Clustering the Mixed Numerical and Categorical Datasets using Similarity Weight and Filter Method

Srinivasulu Asadi\textsuperscript{1} \hspace{1cm} Dr.Ch.D.V.Subba Rao\textsuperscript{2} \hspace{1cm} C.Kishore\textsuperscript{3} \hspace{1cm} Shreyash Raju\textsuperscript{4}

\textsuperscript{1}Associate Professor, Department of IT, Sree Vidyanikethan Engineering College, A.Rangampet, Chittoor (dt). A.P. INDIA-51710, srinu_asadi@yahoo.com

\textsuperscript{2}Professor of Computer Science Engg, Department of CSE, S.V.University Engg. College, Tirupathi, Chittoor (dt). A.P, INDIA-51710, subbarao_chdv1@hotmail.com

\textsuperscript{3}M.Tech Student, Department of IT, Sree Vidyanikethan Engineering College, A.Rangampet, Chittoor (dt). A.P. INDIA-51710, kishore.btech16@gmail.com

ABSTRACT

Clustering is a challenging task in data mining technique. The aim of clustering is to group the similar data into number of clusters. Various clustering algorithms have been developed to group data into clusters. However, these clustering algorithms work effectively either on pure numeric data or on pure categorical data, most of them perform poorly on mixed categorical and numerical data types in previous k-means algorithm was used but it is not accurate for large datasets. In this paper we cluster the mixed numeric and categorical data set in efficient manner. In this paper we present a clustering algorithm based on similarity weight and filter method paradigm that works well for data with mixed numeric and categorical features. We propose a modified description of cluster center to overcome the numeric data only limitation and provide a better characterization of clusters. The performance of this algorithm has been studied on benchmark data sets.

Keywords — Data Mining, Clustering, Numerical Data, Categorical Data, similarity weight, filter method

1. INTRODUCTION

Clustering is a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters. It is a challenging task in data mining techniques [7]. Many data mining applications require partitioning of data into homogeneous clusters from which interesting groups may be discovered, such as a group of motor insurance policy holders with a high average claim cost, or a group of clients in a banking database showing a heavy investment in real estate. To perform such analyses at least the following two problems have to be solved; (1) efficient partitioning of a large data set into homogeneous groups or clusters, and (2) effective interpretation of clusters. This paper proposes a solution to the first problem and suggests a solution to the second. With the amazing progress of both computer hardware and software, a vast amount of data is generated and collected daily. There is no doubt that data are meaningful only when one can extract the hidden information inside them. However, “the major barrier for obtaining high quality knowledge from data is due to the limitations of the data itself”. These major barriers of collected data come from their growing size and versatile domains. Thus, data mining that is to discover interesting patterns from large amounts of data within limited sources (i.e., computer memory and execution time) has become popular in recent years. Clustering is considered an important tool for data mining. The goal of data clustering is aimed at dividing the data set into several groups such that objects have a high degree of similarity to each other in the same group and have a high degree of dissimilarity to the ones in different groups. Each formed group is called a cluster. Useful patterns may be extracted by analyzing each cluster. For example, grouping customers with similar characteristics based on their purchasing behaviors in transaction data may find their previously unknown patterns. The extracted information is helpful for decision making in marketing.

Various clustering applications have emerged in diverse domains. However, most of the traditional clustering algorithms are designed to focus either on numeric data [13] or on categorical data [4][8-11]. The collected data in real world often contain both numeric and categorical attributes. It is difficult for applying traditional clustering algorithm directly into these kinds of data. Typically, when people need to apply traditional distance-based clustering algorithms [3][18] (ex., k-means[1]) to group these types of data, a numeric value...
will be assigned to each category in this attributes. Some categorical values, for example “low”, “medium” and “high”, can easily be transferred into numeric values. But if categorical attributes contain the values like “red”, “white” and “blue” … etc., it cannot be ordered naturally. How to assign numeric value to these kinds of categorical attributes will be a challenge work.

In this paper we first divide [2] the original data set into pure numerical and categorical data set. Next, existing well established clustering algorithms [8][13][19] designed for different types of datasets are employed to produce corresponding clusters. Last, the clustering results on the categorical and numeric dataset are combined as a categorical dataset on which the categorical data algorithm [14] is employed to get the final output.

The reminder of this paper is organized as follows. Next we show the background and related works and the proposed method for clustering on mixed categorical and numerical data, finally the conclusion of our work.

### 2. RELATED WORK

#### 2.1 Cluster Ensemble approach for mixed data

Dataset with mixed data type are common in real life. Cluster Ensemble [3] is a method to combine several runs of different clustering algorithm to get a common partition of the original dataset. In the paper divide and conquers technique [2] is formulated. Existing algorithm use similarity measures like Euclidean distance [12] which gives good result for the numeric attribute. This will not work well for categorical attribute. In the cluster ensemble approach numeric data [13] are handled separately and categorical data are handled separately. Then both the results are then treated in a categorical manner. Different types of algorithm used for categorical data are K-Modes [8][19], K-Prototype, ROCK [4] and squeezer algorithm. In K-Mode the total mismatch of categorical attributes of two data record is projected. The Squeezer algorithm yields good clustering result, good scalability and it handles high dimensional data set efficiently.

#### 2.2 Methodology

1. Splitting of the given data set into two parts. One for numerical data and another for categorical data
2. Applying any one of the existing clustering algorithms for numerical data set
3. Applying any one of the existing clustering algorithms for categorical data set
4. Combining the output of step 2 and step 3
5. Clustering the results using squeezer algorithm

The credit approval and cleve (heart diseases) dataset are used and measures the cluster accuracy and cluster error rate. The cluster accuracy ‘r’ is defined by

\[ r = \frac{\sum_{i=1}^{K} a_i} {n} \]

Where K represents number of clusters \( a_i \) represents number of instance occurring in both the cluster I and its class and n represents number of instance in the dataset. Finally cluster error rate ‘e’ defined by

\[ e = 1 - r \]

Where r represents cluster accuracy.

Fig 1: Overview of Cluster Ensemble algorithm framework

Fig 1. Shows that original dataset has to be splitted into categorical dataset and numerical dataset and clustering them. The output of clustering datasets are clustering using cluster ensemble algorithm.

The algorithm is compared with k-prototype algorithm.

Fig 2. Shows error rate among the k-prototype and Cluster Ensemble algorithm.

### 3. REVIEW OF K-MEANS ALGORITHM

K-means [1] is a clustering algorithm that deals with categorical data only. The k-means clustering algorithm[1] requires the user to specify from the beginning the number of clusters to be produced and the algorithm builds and refines the specified number of
Clustering the Mixed Numerical and Categorical Datasets…

clusters. Each cluster has a mode associated with it. Assuming that the objects in the data set are described by \( m \) categorical attributes, the mode of a cluster is a vector \( Q^c = (q_1, q_2, \ldots, q_m) \) where \( q_i \) is the most frequent value for the \( i \)th attribute in the cluster of objects.

Given a data set and the number of clusters \( k \), the k-means algorithm clusters the set as follows:

1. Select initial \( k \) means for \( k \) clusters.
2. For each object \( X_i \):
   a. Calculate the similarity between object \( X \) and the means of all clusters.
   b. Insert object \( X_i \) into the cluster \( c \) whose mode is the most similar to object \( X \).
   c. Update the mode of cluster \( c \).
3. Retest the similarity of objects against the current modes. If an object is found to be closer to the mode of another cluster rather than its own cluster, reallocate the object to that cluster and update the means of both clusters.
4. Repeat 3 until no or few objects change clusters after a full cycle test of all the objects.

In most clustering algorithms, an object is usually viewed as a point in a multidimensional space. It can be represented as a vector \((x_1, \ldots, x_d)\), a collection of values of selected attributes with \( d \) dimensions; and \( x_i \) is the value of \( i \)-th selected attribute. The value of \( x_i \) may be numerical or categorical.

Most pioneers of solving mixed numeric and categorical value for clustering problem is to redefine the distance measure and apply it to existing clustering algorithms.

**K-prototype:**

K-prototype is one of the most famous methods. K-prototype inherits the ideas of k-means [1], it applies Euclidean distance [12] to numeric attributes and a distance function is defined to be added into the measure of the closeness between two objects. Object pairs with different categorical values will enlarge the distance between them. The main shortcomings of k-prototype may fall into followings:

1. Binary distance is employed for categorical value. If object pairs with the same categorical value, the distance between them is zero; otherwise it will be one. However, it will not properly show the real situation, since categorical values may have some degree of difference. For example, the difference between “high” and “low” shall not equal to the one between “high” and “medium”.
2. Only one attribute value is chosen to represent whole attribute in cluster center. Therefore, the categorical value with less appearance seldom gets the chance to be shown in cluster center, though these items may play an important role during clustering process. Additionally, since k-prototype inherits the ideas of k-means, it will retain the same weakness of k-means.

### 4. PROPOSED ALGORITHM

In this paper we propose a new algorithm for clustering mixed numerical and categorical data. In this algorithm we do the following: First we read the large data set \( D \). Split [2] the Data Set \( D \) into Numerical Data and Categorical Data. Store all the Data Set.

Cluster the Numerical data set and categorical using Similarity Weight.

\[
Sim(a, b) = \sum_{i=1}^{n} t \sqrt{(a_i - b_i)^2}
\]

Combine[2] the clustered categorical dataset and clustered numerical dataset as a categorical dataset using Filtered method.

In this algorithm we cluster the numerical data, categorical data and mixed data. The above process is shown in fig 3.

### 4.1 SIMILARITY WEIGHT METHOD

Cluster validity functions are often used to evaluate the performance of clustering in different indexes and even two different clustering methods. A lot of cluster validity criteria were proposed during the last 10 years. Most of them came from different studies dealing with the number of clusters. To test validity indices, we conduct the Iris data sets and Wine data sets. Iris data sets are perhaps the best known database to be found in the pattern recognition literature. The data set contains 3 classes of 50 instances each, where each class refers to a type of Iris plant. One class is linearly separable from the other 2; the latter is not linearly separable from each other. Wine data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determines the quantity of 13 constituents found in each of the three types of wines. Essentially, for a given data set and “K” value, the DHCMA start with K randomly selected centres, and then assigns each data point to its closest centre, creating K partitions. At each stage in the iteration, for each of these K partitions, DHCMA recursively selects K random centres and continues the clustering process within each partition to form at most \( K^N \) partitions for the \( N^{th} \) stage [1]. In our implementation, the procedure continues until the number of elements in a partition is below \( K+2 \), at which time, the distance of each data item to other data items in that partition can be updated with a smaller value by a brute-force nearest neighbour search.

The Divisive [2] Hierarchical Clustering Algorithm partitions the data set into smaller partitions so that the number of data items in each partition must be less than the maximum partition size i.e., “K+2”. In the first iteration the entire data set is stored as the initial partition. After that, at each stage all the partitions are stored irrespective of their “K+2” condition.
For a set of s-dimensional data, i.e., each data item is a point in the s-dimensional space, there exists a distance between every pair of the data items. In this sequential initialization, all the pair wise distances are calculated after reading their details from the database. The threshold value calculation consists of Distance and Index Arrays. The distance array is used to record the distance of each data point to some other data point in the sequentially stored data set. The index array records the index of the data item at the other end of the distance in the distance array. The number of partition centers at each stage of the DHCA, performance of the algorithm with and without a given number of clusters. K varies from min to max. For max k, the algorithm produces into larger number of distance computations in the DHCA processes and the running time increases with k. As K increases, the change in running time increases along. For minimum value of K, the changes in the running time are small because when K is minimum for the construction of DHCA a small increase in k, decreases the total number of nodes which makes better in performance than using small k. When K gets larger and larger, more distance processes to partition center and the increase in processes eventually improves.

This incorporates the MST into the k-means algorithm with enhance cost function to handle the categorical data, and our experimental results show that it is effective in recovering the underlying cluster structures from categorical data if such structures exist. Modified MST representation for the cluster centre is used. This representation can capture cluster characteristics very effectively because it contains the distribution of all categorical values in cluster.

4.2. FILTER ALGORITHM
For clustering the mixed numerical and categorical datasets, we proposed an algorithm called filter method. First, the original dataset is divided [2] into two sub-datasets i.e., pure categorical dataset and the pure numerical dataset. Next, we apply clustering algorithms on sub-datasets[8][13] based upon their type of dataset to get the corresponding clusters. Last, the clustering results of numerical and categorical datasets are combined as a categorical dataset, on which the categorical data clustering algorithm is exploited to get the final cluster.

Now we discuss the last step of above process. With the clustering results on the categorical and numerical datasets, we also get the final clustering results by exploiting filter method. The process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints and data sources, etc. Applications of collaborative filtering typically involve very large data sets. Collaborative filtering methods have been applied to many different kinds of data including sensing and monitoring data - such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data - such as financial service institutions that integrate many financial sources; or in electronic commerce and web 2.0 applications where the focus is on user data, etc. The remainder of this discussion focuses on collaborative filtering for user data, although some of the methods and approaches may apply to the other major applications as well.

Step(1): Start with a tree built by the sequential initialization.
Step(2): Calculate mean and standard deviation of the edge weights distance array.
Step(3): Use their sum as the threshold.
Step(4): Perform multiple runs of Similarity Algorithm.
Step(5): Identify longest edge using Similarity.
Step(6): Remove this longest edge.
Step(7): Check Terminating Condition and continue.
Step(8): Put that number of clusters into Filter Method.

To summarize, the numerical parameters the algorithm needs from the user include the data set, the loosely
estimated minimum and maximum numbers of data points in each cluster, the input “K” to the filter method and similarity weight method, and number of nearest neighbours to keep for each data item in the auxiliary arrays, while the outputs will be the final distance and index arrays, and a labelling array that remembers the cluster label each data item belongs to.

**Filter Algorithm**

**Input:** Categorical Data set DS

**Output:** Set of Clusters

\[ D = \{X_1, X_2, ..., X_n\} \]

\[ C = \text{No of Clusters} \]

\[ m = \text{No of Attributes} \]

\[ n = \text{No of Records} \]

\[ F(D, C) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{i,j} d(X_i, C_j) \]

1. Read Data set D
2. Compute Dissimilarity
   \[ d(x, y) = \frac{x^2 + y^2}{|x| + |y|} \]
3. Compute F
   \[ F(D, C) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{i,j} d(X_i, C_j) \]
4. Cluster data according to F.
5. Submit the F into post Clustering Technique i.e. Filtered Algorithm

**Fig 4:** Applying clustering technique Similarity Weight and Filter Method

**Fig 5:** Results of clustering showing groups divided into clusters

**Fig 6:** Initialization and Input

**4.3 Advantages of Proposed System**

- Efficient use of the cut and cycle properties by our Fast Filtered-Inspired clustering algorithm.
- Shape of a cluster has very little impact on the performance of this Filtered clustering algorithm.
- Efficient for dimensionality more than 5 and reduced time complexity.
- Nearest neighbor search is used to construct efficient Filtered.
- Works efficiently even if the boundaries of clusters are irregular.
4. CLUSTERING RESULTS
We used the filter and k-prototypes algorithms to cluster the credit approval dataset and the clever dataset into different numbers of clusters, varying from 1 to 10. For each fixed number of clusters, the clustering errors of different algorithms were compared. For both of datasets, the k-prototypes algorithm, just as has been done. All numeric attributes are rescaled to the range of [0,1]. On the credit approval dataset.

![Fig 8: Clustering Error vs. number of clusters](image)

Figure 8 shows the results on the credit approval dataset set of different clustering algorithms from figure 8, we can summaries the relative performance of our algorithms as follows.

Table 4.1: Relative performance of different clustering algorithms (Credit approval dataset)

<table>
<thead>
<tr>
<th>Method</th>
<th>Average clustering error</th>
</tr>
</thead>
<tbody>
<tr>
<td>k-prototype</td>
<td>0.211</td>
</tr>
<tr>
<td>Similarity &amp; Filter</td>
<td>0.181</td>
</tr>
</tbody>
</table>

That is, comparing with the k-prototypes algorithm, our algorithm performed the best in all cases. It never performed the worst. Furthermore, the average clustering errors of our algorithm are significantly smaller than that of the k-prototypes algorithm. The above experimental results demonstrate the effectiveness of filter method for clustering dataset with mixed attributes. In addition, it outperforms the k-prototypes algorithm with respect to clustering accuracy.

5. CONCLUSION
The method of comprehensive assessment using similarity weight in the attribute synthetic evaluation system seemed to be objective and rational. Not only it embodied the weights of variables involved, but also exploiting the information presented by the sample. Our system is efficient for any number of Dimensions and reduces Time Complexity. Also Irregular boundaries can be efficiently handled using our filtered algorithm Divide and Conquer Technique. The future work is that we mix the different clustering datasets (labeled, unlabeled, nominal, and ordinal) with different algorithms.

6. REFERENCES


[10] Iam-On, N.; Boongeon, T.; Garrett, S.; Price, C.;Aberystwyth University, Aberystwyth. A Link-
Based Cluster Ensemble Approach for Categorical Data Clustering. vol. PP 1.


[12] Charu C. Aggarwal. Towards Systematic Design of Distance Functions for Data Mining Applications. SIGKDD '03, August 24-27, 2003, Washington, DC, USA


[16] Bainian Li; Kongsheng Zhang; and Jian Xu. Similarity measures and weighted fuzzy c-mean clustering algorithm. World Academy of Science, Engineering and Technology 76 2011


[18] Sotirios P. Chatzis. A fuzzy c-means-type algorithm for clustering of data with mixed numeric and categorical attributes employing a probabilistic dissimilarity functional. Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Road, South Kensington Campus SW7 2BT, UK


