Analysis of Different Similarity Measure Functions and their Impacts on Shared Nearest Neighbor Clustering Approach

Anil Kumar Patidar
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

Jitendra Agrawal
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

Nishchol Mishra
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

ABSTRACT

Clustering is a technique of grouping data with analogous data content. In recent years, Density based clustering algorithms especially SNN clustering approach has gained high popularity in the field of data mining. It finds clusters of different size, density, and shape, in the presence of large amount of noise and outliers. SNN is widely used where large multidimensional and dynamic databases are maintained.

A typical clustering technique utilizes similarity function for comparing various data items. Previously, many similarity functions such as Euclidean or Jaccard similarity measures have been worked upon for the comparison purpose. In this paper, we have evaluated the impact of four different similarity measure functions upon Shared Nearest Neighbor (SNN) clustering approach and the results were compared subsequently. Based on our analysis, we arrived on a conclusion that Euclidean function works best with SNN clustering approach in contrast to cosine, Jaccard and correlation distance measures function.

Keywords

Data mining, Clustering, SNN (Shared Nearest Neighbor), Density, Noise, Outlier, Similarity Measure.

1. INTRODUCTION

1.1 Data Mining

Data mining is new technology/process of finding novel, hidden, interesting, and useful information, or knowledge from the large volumes of raw data [6]. This useful information or knowledge can be used to predict or to tell us something new. Data is an essential entity or fact of our corporation, but only if we know how to retrieve or extract useful data from the large volumes of raw data. Data mining technique helps us in accomplishing this [7].

1.2 Clustering

Clustering is the most important technique of data mining. Clustering is a technique of grouping of similar data objects together, so that the objects in each group (called cluster) share the same pattern of information. Clustering technique is widely used in financial data classification, spatial data processing, satellite photo analysis, engineering and medical figure auto-detection, Social network analysis etc. [5]. There are two types of clustering techniques [8] - partitioning and hierarchical clustering technique.

In this paper, we have used density based SNN clustering approach. It is an efficient clustering approach for dynamic database mining. From the previous results, has been inferred that the Density based clustering is very effective for analyzing large amounts of heterogeneous, complex data for example clustering of complex objects [5].

1.3 Similarity Measures

Similarity measure is defined as the distance between various data points. The performance of many algorithms depends upon selecting a good distance function over input data set. While, similarity is a amount that reflects the strength of relationship between two data items, dissimilarity deals with the measurement of divergence between two data items [2] [3].

Here, we present a brief overview of similarity measure functions used in this paper:

1. **Euclidean distance:** Euclidean distance determines the root of square differences between the coordinates of a pair of objects [2]. For vectors x and y distance d (x, y) is given by:

$$\textbf{Sim(x, y) = d} = \sqrt{\sum_{i=1}^{n}(\mathbf{x_i} - \mathbf{y_i})^2}$$

   Where x and y are n-dimensional vectors.

2. **Cosine distance:** Cosine distance measure for text clustering determines the cosine of the angle between two vectors given by the following formula [2]:

$$\textbf{Sim}(\mathbf{x_i}, \mathbf{x_j}) = \cos\theta = \frac{(\mathbf{xi{\cdot}xj})}{(|\mathbf{xi}| \times |\mathbf{xj}|)}$$

   Where, θ refers to the angle between two vectors and x_i, x_j are n-dimensional vectors.

3. **Jaccard distance:** The Jaccard distance, involves the measurement of similarity as the intersection divided by the union of the data items [3]. The formulae could be stated as:

$$\textbf{Sim}(\mathbf{x_i}, \mathbf{x_j}) = \frac{(\mathbf{xi{\cdot}xj})}{(|\mathbf{xi}|\mathbf{2} + |\mathbf{xj}|\mathbf{2} - \mathbf{xi{\cdot}xj})}$$

4. **Pearson Correlation distance:** Pearson's correlation distance is another measure of the extent to which two vectors are related [3]. The distance measure could be mathematically stated as:

$$\textbf{Sim(x, y)} = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sqrt{\left(\sum x^2 - \frac{(\sum x)^2}{n}\right)\left(\sum y^2 - \frac{(\sum y)^2}{n}\right)}}$$

2. OUTLINE OF THE PAPER

This paper is composed of 6 sections in addition to the introduction. Section-3 describes the related work (literature survey) done based on the notion of density and similarity measure. The SNN clustering approach is discussed in Section-4. While Section-5 dealt the experimental setup, section-6 confined the results and analysis. A short conclusion and directions for future work is presented in Section-7 and section-8 dealt with references.

3. LITERATURE SURVEY

There are number of clustering algorithms based to the notion of density. However, in this paper our focus confined on the widely used SNN clustering approach. In this section, we represent a brief overview of the work done in the area of Density based clustering and similarity measure. Discovering clusters of different sizes and shapes is difficult in the presence of noise and outliers. Many recent clustering algorithms like DBSCAN [9], CURE [10], ROCK [11] and Chameleon [12], and other variations of DBSCAN clustering approach have tried to address this problem, but these algorithms did not work well with the objects of varying density. Finding clusters of different shape, size, and density, especially in the presence of noise and outlier is a problem dealt most recently with a recent clustering algorithm known as SNN clustering approach.

Jarvis and Patrick [4], first introduced this idea of shared nearest neighbor. In the Jarvis – Patrick approach, a snn (shared nearest neighbor) graph is created from the proximity matrix. A link is constructed from pair of points ‘a’ and ‘b’ if and only if ‘a’ and ‘b’ has their closest k- nearest neighbor lists to each other. This approach is k-nearest neighbor sparsification. The number of near neighbors that two points share derives the weights of the links between two points in the snn graph.

Martin Ester, Hans-Peter Kriegel, Joerg Sander, and Xiaowei Xu [9], demonstrated that the DBSCAN clustering approach find clusters of arbitrary shapes and sizes but it cannot work with data clusters of differing densities, because its density-based definition of core points can’t address the core points of varying density clusters. In DBSCAN clustering approach, if user defines the neighborhood of a point by giving a particular radius and then looks up for core points (core objects) then one of the point that satisfy the conditions for core point is selected as core point while rest of the points will be marked as noise. Else every point connected to that core point will belong to one cluster.

Sudipto Guha, Rajeev Rastogi and Kyuseok Shim [10], represented that CURE (Clustering Using REpresentatives), utilizes representative points to find non-globular clusters. One of the problems of using CURE clustering approach is that it cannot handle many types of globular shapes. This problem is due to the approach of CURE algorithm to finds representative points, i.e., CURE algorithm find points along the boundary, and then shrinks those points towards the center of the cluster.

George Karypis, Eui-Hong Han, and Vipin Kumar [12] verified that while DBSCAN uses the notion of core points, CURE utilizes representative points as criterion, but neither of the core points or representative points was explicitly used by Chameleon. All three approaches (DBSCAN, CURE, and Chameleon) share the common idea (that the challenge) of finding clusters of different shapes and sizes. Main motto of these three clustering approaches is to find points or subsets of points and then constructing clusters around them. Chameleon approach is important for spatial data, as we cannot represent non-globular clusters by their centroid, thus, centroid based scheme cannot handle them [12]. While using DBSCAN, CURE, and Chameleon approaches, we must also give considerable attention to handling of noise and outliers.

Anna Huang [2], evaluated the effects of many similarity functions on k-mean clustering algorithm.

Kazem Taghva and Rushikesh Veni [3], compared and analyzed the effectiveness of these measures in partitional clustering for text document datasets.

In this paper, we described SNN clustering approach with four different similarity measure functions and compared the effects of these similarity measures on SNN clustering approach.

4. SNN CLUSTERING APPROACH

Shared Nearest Neighbor (SNN) [1] is one of the most important and most common clustering approach in engineering and scientific literature, which has the ability to produce clusters of different size, shape, and density. The SNN approach, like DBSCAN approach [9], is based on density-based clustering approach. The main difference between SNN approach and DBSCAN approach is that while SNN deals with varying densities clusters, DBSCAN do not deal with clusters of varying densities. SNN defines the similarity between points by examining the number of nearest neighbors that are shared by two points. Utilizing the similarity measure in the SNN clustering approach, we defined the density as the sum of all the similarities of the nearest neighbors of a point. High-density points become core points, and low-density points become noise points. All other points, greatly similar to particular core points were drew as new clusters.

SNN clustering approach [1] can be explained as under.

1. Compute the similarity matrix: This corresponds to a similarity graph with data points for nodes and edges whose weights are the similarities between data points.
2. Sparsify the similarity matrix: This involves keeping only the *k* most similar neighbors of each data point. This corresponds to only keeping the *k* strongest links of the similarity graph.
3. Construct the shared nearest neighbor graph: SNN graph obtained from the sparsified similarity matrix. Here, we could apply a similarity threshold and find the connected components to obtain the clusters (Jarvis Patrick algorithm)
4. Find the SNN density of each Point: Data points having an SNN similarity greater or equal to *Eps* were obtained.
5. Find the core points: All points that have an SNN density greater than *MinPnt* were designated as Core points.

6. Form clusters from the core points: If two core points are within a radius, *Eps*, of each other, they are placed in the same cluster.
7. Discard all noise points: All non-core points that were not within a radius of *Eps* of a cluster are discarded.
8. Assign all non-noise, non-core points to clusters: All these points are assigned to the nearest cluster.

Following are the inputs and their corresponding outputs as generated by the SNN clustering approach.

Input:

D- Data set
k- Maximum number of nearest neighbors to each point
Eps- Density threshold (radius of cluster)
minPnt- Core point threshold

Output:

K: a set of clusters

In this paper, we used four different similarity measure functions for calculating similarity matrix and compared the similarity graphs and resultant clusters. The similarity measure functions are- Euclidean, Cosine, Jaccard and Correlation function.

SNN clustering approach has many good characteristics. First, the SNN clustering approach does not cluster all the points. In general, this is good, because much of the data is noise and needs to be removed. If the complete clustering is desired, then unclustered data can be inserted to the core clusters discovered by SNN clustering approach by assigning them to the cluster containing the closest representative point. Second, the approach is especially partitional, although we have experimented some by creating a hierarchy of clusters. Finally, the time complexity is O(n^2) where *n* is the number of points, because the similarity matrix has to be computed [1] [4].

5. EXPERIMENTAL SETUP

We have used some of different types of datasets including test data sets of Synthetic databases, KDD cup'99 and Mushroom dataset and some randomly generated datasets by which we can described the effects of four different similarity measure functions upon Shared Nearest Neighbor (SNN) clustering approach. All these experiments were performed with the help of MATLAB 2010a (MATLAB 7.10).

Here, for experimentation, we used a 2D dataset containing 107 data points as shown in Figure- 1. We compute each result shown here by taking the following input parameters- k=7, Eps=4 and minPnt=5.

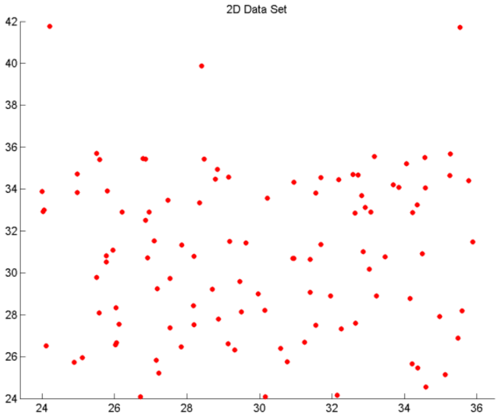
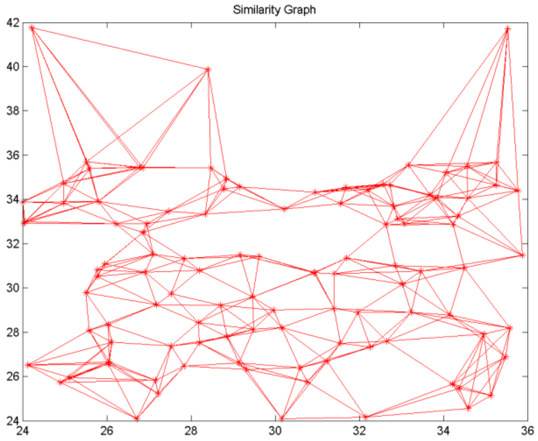


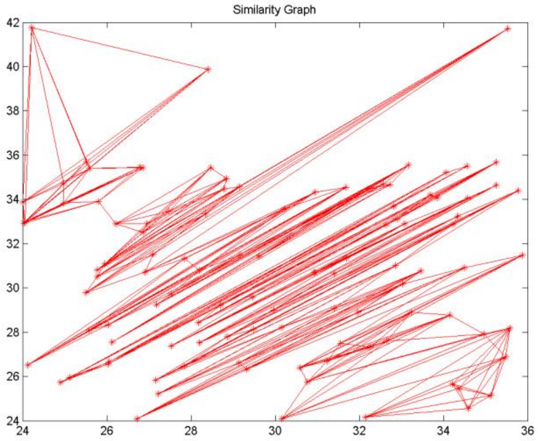
Fig 1: 2D Data Set

6. RESULT AND ANALYSIS

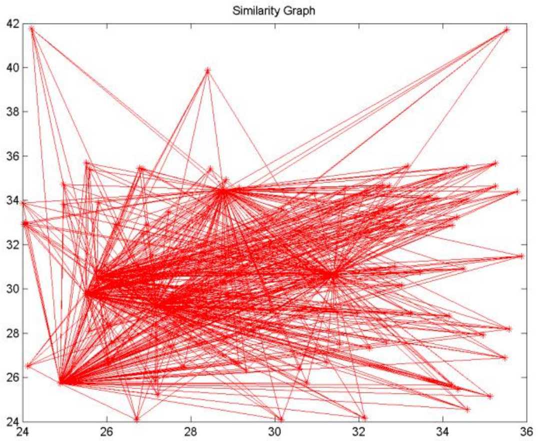
From data set shown in figure- 1, we first compute the similarity matrix by using the similarity measure functions- Euclidean, Cosine, Jaccard and Correlation functions and construct the sparsified similarity graph based on the k nearest neighbor criteria. Similarity graph generated by different similarity measure functions are shown in figure- 2.



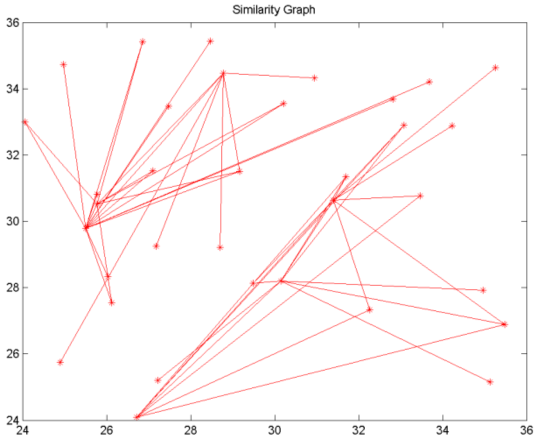
2(a) Similarity Graph generated by Euclidean function



2(b) Similarity Graph generated by Cosine function



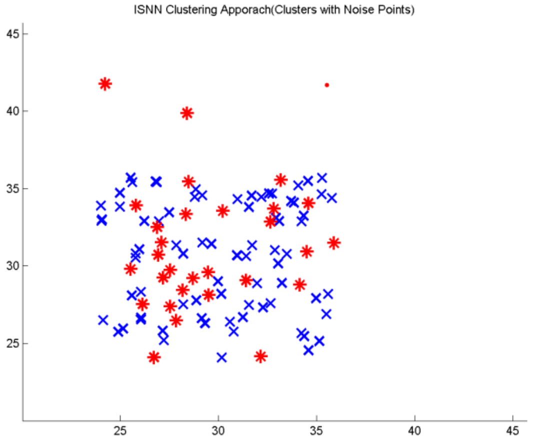
2(c) Similarity Graph generated by Jaccard function



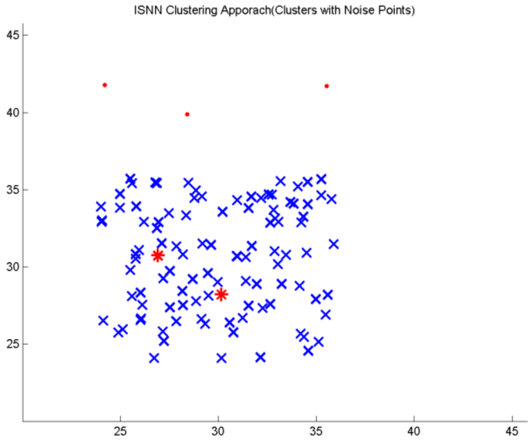
2(d) Similarity Graph generated by Correlation function

Fig 2: Similarity Graph generated by different similarity functions

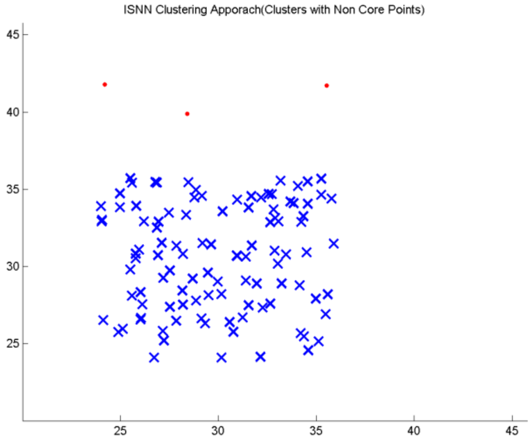
Similarity matrix calculation is most important part of SNN clustering approach. The comparison between similarity graphs is clear by their figures.



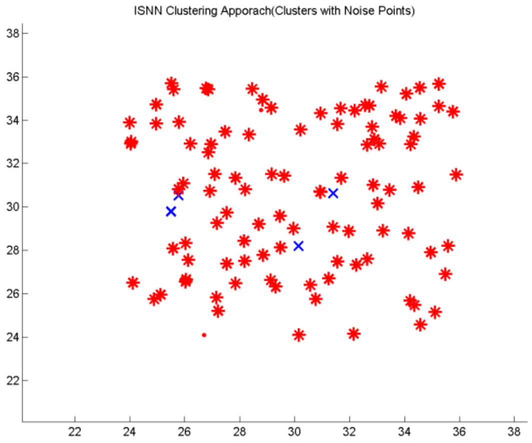
3(a) Clusters generated by Euclidean function



3(b) Clusters generated by Cosine function



3(c) Clusters generated by Jaccard function



3(d) Clusters generated by Correlation function

Fig 3: Clusters constructed by different similarity functions

After construction of similarity graph, we generate SNN graph and by applying user specified criteria- Eps and minPnt on this SNN graph, we compute core, noncore, and noise points. The clusters of core, noncore, and noise points by using different similarity functions are shown in figure- 3.

In figure- 3, X depicts the core point, dot (.) shows the noncore point, and star (*) conveys the noise points. We compared the Clusters constructed using different similarity function by their accuracy of generating clusters of core points.

We observed the following facts-

1. Clusters constructed by Jaccard and Cosine functions had no or very less noise points, Euclidean function had some noise points while clusters constructed using correlation function had lot of noise points, as shown in figure- 3.

2. In SNN clustering approach, Euclidean distance function performed better because not all the points are clustered in SNN clustering approach. Most of the data points are noises and hence removed.

3. If the complete clustering is desired, then it can be done by following two ways-

    a. Using Euclidean distance function, unclustered data can be inserted to the core clusters, discovered by SNN clustering approach and assigning them to the clusters containing the closest representative point.

    b. Using Jaccard or Cosine distance function, clusters can be constructed using SNN clustering approach.

4. We observed that generation of core, noncore, and noise points is dependent upon data points included in dataset and the user specified criteria k, Eps and minPnt.

5. If some points are clustered and others are removed as noise according to given specified criteria, then the clustering process performed faster.

7. CONCLUSION AND FUTURE WORK

In this paper, we have analyzed the impact upon SNN clustering approach (SNN) of different similarity computation functions and compared the resultant similarity graphs and clusters. From the above results, we can infer that the SNN clustering approach with Euclidean similarity measure function provides better and faster results as compared to the other distance functions described here. In future, we hope to analyze impacts of other different similarity measure functions upon various popular clustering techniques.

8. REFERENCES

[1] Levent Ertoz, Michael Steinback, Vipin Kumar, “Finding Clusters of Different Sizes, Shapes, and Density in Noisy, High Dimensional Data”, Second SIAM International Conference on Data Mining, San Francisco, CA, USA, 2003.

[2] Anna Huang, “Similarity Measures for Text Document Clustering”, *NZCSRSC 2008*, April 2008, Christchurch, New Zealand.

[3] Kazem Taghva and Rushikesh Veni, “Effects of Similarity Metrics on Document Clustering”, 2010 Seventh International Conference on Information Technology.

[4] R. A. Jarvis and E. A. Patrick, “Clustering Using a Similarity Measure Based on Shared Nearest Neighbors,” IEEE Transactions on Computers, Vol. C-22,

[5] M. R. Anderherg, “Cluster Analysis for Application”, Academic Press, New York, 1973.

[6] Jiawei Han, Micheline Kamber, “Data Mining: Concepts and Techniques”, Morgan Kaufmann Publishers, San Francisco, USA, 2001, ISBN 1558604898.

[7] Lori Bowen Ayre, ”Data Mining for Information Professionals”, 2006.

[8] Arun K Pujari, “Data Mining Techniques- Second Edition”, Universities Press. No. 11, November 1973.

[9] Martin Ester, Hans-Peter Kriegel, Jorg Sander, Xiaowei Xu, “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,” KDD 96, Portland, OR, pp. 226-231, 1996.

[10] Sudipto Guha, Rajeev Rastogi, Kyuseok Shim,“CURE: An Efficient Clustering Algorithm for Large Databases”, ACM, 1998.

[11] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim, *“*ROCK: A Robust Clustering Algorithm for Categorical Attributes”, In Proceedings of the 15th International Conference on Data Engineering, 1998.

[12] George Karypis, Eui-Hong Han, and Vipin Kumar, **“**CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling,” IEEE Computer, Vol. 32, No. 8,. pp. 68-75, August 1999.