Optimal Bayes Estimators for Latent Position Network Models

Marco Lam

July 2019

The latent position model is a widely used statistical model for the analysis of social network interactions. This model postulates that the nodes are embedded as points in a Euclidean social space, and that nodes that are close in this space are more likely to exhibit social interactions. One of the main difficulties encountered is that the likelihood of a latent position model is unaffected by rotations, translations and reflections of the latent positions. This in turn creates a non-identifiability problem. The goal of this research project is to adopt a decision theoretic approach to define an optimality criterion that can be used to summarise a posterior sample of latent positions. This would allow one to extract a meaningful point estimate for the model's parameters which would overcome said identifiability issues.

Contents

1 Introduction

1.1 Latent Position Models

Network models are used to study actors and the relationship between them. Latent position model was first introduced by Hoff, Raftery and Handcock [1] as an alternative approach to network modelling. The estimation of the latent positions is usually performed in a Bayesian setting using Markov chain Monte Carlo methods to sample from a posterior distribution of interest. Samples of positions for each actor can be captured from this algorithm. In this paper, we will primarily focus on the 2 dimensional space using the Euclidean distances.

1.2 Procrustes Analysis

Procrustes analysis is a popular method to compare shapes between data while allowing superimposition on each individual set of points in the data set. Procrustes imposition includes scaling, rotating, translating and reflecting. In this paper, we will be using Partial Procrustes analysis which does not allow scaling from the superimposition.

Partial Procrustes analysis does not affect the representativeness of a latent position model. Two latent position models can have different coordinates for their actors but they would represent the exact same information if the relative positions of a specific actor to the others are the same in both models.

1.3 The Non-Identifiability Problem

Generally, the posterior mode of the samples generated from the MCMC algorithm is chosen as the optimal positions for the actors [1]. Alternatively, one could apply Partial Procrustes superimposition on each sample to achieve maximum likeliness between these MCMC samples, then the optimal position for each actor is the average position from the transformed positions.

In the latent position model, the position of each actor is meaningless without considering the relative position of this actor to the others in the latent space, since each latent position model can be modified under Partial Procrustes superimposition as mention in 1.2. Therefore, we will try to come up with a method that will determine the optimal positions for the actors by looking at the distances between them, rather then the actual positions.

The following approach is suggested using the Bayes Estimator to give penalties (or losses) to the difference in total distances between different MCMC samples.

2 Bayes Estimator

2.1 Definition

Suppose an unknown parameter θ is known to have a prior distribution π. i.e. $\theta \sim \pi$. Let $\hat{\theta}$ be an estimator of θ, and L be a loss function.

Then the **Bayes Risks** is given by

$$\mathop{\mathbb{E}}_{\pi}(L(\theta, \hat{\theta}))$$

and the **Bayes Estimator** is

$$\arg\min_{\hat{\theta}} \mathop{\mathbb{E}}_{\pi}(L(\theta, \hat{\theta}))$$

Generally, a loss function $L(\theta, \hat{\theta})$ returns a loss or a cost that describe the difference or the error of the predictor $\hat{\theta}$ to the true value θ. There is not a general rule of how to pick or build a loss function. However, the two loss functions below are widely used.

Mean Absolute Error:

$$\mid \theta - \hat{\theta} \mid$$

$$\frac{1}{n}\sum_{i=1}^{n} \mid \theta_i - \hat{\theta}_i \mid$$

Mean Square Error:

$$(\theta - \hat{\theta})^2$$

$$\frac{1}{n}\sum_{i=1}^{n}(\theta_i - \hat{\theta}_i)^2$$

Ultimately, we are interested in finding out the sample with the optimal positions from all our samples based on the total distances. In the other words, the total distances between all actors in this optimal sample has the least difference to other samples, compared to picking any other samples as the optimal solution.

If we set

$$d_{ij} = \|z_{i\cdot} - z_{j\cdot}\|_2 \quad \text{and} \quad d_{ij}^{(t)} = \|z_{i\cdot}^{(t)} - z_{j\cdot}^{(t)}\|_2$$

where $z_{k\cdot} = (z_{k1}, z_{k2})$ denotes the true latent position of k^{th} actor and $z_{k\cdot}^{(t)} = (z_{k1}^{(t)}, z_{k2}^{(t)})$ denotes the latent position of k^{th} actor in the t^{th} sample from the MCMC algorithm. We have

Mean Absolute Error:

$$\frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n} \mid d_{ij} - d_{ij}^{(t)} \mid$$

Mean Square Error:

$$\frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}(d_{ij} - d_{ij}^{(t)})^2$$

Denote $\boldsymbol{d} = \{d_{ij}\}_{i,j}$, and set $\hat{\theta} = \boldsymbol{d}$ and $\theta = \boldsymbol{d^{(t)}}$. Our Bayes Estimator is then

$$\arg\min_{\boldsymbol{d}} \mathbb{E}(L(\boldsymbol{d}, \boldsymbol{d^{(t)}}))$$

$$\arg\min_{\boldsymbol{d}} \frac{1}{T}\sum_{t}^{T}\frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}(d_{ij} - d_{ij}^{(t)})^2$$

$$\arg\min_{\boldsymbol{d}} \sum_{t}^{T}\sum_{i}^{n}\sum_{j}^{n}(d_{ij} - d_{ij}^{(t)})^2$$

and the loss function here is

$$\sum_{i}^{n}\sum_{j}^{n}(d_{ij} - d_{ij}^{(t)})^2 \tag{1}$$

2.2 Minimizing distances

Furthermore, the equation of this particular Bayes Estimator can be simplified.

To minimise $\boldsymbol{d}$:

$$\frac{\partial}{\partial \boldsymbol{d}} \sum_t^T \sum_i^n \sum_j^n (d_{ij} - d_{ij}^{(t)})^2 = 0$$

$$T \cdot \sum_i^n \sum_j^n d_{ij}^* = \sum_t^T \sum_i^n \sum_j^n d_{ij}^{(t)}$$

$$\sum_i^n \sum_j^n d_{ij}^* = \frac{1}{T} \sum_t^T \sum_i^n \sum_j^n d_{ij}^{(t)}$$

$$\sum_i^n \sum_j^n d_{ij}^* = \sum_i^n \sum_j^n \overline{d_{ij}^{(t)}}$$

$$\Rightarrow \boldsymbol{d}^* = \overline{\boldsymbol{d}}$$

It is important to note that we cannot use the results from the derivation above as our solution. This derivation provides a matrix that contains the means of distances between each pair of actors. In doing this, the solution is no longer a proper distance matrix. The triangle inequality is not guaranteed to hold for this matrix necessarily. However, this derivation does help to simplify our equation for the Bayes Estimator. Thus, the following theorem is proposed.

Theorem 1.

$$\arg\min_{\boldsymbol{d}} \mathbb{E}(L(\boldsymbol{d}, \boldsymbol{d^{(t)}})) = \arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (d_{ij} - \overline{d_{ij}})^2$$

Proof.

$$\arg\min_{\boldsymbol{d}} \sum_{t}^{T} \sum_{i}^{n} \sum_{j}^{n} (d_{ij} - d_{ij}^{(t)})^2$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (T \cdot d_{ij}^2 + \sum_{t}^{T} (d_{ij}^{(t)})^2 - 2 \cdot d_{ij} \sum_{t}^{T} d_{ij}^{(t)})$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (T \cdot d_{ij}^2 - 2 \cdot d_{ij} \sum_{t}^{T} d_{ij}^{(t)})$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (T \cdot d_{ij}^2 - 2 \cdot d_{ij} \cdot T \cdot \overline{d_{ij}})$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (d_{ij}^2 - 2 \cdot d_{ij} \cdot \overline{d_{ij}})$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (d_{ij}^2 + \overline{d_{ij}}^2 - 2 \cdot d_{ij} \cdot \overline{d_{ij}})$$

$$\arg\min_{\boldsymbol{d}} \sum_{i}^{n} \sum_{j}^{n} (d_{ij} - \overline{d_{ij}})^2$$

□

This equation is computationally cheaper than before. It is t times quicker as it is not required to deal with one of the summations as before. This is particularly helpful for a large sample size.

3 Extension from Bayes Estimator

3.1 Classical Multidimensional Scaling

So far, the approach we have looked at uses one of the iterations from the MCMC algorithm as the solution. This approach would be very limited with a small number of samples to work with. To improve performance, the amount of samples must increase but this will also increase computational expense together. As the theorem suggests above, a proper distance matrix is always preferred if it's closer to $\overline{\boldsymbol{d}}$ than any other option of matrices does. Rather than using the Bayes Estimator method, we can consider to find a matrix of distances that is as similar to $\overline{\boldsymbol{d}}$ as possible while the matrix itself satisfies the triangle inequality. To achieve this, classical multidimensional scaling (CMD scaling) can be used. By definition, CMD scaling find a set of points that have distances as close to the ideal set of distances (in our case, $\overline{\boldsymbol{d}}$) as possible.

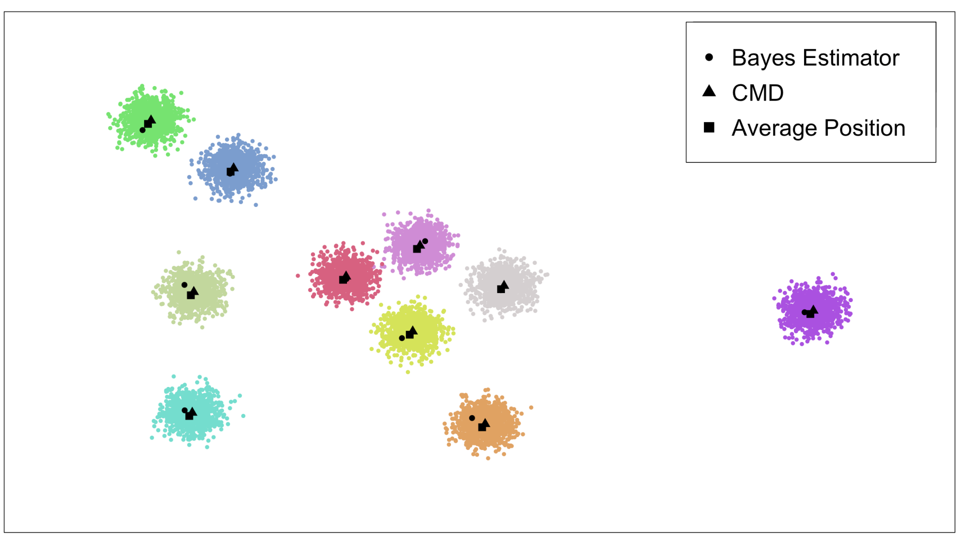
By using CMD scaling, we now would only need to find $\overline{\boldsymbol{d}}$ and apply CMD scaling to $\overline{\boldsymbol{d}}$ to obtain our optimal latent positions, this should be an even more efficient and more accurate approach in terms of our loss function, compared to the methods that were mentioned above.

4 Examples

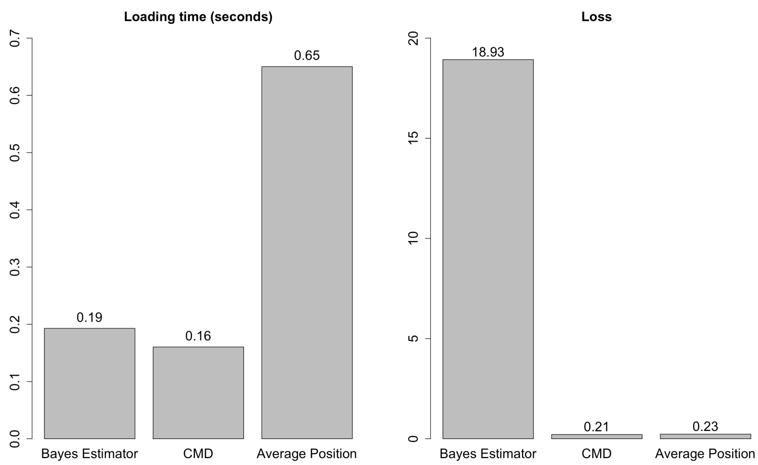
We will now look at two examples below that have been implemented in R studio. In this section, the Bayes Estimator method refers to the result from Theorem 1, with the loss function from the expression 1. The CMD scaling method refers to the approach mentioned in 3.1, the function `cmdscale` from R is used [4].

4.1 Generated Samples

In the following example, we will generate samples of latent positions for 10 actors. The two methods listed above in section 4 are then used to compared with the method of taking average positions after applying Procrustes superimposition to each sample. The efficiencies and accuracies of these methods will be compared according to our loss function. The "true" latent positions of these 10 actors are generated using a Gaussian distribution with a zero mean and variance of 10. 1000 samples are then created by adding an error term to the true positions for each actor, these error terms are generated using another Gaussian distribution with zero mean and variance of 1.

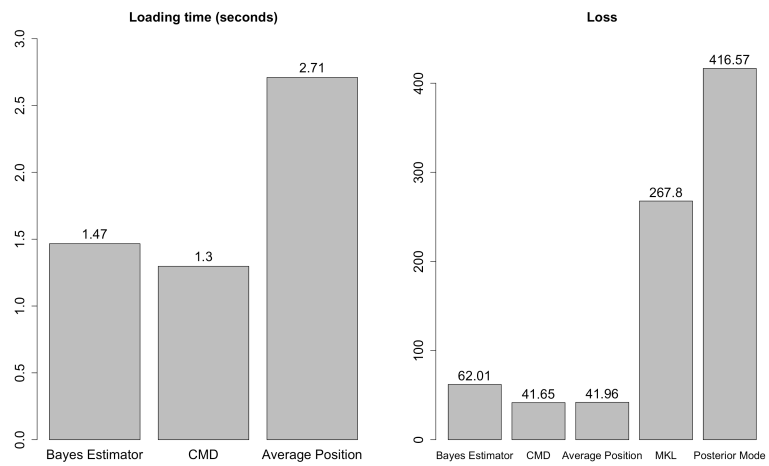


After running the three algorithms, it is clear that the CMD scaling method is the quickest and the most accurate (according to our loss function) as expected. Meanwhile, the Bayes Estimator is providing a terrible result despite not being too much slower to run then the CMD scaling. Lastly, the average positions has a good accuracy, but it is very expensive computationally compared to the other methods.



4.2 Monks Data

An ethnographic study of community structure in a New England monastery between 18 Monks was collected by Samuel F. Sampson during the 1960's. By using the `latentnet` package in R, 4000 samples of latent positions are captured for these 18 Monks [2]. The package also provided results of the optimal positions with two other approaches, the posterior mode from the MCMC algorithm and the Minimum Kullback-Leibler (MKL) method [3]. Although the efficiency of these two methods cannot be referenced in this case, the accuracies of these 2 methods according to our choice of loss function can be obtained.



In terms of the efficiency, CMD scaling was the fastest to run in R, as expected. Similar to the previous example, the Bayes Estimator is almost as efficient as the CMD scaling. The average position method was very computational expensive here.

Between these three methods, the accuracies are similar to the ones from the last example. However, the performances of the 2 methods provided in the package are very poor. After applying Procrustes Superimposition to the latent positions calculated by all five different methods, it is clear that the optimal positions from the 2 methods we propose provide a denser set of latent positions compared to the methods from the package.

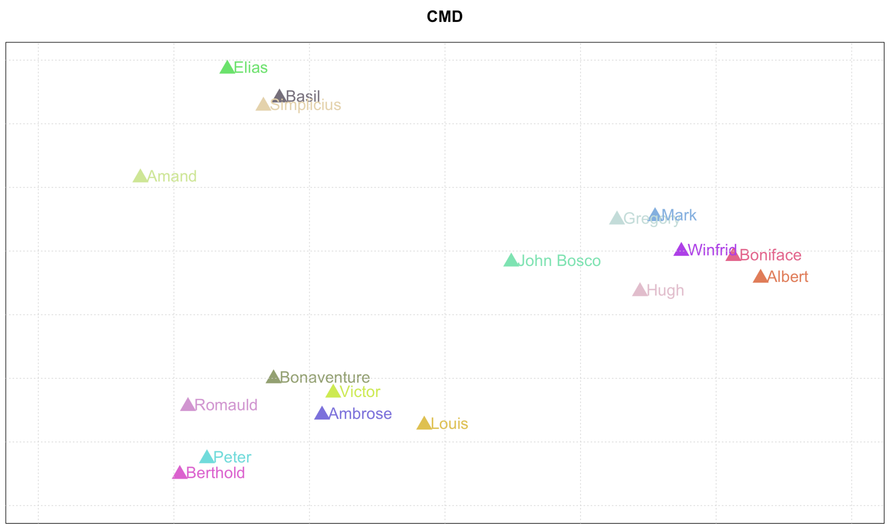


Figure 1: Latent Positions determined by CMD scaling

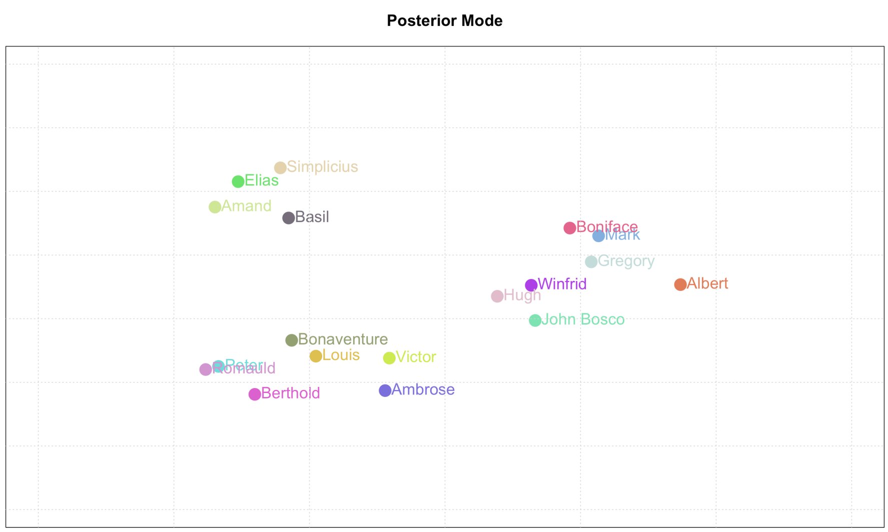


Figure 2: Latent Positions determined by Posterior Mode

5 Simulation

Once the optimal latent positions are chosen, by using

$$p_{ij} = \frac{e^{\beta - d_{ij}}}{1 + e^{\beta - d_{ij}}}$$

the probability of any two actors having a relationship can be found whether it is undirected or directed [1]. After, simulations can be done based on these probabilities and thus we have the total number of edges within the latent position model.

6 Conclusion

In this report, we have tried to propose new methods to select an optimal position for each actor in a latent space. These methods have an advantage such that the optimal positions are selected based on a theoretical results. At first, we decided to use Bayes Risk and Bayes Estimator to select a sample from our data with the least difference in total distances between actors compared to the rest of the data. We chose to use the MSE of distances as our loss function. As we looked into the computation area of this algorithm, we managed to propose a theorem and managed to simplify the algorithm even further. Finally, we suggested an additional approach that the classical multidimensional scaling can be used on the average distances between actors from our samples. The last algorithm makes it even cheaper computationally compared to the previous approaches. We have also looked at two examples to see how these algorithms would behave in these two scenarios. The classical multidimensional scaling approach was the most efficient as expected.

Acknowledgement

I would like express my great appreciation to Dr. Riccardo Rastelli, my supervisor, for his valuable and constructive suggestions and guidelines during the planning and development of this research work. I would also like to thank the University College Dublin School of Mathematics & Statistics for the opportunity and support for the Undergraduate Summer Research Project 2019.

References

[1] Peter D. Hoff, Adrian E. Raftery, and Mark S. Handcock. *Latent Space Approaches to Social Network Analysis*. Journal of the American Statistical Association 97 (460), 1090-1098, December 2002.

[2] Pavel N. Krivitsky *Latent space models with latentnet* February 2014.

[3] Latent Position and Cluster Models for Statistical Networks. `https://cran.r-project.org/web/packages/latentnet/latentnet.pdf` 2018

[4] Classical (Metric) Multidimensional Scaling. `https://www.rdocumentation.org/packages/stats/versions/3.6.1/topics/cmdscale`