A Novel Algorithm for Clustering High Dimensional Data

Isa Inuwa-Dutse, Xinyue Liu and Dejan Milojicic

***Abstract*---**The Challenges of Cluster Analysis and Related Work K-means is one of the most commonly used clustering algorithm, but it does not perform well on data with outliers or with clusters of different sizes or non-globular shapes. The single link agglomerative clustering method is the most suitable for capturing clusters with non-globular shapes, but this approach is very sensitive to noise and cannot handle clusters of varying density. However, most of the clustering challenges, particularly those related to "quality," rather than computational resources, are the same challenges that existed decades ago: how to find clusters with differing sizes, shapes and densities, how to handle noise and outliers, and how to determine the number of clusters. The general idea of our novel subspace outlier model is to analyze for each point, how well it fits to the subspace that is spanned by a set of reference points. The experimental evaluation showed that proposed method can find more interesting and more meaningful outliers in high dimensional data with higher accuracy than full dimensional outlier models by no additional computational costs.

***Keywords*---**Clustering, High-Dimensional, Nearest Neighbours, Data Points, Root Mapping.

I. Introduction

CLUSTERING in general is an unsupervised process of grouping elements together, so that elements assigned to the same cluster are more similar to each other than to the remaining data points [1]. This goal is often difficult to achieve in practice. Over the years, various clustering algorithms have been proposed, which can be roughly divided into four groups: partitional, hierarchical, density based, and subspace algorithms. Algorithms from the fourth group search for clusters in some lower dimensional projection of the original data, and have been generally preferred when dealing with data that are high dimensional [2], [3], [4], [5]. The motivation for this preference lies in the observation that having more dimensions usually leads to the so-called curse of dimensionality, where the performance of many standard machine-learning algorithms becomes impaired.

The difficulties in dealing with high-dimensional data are omnipresent and abundant. However, not all phenomena that arise are necessarily detrimental to clustering techniques. This paper that data points, which is the tendency of some data points in high-dimensional data sets to occur much more frequently in k-nearest neighbour lists of other points than the rest of the points from the set, can in fact be used for clustering. To our knowledge, this has not been previously attempted. In a limited sense, data points in graphs have been used to represent typical word meanings in [6], which were not used for data clustering. Our current focus was mostly on properly selecting cluster prototypes, with the proposed methods tailored for detecting approximately outlier spherical clusters.

II. Related Work

A. Density Based Clustering

Density based clustering [8] differentiates regions which have higher density than its neighbourhood and does not need the number of clusters as an input parameter. Regarding a termination condition, two parameters indicate when the expansion of clusters should be terminated: given the radius of the volume of data points to look for a minimum number of points for the density calculations has to be exceeded. Local scaling is a technique which makes use of the local statistics of the data when identifying clusters. This is done by scaling the distances around each point in the dataset with a factor proportional to its distance to its k^{th} nearest neighbour. Locally scaled densitybased clustering algorithm clusters points by connecting dense regions of space until the density falls below a threshold determined by the center of the cluster. In high-dimensional spaces this is often not easy to estimate, due to data being very sparse. There is also the issue of choosing the proper neighbourhood size, since both small and large values of *k* can cause problems for density based approaches [9].

B. K-means++

The *K-means++* is a specific way of choosing centers for the *k*-means algorithm. The relationship between *k-means++* clustering and data points was briefly examined in [10], where it was observed that data points may not cluster well using conventional prototype-based clustering algorithms (*K-means ++*) [7], since they not only tend to be close to points belonging to the same cluster (i.e., have low intra-cluster distance) but also tend to be close to points assigned to other clusters (low inter-cluster distance). The demonstrable gains of k-means++ over random initialization is precisely in the constantly updated non-uniform selection. The algorithm that works in a small number of iterations, selects more than one point in each iteration but in a non-uniform manner, and has provable approximation

Manuscript received on August 05, 2019, review completed on August 05, 2019 and revised on August 19, 2019.

Isa Inuwa-Dutse is with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.

Xinyue Liu is with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.

Dejan Milojicic is with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.

Digital Object Identifier: AIML082019002.

guarantees. Data points can, therefore, be viewed as (opposing) analogues of outliers, which have high inter- and intra-cluster distance, suggesting that data points should also receive special attention [10].

III. Proposed System

The proposed method identifies the patterns among data points and forms clusters of data points around these patterns. It operates by simultaneously considering all data point as potential patterns and exchanging messages between data points until a good set of patterns and clusters emerges. The root mapping and neighbour cluster is used to find the fitness value data points are exchanged between data points until a high-quality set of patterns and corresponding clusters gradually emerges.

A. Feature Selection

A "feature" or "attribute" or "variable" refers to a portion of the data points. Typically before collecting data, features are specified or preferred. Features can be discrete, continuous, or insignificant. Feature selection for high-dimensional data clustering is the task of disregarding irrelevant and redundant terms in the vectors that represent the data points, aiming to find the smallest subset of terms that reveals "natural" clusters of data points. To Searching for the small subset figure: 1 of relevant terms will speed up the clustering process, while avoiding the curse of dimensionality.

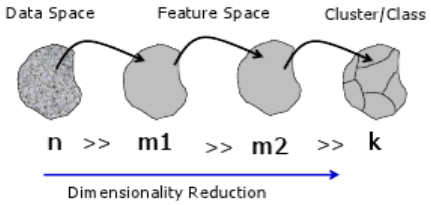

Fig. 1: Dimensionality Reduction

The Irrelevance filter removes irrelevant features using a modified form of the Relief algorithm, which assigns relevance values to features by treating training samples as points in feature space. For each sample, it finds the nearest "hit" (another sample of the same class) and "miss" (a sample of a different class), and adjusts the significance value of each feature according to the square of the feature difference between the sample and the hit and miss. Irrelevance Filter feature selection methods evaluate attributes prior to the learning process, and without specific reference to the clustering algorithm that will be used to generate the final result. The filtered dataset may then be used by any clustering algorithms.

B. Correlation of Root Mapping to Data Clusters

A correlation between low data points elements and outliers was also observed. A low-points score indicates that a point is on average far from the rest of the points and hence probably an outlier. In high-dimensional spaces, however, low data point elements are expected to occur by the very nature of these spaces and data resource. The root mapping can be applied using more general notions of similarity, and the similarities may be positive or negative. The output of the algorithm is unchanged if the similarities are scaled and/or offset by a constant (as long as the preferences are scaled and/or offset by the same constant). To compute fitness measure over the set of possible clusters and then chooses among the set of cluster candidates points those that optimize the measure used. To identify the cluster of a specific vertex or to group all of the vertices into a set of clusters, and then present possible cluster fitness measures that serve for methods that produce the clustering by comparing different groupings and selecting one that meets or optimizes a certain criterion. The ratio of the cluster is to minimum sums of degrees either inside the cluster or outside it. A fitness function is evaluated for all neighbours and the outcome is used to choose to which neighbour the search will proceed.

C. Neighbour clustering Algorithm

The neighbour clustering algorithm works message passing among data points. Each data points receive the availability from others data points (from pattern) and send the responsibility message to others data points (to pattern). Sum of responsibilities and availabilities for data points identify the cluster patterns.

The high-dimensional data point availabilities A (i, k) are zero: A(i, k) = 0, R (i, k) is set to the input similarity between point i and point k as its pattern, minus the largest of the similarities between point i and other candidate patterns.

The cluster responsibilities are computed using the equation,

$$R(i,k) \leftarrow S(i,k) - \max_{k' s.t. k' \neq k} \{A(i,k') + S(i,k')\} \quad (1)$$

In later iterations, when some data points are effectively assigned to other patterns, their availabilities will drop below zero. These negative availabilities will decrease the effective values of some of the input similarities S (i, k') in the above rule, removing the corresponding candidate from competition.

The above responsibility in equation (1) is update lets all data point patterns are compete for ownership of a data point, the following availability update gathers confirmation from data points as to whether each datas would make a good pattern:

$$A(i,k) \leftarrow min\left\{0, R(k,k) + \sum_{i' s.t. i' \in \{i,k\}} \max\{0, R(i',k)\}\right\} \quad (2)$$

The data links are sent from cluster members (data points) to candidate patterns (data points), indicating how well-suited the data point would be as a member of the candidate pattern cluster. The rot mapping and Neighbour clustering is iteratively computes data responsibilities and data availabilities to overcome the outlier points. The algorithm terminates if decisions for the patterns and the cluster boundaries are

unchanged for convict's iterations, or if maximum iterations are reached. The responsibilities and availabilities are messages that provide evidence for whether or not each data point should be in data points and if not to what outlier that data point should be assigned.

Algorithm 1: Neighbour Clustering Algorithm

Require: A**,** R**,** i, k

1. Initialize A (i, k) =0, R (i, k) = 0, k=0, and S (i, k) = 0 randomly
2. repeat
3. Update the data point responsibility by (1) where S (i, k) is the similarity of data points and root map pattern *k.*
4. Update the data point availabilities by (2)
5. Update self-availability by using (3)
6. Compute sum = A (i, k) + R (i, k) for data point *i* and find the value of *k* that maximize the sum to identify the data points.
7. If outlier points do not change for fixed number of iterations go to step 7 else go to step 1.

IV. Experimental Results

The proposed root mapping with neighbour clustering algorithm on Real-world data are usually much more complex and difficult to cluster, therefore such tests are of a higher practical significance. As not all data exhibit data points, the algorithms is tested both on intrinsically high-dimensional, high- data points and intrinsically low-to-medium dimensional, low-data. There were two different experimental setups. In the first setup, a single data set was clustered for many different K-s (number of clusters), to see if there is any difference when the number of clusters is varied. In the second setup, 20 different data sets were all clustered by the number of classes in the data (the number of different labels).

The clustering quality in these experiments was measured by two quality indices, the silhouette index and the isolation index [11], which measures a percentage of *k*-neighbour points that are clustered together. In the experimental setup, the two-part Miss-America data set (cs.joensuu.fi/sipu/datasets/) was used for evaluation. Each part consists of 6,480 instances having 16 dimensions. Results were compared for various predefined numbers of clusters in algorithm calls. Each algorithm was tested 50 times for each number of clusters. Neighbourhood size was 5. The highest level of noise for which we tested was the case when there was an equal number of actual data instances in original clusters and noisy instances. At every noise level, RMNC (root map with neighbour cluster), KM++, GHPC, and Global Hubness-Proportional K-Means (GHPKM) were run 50 times each.

The results for both parts of the data set are given in Table 1 and Table 2.

The Root Map and Neighbour Cluster (RMNC) is clearly outperformed GHPC, KM and other data-based methods. This shows that hubs can serve as good cluster center prototypes.

TABLE 1
CLUSTERING QUALITY OF SILHOUETTE INDEX ON THE MISS-AMERICA DATA SET

K	2	4	6	8	10	12	14	16
RMNC	**0.59**	0.42	0.31	0.28	0.19	0.17	0.13	0.1
GHPC	0.38	0.29	0.25	0.21	0.15	0.10	0.10	0.09
KM++	0.14	0.12	0.09	0.08	0.07	0.07	0.07	0.07
GHPKM	0.28	0.18	0.17	0.14	0.13	0.11	0.10	0.08

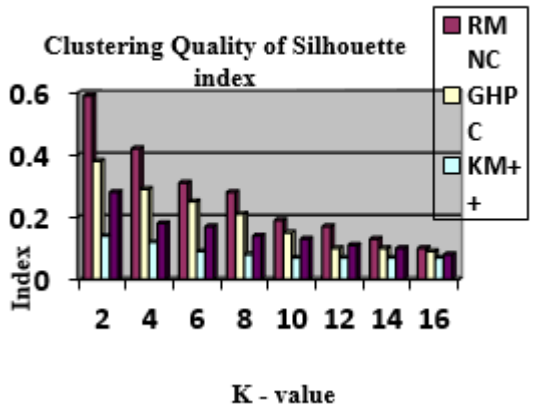

Fig.2: Clustering Quality of Silhouette Index.

TABLE 2
CLUSTERING QUALITY OF ISOLATION INDEX ON THE MISS-AMERICA DATA SET

K	2	4	6	8	10	12	14	16
RMNC	**0.94**	0.92	0.79	0.58	0.51	0.49	0.36	0.29
GHPC	0.91	0.89	0.71	0.53	0.42	0.33	0.30	0.26
KM++	0.62	0.46	0.34	0.23	0.19	0.16	0.13	0.12
GHPKM	0.85	0.54	0.45	0.38	0.29	0.26	0.24	0.23

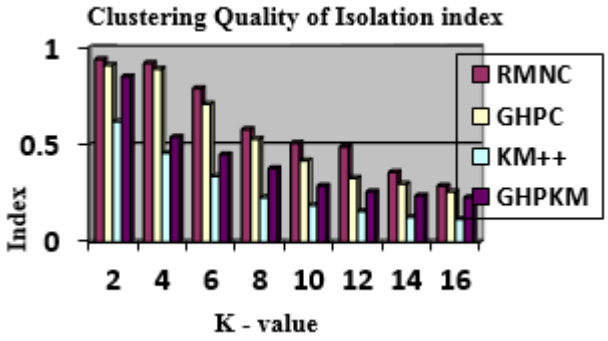

Fig. 3: Clustering Quality of Isolation Index.

V. Conclusion

The proposed method of RMNC method had proven to be more robust than the GHPKM and K-Means++ baseline on both

synthetic and real-world data, as well as in the presence of high levels of artificially introduced noise. The root map with neighbour clustering can easily be extended to incorporate additional pair-wise constrains such as requiring points with the same label to come into view in the same cluster with just an extra layer of function hubs. The model is flexible enough for information other than explicit constraints such as two points being in different clusters or even higher-order constraints (e.g., two of three points must be in the same cluster).

REFERENCES

[1] Gorunescu, Florin. Data Mining: Concepts, models and techniques. Vol. 12. Springer Science & Business Media, 2011.

[2] Günnemann, Stephan, et al. "Subspace correlation clustering: finding locally correlated dimensions in subspace projections of the data." Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012.

[3] Kriegel, Hans-Peter, Peer Kröger, and Arthur Zimek. "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering." ACM Transactions on Knowledge Discovery from Data (TKDD) 3.1 (2009): 1.

[4] Kriegel, Hans-Peter, et al. "Density-based clustering." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1.3 (2011): 231-240.

[5] Yuan, Xiaoru, et al. "Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data." IEEE Transactions on Visualization and Computer Graphics 19.12 (2013): 2625-2633.

[6] Nasiruddin, Mohammad. "A state of the art of word sense induction: A way towards word sense disambiguation for under-resourced languages." arXiv preprint arXiv: 1310.1425 (2013).

[7] Bachem, Olivier, et al. "Fast and provably good seedings for k-means." Advances in Neural Information Processing Systems. 2016.

[8] Correa, Carlos D., and Peter Lindstrom. "Locally-scaled spectral clustering using empty region graphs." Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012.

[9] Meyer, Fernand, and Jean Stawiaski. "Morphology on graphs and minimum spanning trees." International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing. Springer, Berlin, Heidelberg, 2009.

[10] Muja, M., & Lowe, D. G. (2014). Scalable nearest neighbor algorithms for high dimensional data. IEEE transactions on pattern analysis and machine intelligence, 36(11), 2227-2240.

[11] Tao, Jun, Chaoli Wang, and Ching Kuang Shene. "FlowString: Partial streamline matching using shape invariant similarity measure for exploratory flow visualization." 2014 IEEE Pacific Visualization Symposium. IEEE, 2014.

[12] Angiulli, Fabrizio, and Clara Pizzuti. "Fast outlier detection in high dimensional spaces." European Conference on Principles of Data Mining and Knowledge Discovery. Springer, Berlin, Heidelberg, 2002.

[13] Zhang, Y., & Telesca, D. (2014). Joint clustering and registration of functional data. arXiv preprint arXiv:1403.7134.