The Thirty-First International Florida
Artificial Intelligence Research Society Conference (FLAIRS-31)

Improving Topic Model Visualization via Multi-Dimensional Scaling and Cliques

King-Ip (David) Lin, H. Andrew Kim
Computer Science Department
Baylor University
Waco, TX 76798

Abstract

We propose representing high-dimensional data in 2-dimensions using cliques mapped onto several planes. Currently, Multidimensional Scaling (MDS) projects every point onto an R-space medium. However, this may not produce the most ideal result some relations between points may exhibit higher stress than others. We propose utilizing cliques to extract a complete subset of points into separate facets in order to convey the most accurate distance representation as possible therefore achieving low stress in each instance.

Introduction

One aspect in machine learning/data mining that is often overlooked is the presentation of the results. While advances in the field have led to new models for knowledge, presenting them in an understandable and effective manner is crucial for them to gain wide acceptance and usefulness.

For instance, topic modeling methods such as Latent Dirichlet allocation (LDA) (Biel and Lafferty 2009) have emerged as an important analytical tool for text mining and analysis. LDA discover a set of topics from a large corpus of documents. Each document is represented as a probability distribution of the words in the corpus. This capture the idea of different topics are described via different sets of words. LDA is an effective and automated method to make sense of the various themes from a non-annotated corpus. However, one problem is how to present the topics to the user. One can show the list of words and their probabilities, but that may not appeal to many potential users without a strong sense of probability distributions.

One natural way of visualizing the topics is to make use of the statistical distance measures defined for probability distributions such as Bhattacharyya distance and Hellinger distance. Given that, one can then apply multidimensional scaling (MDS) to map the topics into points in a (2 or 3)-dimensional Euclidean space, which can be visualized in a very intuitive fashion. This also allows us to get around the problem of high dimensionality, as each topic is a probability distribution of a large number of words. Examples of applying MDS to topic visualization are found in (Fortuna, Grobelnik, and Mladenic 2005; Sievert and Shirley 2014).

In this paper, we propose improvements of applying MDS to topic visualization. One observation is that no matter how good the mapping is, there are bound to be pairs of points which distance in the Euclidean space is very different from the actual distance of their corresponding topics. That means very often the mapped points in the MDS convey a false sense of relationship about the actual data. This is a recurring problem for MDS regardless of whether the objects mapped are topics or not. To overcome this, we propose using multiple mappings for visualization. Instead of mapping all the points into a single space, we partition the data into groups, and visualize each group individually. The goal is that each group is mapped to a space such that distances between every pair of points within the group are well preserved.

Our proposed method is iterative. At each iteration, we first run MDS on the current group to map the topics to points. Then we create a graph linking pairs of points in which the mapped distance is close to the actual distance. Each group that we are looking for corresponds to cliques in such graph. Once we find a clique we remove them from the set and form a group. We then repeat the process until we either run out of points, or the number of groups generated becomes too large. This process enable us to generate multiple mapping that are highly reliable and minimize potential misleading information.

Subsequent sections will describe the method in more detail, as well as present initial experimental results to provide a sense of how our method performs.

Our approach

Multi-dimensional Scaling (MDS)

Multidimensional scaling was originally introduced as a mathematical technique in the field of psychology and political science to aid researchers to uncover the hidden structure of databases (Kruskal and Wish 1978). In MDS, the input is a set of objects such that for any pair O_i, O_j, there is a measure of dissimilarity p_{ij} between them. MDS maps the objects into an R dimensional Euclidean space where every object becomes a point. Thus each pair of objects has an Euclidean distance d_{ij}. The mapping is chosen such as to minimize the stress function:

$$\text{Stress-1} = \sigma_1 = \sqrt{\frac{\Sigma[f(p_{ij}) - d_{ij}(\mathbf{X})]^2}{\Sigma d_{ij}^2(\mathbf{X})}} \quad (1)$$

Our objective here is to minimize `Stress-1` in order to obtain an ideal spatial configuration **X** which would accurately represent the proximity as closely as possible. Algorithms such as SMACOF has been developed to minimize the function above and map the objects into the corresponding points. The interested reader may want to consult (Borg and Groenen 2005) for more information.

We also want to note that there has been effort in visualize results of MDS, including those of GGvis/GGobi (Buja et al. 2008), and the work by Chen et. al. (Chen, Hrdle, and Unwin 2008)

Our approach: MDS Clique

As mentioned, our goal is to make the visualization more reliable/representative. We accept the fact that MDS cannot perfectly match the objects into the Euclidean space perfectly (with no stress). However, we assume there is a level of tolerance, k, such that the algorithm will accept a mapping such that for any pair of objects i and j, the difference d_{ij} and p_{ij} must be $\leq k$.

To achieve this, we start out by running MDS to map all the objects into a Eucliedan space. After that we construct a graph M such that every object is a vertex, and an edge exists between a pair of vertices if the difference ind_{ij} and p_{ij} between them is less than some constant k. Then we run a maximal clique algorithm on M. This clique found from the algorithm corresponds to the set of points that satisfies the requirement above. Note that we are not simply removing edges that are far apart from each other in Euclidean space. Rather, we are removing edges that corresponds to pairs of objects that the mapped points does not match the original objects in terms of distance.

After that we take the remaining items to repeat the process. Notice that we have to map those points into a new multi-dimensional space, as the space generated in the last iteration does not provide a completely faithful representation of them. We repeat the same process above, until all the points are in one of the groups. This is summarized as 1

Algorithm 1: MDS Clique Algorithm

```
Result: Separate cliques given $k$
Let $T$ be the set of all points
do
    Run MDS on $T$, generate graph $M$ with edges
      representing stress between points
    Generate graph $M'$ from graph $M$ by removing
      edges with stress > k (k represents a
      particular threshold)
    Find the maximum clique $C$
    Record clique $C$
    Consider all vertices $V$ that is in $M$ but not in $C$
    $T :=$ all points corresponding to $V$
while $T$ is not empty;
```

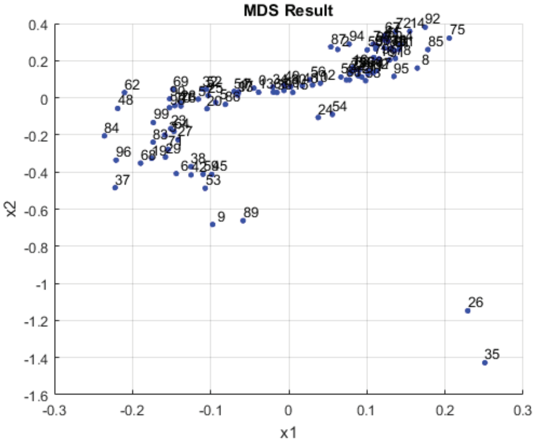

Figure 1: Single MDS map for the 100 topics on the Reuters-27158 data set

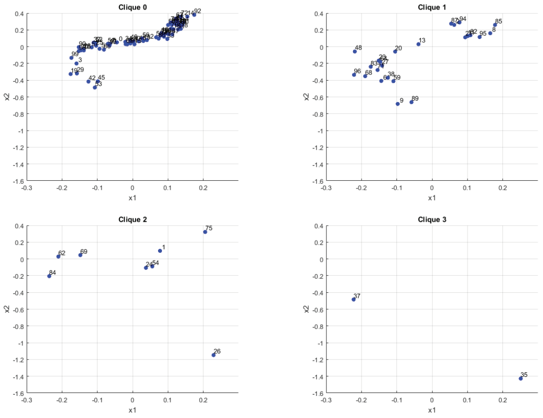

Figure 2: MDS-Clique map for the 100 topics on the Reuters-27158 data set

There are multiple ways of specifying k. If one have knowledge about distance values, one can set k to be a ratio between the difference between p_{ij} and d_{ij} and d_{ij} . On the other hand, one can set k to be the mean of the difference between p_{ij} and d_{ij}, and if a tighter bound is needed, substract a number of the standard deviation from it.

We illustrate our algorithm with an example. We applied LDA on the standard Reuters-21758 data set, generating 100 topics. We then apply MDS to map every topic to a single space, as well as applying our algorithm. Figure 1 shows the result of mapping all 100 topics in a single space. Figure 2 shows the 4 cliques that are generated by running our algorithm. In this case we set the threshold to be the (mean - standard deviation) of the difference between the distance of each pair of topics and the distance of the corresponding mapped points.

Most of the data resides on the first 2 cliques. While there are similarity between the two mappings (e.g. clique-1 is the exact mapping to the original mapping), there are also some differences. For instance, figure 3 shows the second clique

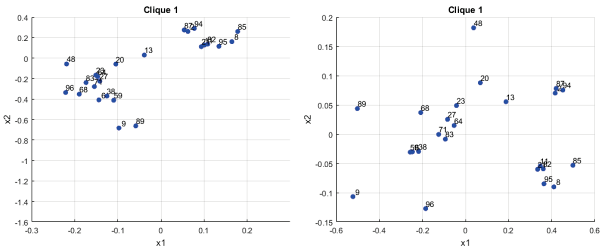

Figure 3: Compare cliques on the Reuters-27158 data set

(right hand side) and the corresponds points in the original mapping (left hand side). While the general grouping of the points are similar in both cases, there are some significant difference in relative location of the points. For example, topic 48, 9, 89 and 96's relative position and distance are quite different between the two mappings.

Experiments

Overview

We test our implementation of MDS-clique to evaluate its performance. We want to observe how much do our algorithm improve on the mapping. We also want to observe if the number of cliques generated is reasonable, as well as how the points are distributed.

Data sets Two types of data sets are used. The first (denoted as MATRIX) is defined by generating a random dissimilarity matrix of objects with a specified number of clusters, each with the same number of elements). Intra-cluster distance of objects is set to a small value (between 0.01 and 0.1) while the inter-cluster distance set to a large value (between 1.75 and 2.0).

The second dataset (denoted as 3D) is formed by generating points in a 3-dimensional space where their dissimilarities are measured by the Kullback–Leibler dissimilarity function (so to ensure the MDS mapping will be different). We also generate the points such that the lies in separate (spatial) clusters.

For each type of data we generate 1,067 samples, each with 100 points. With space limitation, we report the results only on the MATRIX data as the results of the 3D data shows similar trends.

Evaluation of results The main criteria for evaluation is the Kruskal's stress measure as noted in Equation 1. We measure the stress of data generated by MDS-clique vs. single MDS (control). we are interested in comparing the overall mean between the two groups to prove that the test group has a statistically significant using the t-test (with confidence level being 95% ($\alpha = 0.05$) two-tailed) signifying whether a particular algorithm performs better (or worse) than regular MDS.

We took two ways to compare the stress. For *Overall Contral Stress*, measures the global mean of all the stress gathered from all the experiments irrespective of which clique it originated from. Notice that the MDS-clique generate fewer pairs of objects for comparison (as objects from different clique are not compared) and thus have a big advantage if we take the overall value. So we average out the stress on a per edge basis for fair comparison.

Average	Mean	$\sigma_{\bar{x}}$
Control Average Stress Per Edge	2.3364	±0.1483
Clique Average Stress Per Edge	0.1313	±0.0044
Sample Control Stress	6.7546	±0.8000
Sample Clique Stress	0.9588	±0.0159
# of Cliques	10.7917	±0.1097

Table 1: Removing edges by standard deviation for MATRIX

Average	Mean	$\sigma_{\bar{x}}$
Control Average Stress Per Edge	2.5397	±0.3028
Clique Average Stress Per Edge	0.0598	±0.0020
Sample Control Stress	3.8419	±0.2911
Sample Clique Stress	0.2596	±0.0058
# of Cliques	15.2396	±0.1050

Table 2: Removing edges by distance measure, $N = 100$, $C = 3$, $k = 0.1$

The *Sample Control Stress* first takes the average of the stress in each clique of the MDS-clique and then reports the mean of all the concatenated means together along with the margin of error. In this case we only compares the stress values between the pairs in each clique with only their respective pairs in the control group for an apples-to-apples comparison.

Other metrics we report are the number of cliques on average each sample generates by MDS-Clique. Generally speaking, these smaller the number of cliques (while keeping the overall stress low), the better for an ever better user experience. Ideally, we would have only one plane with a stress of 0 which is the objective of MDS itself.

Experiment: MDS Clique

We first look at the stress value of the data generated by MDS-clique. We looked at the results for both ways of setting k (mean/standard deviation vs. constant). .

In both cases we see a significant decrease in stress, both on a per edge basis and a per clique basis. The result also suggest that there are significant discrepancies of the distances and dissimilarity in the basic MDS (that MDS-clique is able to exploit). As for the two methods, the distance measure method provides the better performance, but come as a price of splitting the data into more cliques.

Object distribution among cliques Most of the points generated by both the distance measure and the standard deviation reveals that most of the points are included in the initial cliques and gradually tapers off. Figure 4 reveals the standard deviation measure typically has around 15 points on average in its first clique and has a steeper drop-off than the distance measure which is more evenly spread out.

A couple of interesting observations emerge from these numbers; the first clique (which corresponds to the origi-

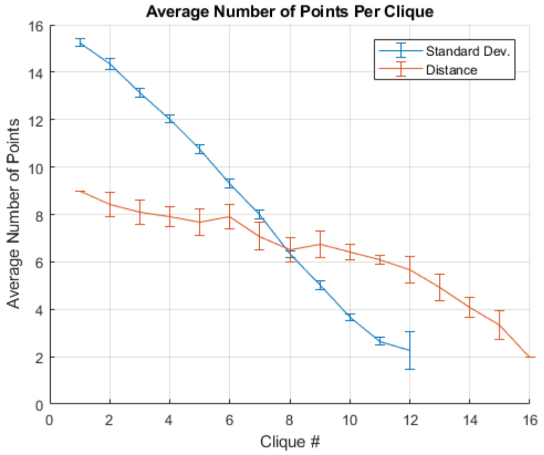

Figure 4: Average number of points per clique generated from 384 samples with Standard Dev. $k = \bar{x} - s$ and Distance $k = 0.25$

nal MDS) has roughly 15% of the points. This re-emphasize the fact the there are quite a bit of discrepancies of the distances between the MDS result with the original dissimilarity. On the other hand, at least via the mean/standard deviation method, most of the points are captured in the first few cliques. For instance, by clique 6 roughly 80% of the points are captured. So the user does not have to look at a large number of cliques to get the full view of the data.

Conclusion and Future Work

Overall, MDS-Clique offers us a viable alternative to display high-dimensional data easily in a human-readable format. The primary objective of MDS-Clique is to overcome the limitation of MDS where not all the points projected onto a single medium necessarily has a faithful representation. Not all projections are created equal in this case, so taking a multifaceted approach where each clique represents a faithful representation has been proven by the Kruskal stress value as shown in the experiments above.

Visually optimizing the spatial configuration of the various cliques which are represented as planes can offer an ideal representation of the separated data to the end user. Previous work have focused on a multifaceted view where the internal and external relations are characterized by the similarity of the data according to their pre-assigned cluster group label. In this case, the relations are characterized by the stress formula and further work can be done in order to present a better view to the end user by arranging the clique in a visual space in an optimal manner that reduces the overall stress among the cliques.

References

Biel, D. M., and Lafferty, J. D. 2009. Topic models. In Ashok N. Srivastava, M. S., ed., *Text mining: classification, clustering, and applications*. Chapman and Hall/CRC. chapter 4, 71–105.

Borg, I., and Groenen, P. J. 2005. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media.

Buja, A.; Swayne, D. F.; Littman, M. L.; Dean, N.; Hofmann, H.; and Chen, L. 2008. Data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics* 17(2):444–472.

Chen, C.-h.; Hrdle, W.; and Unwin, A. 2008. *Handbook of data visualization*. Berlin;London;: Springer.

Fortuna, B.; Grobelnik, M.; and Mladenic, D. 2005. Visualization of text document corpus. *Informatica* 497.

Kruskal, J. B., and Wish, M. 1978. *Multidimensional Scaling*. Number no. 07-011 in Sage University Paper. Quantitative Applications in the Social Sciences. SAGE Publications, Inc.

Sievert, C., and Shirley, K. 2014. Ldavis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, 63–70. Association for Computational Linguistics.