Generalized Dot Product Models for Weighted Networks

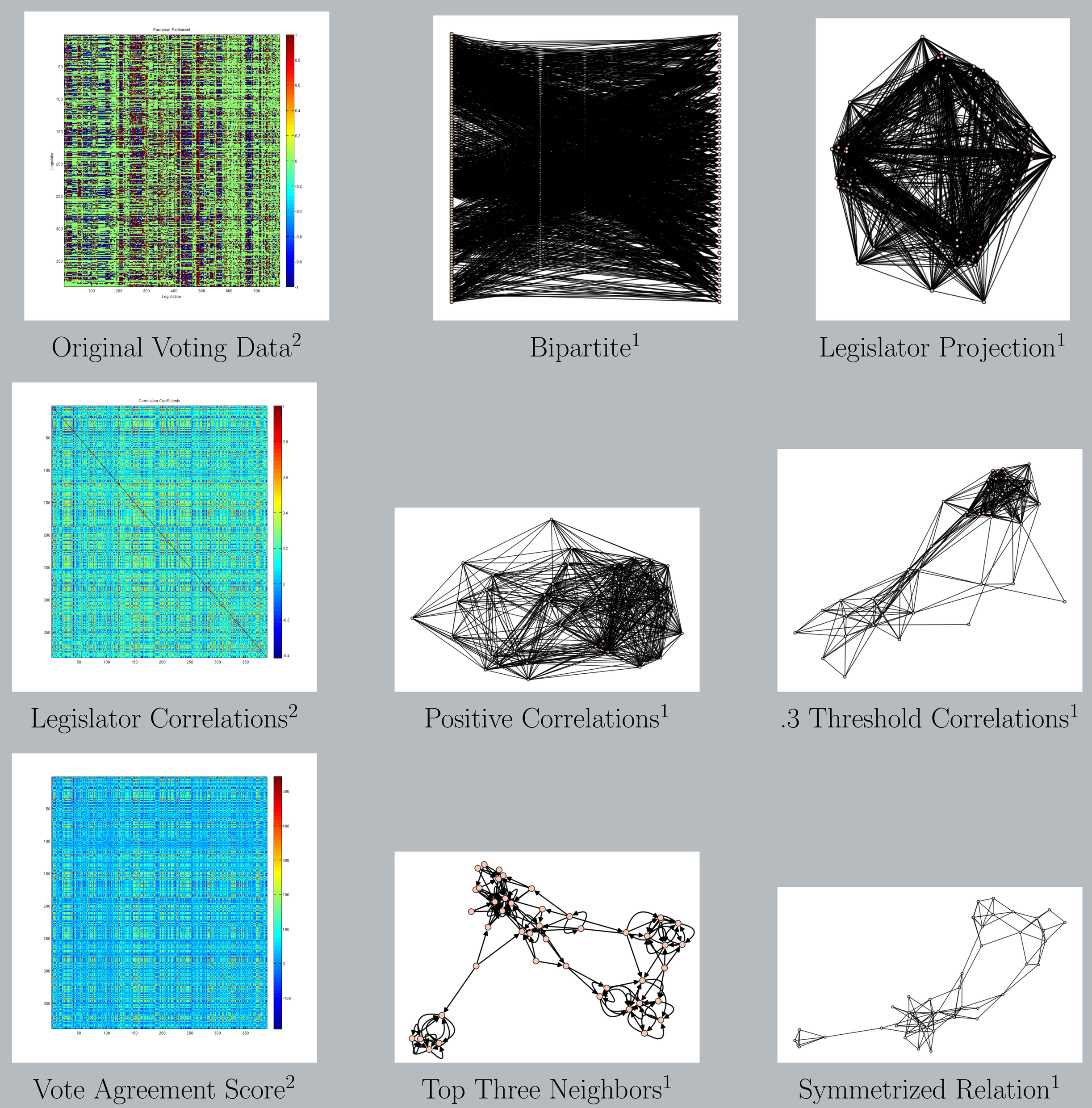
Daryl DeFord
Advisor: Dan Rockmore
Department of Mathematics

Abstract

Random network models provide an important tool for understanding complex systems by generating null models for metric comparisons. However, the most common random network models, such as Erdös–Rényi, Barabasi–Albert, and Watts–Strogattz, construct unweighted networks. We present a new generative model for multigraphs and weighted graphs, based on the Random Dot Product Model, that allows us to analyze these networks from a geometric perspective. Building on previous results, we show that this model admits several desirable generative properties that match expected results about many families of networks. As an inferential model, the geometric interpretation admits a linear algebraic algorithm for associating vectors to nodes and we introduce a natural stress function that allows for appropriate dimension selection.

Problem: Network Formulations

Complex networks are used to model all sorts of physical and social systems. However, the process of extracting a useful network model from a noisy data set requires making a series of decisions that determine the properties of the eventual network. Some of these choices are suggested by the data, such as the categorization of nodes and edges, while others may be determined by the mathematical tools available, such as the choice between digraphs and undirected models or between simple networks and weighted networks. Finally, some choices, such as the selection of thresholding parameters, are influenced by many factors and can change the resulting network in subtle and complicated ways. The figures below show several different representations of voting patterns in the European Parliament in 1979–1984 [3].



Original Voting Data[2] | Bipartite[1] | Legislator Projection[1]
Legislator Correlations[2] | Positive Correlations[1] | .3 Threshold Correlations[1]
Vote Agreement Score[2] | Top Three Neighbors[1] | Symmetrized Relation[1]

Figure 1: Network representations of European Parliament voting

Perturbations of the chosen network representation do not necessarily accurately reflect changes in the underlying data. Thus, null models that match parameters from a particular network may fail to give accurate comparisons.

Related Generative Models

Our model is an extension of the Random Dot Product Model (RDPM) introduced by Scheinerman and Young [9]. The RDPM is a latent space model, with pairwise connection probabilities defined by the dot products of the associated vectors. Scheinerman and Young showed that, for a broad class of initial distributions, the RDPM generates networks that have properties, such as short average path length and high clustering, that are commonly seen in social networks [9]. Later, Scheinerman and Tucker gave an efficient algorithm for estimating the latent vectors from a given network [6]. The RDPM process also motivates a particular adjacency spectral embedding that has proved to be useful for proving consistency results about stochastic block model derived graphs [8].

Recently, several other generative models have been developed for weighted networks [1, 5, 7]. These methods can be realized as special cases of our model, either by limiting the dimension of the latent space or restricting to finite distributions.

Our Model (WDPM)

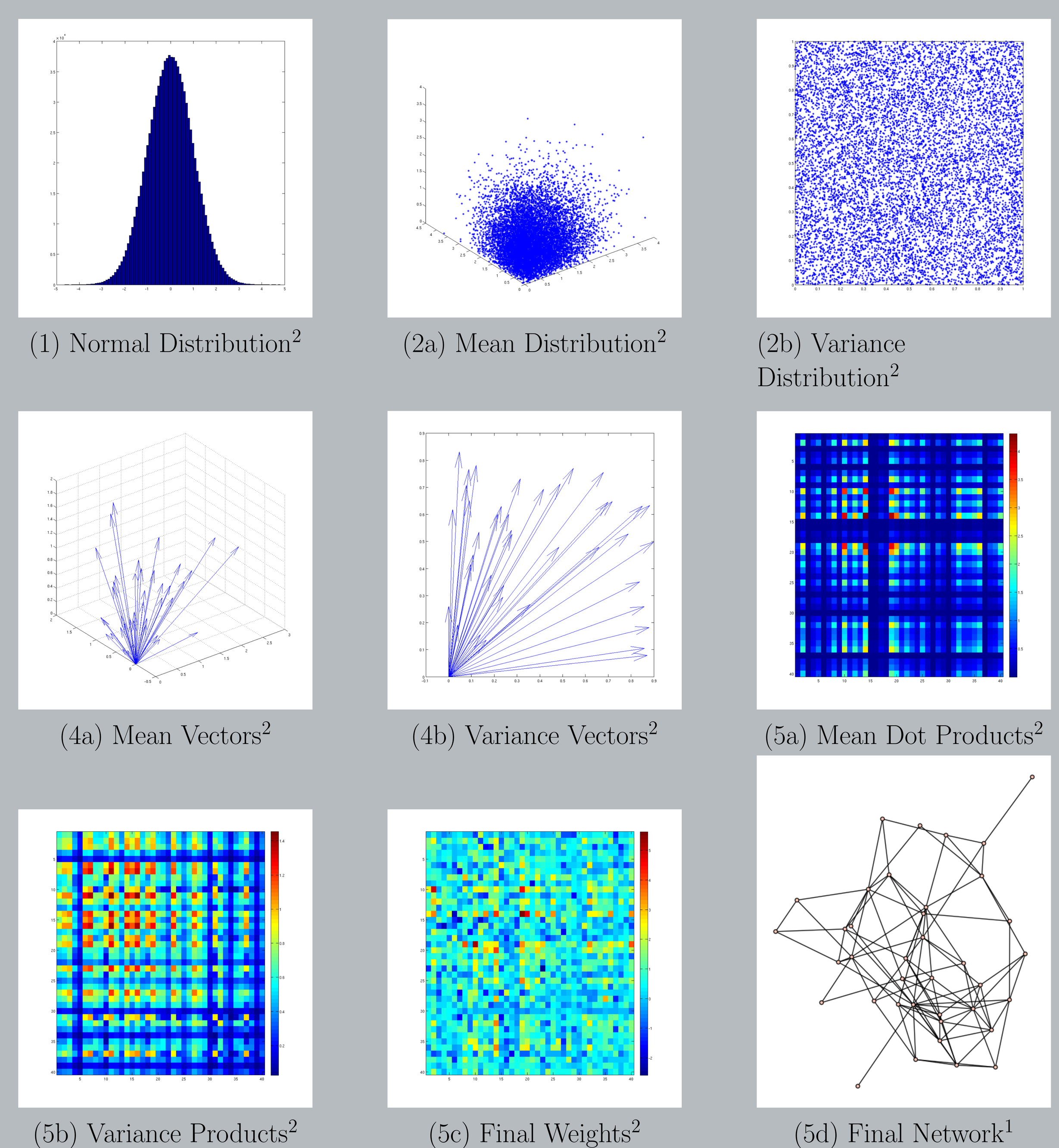
In order to generalize the RDPM for weighted networks we allow the edges to be drawn from an arbitrary parametrized probability distribution instead of a Bernoulli trial. In order to accommodate more complex distributions, we incorporate several latent vectors for each node, one for each parameter. Our final generative process proceeds as follows:

1. Select a parametrized probability distribution, P, with k parameters.
2. For each parameter, $1 \leq i \leq k$, select a dimension d_i for the corresponding latent space and a distribution W_i over $\mathbb{R}^{d_i}$.
3. Select the desired number of nodes, n.
4. For each node, $1 \leq j \leq n$, select k vectors (one from each parameter space), $V_j^i \in \mathbb{R}^{d_i}$ according to W_i.
5. For each pair of nodes, $1 \leq j, \ell \leq n$, place an edge between them with weight drawn from the probability distribution $P(\langle V_j^1, V_\ell^1\rangle, \langle V_j^2, V_\ell^2\rangle, \ldots, \langle V_j^k, V_\ell^k\rangle)$.

We call this process the weighted dot product model (WDPM). Depending on the application, after generating the network, it may be desirable to do some post–processing to obtain specified properties, such as normalizing the columns to obtain a stochastic matrix or taking the Hadamard product with a sparse connectance matrix to achieve a desired density.

Generative Example

The WDPM is a flexible model, as the weight distribution P as well as the parameter dimensions d_i and distributions W_i can be selected arbitrarily. The example below shows the construction of a WDPM network on 40 nodes with P chosen to be the normal distribution.



(1) Normal Distribution[2] | (2a) Mean Distribution[2] | (2b) Variance Distribution[2]
(4a) Mean Vectors[2] | (4b) Variance Vectors[2] | (5a) Mean Dot Products[2]
(5b) Variance Products[2] | (5c) Final Weights[2] | (5d) Final Network[1]

Figure 2: An example WDPM network generated with a normal probability distribution.

In this example, the final network contains several features, high clustering and short path lengths, that resemble standard social network models.

Why WDPM?

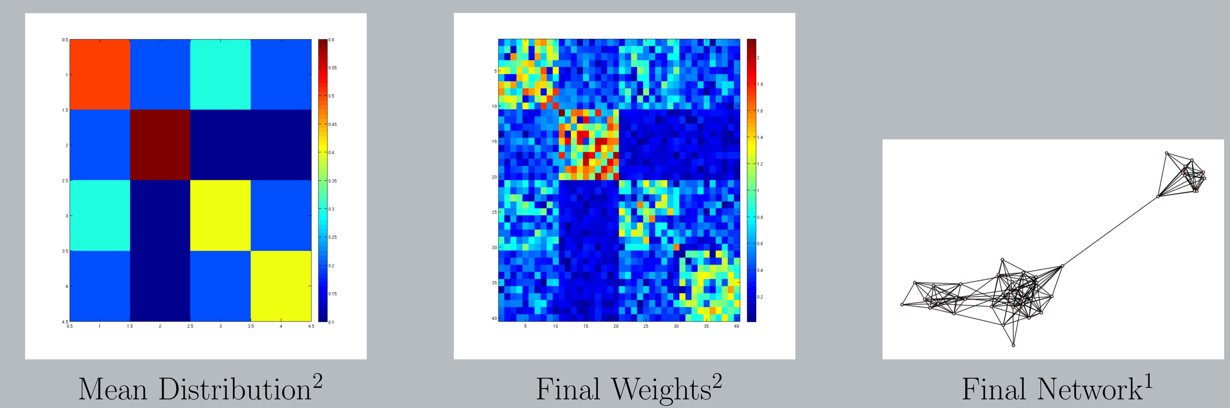
As a generative model, the WDPM offers three key advantages compared to other processes: generality, geometric leverage, and interpretability. Since P can be any parametrized probability distribution, the WDPM can be used to model networks derived from a wide variety of real–world data. As a latent space model, the WDPM provides an embedding of the network into Euclidean space, allowing us to use tools from linear algebra to analyze our networks.

Using the dot product to parametrize the network distinguishes the WDPM (and the RDPM before it) from other latent space models where distance is the standard measure. This approach allows us to understand the embedding in terms of the magnitude of each vector, which captures the corresponding node's propensity to communicate, and the direction of each vector, which captures the more standard latent space notion of node similarity [6, 9].

Special Cases and Variations

The WDPM permits several parameters to be tuned independently. As such, several simplifications and special cases of our model are of particular interest for computational or theoretical reasons.

- Restricting the distributions W_i to S^{d_i-1} gives a model where connections only depend on similarity.
- Conversely, selecting $d_i = 1$ gives a model that uses no similarity information as in [5].
- When W_i is a distribution over a finite set of vectors in $\mathbb{R}^{d_i}$ we have a generalized stochastic block model as in [1].
- Further restricting W_i to a single vector describes a generalized Erdös–Rényi model.
- Selecting P to be a constant distribution allows the network to reflect the latent embedding exactly.

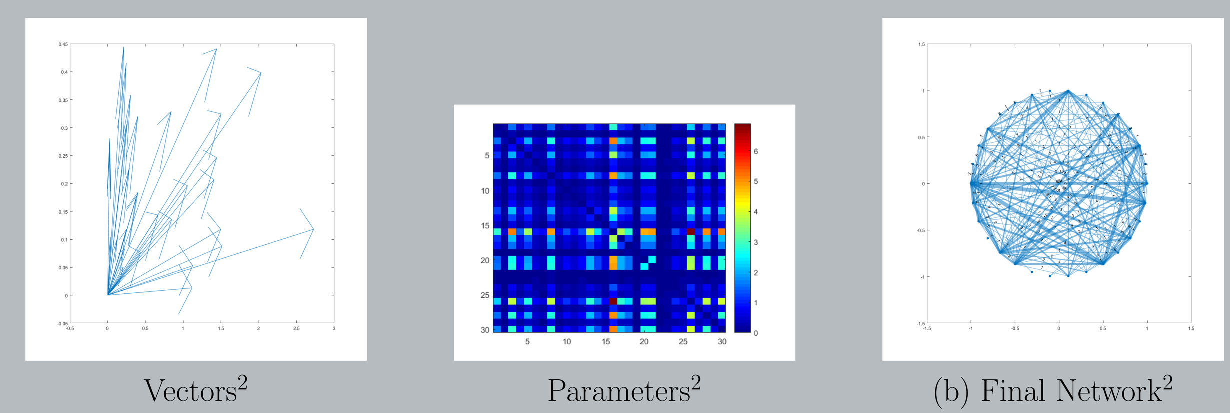


Mean Distribution[2] | Final Weights[2] | Final Network[1]

Figure 3: Stochastic block version of the WDPM.

Poisson Multi–Networks

The choice of P in the WDPM is not restricted to continuous distributions. As an example, collaboration and co–authorship are commonly studied using network models [4]. One way to construct a network from such a dataset is to place an edge of weight k between two scientists who written k papers together. To model such a network process using the WDPM we can take P to be the Poisson distribution. This assumes that the number of papers authored by any pair of scientists depends on their similarity as researchers and propensity to publish. Since the Poisson distribution has only one parameter, we only need one set of vectors to construct the network, unlike the normal distribution in Figure 2.



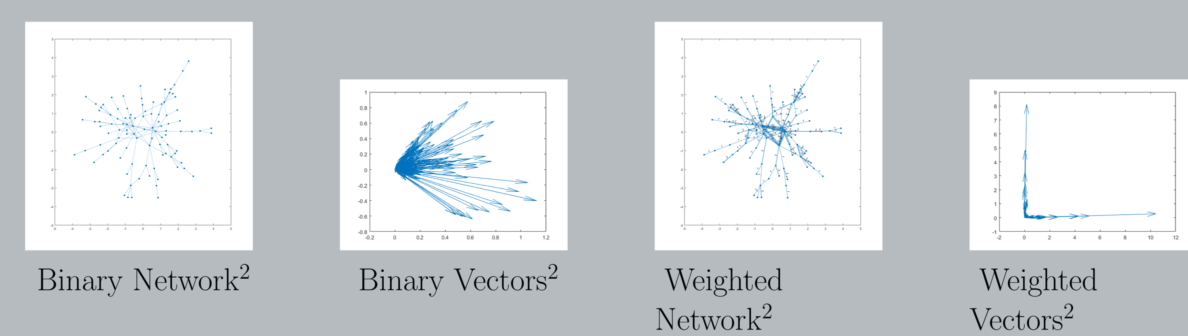
Vectors[2] | Parameters[2] | (b) Final Network[2]

Figure 4: Multinetwork constructed using the WDPM.

Once constructed, these networks can serve as null models for comparison to observed networks. Since P has a single parameter, the results of [9] can be extended to provide theoretical bounds for analysis.

WDPM Inference

Given a particular network of interest we can estimate the most likely vectors corresponding to our network using a version of the iterated algorithm presented in [6]. This algorithm allows us to use the latent vectors as a proxy for the network and analyze it using linear algebra. In particular, an angular k–means approach allows us to detect community structures in these weighted networks. The figure below shows this process for co–authorship data derived from combinatorial geometry researchers [2].



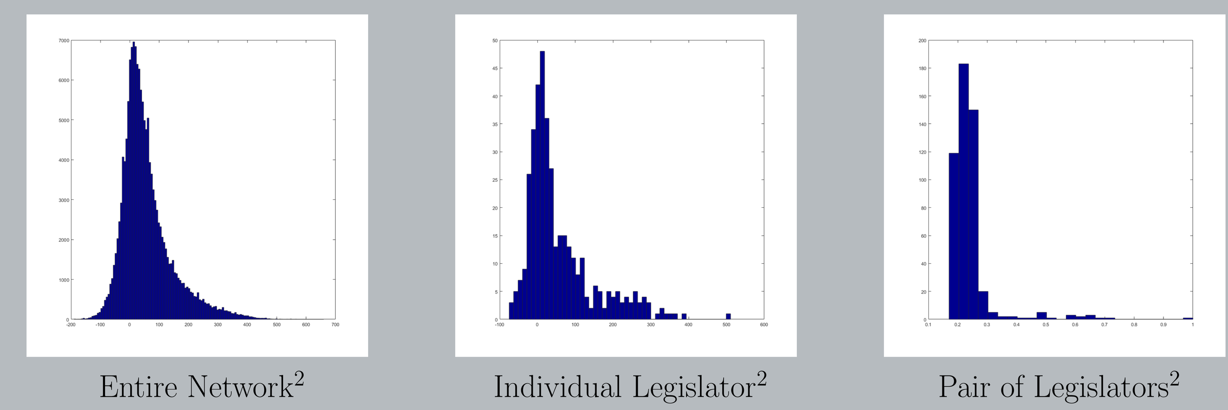
Binary Network[2] | Binary Vectors[2] | Weighted Network[2] | Weighted Vectors[2]

Figure 5: Networks and vectors learned from co–authorship data.

As demonstrated in Figure 5, our results have consistently found better orthogonal clustering using the WDPM as opposed to the binary RDPM.

Time Series Networks

Many sources of data, such as stock market closing prices and neural firing patterns, are reported as time series data. These datasets allows us to fit expected values to multiple parameters using the evolving distribution of the individual entry values. The figures below show these values for the European Parliament voting data.

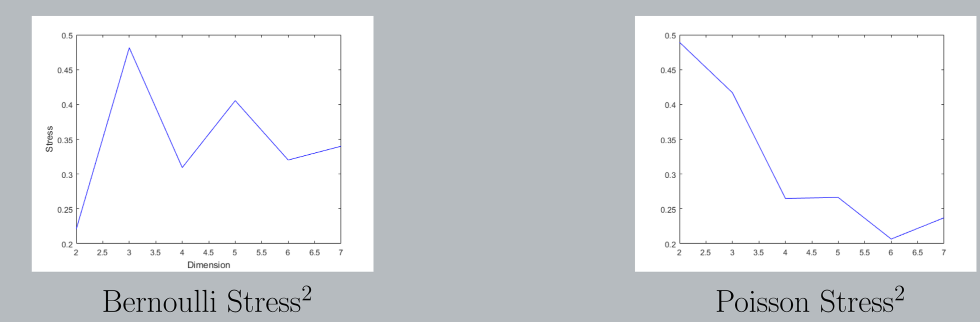


Entire Network[2] | Individual Legislator[2] | Pair of Legislators[2]

Figure 6: Histogram of weight distributions for the European Parliament voting data.

Stress Function

Since the method for learning vectors for a given network is efficient, we can select the appropriate dimension for each parameter space using a stress function. A natural stress function for similarity clustering, as described in [6], is the sum of the reference vector dot products. The figure below shows the stress values for the collaboration network using the Bernoulli and Poisson WDPM.



Bernoulli Stress[2] | Poisson Stress[2]

Figure 7: Stress function values for the collaboration network.

Confirming what we demonstrated in Figure 5, the clustering analysis of the weighted network is much more effective than the binary case, improving as the dimension increases to six.

Conclusions and Future Work

Our generative model allows us to construct null models for a wide variety of networks derived from real–world data. Additionally, the vector space structure of the embedding permits us to learn vectors from a given network and provides an interpretation of the latent variables. This provides for an efficient and effective community detection technique.

We intend to extend this research in several ways:

- Considering non–Euclidean embeddings motivated by information geometry
- Proving expected bounds for specific families of distributions
- Extending this process to multiplex networks
- Constructing factorization algorithms for specific classes of networks
- Describing the relation of this model to graphons

References

[1] C. Aicher, A. Jacobs, and A. Clauset: *Learning Latent Block Structure in Weighted Networks*, Journal of Complex Networks, 3(2), (2015), 221–248.

[2] V. Batagelj and A. Mrvar: *Pajek datasets*, (2006) URL: http://vlado.fmf.uni-lj.si/pub/networks/data/.

[3] S. Hix, A. Noury, and G. Roland: *Dimensions of Politics in the European Parliament*, American Journal of Political Science, 50(2), (2006), 494–511.

[4] M. Newman: *The structure of scientific collaboration networks*, PNAS, 98 (2), (2001), 404–409.

[5] J. Ranola, S. Ahn, M. Sehl, D. Smith, and K. Lange: *A Poisson Model for random multigraphs*, Bioinformatics, 26, (2010), 2004–2011.

[6] E. Scheinerman and K. Tucker: *Modeling graphs using dot product representations*, Computational Statistics, 25, (2010), 1–16.

[7] T. Shafie: *A Multigraph Approach to Social Network Analysis*, Journal of Social Structure, 16, (2015), 1–21.

[8] D. Sussman, M. Tang, D. Fishkind, and C. Priebe: *A consistent adjacency spectral embedding for stochastic blockmodel graphs*, Journal of the Americal Statistical Association, 107, (2012), 1119–1128.

[9] S. Young and E. Scheinerman: *Random Dot Product Models for Social Networks*, Algorithms and Models for the Web-Graph, Lecture Notes in Computer Science, 4863, (2007), 138–149.