if given a record of past transactions then the probability of future transactions can also be predicted. With sufficient association rules, customer buying pattern can be studied, analysed and interpreted and it becomes possible to predict which products the customer will purchase next along with the given set of purchase of particular products.

References


