K-MEANS AND HIERARCHAL CLUSTERING FILES ON SIMILARITY MEASURE

D.Ashok Kumar[1] and S. Vishnu Murthy[2]

1. M.Tech Final, Department of CSE, Aditya Institute of Technology and Management, Tekkali
2. Sr.Assistant Professor, Department of CSE, Aditya Institute of Technology and Management, Tekkali

Abstract: The K-means algorithm is a popular data-clustering algorithm. However, one of its drawbacks is the requirement for the number of clusters, K, to be specified before the algorithm is applied. This paper first reviews existing methods for selecting the number of clusters for the algorithm. Factors that affect this selection are then discussed and a new measure to assist the selection is proposed. The paper concludes with an analysis of the results of using the proposed measure to determine the number of clusters for the K-means algorithm for different data sets. In order to decide which clusters should be combined (for agglomerative), or where a cluster should be split (for divisive), a measure of dissimilarity between sets of observations is required. In most methods of hierarchical clustering, this is achieved by use of an appropriate metric (a measure of distance between pairs of observations), and a linkage criterion which specifies the dissimilarity of sets as a function of the pair wise distances of observations in the sets.

Keywords: Clustering, K-means algorithm, cluster number selection

Introduction:

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”. A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters. Clustering is the unsupervised classification of patterns (data items) into groups. This is most interesting and important topics in data mining. The aim of clustering is clustering is to determine the intrinsic grouping in a set of unlabeled data. There have been many clustering algorithms published every year. They can be proposed for very distinct research fields, and developed using totally different. Techniques and approaches. It is the most frequently used partitioned clustering algorithm in practice. Another recent scientific discussion states that k-means is the favorite algorithm that practitioners in the related fields choose to use. Unnecessary to mention, k-means has more than a small number of basic limitations, such as sensitiveness to initialization and to cluster size, and its performance can be worse than other state-of-the-art algorithms in many domains. An algorithm with sufficient performance and usability in most of application. Scenarios could be preferable to one with better performance in some cases but limited usage due to high complexity. While offering reasonable results, k-means is fast and easy to combine with other methods in larger systems. A common approach to the clustering problem is to treat it as an optimization procedure. An optimal partition is found by optimizing a particular function of similarity among data. Basically, there

is an implicit Assumption that the true basic structure of data could be correctly described by the similarity formula defined and embedded in the clustering criterion function. For instance, the original k-means has sum-of-squared-error objective function that uses Euclidean distance. In a very sparse and high-dimensional domain like text documents, spherical k-means, this uses cosine

Similarity instead of Euclidean distance as the measure is deemed to be more suitable. Showed that Euclidean distance was indeed one particular form of a class of distance measures

They concluded that non-Euclidean and no metric measures could be informative for statistical learning of data. Clustering still requires more robust dissimilarity or similarity Measures; recent works such as illustrate this need.

Clustering methods are classified into hierarchical clustering, data partitioning, data grouping. The hierarchical clustering is used to establish cluster taxonomy. Data partitioning is used to build a set of flat partitions. They are also known as non-overlapping clusters. Our first objective is to derive a novel method for measuring similarity between data objects in sparse and high-dimensional domain, particularly text documents.

K-Means Clustering:

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an objective function, in this case a squared error function. The objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centers.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers.

The k-means algorithm can be run multiple times to reduce this effect.

K-means is a simple algorithm that has been adapted to many problem domains. As we are

going to see, it is a good candidate for extension to work with fuzzy feature vectors.

This is a simple version of the k-means procedure. It can be viewed as a greedy algorithm for partitioning the n samples into k clusters so as to minimize the sum of the squared distances to the cluster centers. It does have some weaknesses:

- The way to initialize the means was not specified. One popular way to start is to randomly choose k of the samples.
- The results produced depend on the initial values for the means, and it frequently happens that suboptimal partitions are found. The standard solution is to try a number of different starting points.
- It can happen that the set of samples closest to mi is empty, so that mi cannot be updated. This is an annoyance that must be handled in an implementation, but that we shall ignore.
- The results depend on the metric used to measure || x - mi ||. A popular solution is to normalize each variable by its standard deviation, though this is not always desirable.
- The results depend on the value of k.

The Goals of Clustering:

The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. But how to decide what constitutes a good clustering? It can be shown that there is no absolute “best” criterion which would be independent of the final aim of the clustering. Consequently, it is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs. For instance, we could be interested in finding representatives for homogeneous groups (data reduction), in finding “natural clusters” and describe their unknown properties (“natural” data types), in finding useful and suitable groupings (“useful” data classes) or in finding unusual data objects (outlier detection).

Requirements:

The main requirements that a clustering algorithm should satisfy are:

- Scalability.
- Dealing with different types of attributes
- Discovering clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters
- Ability to deal with noise and outliers
- Insensitivity to order of input records
- High dimensionality
- Interpretability and usability.

Hierarchical Clustering:

Hierarchical clustering is an agglomerative (top down) clustering method. As its name suggests, the idea of this method is to build a hierarchy of clusters, showing relations between the individual members and merging clusters of data based on similarity. In the first step of clustering, the algorithm will look for the two most similar data points and merge them to create a new "pseudo-datapoint", which represents the average of the two merged datapoints. Each iterative step takes the next two closest datapoints (or pseduo-datapoints) and merges them. This process is generally continued until there is one large cluster containing all the original datapoints. Hierarchical clustering results in a "tree", showing the relationship of all of the original points. Hierarchical clustering of spike events is a method of grouping events that are similar in topology, morphology, or both, and it provides a method of efficient, detailed analysis of

interracial events. Information about the relative populations of spikes at multiple foci is presented, and artifact events are grouped and eliminated en masse. The process of hierarchical clustering is explained, and a set of simulated traces is used to illustrate the process of hierarchical clustering and the development of a cluster tree to display the relative populations of similar spike events. Using EEG data from long-term monitoring, the use of a "review wizard" is explored as a means of structuring the process of hierarchical clustering and traversing the cluster tree. This aid is also used to streamline the process of determining the similarity of events within each group and of verifying that events exhibiting clinically important differences are not hidden within the groups comprising the average traces.

We have a number of data points in an n-dimensional space, and want to evaluate which data points cluster together. This can be done with a hierarchical clustering approach

It is done as follows:

- Find the two elements with the small distance (that means the most similar elements)
- These two elements will be clustered together. The cluster becomes a new element
- Repeat until all elements are clustered

Important parameters in hierarchical clustering are:

The distance method: This measure defines how the distance between two data points is measured in general

Available options: Euclidean (default), Cosine, Correlation,

Spearman The linkage method this defines how the distance between two clusters is measured.

Available options: Average (default) and Ward

Use PCA data: Determines if the data is pretreated with a PCA.

Hierarchical Clustering Approach:

A typical clustering analysis approach via partitioning data set sequentially

Construct nested partitions layer by layer via grouping objects into a tree of clusters (without the need to know the number of clusters in advance)

Uses distance matrix as clustering criteria

- Agglomerative vs. Divisive:

(Two sequential clustering strategies for constructing a tree of clusters)

- Agglomerative: a bottom-up strategy
- Initially each data object is in its own (atomic) cluster
- Then merge these atomic clusters into larger and large clusters
- Divisive: a top-down strategy

(Initially all objects are in one single cluster)

- Then the cluster is subdivided into smaller and smaller clusters

To perform agglomerative hierarchical cluster analysis on a data set using Statistics Toolbox functions, follow this procedure:

1. Find the similarity or dissimilarity between every pair of objects in the data set. In this

step, you calculate the *distance* between objects using the pdist function. The pdist function supports many different ways to compute this measurement. See Similarity Measures for more information.

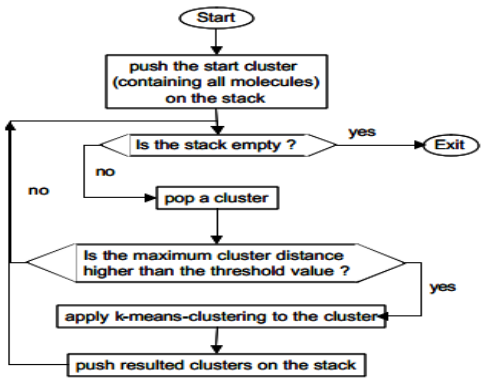
2. Group the objects into a binary, hierarchical cluster tree. In this step, you link pairs of objects that are in close proximity using the linkage function. The linkage function uses the distance information generated in step 1 to determine the proximity of objects to each other. As objects are paired into binary clusters, the newly formed clusters are grouped into larger clusters until a hierarchical tree is formed.

3. Determine where to cut the hierarchical tree into clusters. In this step, you use the cluster function to prune branches off the bottom of the hierarchical tree, and assign all the objects below each cut to a single cluster. This creates a partition of the data. The cluster function can create these clusters by detecting natural groupings in the hierarchical tree or by cutting off the hierarchical tree at an arbitrary point.

Find Natural Divisions in Data:

The hierarchical cluster tree may naturally divide the data into distinct, well-separated clusters. This can be particularly evident in a dendrogram diagram created from data where groups of objects are densely packed in certain areas and not in others. The inconsistency coefficient of the links in the cluster tree can identify these divisions where the similarities between objects change abruptly. You can use this value to determine where the cluster function creates cluster boundaries.

Flow chart for hierarchical k-means clustering:



Parameter selection:

After a descriptor file and a class file are chosen the following parameters can be set:

1. relevant classes,

2. k-value,

3. Threshold value.

At least one of the target classes has to be chosen and values for k and the threshold to be set. If illegal values for k and threshold were entered, nothing will happen. If another k value than two is chosen, no visualization, will be displayed. Nevertheless the clustering will be carried out and the save modes will stay applicable.

The k-value should be of type integer and >1 and the threshold value can be of type double (example: 2.7) and >0. If illegal values for k and the threshold were entered, nothing will happen. If another k value than two is chosen no visualization will be displayed. Nevertheless the clustering will be carried out and the save modes will stay applicable.

Working with the clustering tree:

The right part of the main window displays information about the tree and makes it possible to navigate in the tree.

Overview

In the first area (from top to down) an overview of the whole tree. The part of the tree, which is displayed on the left side of the window, is painted in white. In case the "diagram" checkbox is not enabled it is possible to choose a target class. If a target is chosen, the nodes in the tree that contain this target class will change their color to that of the target class.

Module Work:

File clustering is one of the essential text mining techniques. It has been around since the start of the text mining. It is the process of grouping objects into some groups in such a way that there is maximization of intra cluster object similarity and inter-cluster dissimilarity. Here an object does mean the file and term refers to a word in the file. Each file considered for clustering is represented as an m – dimensional vector "d". The "m" represents the total number of terms present in the given document. Many approaches came into existence for the document clustering. They include the information theoretic co-clustering, non – negative matrix factorization, and probabilistic model based method and so on. However, these approaches did not use specific measure in finding the document similarity. In this paper we consider methods that specifically use the certain measurement. From the literature it is found that one of the popular measures is the Euclidian distance:

$$\text{Dist}(di, dj) = \|di - dj\| \text{ -------} \quad (1)$$

K-means is one of the important clustering algorithms in the world. Due to its simplicity and ease of use it is still being used in the data mining domain. Euclidian distance measure is used in k-means algorithm. The main purpose of the k-means algorithm is to minimize the distance, as per the Euclidian measurement, between objects in clusters.

The centroid of such clusters is represented as follows:

$$\text{Min} \sum^{k} \sum \|d_i - C_r\|^2 \text{ --------} 2$$

$$r=1 \; d_i \epsilon S_r$$

In text mining domain, cosine similarity measure is also widely used measurement for finding document similarity, especially for hi-dimensional and sparse document clustering. The cosine similarity measure is also used in one of the variants of k-means known as spherical k-means. It is mainly used to maximize the cosine similarity between cluster's centroid and the documents in the cluster. The difference between k-means that uses Euclidian distance and the k-means that make use of cosine similarity is that the former focuses on vector magnitudes while the latter focuses on vector directions. Another popular approach is known as graph partitioning approach. In this approach the document corpus is considered as a graph. Min – max cut algorithm is the one that makes use of this approach and it focuses on minimizing centroid function.

$$\text{Min} \sum^{k} \frac{\text{Dtr D (3)}}{} \text{ ------- (3)}$$

$$r=1 \quad \|Dr\|^2$$

Other graph partitioning methods include Normalized Cut and Average Weight are used for document clustering purposes successfully. They used pair wise and cosine similarity score for document clustering. For document clustering analysis of criterion functions is made.

Another Clustering method of document clustering based on graph separation is implemented. It builds nearest neighbor graph first and then makes

clusters. In this approach for given non-unit vectors of document the extend Jaccard coefficient is:

$$Conn_{jaccoeff}(U_i, U_j) = \frac{U_i U_j}{\|U_i\|^2+\|U_j\|^2-U_iU_j} \quad \text{---- (4)}$$

Saving the Results:

Under File three save settings are available:

1. save clusters,
2. save virtual means,
3. save all virtual means.

In order to save the result clusters go to File | save clusters. A file chooser will pop up, where the destination path can be entered. The program will write the indices of the cluster members, one index per line, into the chosen file beginning with the leftmost cluster in the "hierarchical cluster tree". The result clusters are leaves of the "hierarchical cluster tree". Clusters are separated with an empty line

Initiate Work:

Identify files similarity

In this paper, there are many methods to judge

Similarity between documents. A brute force approach will compare the subject document with investigated documents word by word. However, in most cases, such approach is time and resources' consuming. In addition, such approach can be easily fooled through editing a small number of words in the document. A more effective approach depends or is based on metrics related to the documents such as the number of statements, paragraphs, punctuation, etc. A similarity index is calculated to measure the amount of similarity between documents based on those metrics.

Comparing the approach of taking the document word by word in comparison to statement. On one side, word by word comparison can minimize the effect of changing one or a small number of words relative to the total document. However, this can be time consuming and word to word document similarity may not necessarily means possible Plagiarism especially if the algorithm did not take the position of the words into consideration. Documents' similarity can be classified in different categories. In one classification, they can be classified into: word based, keyword based, sentence based, etc. phrase by paragraph approach is also affected by several variances such as the difference in size between the compared documents and the amount of words edited in those statements or paragraphs. Hashing algorithms are also used to measure Documents similarity. Hashing algorithms are used originally in security to verify the integrity of an Investigated disk drive and protected it from being tampered. Hashing can be calculated for a word, a Paragraph, a page, or a whole document.

Calculation:

Performing a hierarchical clustering algorithm, usually a distance threshold is predefined, until which data points are treated similar and should remain in a cluster. This threshold depends on the chosen data set and the chosen data description. To help the user in choosing this threshold, a calculation can be run where for a specified data range and a specified step size the number of singletons, the number of clusters (note that a clusters has at least more than one data point), and the sum of the maximum inner leave cluster distances (SMID) is calculated, normalized between zero and one and displayed in one graph

the results for a calculation. With increasing low thresholds the number of singletons decreases

whereas the number of clusters and the SMID increases. Singletons are mainly fused forming clusters and have by that a positive contribution on the number of clusters and the SMID value. At a certain threshold the number of clusters and a little delayed the SMID reach a maximum. After that both values decrease constantly. From this maximum point on singletons are mainly fused to existing clusters and clusters are merged, forming arger clusters. The reason why the SMID values are delayed compared to the number of clusters, is due to the fact that two properties

contribute to this value – the positive influence of clusters with larger diameter and the negative influence of the reduced number of clusters. We propose a threshold, where a maximum in the SMID value is obtained as "good" threshold, since it always occurs

independent of the data set and data description - and "meaningful" results were obtained with this threshold.To calculate this threshold go to File | Threshold calculation. First a file chooser will pop up. After a path is defined an input window will be displayed where the upper and lower bound for the threshold and the step size have to be specified. After the parameters are set, the hierarchical k-means will be started. The algorithm will be run starting at the entered lower bound, increasing the threshold by the entered step size until the upper bound is reached. The result is displayed as a diagram whe re the number of singletons, the number of clusters and the SMID are shown as curves.

Conclusion:

Clustering is one of the data mining and text mining techniques used to analyze datasets by dividing it into various meaningful groups. The objects in the given dataset can have certain relationships among them. All the clustering algorithms assume this before they are applied to datasets. The existing algorithms for the text mining make use of a single point of view for measuring similarity between items. Their drawback is that the clusters cannot exhibit the complete set of relationships among items. To overcome this drawback, we propose a new similarity measure known as the Hierarchal clustering Files based similarity measure to ensure the clusters show all relationships among objects. This approach makes use of different point of view from different objects of the multiple clusters and more useful assessment of similarity could be achieved

References

- Cartwright, H. M. and Sztandera, L. M. Soft Computing Approaches in Chemistry; Springer-Verlag: Heidelberg, 2003.

- J. B. MacQueen (1967): "Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability"*, Berkeley, University of California Press, 1:281-297

- A. Ahmad and L. Dey, "A Method to Compute Distance Between Two Categorical Values of Same Attribute in Unsupervised Learning for Categorical Data Set," Pattern Recognition Letters, vol. 28, no. 1, pp. 110-118, 2007

- S. Zhong, "Efficient Online Spherical K-means Clustering," Proc. IEEE Int'l Joint Conf. Neural Networks (IJCNN), pp. 3180-3185, 2005

- I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in KDD, 2003, pp. 89–98.
- Grier, S., "A tool that detects plagiarism in Pascal programs",
- ACM SIGCSE Bulletin, vol. 13(1), 1981, pp. 15-20.
- Faidhi, J.A.W., Robinson, S.K., "An empirical approach for detecting program similarity within a university programming environment", Computers & Education, vol. 11(1), pp. 11-19.
- Duda, R. O.; Hart, P. E.; and Stork, D. G. Pattern Classification; John Wiley & Sons: 2000
- Otto, M. Chemometrics. Statistics and Computer Application in Analytical Chemistry.; Wiley-VCH: 1998.
- Jain, A. k.; Murty, M. N.; Flynn, P. J.; and Data Clustering: A Review. ACM Computing Surveys 1999, 31, 265-323.