IEEE Visualization Conference, Poster Extended Abstract. Phoenix, AZ, October 2017

Visualization of Multivariate Data with Network Constraints using Multi-Objective Optimization

Bhavya Ghai[*] Alok Mishra[†] Klaus Mueller[‡]

Computer Science Department, Stony Brook University

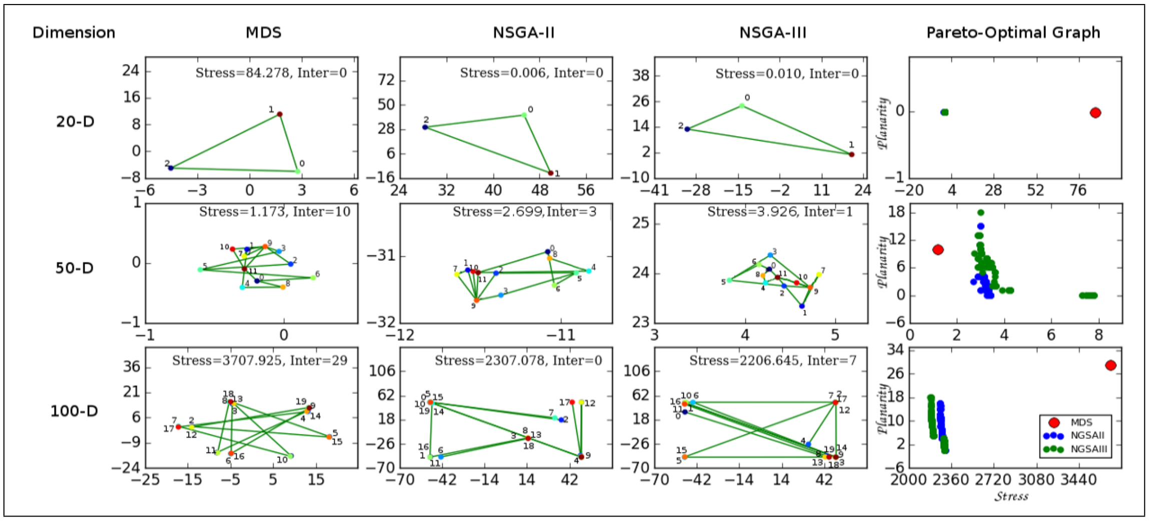

Figure 1: Comparasion of Metric MDS, NSGA-II and NSGA-III using 3 datasets with 20, 50 and 100 dimensions respectively. Pareto-Optimal graph shows the tradeoff between Planarity and Stress

Abstract

Dimensionality reduction techniques play a key role in data processing and data visualization. Common dimensionality reduction techniques such as MDS, PCA, etc. do a decent job in projecting high-dimensional data into lower dimensions. But in the case of network data, they simply ignore the relationship between nodes which might result in non-planar graphs with many intersecting edges. In this paper, we have tried to model dimensionality reduction for network data as a multi-objective optimization problem. We have tried to draw graph/network in lower dimensions such that planarity is maximized and stress function is minimized simultaneously. We have used two genetic algorithms namely, NSGA-II and NSGA-III. For them, both objectives are equally important and they optimize them together. These techniques return a set of non-dominated solutions represented by pareto-Optimal front. We observed that genetic algorithms outperformed MDS for some cases. In other cases, genetic algorithms gave solutions with significantly lower number of intersections for slight increase in stress value.

Index Terms: Network Visualization, Graph Visualization, High-Dimensional data, Genetic Algorithms, Dimensionality Reduction Techniques, Multivariate Data

1 Introduction

In today's digital era, Its crucial to use effective information visualization (InfoVis) techniques to make sense of different categories of data. In this study, we'll be focusing on multivariate network/graphs i.e. each node in the graph/network might lie in n-dimensional space. Popular dimensionality reduction techniques like MDS [5], PCA, etc. have been effective in dealing with high-dimensional data. When applied to multivariate network data, these techniques completely ignore the relationship between points/nodes. Hence, we might get a non-planar graph which has large number of edge intersections. Such a plot might be difficult to understand and analyze the patterns. In this work, we have tried to address this problem by developing a new model to reduce network/graph data via genetic algorithms. Our model takes n-dimensional points along with edges as Input. We have used NSGA-II [3] and NSGA-III [4] such that resulting graph resembles original graph while having minimum number of edge intersections. This is achieved by optimizing planarity and stress function simultaneously. Our model can be employed in visualizing numerous real world multivariate networks like communication, social, financial, internet networks, etc. For example, in social network each person can be modeled as a node. Each node can have multiple attributes like name, age, gender, number of friends/followers, number of posts, etc. Nodes can be linked together if one person is a friend/follower of other. Our model is designed to visualize such multivariate networks effectively.

2 Objective

This problem of dimensionality reduction can be visualized as searching for an optimal configuration of points in lower dimension space. The term 'Optimal' here means the configuration which satisfies two objectives i.e. minimize stress and maximize planarity.

2.1 Stress

Stress is a loss function which quantifies the preservation of relative distances in higher dimensional space to lower dimension space. If relative distances between points in lower dimension is exactly

[*] e-mail: bghai@cs.stonybrook.edu

[†] e-mail: almishra@cs.stonybrook.edu

[‡] e-mail: mueller@cs.stonybrook.edu

same as higher dimension space, then stress will be 0. In all other cases, it will have a positive value which will increase as pairwise distances deviates from original distances. There are multiple ways to calculate Stress [6]. In our case, we are using metric MDS as base model. Hence, we are using metric MDS stress function as shown in Eq 1.

$$Stress = \sqrt{\sum_{i\neq j=1,..,N} (D_{ij} - d_{ij})^2} \tag{1}$$

Here, D_{ij} and d_{ij} represents the (i,j) element of the dissimilarity matrix between points in higher dimension and lower dimensional space respectively. Generally, Euclidean distance is used to calculate the dissimilarity matrix. Since, we are dealing with high dimensional space, Euclidean distance might not be the most appropriate due to curse of dimensionality. Based on the research findings in [1], we have used Manhattan distance over Euclidean distance for calculating stress function.

2.2 Planarity

In graph theory, a graph is said to be planar if no two edges intersect. We have tried to quantify planarity in terms of number of intersections. In our algorithm, we have tried to maximize planarity by minimizing number of intersections. Each edge in the graph can be considered as a line segment with given end points. We found total number of intersections by comparing each pair of such line segments [7].

3 Methodology

Suppose the input graph has n-points with x dimensions each and we need to reduce them to y dimensions. We have used NSGA-II and NSGA-III to perform this optimization task [1] using $y = 2$. We choose population size P and calculate the dissimilarity matrix D for n-points in higher dimensional space. Each population member p_i represents a solution in reduced space. Each p_i is encoded as a concatenated set of n-points points in lower dimension space as shown in Eq 2.

$$p = \{a_1, a_2, a_3, ..., a_n\} \tag{2}$$

In our case where $y = 2$, a_i will represent a point in 2-D space. Initially, each p_i is assigned a set of points randomly. In every generation/iteration, we calculate number of intersections(planarity) and dissimilarity matrix d for each population member. Using d and D, we calculate the stress value for each p_i. Based on stress and number of intersections, each p_i moves in the search space towards the optimal solution. The algorithm terminates after a pre-defined number of iterations and returns a set of population members at the pareto-optimal front. Each of such population members represent a non-dominated solution in lower dimension space.

4 Results

In Figure 1, we have to tried to compare Metric MDS, NSGA-II and NSGA-III using 3 self-constructed datasets. Each of the 20D, 50D and 100D datasets are reduced to 2-dimensions. The first 3 columns represent the reduced output graph for MDS, NSGA-II and NSGA-III. The fourth column tries to compare the 3 algorithms based on planarity and stress value. In the first case, When there are no intersections in input graph, Genetic algorithm returns similar result to MDS. In other words, Since input graph has no intersections, Genetic algorithms just optimized stress. Hence, they produce similar result to MDS. In the second case, Our algorithm performed better on planarity for a slight increase in stress. In the last case, NSGA-II and III outperformed MDS on both objectives.

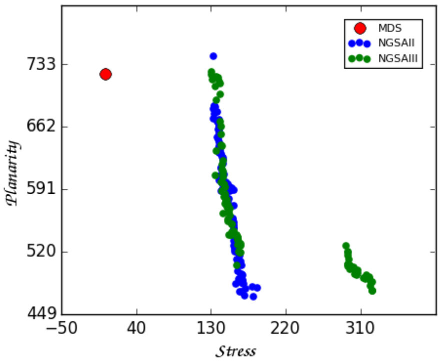

Figure 2: Pareto-Optimal graph representing the comparasation between Metric MDS, NSGA-II and NSGA-III for World Soccer Dataset

Apart from our own datasets, we also tested our algorithm on World Football Dataset [2]. As expected, Genetic algorithms performed fairly well on planarity for slight increase in stress as can be seen in Figure 2.

5 Conclusion

In this work, we have tried to develop a dimensionality reduction technique for network data inspired by metric MDS model using Genetic algorithms. We tried to optimize stress and planarity simultaneously. Given suitable time and parameter tuning, we observed that our model performed significantly better than MDS for some cases (in terms of stress and planarity). In other cases, our model gave solutions with significantly lower number of intersections for slight increase in stress value. Based on the requirement, end user is free to choose any solution from pareto-optimal front.

Genetic algorithms are computationally expensive so we intend to develop parallelized version of genetic algorithms and execute them on GPU for better performance. Future work might include exploring other optimization models which might achieve comparable accuracy in less time. We would also like to investigate the effectiveness of this model for different types of graphs. Lastly, we'll like to incorporate edge bundling for dense graphs.

6 Acknowledgments

This research was partially supported by NSF grant IIS 1527200.

References

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional space. In *International Conference on Database Theory*, pp. 420–434. Springer, 2001.

[2] V. Batagelj and A. Mrvar. Pajek datasets. `http://vlado.fmf.uni-lj.si/pub/networks/data/sport/football.htm`, 2006.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[4] H. Jain and K. Deb. An improved adaptive approach for elitist nondominated sorting genetic algorithm for many-objective optimization. In *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 307–321. Springer, 2013.

[5] J. B. Kruskal and M. Wish. *Multidimensional scaling*, vol. 11. Sage, 1978.

[6] A. Lee. pyswarm: Particle swarm optimization (pso) with constraint support. `https://pythonhosted.org/pyswarm/`, 2014.

[7] J. Qu, Y. Song, and S. Bressan. Psogd: A new method for graph drawing. In *Data Engineering Workshops (ICDEW), 2013 IEEE 29th International Conference on*, pp. 229–235. IEEE, 2013.

[1] Source Code - https://github.com/bhavyaghai/Visualizing-Network-Data