Bachelor's thesis

Recovering network structure through Latent Space Models

Department of Statistics
Ludwig-Maximilians-University Munich

by Lea Schulz-Vanheyden
supervised by Prof. Dr. Göran Kauermann &
Giacomo De Nicola
Munich, September 23, 2020

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst und dabei keine anderen als die angegebenen Hilfsmittel benutzt habe. Sämtliche Stellen der Arbeit, die im Wortlaut oder dem Sinn nach Publikationen oder Vorträgen anderer Autoren entnommen sind, habe ich als solche kenntlich gemacht. Die Arbeit wurde bisher weder gesamt noch in Teilen einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

München, 22.09.2020 ....
Ort, Datum

Lea Schulz-Vanheyden ....
Lea Schulz-Vanheyden

Abstract

The latent space model is based on the idea that actors exist in a social space. It fits a model to a network by assigning the actors positions. This thesis evaluates how well the model is able to recover the positions of the underlying network. To achieve this, a simulation study is conducted. Networks with varying underlying distributions, numbers of nodes and dimensions in space are simulated. Latent space models with different dimensions are fitted to them. Next, the distances between actors estimated by the model are compared to the true distances. It can be seen that the type of distribution and the number of nodes barely have an effect on how good the model recovers the network structure. What makes a difference is firstly the dimensions of the space the network was simulated in and secondly the difference between the true dimensions of the network and the ones used for the model. The model is able to best recover the network structure if the network and the model dimension are the same. It can also be observed that overfitting the model yields better results than underfitting.

Contents

List of Figures

1 Introduction

Networks help us represent data on relations between actors. They are used in a wide variety of areas: to describe the behaviour of epidemics, the interconnectedness of corporate boards, wars between nations or links between websites. This thesis focuses only on social networks. Social networks as described by Kolaczyk (2009) are networks that focus on relationships between people or groups of people, therefore the relations are social. One could be looking at friendships, business connections or even trade agreements of nations. Data on social networks consists of information on pairs of actors or nodes. Often this data represents the existence, absence or value of a relationship between pairs of actors, such as friendship, shared membership in a group of individuals or the volume of emails between business partners. Here we consider binary social network data representing the presence or absence of a relationship. A so-called tie or edge exists between two nodes if they interact in some way. Ties can be directed or undirected. For example, looking at a disease where one person infects the other, this would be a directed tie. When we observe friendships, we could treat them as undirected: If one person is friends with another person, we also assume that the other person reciprocates this friendship. This thesis will only deal with undirected/reciprocal ties.

To assess and compare the relationships between members of a network one uses Social Network Analysis. Companies use it to choose which products they suggest to you, and the NSA used it to figure out the leadership structure of the hijackers of 9-11 (Satell, 2013) (Krebs, 2002). The current Covid-19 pandemic shows other examples where network analysis might be of use. As shown by Wang et al. (2020) it helps us understand how the disease spreads and what role hospitals played in it. Other papers analyse the network of the twitter users sharing 5G-conspiracy theories related to the pandemic (Ahmed et al., 2020) or the SARS-Cov-2 genomes (Forster et al., 2020).

When we want to see the network as a whole and not risk overlooking important features a good strategy is to build a statistical social network model and formalise the likelihood of observing a certain network from the space of all possible networks. One of these models is the *latent space model* introduced by Hoff et al. (2002). The latent space model thinks of the actors of the network as points in "social space". The distance between two actors in this unobserved Euclidean space corresponds to the strength of the relationship between the two. The more similar they are, the closer they are to each other in the social space. The probability of a tie increases as these locations become closer together. (It may also be affected by observable covariates, but there will be none in this thesis.) That also means that the latent space model is inherently transitive as it follows as a result of actors being close to each other. The concept of transitivity is that if person i and person j are friends and person j and person l are friends, person i and person l are more likely to also be friends. Person i and person j would be close to each other as would person j and person l. That means person i and persons l would also be close to each other and have a higher probability of a tie.

The idea of the social space has been around much longer. According to Handcock et al. (2007), the idea of representing a social network by assigning positions in a continuous space to the actors was introduced in the 1970s. Wasserman et al.

(1994) mention using Multidimensional scaling. Its goal is to represent the data that is given as dissimilarity measures between pairs of objects or individuals by points in an (usually) Euclidean space. Hoff et al. (2002) used the same idea as the basis of their model. A similar model was proposed by Schweinberger and Snijders (2003), but using an ultrametric space rather than a Euclidean space. Hoff (2005) introduced the bilinear effects model which resembles the latent space model but incorporates third-order dependence via a bilinear effect.

Since the introduction of the latent space model, others have taken up the idea and continued working on it. For example, Handcock et al. (2007) who introduced the *latent position cluster model.* Networks often show clustering, i.e. actors cluster into (unobserved) groups, within which links are more likely. One reason for this is transitivity. Another one is a tendency called homophily by attributes: ties are often more likely to occur between actors that have similar attributes than between those who do not. According to Handcock et al. (2007), many social networks exhibit clustering beyond what can be explained by transitivity and homophily on observed attributes. The latent position cluster model takes account of transitivity, homophily on attributes and clustering simultaneously in a natural way by allowing the latent space positions to follow a mixture of distributions, each corresponding to a cluster.

Krivitsky et al. (2009) built onto Hoff et al.'s model by including additive random individual effects. Apart from transitivity, homophily on observed attributes and clustering social network data often exhibits heterogeneity of actors. Heterogeneity is the tendency of some actors to have more ties than others. To take account of this the *Latent Cluster Random Effects Model* by Krivitsky et al. (2009) expands the latent position cluster model by Handcock et al. (2007) by adding random effects as proposed by Hoff (2005).

The models presented so far do not allow for hierarchical networks. Hierarchical networks are multiple partially exchangeable networks that arise for example when studying school systems. Instead of fitting separate single-network models for each school, it would make much more sense to be able to link these models to be able to fit them together and pool information from multiple networks to assess treatment and covariate effects. Sweet et al. (2013) introduce the *Hierarchical Network Models* framework that can be used to extend the single-network statistical network models to multiple networks. Even though the framework is general, they focus specifically on hierarchical latent space models.

Sarkar and Moore (2006) extended Hoff et al. (2002)'s Euclidean latent space model to dynamic networks. Dynamic networks are networks, that are able to represent the structure and the evolution of the ties between nodes. The ties between the nodes are measured at multiple time points to map the changes. Embedding this longitudinal network data into the model helps understand how relationships form, dissolve and change, for example how politicians form loyalties or break ranks with their parties or how co-authorship patterns develop and change over time. To be able to incorporate dynamic networks into the latent space model Sarkar and Moore (2006) developed a generalized multidimensional scaling to find the initial latent actor positions across discrete time points. Since the estimation is an ad hoc method which makes limited use of the available data, Sewell and Chen (2015) proposed a different approach. Each actor has a temporal trajectory in the latent space, the estimation then occurs within a Bayesian framework using Markov chain Monte Carlo.

Even though there are so many continuations of the model, this thesis will only deal with the latent space model of Hoff et al. (2002). The goal is to evaluate how well the latent space model recovers the original structure of the data for different dimensional spaces, numbers of nodes and distributions. We will only be looking at dichotomous and undirected ties and fit the model using a maximum likelihood estimator.

Chapter 2 focuses on the model used and lays the theoretical groundwork. Chapter 3 focuses on the simulation study. Various networks with different numbers of nodes, dimensions in space and following different distributions are simulated and latent space models in same, lower and higher dimensional space are fitted to them. Then we compare how well the models have recovered the true structure of the networks. A focus is placed on the choice of dimensions for the model. Lastly, the results are evaluated and suggestions are made on how one could further extend this study.

2 Latent Space Model

In order to carry out statistical analyses, there are a few terms and definitions that are practical and often used in social network analysis: Following Luke (2015) we call our set of actors “nodes”. The nodes are connected to one another via some type of social relationship which we call “tie”. Our network consists of n nodes. Between each ordered pair of actors/nodes i and j there is a tie $y_{i,j}$, $i, j = 1, \ldots, n$ which indicates the relationship between the pair of actors. While it does not have to be, in this thesis ties are dichotomous and indicate the presence or absence of some sort of relation. That means that $y_{i,j}$ is 0 if there is no tie and 1 if there is a tie. We will also only deal with undirected/reciprocal ties, $y_{i,j} = y_{j,i}$. All the ties together form the so-called sociomatrix $Y = \{y_{i,j}\}$ $(n \times n)$.

The sociomatrix is then the basis for the latent space model by Hoff et al. (2002). We think of each node $i = 1, \ldots, n$ as a point with an unknown position z_i in social space. The probability of a specific tie then depends on some function of the positions of the actors, here the Euclidean distance as used by Hoff et al. (2002). Generally, it may also be affected by observable covariates, but there aren't any in this thesis due to simplicity. To estimate whether or not there is a tie between two nodes we need to estimate their positions in social space.

The ties only depend on the positions and are otherwise independent. This is called the conditional independence approach: we assume that the presence or absence of a tie between two nodes is independent of all other ties in the network, given the unobserved positions in social space of the two nodes,

$$P(Y|Z, \theta) = \prod_{i \neq j} P(y_{i,j}|z_i, z_j, \theta), \tag{1}$$

where Y is a $n \times n$ sociomatrix with entries $y_{i,j}$. θ and Z are parameters and positions to be estimated.

Using a logistic regression model, we get:

$$\begin{aligned} \eta_{i,j} &= \text{log odds}(y_{i,j} = 1|z_i, z_j, \alpha) && (2) \\ &= \alpha - |z_i - z_j| && (3) \end{aligned}$$

The log-likelihood of our model is given by,

$$logP(Y|\eta) = \sum_{i \neq j} (\eta_{i,j} y_{i,j} - log(1 + e^{\eta_{i,j}})) \tag{4}$$

where η is a function of α and the unknown positions.

We can then estimate either with maximum likelihood or with Bayesian inference. We will only use the maximum likelihood approach as it is much simpler. As described by Hoff et al. (2002) and Handcock et al. (2007) this is straightforward. The first step is to estimate a set of distances between the nodes that maximize the likelihood. The next step is to then find a set of latent positions that approximate the distances. This can be done using multidimensional scaling. From there on we can start a non-linear optimization method.

3 Simulation Study

Now that the theoretical background has been clarified, it is of interest how well the model is able to recover the original structure of the network depending on the number of nodes, the dimension of space of the network and the model and the distribution of the nodes in the network. We want to know whether for example networks with more nodes are recovered better or whether the way the nodes are distributed in space has an influence. How much do the results vary when the dimensionality changes? What real ramifications does fitting a model with lower dimensions than the network has have?

To determine this, a simulation study was carried out. Networks following five different distributions with four different numbers of nodes and in four different dimensional spaces were simulated. Then models in same, lower and higher dimensional spaces were fitted. Finally the original positions of the nodes were compared with those estimated by the model.

This was done using *R* (R Core Team, 2020). The package *latentnet* by Krivitsky and Handcock (2020) implements the latent space model. For procrustes transformation as explained in 3.3.1 the package *vegan* by Oksanen et al. (2019) was used.

3.1 Simulating the networks

The basis for the simulation study is the simulated network. We generate them by simulating nodes in space and ties between them. The nodes follow different distributions in different dimensional spaces. The distributions used are:

- Uniform distribution
- Normal distribution
- 2 Groups following a normal distribution
- 3 Groups following a normal distribution
- 4 Groups following a normal distribution

For each of those distributions we simulate 20, 50, 100 and 200 nodes in 2, 4, 6 and 8 dimensional space. For each combination of number of nodes, distribution and dimension of space, we will repeat the simulation process five times to get a better estimate of the true effect of those variables.

Let n be the number of nodes to simulate and q the dimension of the space in which to simulate. Following the different distributions, we want to generate Z a $n \times q$-matrix with each row corresponding to a node with the position $z_i, (i = 1, \dots, n)$.

$$Z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1q} \\ z_{21} & z_{22} & \cdots & z_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{nq} \end{pmatrix} \tag{5}$$

This matrix gives us the positions of all nodes. It is the foundation on which the ties are generated. For this a Bernoulli distribution is used, the closer two points are to each other, the higher the probability that they have a tie. The exact process is described in chapter 3.1.4. To make simulating the ties as easy as possible it makes sense to have a maximum distance of 1 between any two nodes. To accomplish this, we either simulate all nodes in a sphere with a diameter of 1 (for the uniform distribution) or scale the set of nodes down after simulating (for the normal and the group distributions).

3.1.1 Uniform Distribution

The first and easiest distribution is the uniform distribution. The nodes are equally probable to be at any position in a q-sphere with its center the point of origin and a radius of 0.5 (which is equal to a diameter of 1). We do this by first generating q numbers following this uniform distribution:

$$z_{il} \sim U(-0.5, 0.5) \text{ with } i = 1, \ldots, n, l = 1, \ldots, q \tag{6}$$

$$f(z_{il}) = \begin{cases} 1 & \text{for} - 0.5 \leq z_{il} \leq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Each point $z_i = (z_{i1}, \ldots, z_{iq})$ is a point in a q-cube. We then only keep points that meet the additional condition

$$d(z_i, 0) \leq 0.5 \tag{8}$$

$$\text{with } d(z_i, z_j) = \sqrt{\sum_{l=1}^{q} (z_{il} - z_{jl})^2} \tag{9}$$

$d(z_i, z_j)$ is the Euclidean distance between point z_i and point z_j.

This leaves us with n points, the nodes, in a q-sphere. To visualize what that might look like there are figures for each distribution showing 200 nodes in 2-dimensional space. The uniform distribution can be seen in figure 1a.

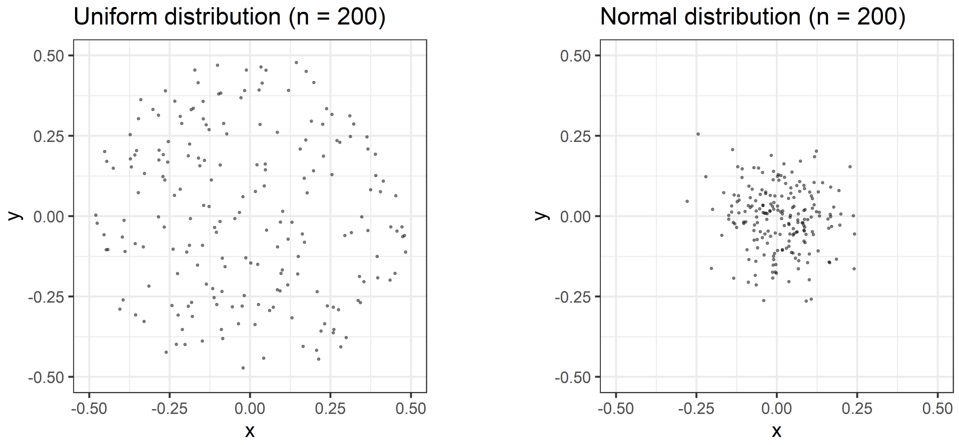
3.1.2 Normal Distribution

To get the nodes normally distributed around the point of origin we simulate n points following this multivariate normal distribution:

$$z_i^* \sim N_q(0, \mathbb{1}) \tag{10}$$

$$f(z_i^*) = \frac{1}{\sqrt{(2\pi)^q}} \exp{(-\frac{1}{2} \sum_{l=1}^{q} {z_{il}^*}^2)} \tag{11}$$

$\mathbb{1}$ denotes the $n \times n$ identity matrix.



(a) Uniform distribution

(b) Normal distribution

Figure 1: Example distributions in 2-dimensional space with $n = 200$

Since some points could have a distance of more than 1 between them they need to be rescaled:

$$z_i = \frac{z_i^*}{2 \cdot d_{\max}} \tag{12}$$

$d_{\max}$ is the biggest observed distance between any two points:

$$\begin{aligned} d_{\max} = max(&d(z_1^*, z_2^*), d(z_1^*, z_3^*), \ldots, d(z_1^*, z_n^*), \\ &d(z_2^*, z_3^*), \ldots, d(z_2^*, z_n^*), \ldots, d(z_{n-1}^*, z_n^*)) \end{aligned} \tag{13}$$

As we can see in figure 1b the points following a normal distribution are much more concentrated and dense around the point of origin compared to the ones following a uniform distribution.

3.1.3 Groups

Let g be the number of groups (here $g = 2, 3, 4$). First, we get the mean of each group:

$$\mu_{Gk} \sim N_q(0, \mathbb{1}) \text{ with } k = 1, \ldots, g \tag{14}$$

Then we want to know how many nodes belong to each group. To calculate n_k, the number of points in group k, we use a multinomial distribution:

$$f(n_1, \ldots, n_g) = \begin{cases} \frac{n!}{n_1! \cdots n_g!} p_1^{n_1} \cdots p_g^{n_g} \text{ for } n_1, \ldots, n_g \in \mathbb{N}_0 \text{ and } n_1 + \ldots + n_g = n \\ 0 \text{ otherwise} \end{cases} \tag{15}$$

$p_1, \ldots, p_g$ are the probabilities to be assigned to each group. Here they are all the same.

$$p_1 = \ldots = p_g = \frac{n/g}{100} \tag{16}$$

We now get all positions of the nodes in group k by sampling n_k points following a normal distribution.

$$z^*_{i_k} \sim N_q(\mu_{Gk}, \frac{0.1}{g} \cdot \mathbb{1}) \text{ with } i = 1, \ldots, n_k \tag{17}$$

The next step is to, again, scale the points with 2 times the biggest observed distance and then obtain our point matrix Z:

$$Z = (z_{1_1}, z_{2_1}, \ldots, z_{n_1}, \ldots, z_{n_g})^T \tag{18}$$

Figure 2 shows example realisations of $g = 2, 3, 4$ group distributions. We can see that in (c) there seem to only be three groups instead of four. This is because two of the groups are so close together, they are no longer distinguishable and form one bigger group. Since this is something that might also happen when working with real data this is a welcomed behaviour.

3.1.4 Generating the network

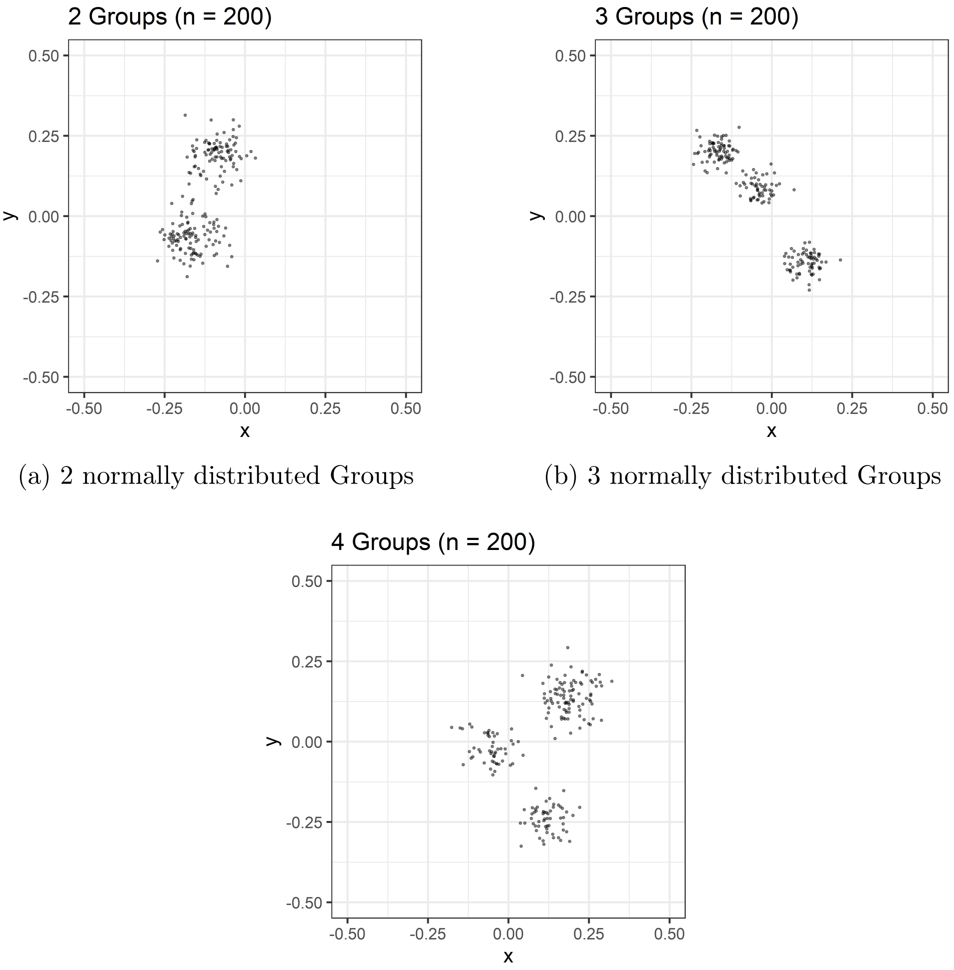
After obtaining our point-matrix Z we now want to compute the symmetric sociomatrix Y.

$$Y = \begin{pmatrix} 0 & & & & \\ y_{2,1} & 0 & & & \\ y_{3,1} & y_{3,2} & 0 & & \\ \vdots & \vdots & & \ddots & \\ y_{n,1} & y_{n,2} & \ldots & y_{n,n-1} & 0 \end{pmatrix} \tag{19}$$

$y_{i,j}$ is the tie between the nodes corresponding to the positions z_i and z_j. To get the ties from the points simulated before we use a Bernoulli distribution. The smaller the distance between the two points the higher the probability of a tie.

$$y_{i,j} \sim B(1 - d(z_i, z_j)) \tag{20}$$

$$f(y_{i,j}) = \begin{cases} (1 - d(z_j, z_j))^{y_{i,j}} d(z_i, z_j)^{1-y_{i,j}} \text{ for } y_{i,j} = 0, 1 \\ 0 \text{ otherwise} \end{cases} \tag{21}$$



(a) 2 normally distributed Groups

(b) 3 normally distributed Groups

(c) 4 normally distributed Groups

Figure 2: Example distributions in 2-dimensional space with $n = 200$

There is no tie between a point and itself: $y_{i,j} = 0$ for $i = j$. And the ties are undirected, meaning that the tie between point i and j is the same as the tie between point j and i: $y_{i,j} = y_{j,i}$.

There are now 400 different sociomatrices/networks simulated (5 distributions $\times$ 4 different number of nodes $\times$ 4 different dimensions of space $\times$ 5 times $= 400$). The next step is to fit latent space models on these networks.

3.2 Fitting the latent space model

We now fit latent space models to all of the networks using maximum likelihood estimation. We not only fit them with the dimension the network was generated in but also in all other dimensions from 2 to 8. That gives us a total of 2800 fitted models ($400 \times 7 = 2800$). From the fitted models we get the estimated positions of the nodes.

We fit the models using the *ergmm*-function in the *latentnet* package in *R* (Krivitsky and Handcock, 2020). When dealing with a network that has groups *ergmm* also has the option to specify the number of groups within the model. Since the goal of this thesis is not to determine how well the latent space model could detect groups we always use the default setting of one group.

3.3 Comparison of fitted and generated data

We want to know how good our fitted model reconstructs the original, true structure of the data. To make this measurable we first need to transform our estimated set of points to be comparable to our original set of points. To do this we are using Procrustes transformation.

3.3.1 Procrustes Transformation

Since the estimated points may be distributed completely different in space than the original ones and for example no longer have a maximum distance of 1, we need to transform them. Procrustes transformation translates, rotates and uniformly scales one set of points to maximum similarity with respect to another set of points (Oksanen et al., 2019). As described by Mardia et al. (1979) we move the points until the sum of squared differences is minimal. This transformation is then optimal in regards to similarity between the two sets.

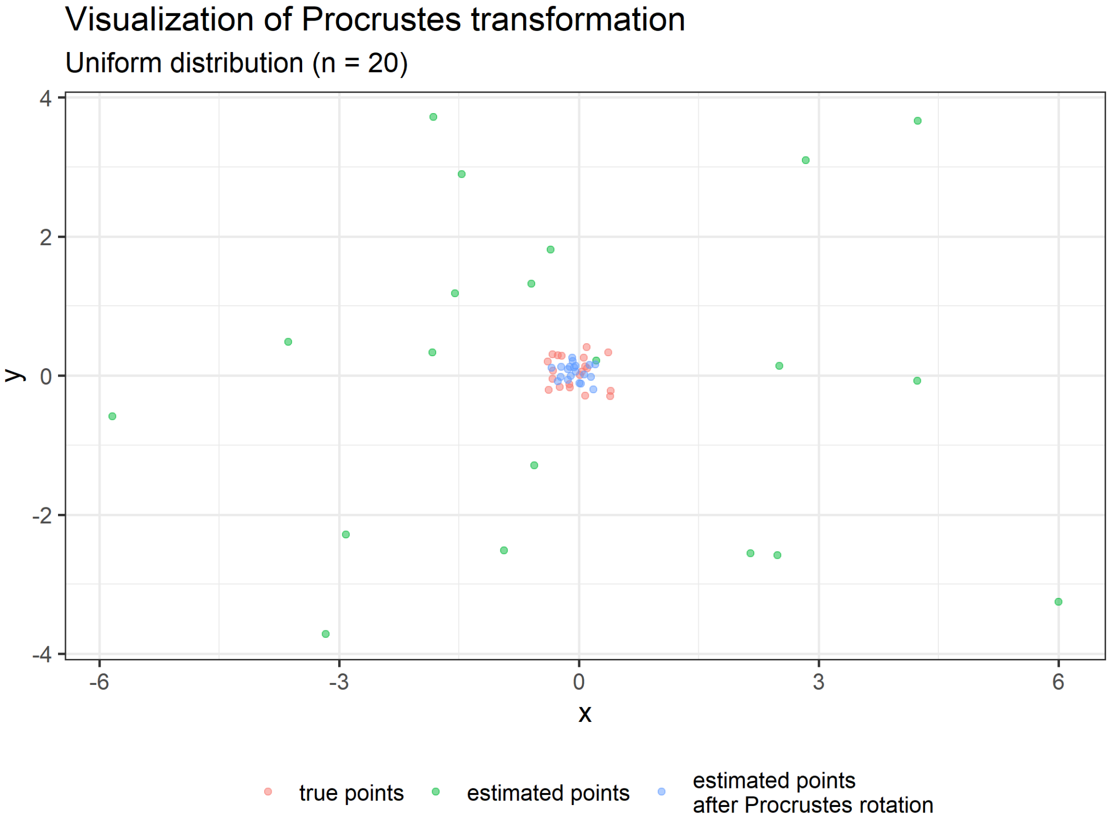


Figure 3: Visualization of Procrustes transformation in a 2-dimensional space with 20 nodes following a uniform distribution

Figure 3 visualizes the effect of a Procrustes transformation. Before the transformation, the estimated points are located somewhere else and have much larger distances than the true points. After, the points lie as close as possible to the original points.

In order to see the effects of the Procrustes transformation on the results of the study in appendix A one can find the results after using simple scaling instead.

One practical problem that arises specifically in our simulation study is that the number of dimensions of the set of estimated points and original points are not always the same. We fit models using both higher and lower dimensions than those of the space in which the true points lie. To still be able to carry out a Procrustes transformation we set all coordinates of the unused dimensions to 0.

We use the *procrustes*-function in the *vegan* package in *R* (Oksanen et al., 2019) to transform the estimated set of points $\hat{Z} = \{\hat{z}_i\}$ to be most similar to the true set of points $Z = \{z_i\}$. The new set of points after transformation is called $\hat{Z}_\mathrm{r} = \{\hat{z}_{i\mathrm{r}}\}$. The next step is to compare $\hat{Z}_\mathrm{r}$ to Z.

3.3.2 Difference of Distances

To measure how different the two sets of points are we use a metric we are going to call the *Difference of Distances*. To obtain it, we compare the Euclidean distances between the true points Z and the distances between the estimated and transformed points $\hat{Z}_\mathrm{r}$.

$$d_\mathrm{diff} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(d(z_i, z_j) - d(\widehat{z_{\mathrm{r}i}}, \widehat{z_{\mathrm{r}j}}))^2} \tag{22}$$

The bigger d_diff is, the bigger is also the difference between the estimated points and the true points. The lower the Difference of Distances the better our model was able to reconstruct the original structure/positions of the nodes.

Since we simulated each combination of distribution, number of nodes and dimension of space five times we also get five Differences of Distances per combination. To better compare them, we calculate the mean of those five. Plots for the standard deviation can be found in appendix B.

3.4 Results

We now have a mean Difference of Distances for each combination of parameters and are able to take a look at them.

Figure 4 shows the results. Figure 5 shows the same plot but the Difference of Distances gets divided by the number of nodes to adjust for n.

Comparing the distributions, we can see that there is barely any difference (given that number of nodes, fitted and original dimensions are the same). When looking at 20 nodes we can see that, while the trend is the same, the graphs differ a bit. This decreases with a rising number of nodes.

Looking at the influence of the number of nodes, we get the sense that the shape is similar between the number of nodes but gets more “defined” the higher the number of nodes. The graphs for 20 nodes are very unstable and, while showing a similar trend, don't show it as clearly as 200 nodes. We can see that the mean Difference of Distances tends to get higher the more nodes we have. This is because with higher numbers of nodes we also have more distances between them and therefore a higher sum overall. When looking at figure 5 we can see that after adjusting for the number of nodes this effect disappears. But we can still see that for 20 nodes the original dimension matters much less than for higher numbers of nodes. The more nodes the bigger are the differences between both the Difference of Distances for the original and for the fitted dimensions.

Looking at the figures we can see that, no matter distribution and number of nodes, the highest original dimension 8 almost always also has the highest Difference of Distances for all fitted dimensions. We can also see that the mean Difference of Distances for 8 original dimensions decreases with higher fitted dimensions.

The graph for 6 original dimensions is in between the graphs for 4 and 8 dimensions most of the time. When looking at 200 nodes, the Difference of Distances decreases up to 6 fitted dimensions and from there on we can see an increase. This seems to also be the case for 50 nodes when looking at the uniform and 4 groups distribution. For 50 nodes and a normal distribution, the graph has a similar behaviour but after 6 fitted dimensions it first increases for 7 fitted dimensions and then decreases for 8 (but is still above 6 fitted dimensions). At 50 nodes and 2 groups and for 100 nodes the graph declines till 6 or (in the case of 3 groups) 7 dimensions and then more or less stagnates. Looking at 20 nodes, we can't observe a trend this distinct but still see a general decrease until 6 fitted dimensions. For higher fitted dimensions the curve still decreases but with a slower slope.

We can notice similar tendencies when studying 4 original dimensions. Except for 20 nodes, the Difference of Distances tends to decline until 4 fitted dimensions and then increases again. The more nodes the stronger this trend. For 20 nodes this isn't observable but we can still see a general decrease that flattens with higher fitted dimensions. The Difference of Distances for 4 original dimensions is almost always lower than the one for 6 (and therefore also 8) original dimensions (with the other variables being the same).

Looking at only 2 original dimensions the graph increases for all fitted dimensions with three exceptions being the uniform, 3 groups and 4 groups distribution at 20 nodes. There it more or less stays the same. Up to 4 fitted dimensions the Difference of Distances for 2 original dimensions is lower than that for higher original dimensions. But for higher fitted dimensions, it gets closer to 4 original dimensions. The bigger the number of nodes the faster this happens. For 200 nodes it is even higher than 4 original dimensions and very similar to or even above 6 original dimensions.

This shows that it would always be best to use the original/true dimension as the fitting dimension. Using a higher or lower dimension both lead to the model being less able to recover the network structure. We can see that using a dimension for the model that is too high is better than using one too low (assuming both are equally far from the original dimension), but the difference it makes is very small. The higher the number of nodes, the bigger this difference. The original distribution/structure of the nodes has little to no effect on this aspect.

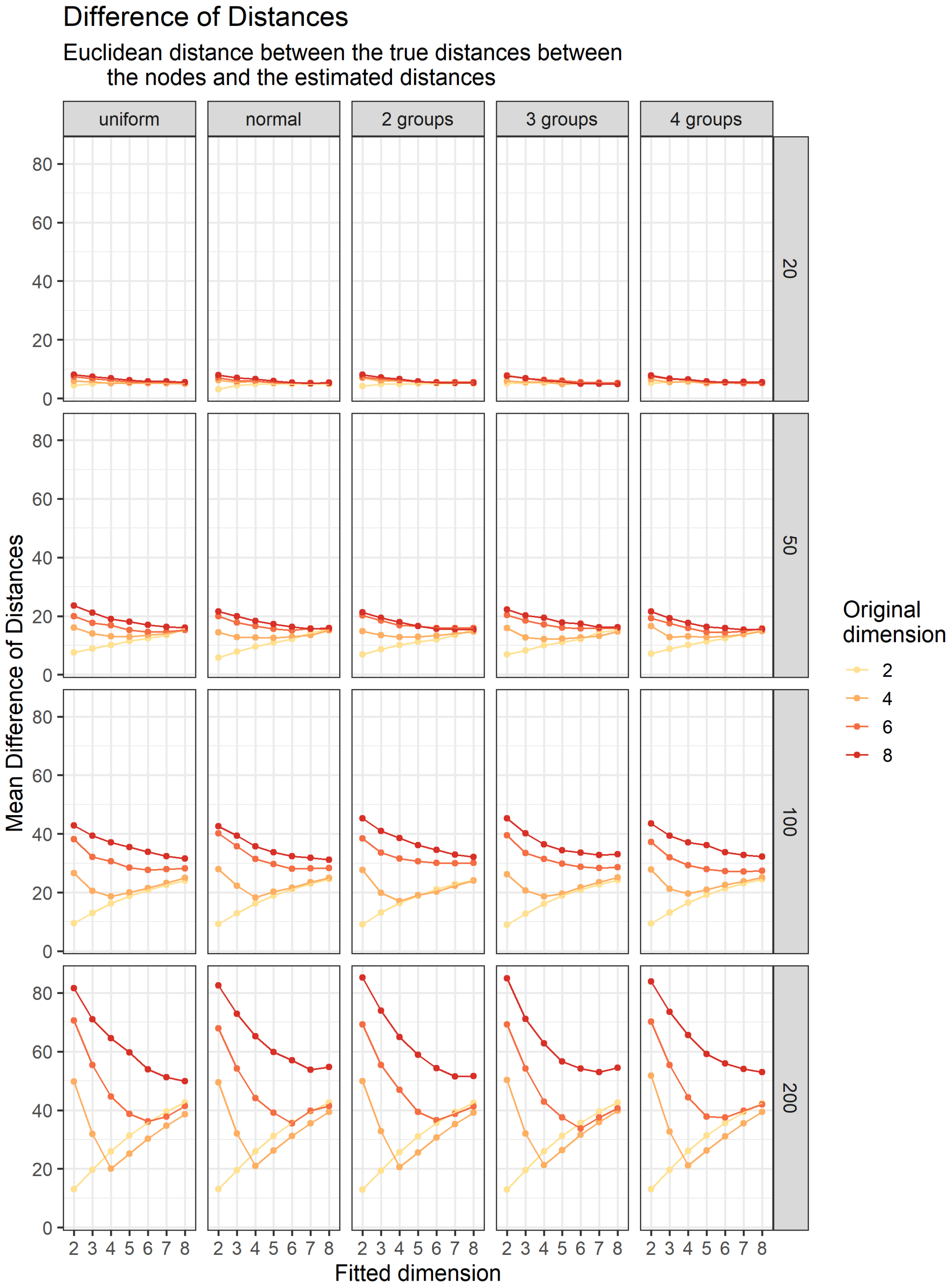


Figure 4: Mean difference of distances between the true, original distances between the nodes and the estimated distances between the nodes after Procrustes transformation for each distribution, number of nodes, number of original dimensions and number of fitted dimensions

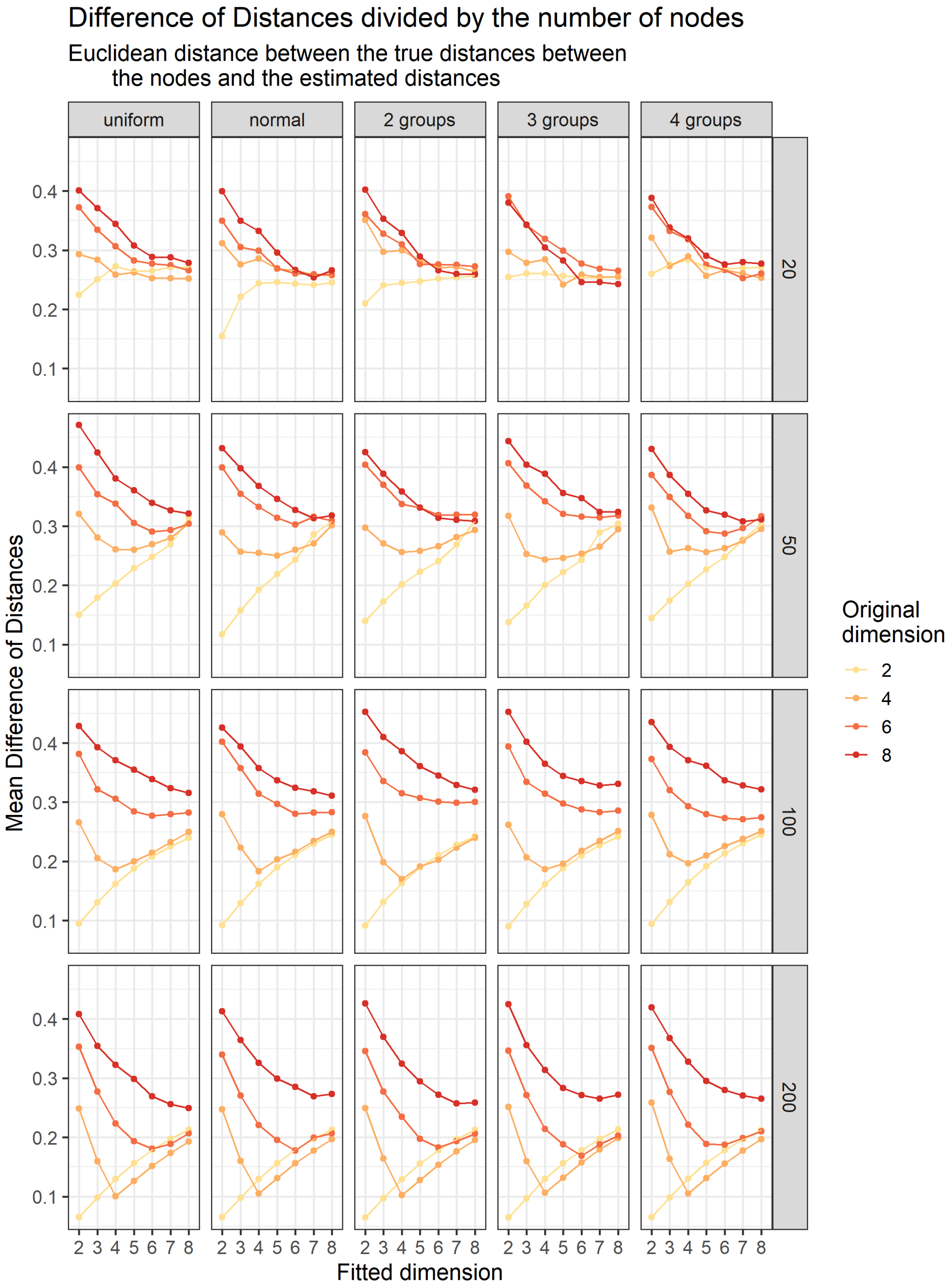


Figure 5: Mean difference of distances between the true, original distances between the nodes and the estimated distances between the nodes after Procrustes transformation for each distribution, number of nodes, number of original dimensions and number of fitted dimensions divided by the number of nodes

4 Conclusion and Discussion

The goal of this thesis was to determine how well the latent space model is able to recover the structure of networks. To do this a number of networks with varying distributions, numbers of nodes and dimensions in space were simulated. Then latent space models with different numbers of dimensions were fit. The positions of nodes from these models were then compared to the true positions. To do this we defined a new metric, the Difference of Distances. Lastly, we took a look at the results of the simulation study.

In summary, it can be noted that the quality of the fit is independent of the distribution. 5 different distributions were tested. The goodness of the recovery is nearly the same for all of the distributions. The number of nodes also don't seem to play a big role. For networks with a lot of nodes, the differences between the different original dimensions are big. Looking at just 20 nodes, the Differences of Distances don't vary that much between the original dimensions. In practice, that means we don't have to make any adjustments to our model based on just the distribution of the nodes and the number of nodes.

What one needs to be careful of is the dimension with which to fit the model. In reality, we typically do not know the original dimensions of the network but the simulation study served as an indication of what one should do if they were known. There is a bias towards using low dimensions to fit the model because the results are easier to interpret and visualize. This study helps us see how much of a problem underfitting really is. It shows that we obtain the best results when fitting with the same dimensions as the network. The lower the dimensions of the network the better the results. In the case of using the wrong dimensions for fitting overfitting returns better results than underfitting. But the difference is so marginal that in practice it makes sense to underfit (due to the advantages it gives us when interpreting the model).

How could one go on from here? In order to understand in more detail the limitations and usage of the latent space model further issues should be addressed: The first and most logical step would be to vary the underlying network structure more. That is, to try out different distributions (for example, with more groups or with groups of varied sizes). Or one could test higher original dimensions or different numbers of nodes.

The second option is to alter the model. In this work, we always used the Euclidean distance. But as Hoff et al. (2002) notes, other metrics are also possible. This could be useful, especially with higher dimensions, as discussed by Aggarwal et al. (2001). Whether other distance metrics show different or similar behaviour would be very interesting for practical applications. But not only the metric could be changed. As mentioned in chapter 2 there are two methods to estimate the parameters: Maximum likelihood estimation, which was used here, and a Bayesian approach. It would be interesting to see if the two estimation methods differ in their results.

After the model has been estimated, it is also possible to change how the similarity of the positions of the nodes is assessed. Perhaps there is a better measure of this than the Difference of Distances. A simple change would be to use the Manhattan metric instead of the Euclidean distance to calculate the distances. According to Aggarwal et al. (2001) this might make sense, especially in higher dimensional spaces. Another option would be to use the fitted model to simulate new networks and compare those to the true networks. This approach wouldn't be focused on the idea behind the latent space model, the social space, but one could still see the impact of different changes of parameters to the performance of the model.

Next, it would be interesting to see how well the latent space model performs compared to other social network models. Since the distinctive feature of the latent space model is the concept of the social space, which makes the comparison of positions as done here practicable, one would need to find another metric to compare the models. An idea would be to use the approach of simulating new networks, as described before.

Another way to compare the performance of the latent space model to other models could be by focusing on community detection. Handcock et al. (2007) introduced the latent position cluster model which extends the latent space model with the possibility to identify clusters of nodes. The latent position cluster model could, for example, be compared to the stochastic block model in regards to how good the community detection is.

References

Aggarwal, C. C., Hinneburg, A. and Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional space, *International conference on database theory*, Springer, pp. 420–434.

Ahmed, W., Vidal-Alaball, J., Downing, J. and Seguí, F. L. (2020). Covid-19 and the 5g conspiracy theory: social network analysis of twitter data, *Journal of Medical Internet Research* **22**(5): e19458.

Forster, P., Forster, L., Renfrew, C. and Forster, M. (2020). Phylogenetic network analysis of sars-cov-2 genomes, *Proceedings of the National Academy of Sciences* **117**(17): 9241–9243.

Handcock, M. S., Raftery, A. E. and Tantrum, J. M. (2007). Model-based clustering for social networks, *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **170**(2): 301–354.

Hoff, P. D. (2005). Bilinear mixed-effects models for dyadic data, *Journal of the american Statistical association* **100**(469): 286–295.

Hoff, P. D., Raftery, A. E. and Handcock, M. S. (2002). Latent space approaches to social network analysis, *Journal of the American Statistical association* **97**(460): 1090–1098.

Kolaczyk, E. D. (2009). *Statistical analysis of network data*, Vol. 1, Springer.

Krebs, V. E. (2002). Mapping networks of terrorist cells, *Connections* **24**(3): 43–52.

Krivitsky, P. N. and Handcock, M. S. (2020). *latentnet: Latent Position and Cluster Models for Statistical Networks*, The Statnet Project (`https://statnet.org`). R package version 2.10.5.
URL: *https://CRAN.R-project.org/package=latentnet*

Krivitsky, P. N., Handcock, M. S., Raftery, A. E. and Hoff, P. D. (2009). Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models, *Social networks* **31**(3): 204–213.

Luke, D. A. (2015). *A user's guide to network analysis in R*, Springer.

Mardia, K. V., Kent, J. T. and Bibby, J. M. (1979). *Multivariate Analysis*, Academic Press.

Oksanen, J., Blanchet, F. G., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., Minchin, P. R., O'Hara, R. B., Simpson, G. L., Solymos, P., Stevens, M. H. H., Szoecs, E. and Wagner, H. (2019). *vegan: Community Ecology Package*. R package version 2.5-6.
URL: *https://CRAN.R-project.org/package=vegan*

R Core Team (2020). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.
URL: *https://www.R-project.org/*

Sarkar, P. and Moore, A. W. (2006). Dynamic social network analysis using latent space models, *Advances in Neural Information Processing Systems*, pp. 1145–1152.

Satell, G. (2013). How the nsa uses social network analysis to map terrorist networks.
URL: *https://www.digitaltonto.com/2013/how-the-nsa-uses-social-network-analysis-to-map-terrorist-networks/*

Schweinberger, M. and Snijders, T. A. (2003). Settings in social networks: A measurement model, *Sociological Methodology* **33**(1): 307–341.

Sewell, D. K. and Chen, Y. (2015). Latent space models for dynamic networks, *Journal of the American Statistical Association* **110**(512): 1646–1657.

Sweet, T. M., Thomas, A. C. and Junker, B. W. (2013). Hierarchical network models for education research: Hierarchical latent space models, *Journal of Educational and Behavioral Statistics* **38**(3): 295–318.

Wang, P., Lu, J.-a., Jin, Y., Zhu, M., Wang, L. and Chen, S. (2020). Statistical and network analysis of 1212 covid-19 patients in henan, china, *International Journal of Infectious Diseases* .

Wasserman, S., Faust, K. et al. (1994). *Social network analysis: Methods and applications*, Vol. 8, Cambridge university press.

A Difference of Distances with Scaling instead of Procrustes transformation

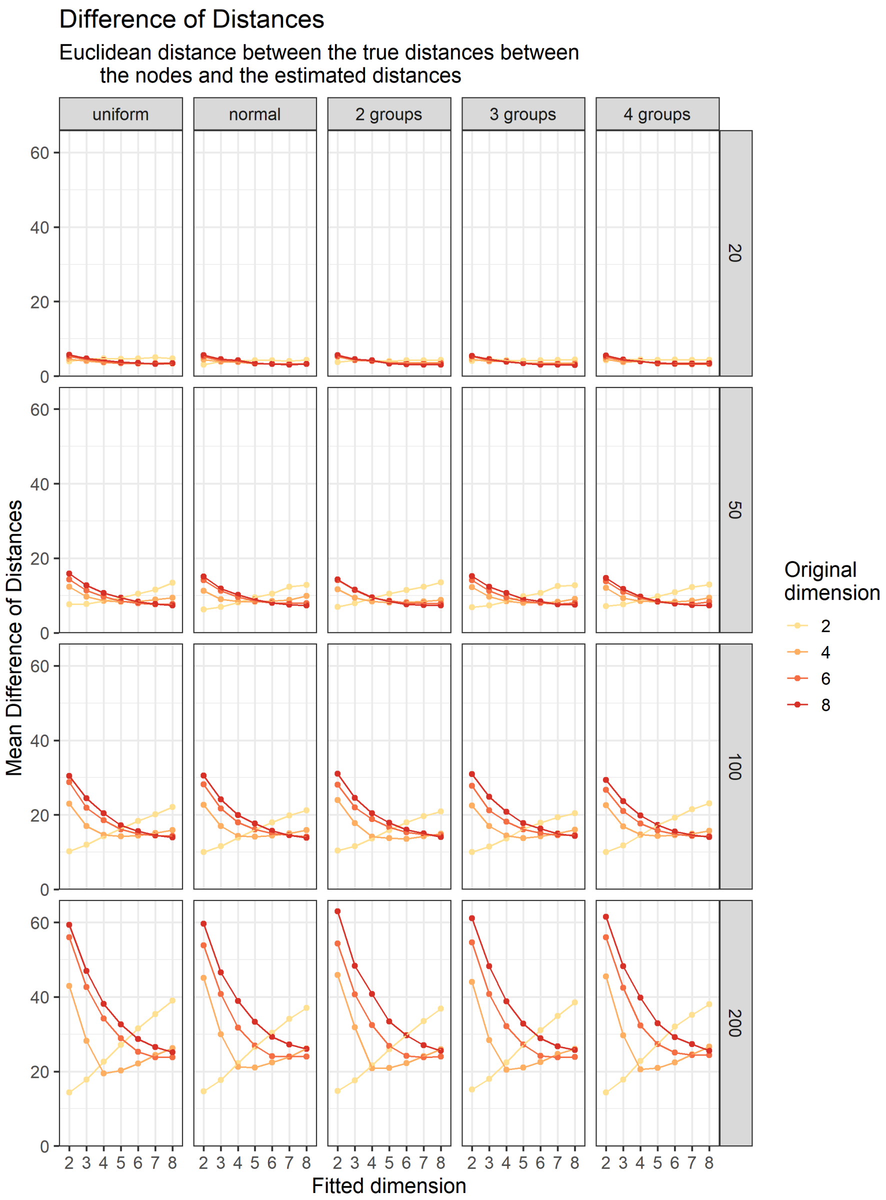


Figure A1: Mean difference of distances between the true, original distances between the nodes and the estimated distances between the nodes after Scaling for each distribution, number of nodes, number of original dimensions and number of fitted dimensions

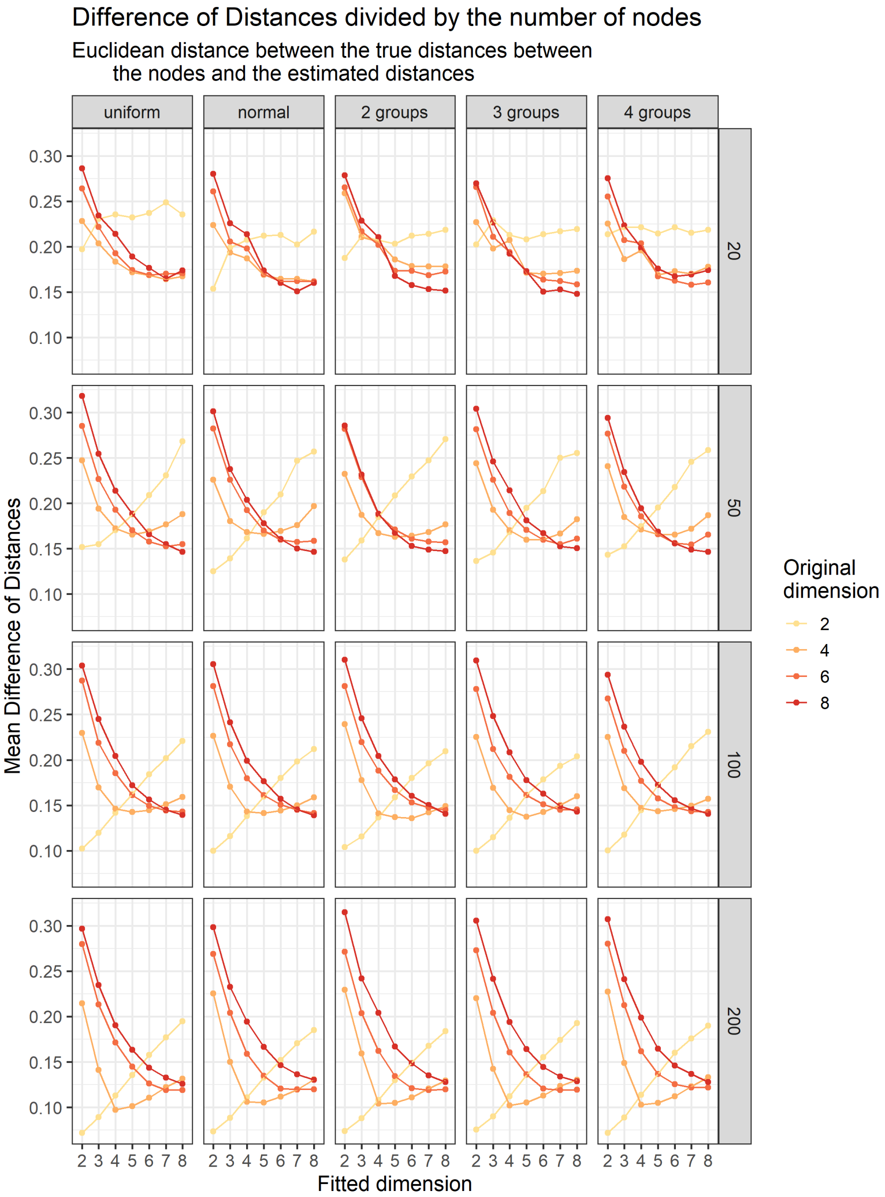


Figure A2: Mean difference of distances between the true, original distances between the nodes and the estimated distances between the nodes after Scaling for each distribution, number of nodes, number of original dimensions and number of fitted dimensions divided by the number of nodes

B Variance of the Difference of Distances

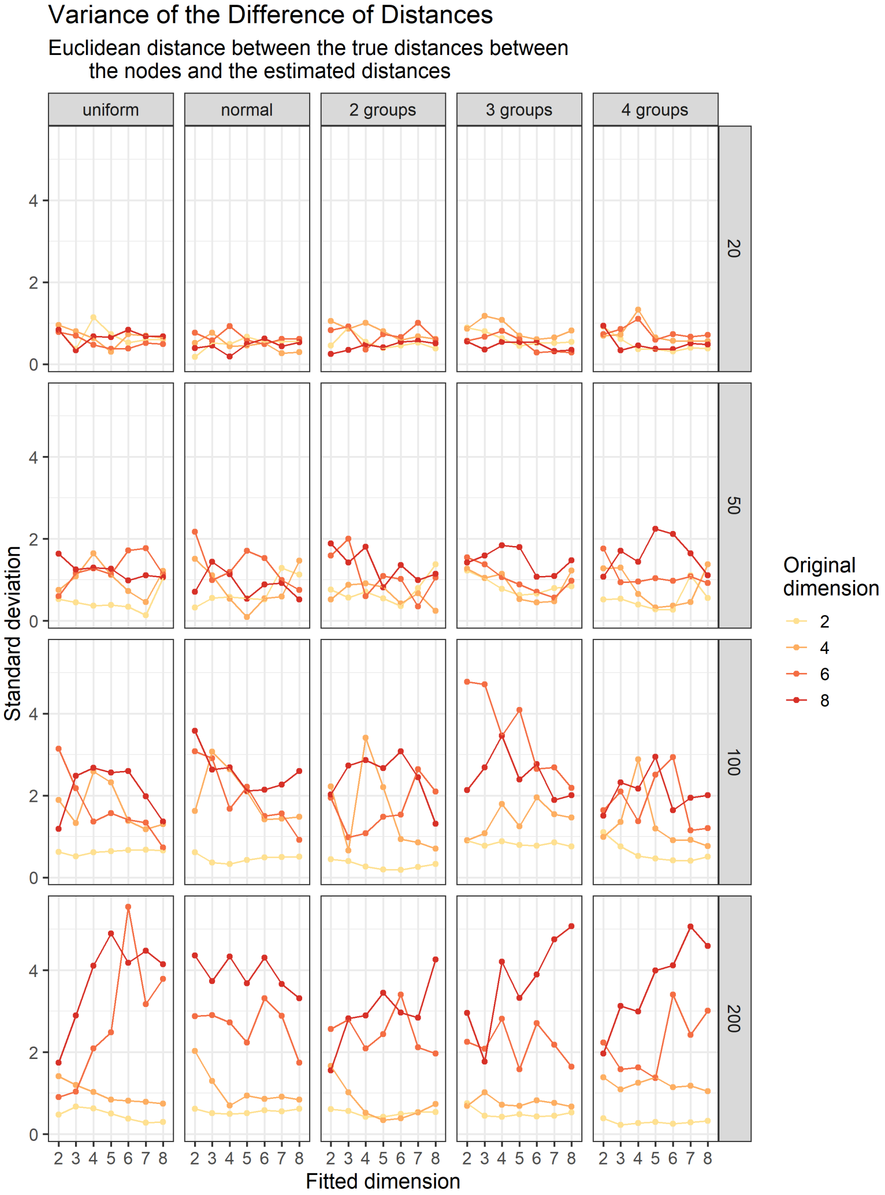


Figure B1: Standard deviation of the Difference of distances between the true, original nodes and the estimated distances between the nodes after Procrustes rotation for each distribution, number of nodes, number of original dimensions and number of fitted dimensions

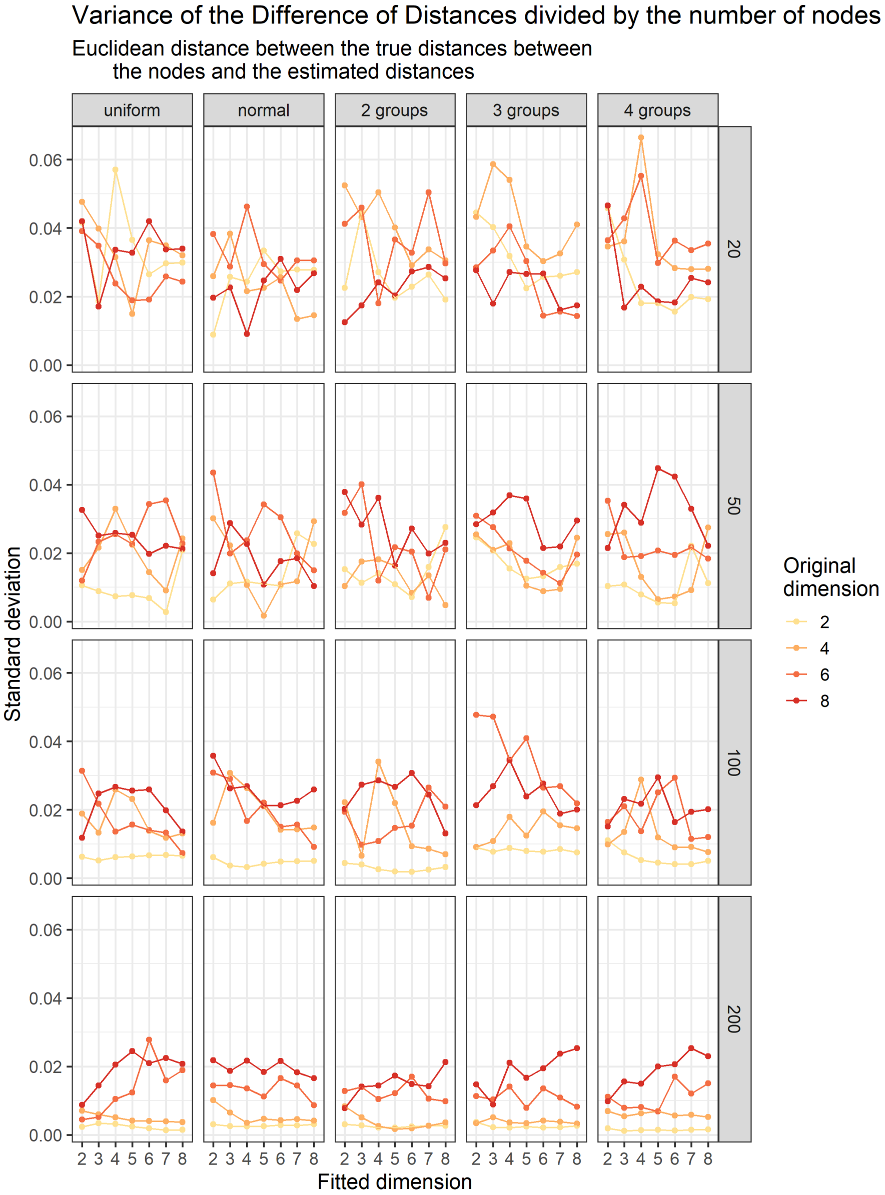


Figure B2: Standard deviation of the Difference of distances between the true, original nodes and the estimated distances between the nodes after Procrustes rotation for each distribution, number of nodes, number of original dimensions and number of fitted dimensions divided by the number of nodes