Review of Statistical Network Analysis: Models, Algorithms and Software Supplementary Material

M. Salter-Townshend[*], A. White, I. Gollini and T.B. Murphy[†‡]

March 22, 2012

Abstract

We provide chosen results of the application of some of the models described in our paper "Review of Statistical Network Analysis: Models, Algorithms and Software" (Statistical Analysis and Data Mining, 2012), when applied to the Caltech University Facebook dataset (Traud et al., 2011). We provide commented R code which may be downloaded from `www.maths.ucd.ie/~mst/networks/R/`. For details on all models, methods and software please see the original paper.

[*] Corresponding Author

[†] Clique Strategic Research Cluster, School of Mathematical Sciences & Complex and Adaptive Systems Laboratory, University College Dublin, Dublin 4, Ireland.

[‡] This material is based upon work supported by the Science Foundation Ireland under Grant No. 08/SRC/I1407: Clique: Graph & Network Analysis Cluster

1 Descriptive Statistics

1.1 Network Summary Statistics

There are very strong local ties within the network, and influential nodes. The following table was obtained using the *statnet* suite of R packages. The R code used to obtain this table and the centrality measures in Section 1.2 can found in `SummaryStats.R`.

edges	triangle	2-star	3-kstar
16651	119562	1231409	40583909

1.2 Centrality and Prestige

Four centrality measures, degree, betweenness, closeness and eigenvector, were applied to the dataset. Each measure identified node 619 as the most central; the actor boasts 496 ties in the network. All measures seemed to roughly agree in terms of the score assigned to actors. Table 1 shows the correlation between centrality scores for each measure. In particular, degree and eigenvector centrality scores have a 98% correlation; note though that the relationship between measures in several cases was non-linear. Figure 1 gives a plot of the relationship between eigenvector and closeness centrality.

	Betweenness	Closeness	Degree	Eigenvector
Betweenness	1.00	0.49	0.75	0.66
Closeness	0.49	1.00	0.84	0.86
Degree	0.75	0.84	1.00	0.98
Eigenvector	0.66	0.86	0.98	1.00

Table 1: Correlation Table of centrality measures applied to Caltech dataset.

1.3 Community Finding and Clustering

Spectral Clustering result: Singular-Value decomposition of the graph Laplacian and clustering of the resulting 8 largest eigenvectors resulted in a clustering that is closely related to the recorded dormitory assignments of the students. Specifically, the contingency between the clustering and the dorms was strongly statistically significant and Cramér's V was found to be 0.642. We provide code in `Spectral.R` for this analysis.

Girvan-Newman Clustering result: We provide code which attempts to find community structure in the Caltech data via the Girvan-Newman algorithm in

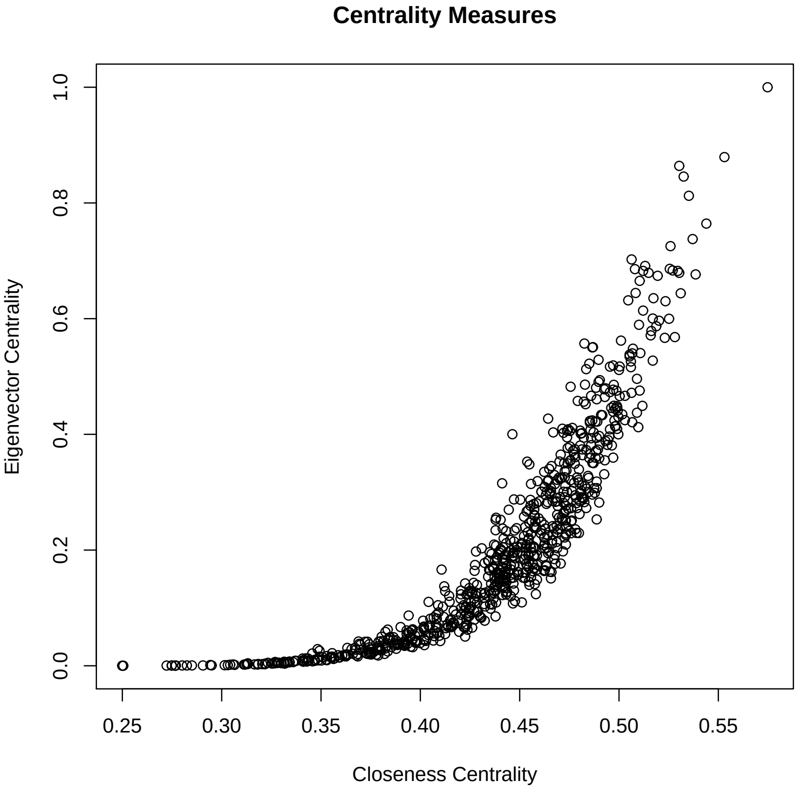

Figure 1: While there is a strong positive relationship between the centrality measures, the relationship is also clearly non-linear.

`GirvanNewman.R`. Clustering with this method did not return a satisfactory community structure, with the nodes being clustered into 517 groups. The three largest groups had 130, 84 and 16 members respectively, while 507 of the clusters had only single membership. The modularity score of the graph as edges were deleted is shown in Figure 2. Note that the maximum modularity score is only about 10%. While this particular algorithm fails to cluster actors in a satisfying manner, approaches with similar but more sophisticated algorithms have been applied to this data with more success (Traud et al., 2011).

2 Visualization

2.1 Adjacency Matrix Visualizations

Figure 3 uses `hclust.R` to create an adjacency matrix visualization of the Caltech network. A strong local ties structure is apparent from the blocks along the diagonal.

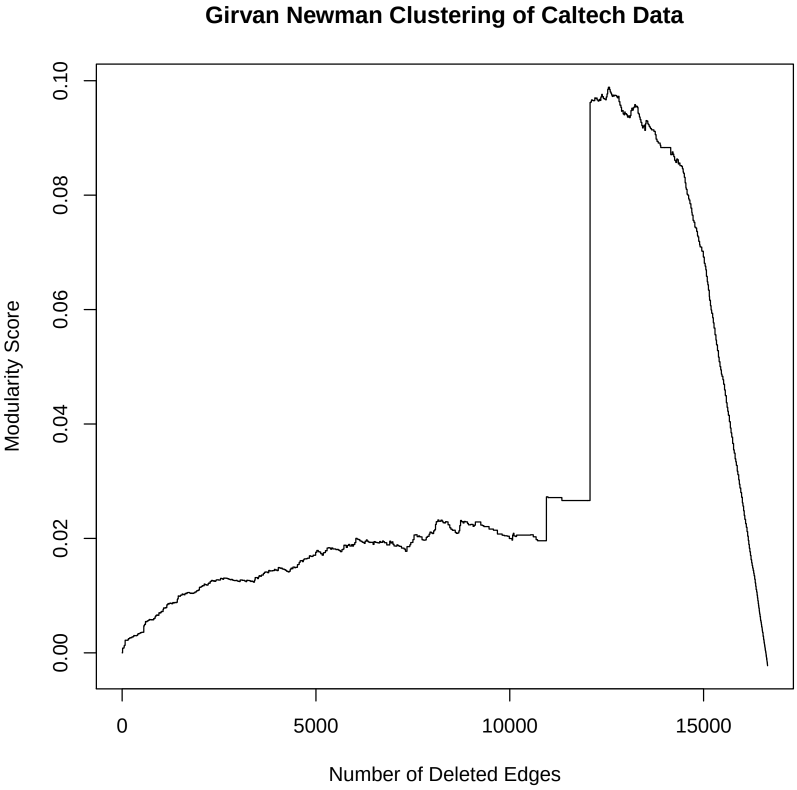

Figure 2: This plot shows the modularity score of the Caltech network as edges are removed, following the Girvan Newman algorithm. This clustering was performed using the *edge.betweenness.community* and *modularity* functions in the `igraph` package.

2.2 Layout Algorithms

We present a subset of the common layout algorithms applied to the Caltech network. We do not include the circle layout or the random layout methods as these result in “hairball” plots for a network of this size and density. Even the layouts that we do include (Reingold-Tilford, Fruchterman-Reingold, MultiDimensional Scaling and Singular Vector Decomposition in Figures 4 to 7 respectively) result in plots that are difficult to discern structure from. This shows the difficulty in visualizing large real-world networks.

3 Classical Models

3.1 Erdős-Rényi

This simplistic model estimates a scalar probability of linkage equal to the mean of the adjacency matrix. In this case it is 0.0574.

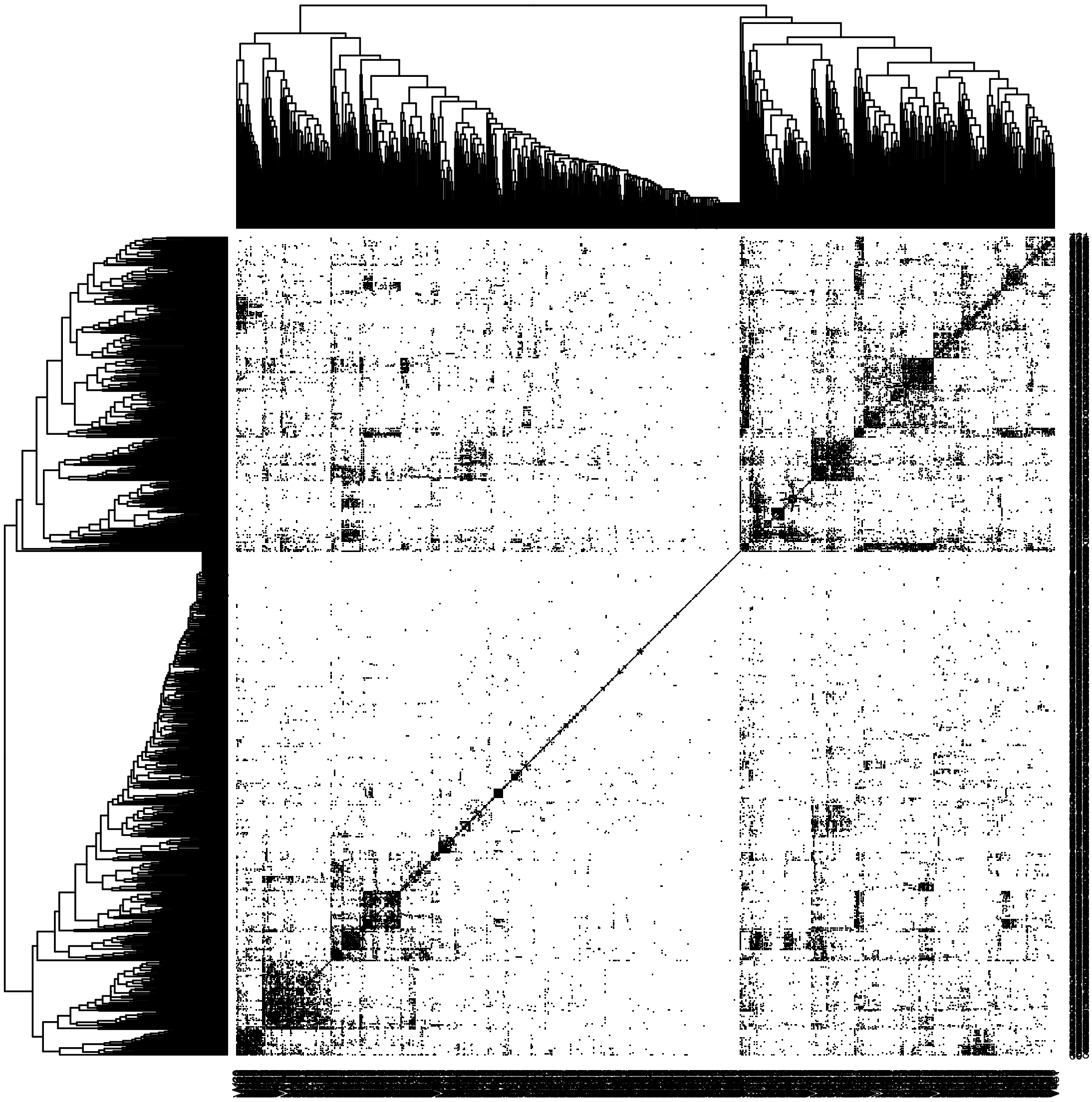

Figure 3: A hierarchical clustering adjacency matrix visualization. This is the result of the *heatmap* function in R on the Caltech dataset. The hierarchical clustering is performed using *hclust*.

3.2 p1 and p2

The p_1 and p_2 models could not be fit to the Caltech network as they are intended only for directed networks. We do fit a modified p_2 model (without the reciprocity parameter) to the Caltech data with `p2.R` using all 8 recorded covariates. Note that the code took several hours to run. Goodness-of-fit is discussed for this code in Section 5.

We also provide Generalized Linear Mixed Model R code for fitting the p_2 model to the Lazega Lawyers friendship dataset (Lazega, 2001) in `p2_lawyers.R`. We use a linear mixed model framework instead of the MCMC algorithm used in van Duijn et al. (2004) but achieve similar results for the same choice of covariates

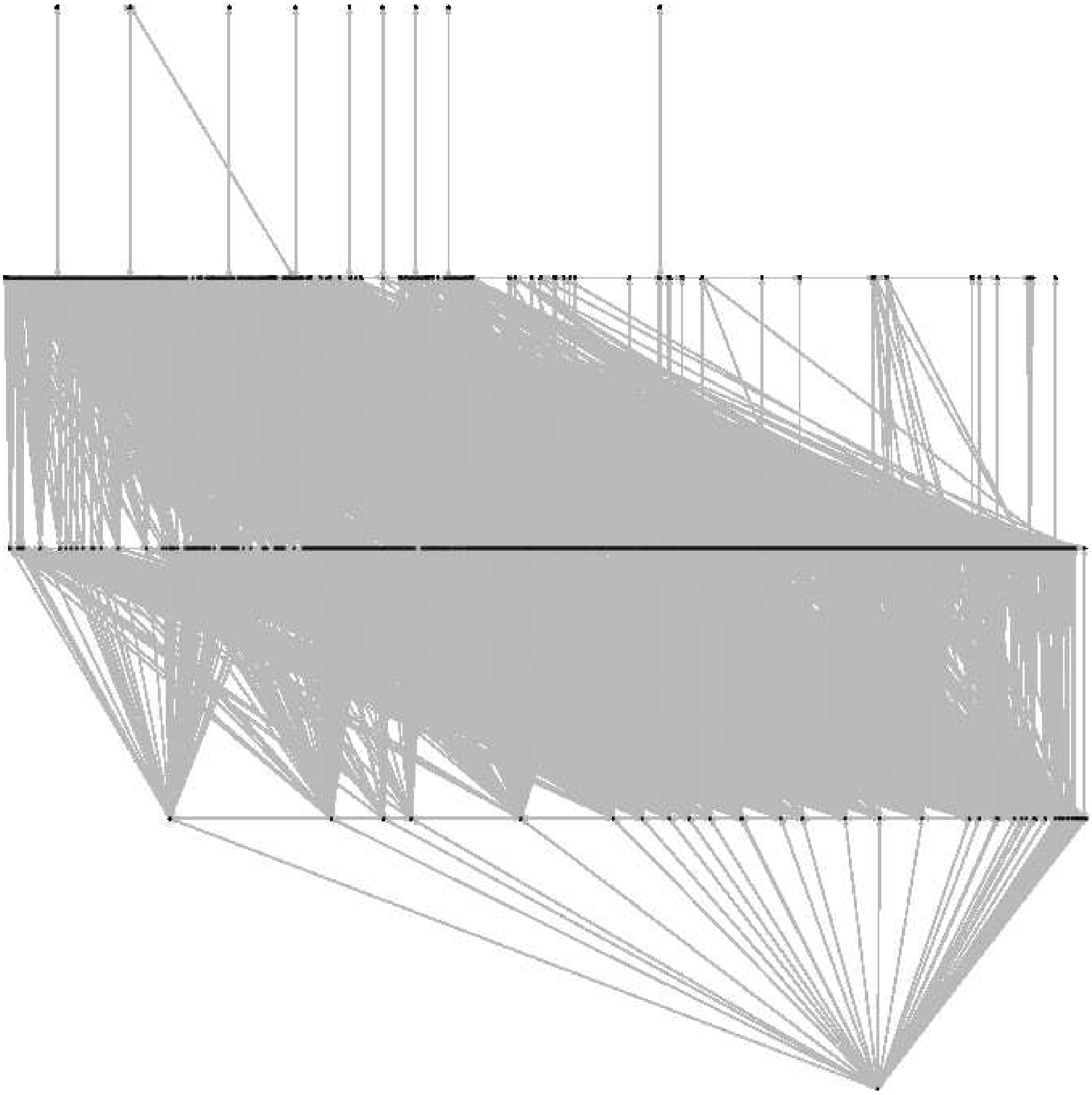

Figure 4: The results of the *plot.igraph* function in R to the Caltech dataset, with Reingold-Tilford layout.

for all three p_2 models in Table 1 of van Duijn et al. (2004).

3.3 Block Models

Using `blockmodel.R` we fit a blockmodel based on observed dorm allocations to the Caltech network. The result is shown in Figure 8.

3.4 Exponential (family) Random Graph Models

We fit an ERGM to the Caltech dataset using the terms: *number of edges, triangles* and *geometrically weighted degree* as these are among the most common choices for ERGM terms. Goodness-of-fit results are presented in Section 5. We acknowledge that better goodness-of-fit could be achieved using a more carefully chosen set of

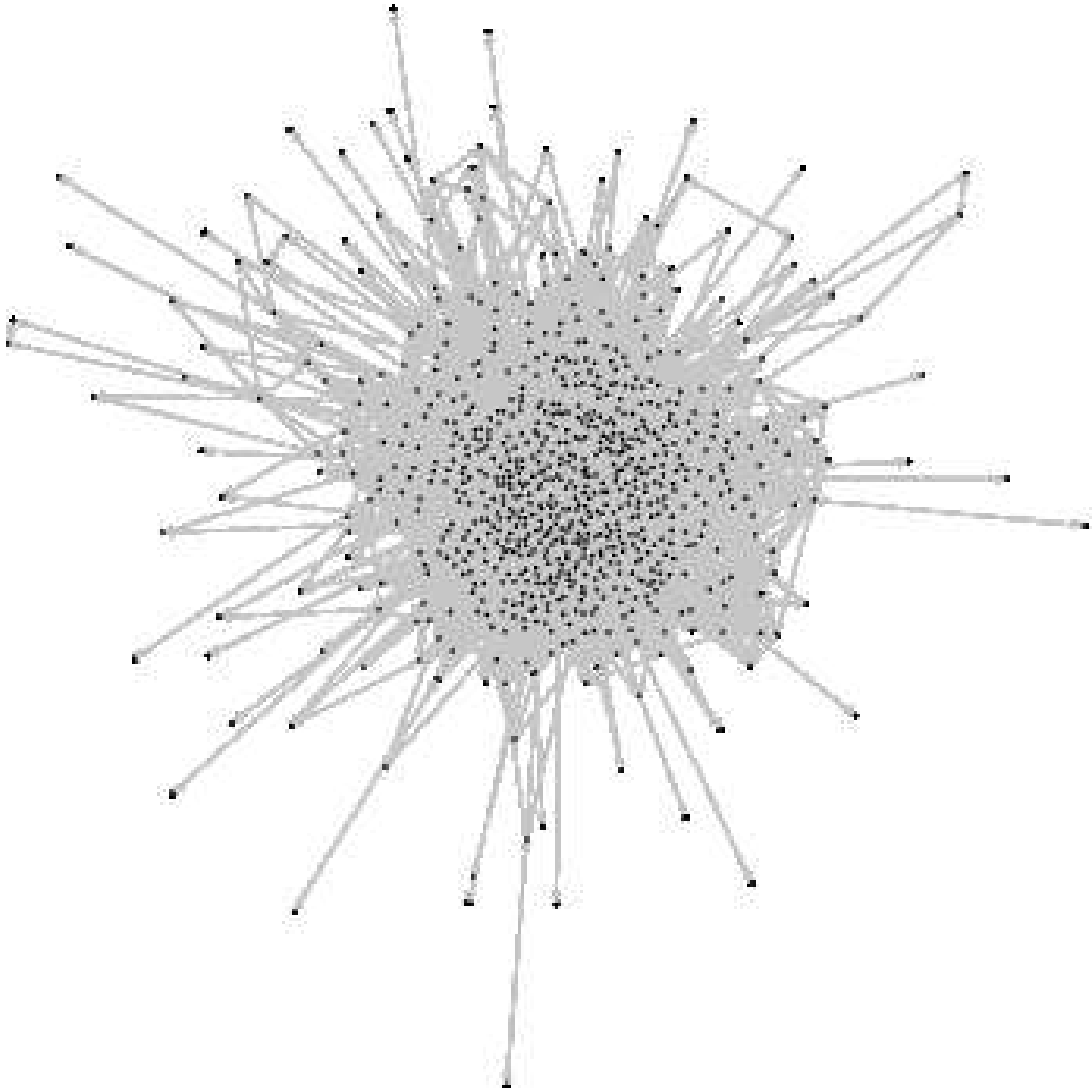

Figure 5: The results of the *plot.igraph* function in R to the Caltech dataset, with Fruchterman-Reingold layout.

network summary statistics. We used the `ergm` package in R and the code took approximately 35 seconds to run on a 3 GHz machine. The code is provided in `ergm.R`.

4 Latent Variable Models

4.1 Latent Block Models

Mixed membership stochastic blockmodels with 2,3,4,5 and 9,10,11 underlying blockmodels were fitted to the Caltech data. Models with smaller underlying classes appeared to fit the data better; this may be due to the manner (collapsed Gibbs sampling) with which the data was fit, however. In all cases a large amount of mixed

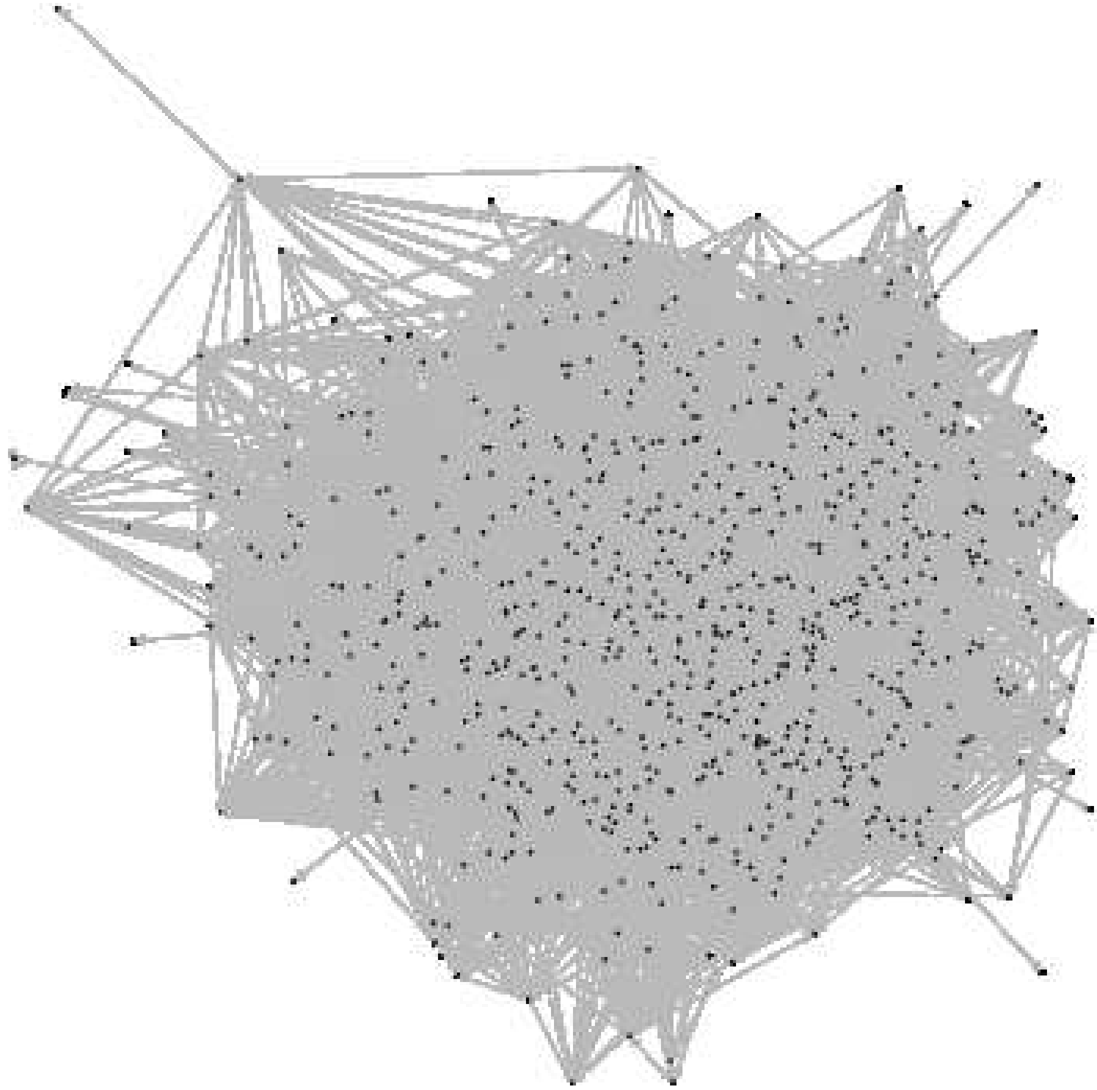

Figure 6: The results of the *plot.igraph* function in R to the Caltech dataset, with mds layout.

block membership occurred; this makes it difficult to identify prominent clusters in the data, although overall network behaviour is predicted accurately, as measured by AUC (see Section 5.1). We provide code using the using the R package *lda* in `MMSB.R`.

4.2 Latent Position Cluster Models

Goodness-of-fit results are presented in Section 5. We used the `VBLPCM` package (the MCMC based `latentnet` package could not cope with a network of this size) in R and the code took approximately 32 minutes to run on a 3 GHz machine to fit a variational Bayes approximation to a social random effects model with 15 groups in a 3 dimensional latent space. `lpcm.R` provides R code to fit the model and assess

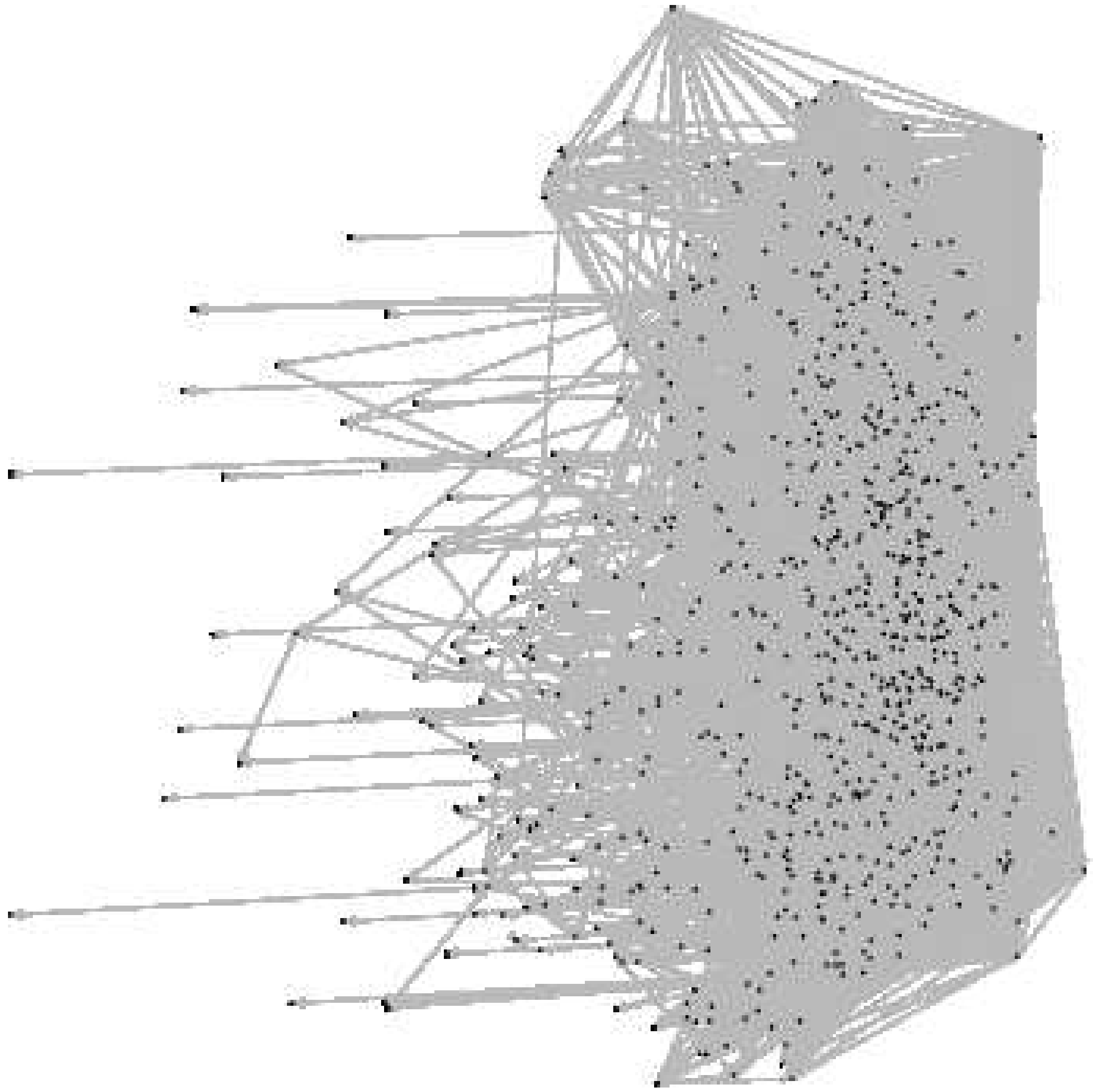

Figure 7: The results of the *plot.igraph* function in R to the Caltech dataset, with svd layout.

the fit using the techniques in Section 5.

5 Goodness-of-Fit and Validation

5.1 ROC Curves and AUC

We assess link prediction accuracy as measured by Area Under the Curve (AUC) of the Receiver-Operating Characteristic (ROC) curve. The ROC curve is the plot of true positive rate against false positive rate as the threshold probability above which a link is predicted is varied. Imputing links as per Erdős-Rényi yields an AUC of 0.5 (the model goes from predict all non-links to predict all links as the threshold passes the observed network density). Any completely random imputation of links

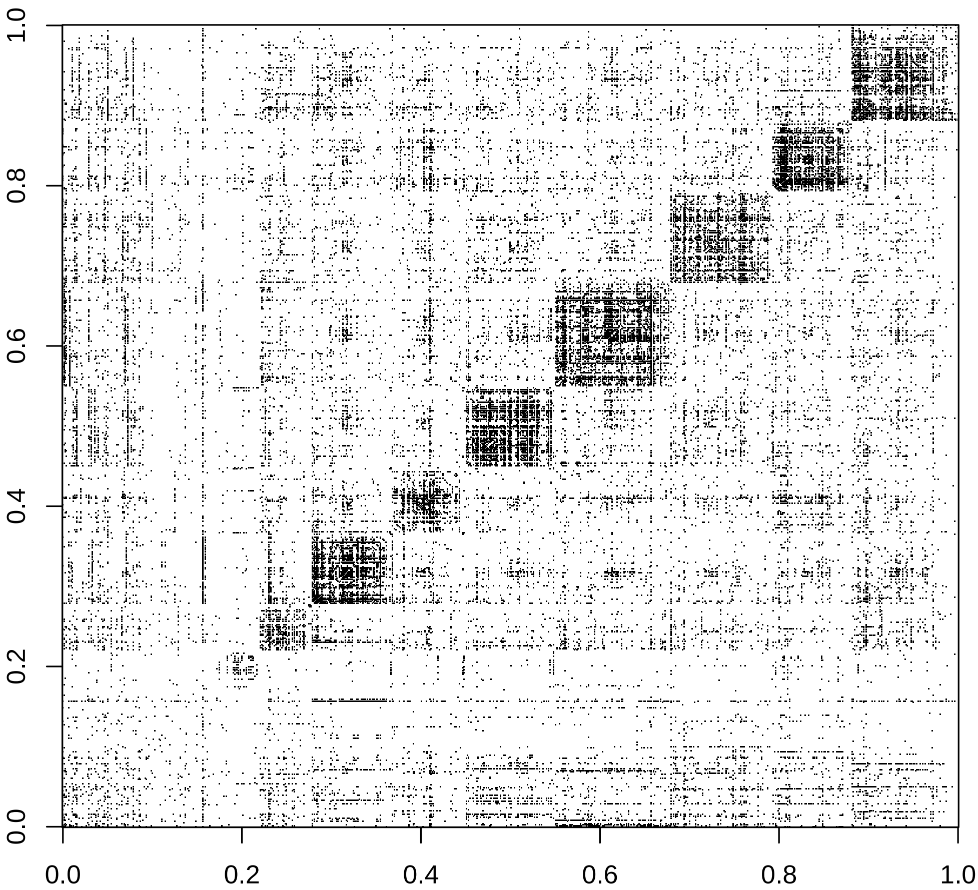

Figure 8: The results of the *blockmodel* function from the `sna` (Butts, 2010) package in R, applied to the Caltech dataset using `blockmodel.R`. The blocks were allocated based on observed dorms. The blocks along the diagonal are an indication that there is a higher density of links between people in the same dorm.

also yields an AUC of 0.5. Only a model that incorrectly predicts all links as non-links and vice-versa will score an AUC of 0 and a model that correctly predicts all possible links correctly will score an AUC of 1.

The symmetric p_2 model without reciprocity and using all 8 recorded nodal attributes achieved an AUC of 0.897. For the MMSB (see `MMSB.R`), the AUC was found to be 0.93 for the $K = 3$ block model and 0.858 for the $K = 10$ block model. For the LPCM, the AUC was 0.883 for 15 clusters in 3 dimensions and 0.879 for the 20 cluster model in 3 dimensional latent space, however the 15 cluster model scored better when comparing the cluster assignments with the observed dormitory assignments (Cramér's V 0.481 as opposed to 0.403).

5.2 Goodness-of-fit via Summary Statistics of Simulated Networks

The R code in the files `ergm.R` and `lpcm.R` include function to produce the graphs in this section. We compare the ERGM and LPCM fits using visual inspection of the distribution of some network summary statistics derived from networks simulated from the models fitted to the Caltech dataset. Figures 9 to 11 depict the fit for the ERGM in terms of the distribution of nodal degree, edgewise shared-partners and geodesic distance between dyads respectively. Similarly, Figures 12 to 14 depict results for the LPCM.

The ERGM performs better than the LPCM on simulating networks with the correct degree distribution but far poorer at capturing the edgewise shared-partner distribution. The ERGM performs better and indeed very well at capturing the geodesic distance distribution. The LPCM captured the largely dormitory driven communities whereas the ERGM does not inform about clustering or community finding.

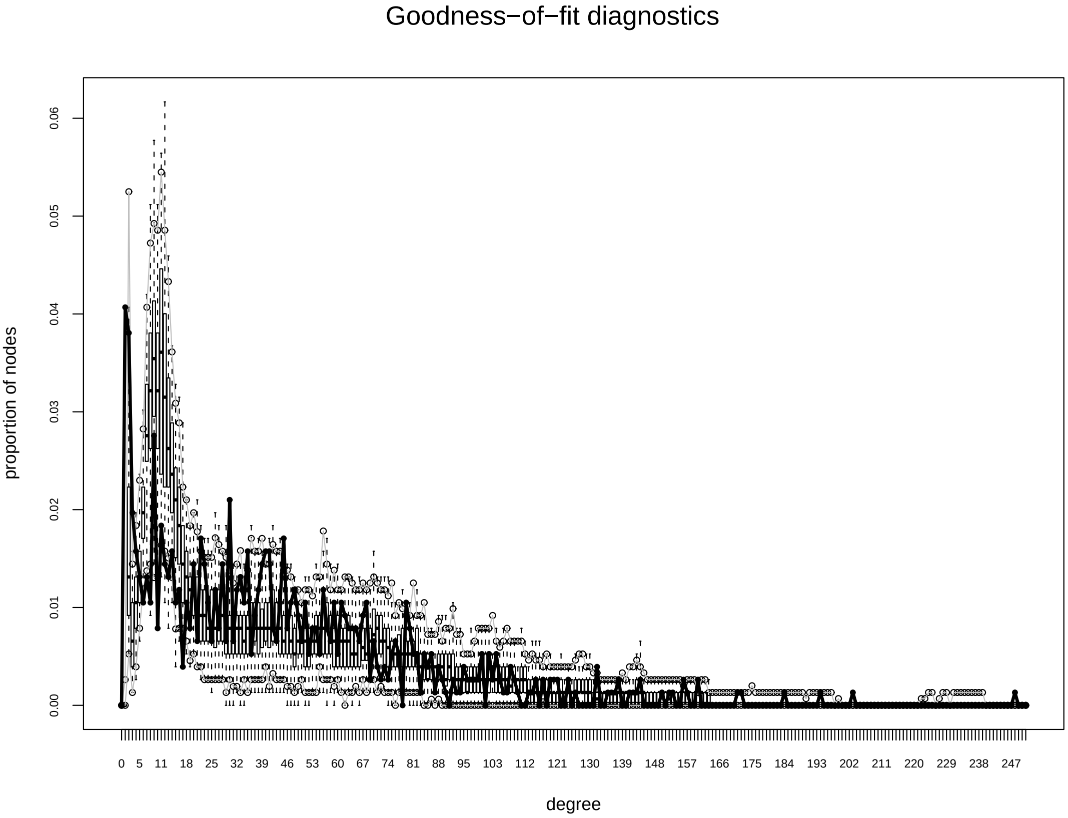

Figure 9: Degree goodness-of-fit plot for the Exponential Random Graph Model using edges triangles and geometrically weighted degree terms.

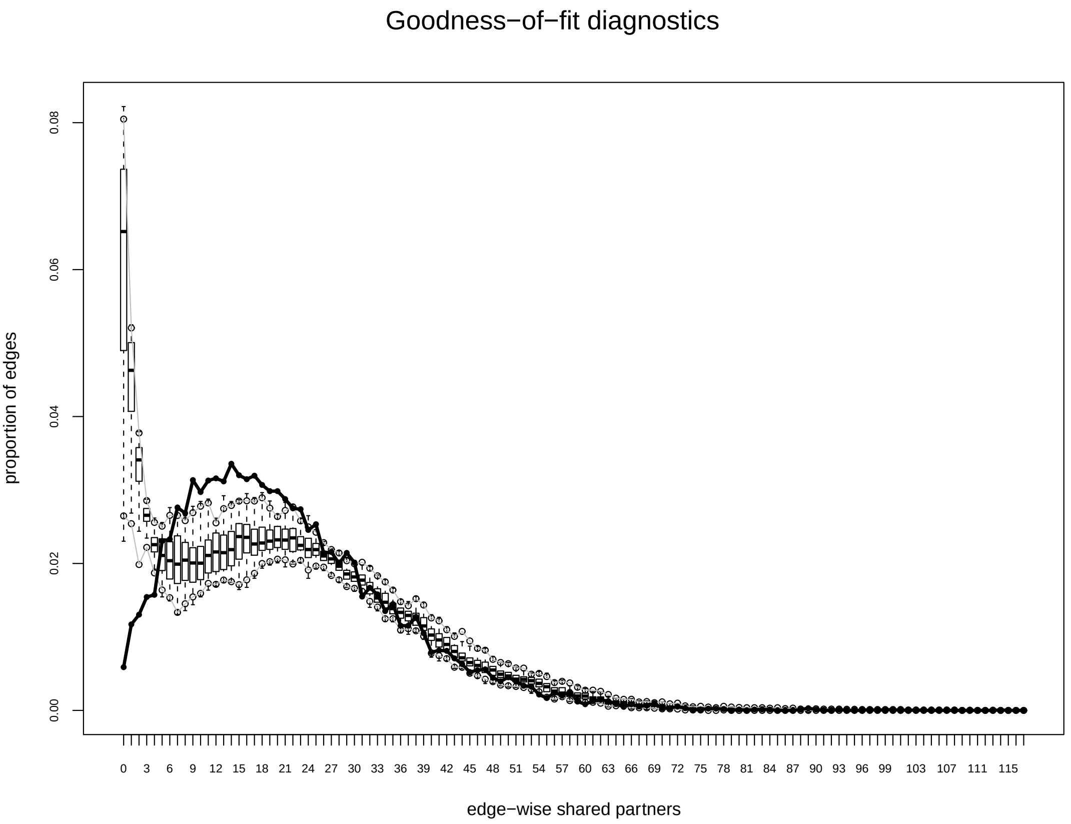

Figure 10: Edgewise-Shared-Partners goodness-of-fit plot for the Exponential Random Graph Model using edges triangles and geometrically weighted degree terms.

References

Butts, C. T. (2010). *sna: Tools for Social Network Analysis.* University of California, Irvine. R package version 2.1-0.

Lazega, E. (2001). *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership.* Oxford University Press.

Traud, A. L., E. D. Kelsic, P. J. Mucha, and M. A. Porter (2011). Comparing community structure to characteristics in online collegiate social networks. *SIAM Review 53*(3), 526–543.

van Duijn, M. A. J., T. A. B. Snijders, and B. J. H. Zijlstra (2004). p_2: a random effects model with covariates for directed graphs. *Statistica Neerlandica 58*(2), 234–254.

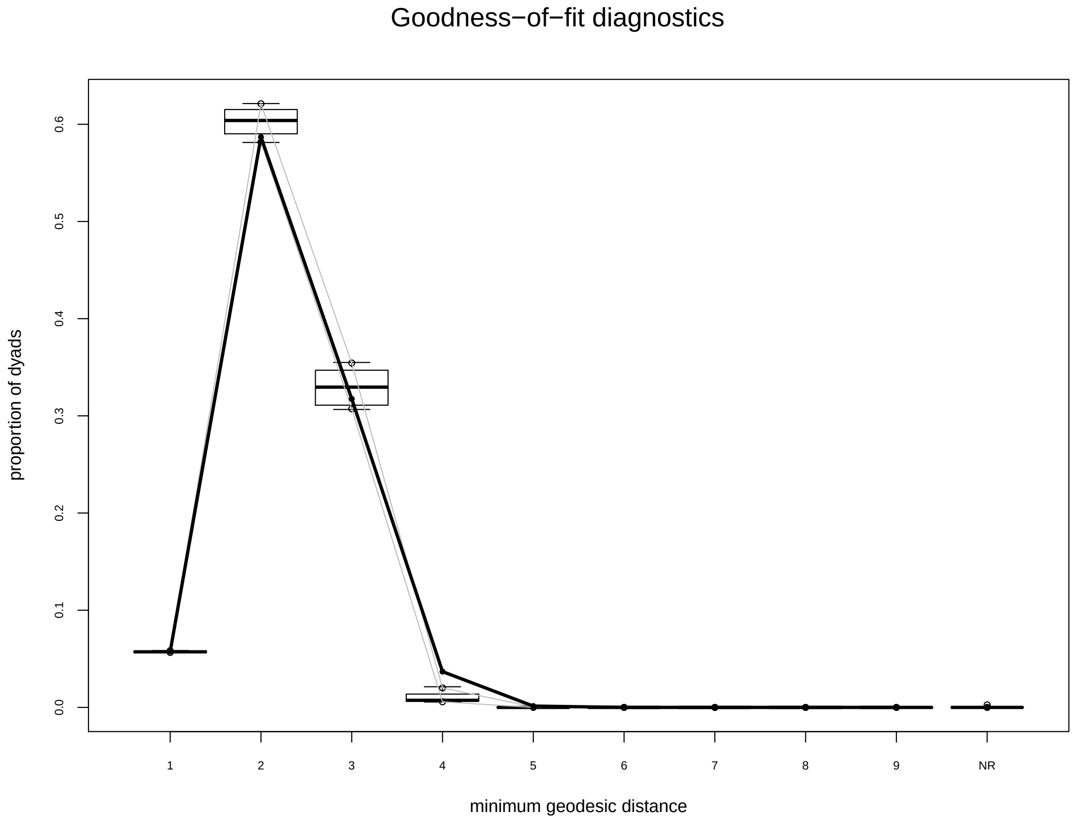

Figure 11: Geodesic distance goodness-of-fit plot for the Exponential Random Graph Model using edges triangles and geometrically weighted degree terms.

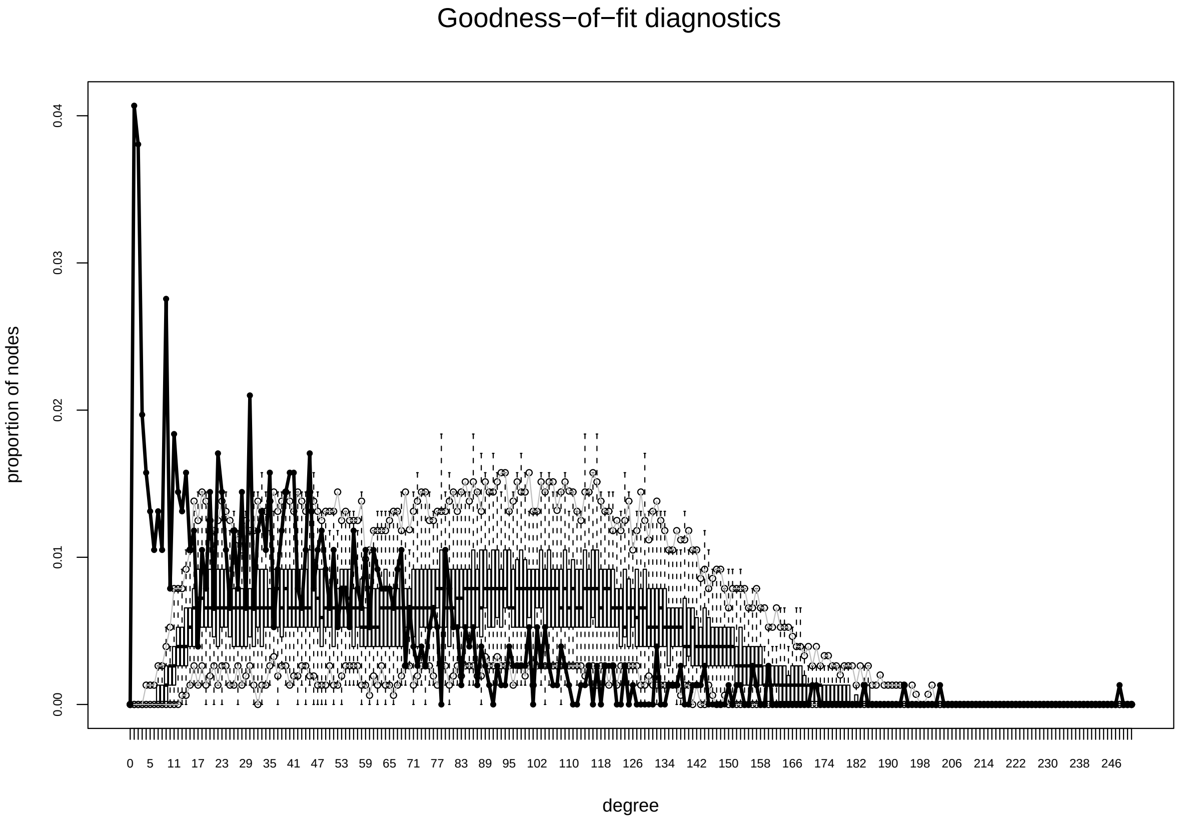

Figure 12: Degree goodness-of-fit plot for the Latent Position Cluster Model in 3 dimensions with 15 clusters.

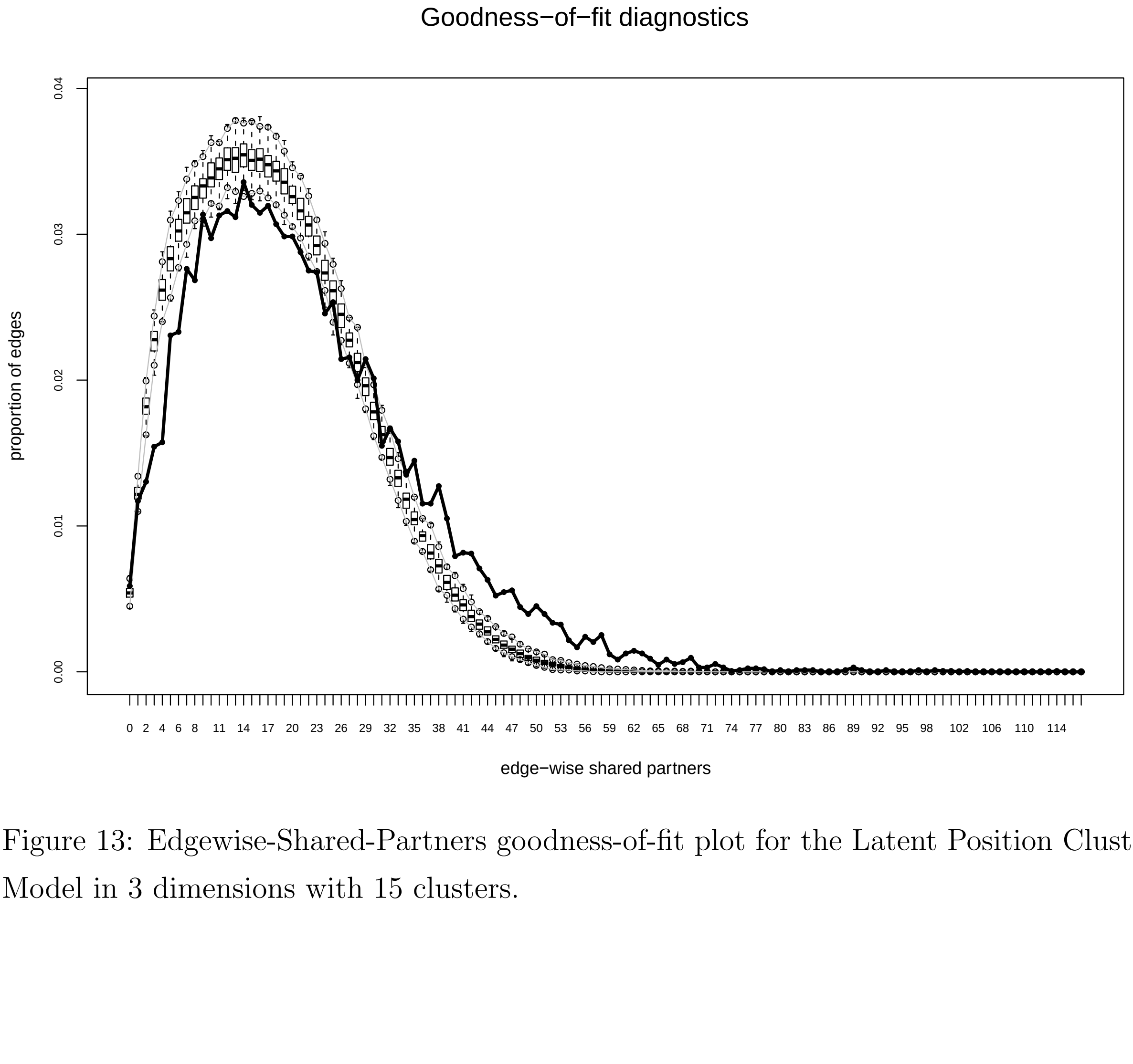

Figure 13: Edgewise-Shared-Partners goodness-of-fit plot for the Latent Position Cluster Model in 3 dimensions with 15 clusters.

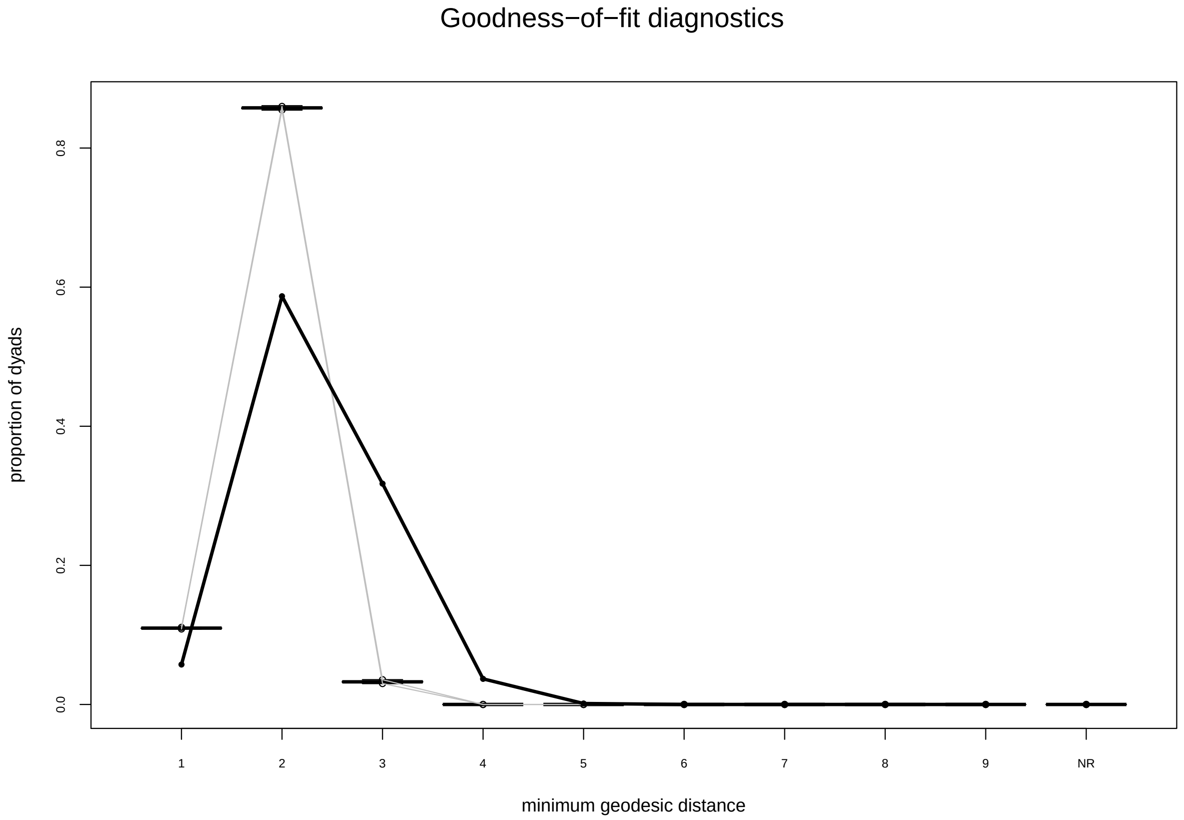

Figure 14: Geodesic distance goodness-of-fit plot for the Latent Position Cluster Model in 3 dimensions with 15 clusters.