Dynamic Latent Space Cluster Model: Modeling US Political Division

Jacob Oard

Department of Mathematical Sciences
Montana State University

May 12, 2023

A writing project submitted in partial fulfillment
of the requirements for the degree

Master of Science in Statistics

APPROVAL

of a writing project submitted by

JACOB OARD

This writing project has been read by the writing project advisor and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the Statistics Faculty.

Date	Ian Laga Writing Project Advisor
Date	John Smith Writing Project Advisor
Date	Katie Banner Writing Projects Coordinator

Abstract

Latent space cluster models have been developed to analyze network data, and have been used for many applications. While a dynamic latent space cluster model has been developed, we propose a more flexible version with a time series on cluster means. This allows for updated latent positions at each time step. After testing this model with simulated data, we apply it to model political division in the United States. We can represent senators as actors in a network and weight their ties by the number of times any pair votes the same on a bill. After accounting for senator party, we then model clusters within the Senate and how they evolve by Congress.

Contents

1 Introduction

Networks are common data structures for relational data. Fundamentally, networks describe a collection of objects (actors) and the connections between actors (ties). An illustrative example is a network of social media users and friendship statuses wherein the users are actors and two users are tied if they share a friendship status. Another example is Wikipedia articles and hyperlinks to other Wikipedia articles. Here, the articles are actors and article i has a tie to article j if there is a hyperlink from article i to article j. We can denote $y_{i,j}$ to be the event that there is a tie between actors i and j. That is,

$$y_{i,j} = \begin{cases} 1, & \text{Actors } i \text{ and } j \text{ have a tie} \\ 0, & \text{Actors } i \text{ and } j \text{ do not have a tie} \end{cases}$$

Specifically, $y_{i,j}$ above comes from an undirected network in which if actor i has a tie with actor j, then conversely, actor j also has a tie with actor i. A directed network is another form of network in which $y_{i,j}$ is the event that actor i has a tie with actor j but not the converse; that is, in a directed network $y_{i,j} = 1$ does not imply that $y_{j,i} = 1$. The Wikipedia article example is an example of a directed network since one article could have a hyperlink to a second article while that second article can lack a hyperlink to the first. In this paper, we will only be considering undirected networks, but each of the mentioned models work for directed networks.

The adjacency matrix $\boldsymbol{Y}$ contains the information about ties between actors, and position (i, j) has the value of $y_{i,j}$. Given the variety of applications of network data, there is interest in creating a statistical model for $\boldsymbol{Y}$. Towards that end, Hoff et al. (2002) proposed a latent space network model. They introduced the notion of a latent space which is an assumed space underlying a network in which the closer any two actors are, the more probable that there is a tie between these two actors.

This underlying space is often euclidean with arbitrary dimension. The latent space network model is given by:

$$y_{i,j}|\boldsymbol{Z}_i, \boldsymbol{Z}_j, x_{i,j}, \alpha, \beta \sim Bernoulli(p_{i,j})$$
$$\text{log odds}(p_{i,j}) = \alpha + \boldsymbol{\beta}'\boldsymbol{x}_{i,j} - |\boldsymbol{Z}_i - \boldsymbol{Z}_j|,$$

where $\boldsymbol{Z}_i$ and $\boldsymbol{Z}_j$ are the latent positions for actors i and j, $x_{i,j}$ is some pairwise covariate between actors i and j, and $p_{i,j}$ is the probability of there being a tie between actors i and j (Hoff et al., 2002). Here, $y_{i,j}$ follows a Bernoulli distribution in which the log-odds of $p_{i,j}$ depends on a linear model in which α is the intercept term and $\boldsymbol{\beta}$ is the coefficient for $\boldsymbol{x}_{i,j}$. In addition, the distance between the latent positions of actors i and j, $|\boldsymbol{Z}_i - \boldsymbol{Z}_j|$, is subtracted to incorporate the inverse relationship between the probability of a tie and the distance between actors within the latent space. It is important to note that this model can be analyzed using a Bayesian framework. We can put Gaussian priors on α, β, and the latent positions, and estimate Y using posterior means obtained from a Monte Carlo Markov Chain algorithm.

Sometimes, actors can have weighted ties between each other. This can be useful for modeling networks in which a pair of actors can have several ties between each other. This only requires a slight adjustment to the latent space network model defined above:

$$y_{i,j}|\boldsymbol{Z}_i, \boldsymbol{Z}_j, x_{i,j}, \alpha, \beta \sim Binomial(n_{i,j}, p_{i,j})$$
$$\text{log odds}(p_{i,j}) = \alpha + \beta' x_{i,j} - |\boldsymbol{Z}_i - \boldsymbol{Z}_j|,$$

where $n_{i,j}$ is the binomial size parameter and is the maximum number of ties between actors i and j, and $p_{i,j}$ is the probability of a single tie between actors i and j. The size parameter is typically known or assumed. The log-odds of the probability parameter is modeled the exact same way.

Handcock et al. (2007) provide a further refinement by incorporating the notion of clustering into the latent space network model and proposed the latent space cluster model. This model allows for the possibility that within the latent space, there are groups of latent positions clustered together. That is, clusters can capture trends of groups of actors having on average a higher probability of having ties among actors within their cluster than without. The latent space cluster model is given by:

$$\begin{aligned}
\text{log odds}(y_{ij} = 1 | \boldsymbol{Z}_i, \boldsymbol{Z}_j, \boldsymbol{X}, \boldsymbol{\beta}) &= \sum_{k=1}^{p} x_{k,i,j}\beta_k - |\boldsymbol{Z}_i - \boldsymbol{Z}_j| \\
\beta_k &\overset{iid}{\sim} N(\xi_k, \phi_k^2), \quad k = 1, \ldots, p \\
\boldsymbol{Z}_i &\overset{ind}{\sim} \sum_{g=1}^{G} \lambda_g MVN_d(\boldsymbol{\mu}_g, \sigma_g^2 I_d), \quad i = 1, \ldots, n \\
\boldsymbol{\mu}_g &\overset{iid}{\sim} MVN_d(0, \omega^2 I_d), \quad g = 1, \ldots, G \\
\sigma_g^2 &\overset{iid}{\sim} \sigma_0^2 Inv - \chi_\alpha^2, \quad g = 1, \ldots, G \\
(\lambda_1, \ldots, \lambda_G) &\sim Dirichlet(\nu_1, \ldots, \nu_G)
\end{aligned}$$

This model assumes there are G clusters in the latent space of dimension d, so $\boldsymbol{Z}_i \overset{ind}{\sim} \sum_{g=1}^{G} \lambda_g MVN_d(\boldsymbol{\mu}_g, \sigma_g^2 I_d)$ where λ_g is the mixing probability of cluster g, $\boldsymbol{\mu}_g$ is the center of cluster g in the latent space, and σ_g^2 is the variance parameter of cluster g. During model fitting G is fixed, but Handcock et al. (2007) note that choosing the number of clusters can be treated as a model selection problem. This model can easily be extended to include weighted ties in the exact same way as described earlier where $y_{i,j} \sim Binomial(n_{i,j}, p_{i,j})$.

Other researchers have developed latent space models that change over time called dynamic latent space network models. That is as time progresses, the ties within a network might also change. Sarkar and Moore (2005) proposed the first dynamic latent space model and others such as Sewell and Chen (2015) have offered other renditions.

2 Model

We propose a new dynamic model that incorporates clustering in which the probability of there being a tie between actors i and j depends on k pairwise covariates at time t and the distance between these actors' latent positions at time t. This model assumes G clusters within the latent space of dimension d. The following is our proposed dynamic latent space cluster model:

$$\begin{aligned}
\text{log odds}(y_{ijt} = 1 | \boldsymbol{Z}_{it}, \boldsymbol{Z}_{jt}, \boldsymbol{X}, \boldsymbol{\beta}) &= \sum_{k=1}^{p} x_{k,i,j,t}\beta_{k,t} - |\boldsymbol{Z}_{it} - \boldsymbol{Z}_{jt}| \\
\beta_{k,t} &\overset{iid}{\sim} N(\xi_k, \phi_k^2), \quad k = 1, \ldots, p \\
\boldsymbol{Z}_{it} &\overset{iid}{\sim} \sum_{g=1}^{G} \lambda_{gt} MVN_d(\boldsymbol{\mu}_{gt}, \sigma_{gt}^2 I_d), \quad i = 1, \ldots, n \\
\sigma_{gt}^2 &\overset{iid}{\sim} \sigma_0^2 Inv - \chi_\alpha^2, \quad g = 1, \ldots, G, \quad t = 1, \ldots, T \\
(\lambda_{1t}, \ldots, \lambda_{Gt}) &\overset{iid}{\sim} Dirichlet(\nu_1, \ldots, \nu_G), \quad t = 1, \ldots, T \\
\boldsymbol{\mu}_{\cdot,t} &= \phi \boldsymbol{\mu}_{\cdot,t-1} + \boldsymbol{\varepsilon}_{\cdot,t} \qquad \boldsymbol{\varepsilon}_{\cdot,t} \overset{iid}{\sim} MVN_d(0, \omega^2 I_d) \\
\boldsymbol{\mu}_{g,1} &\overset{iid}{\sim} MVN_d(0, \omega^2 I_d) \\
\phi &\sim \text{Unif}(-1, 1)
\end{aligned}$$

Here the prior on the latent positions depends on λ_{gt}, the proportion of latent positions belonging to cluster g at time t; $\mu_{g,t}$, the cluster center for cluster g at time t; σ_{gt}^2, the variance parameter for cluster g at time t; and ϕ a first order auto-regressive coefficient. This model allows for clusters to have different sizes and spreads. The cluster centers at time t depend on its center at the previous time, $t-1$. We assume that the variance of the cluster means is stable over time. It is important to include ϕ in the model because without it, the expected squared distances between cluster means from one time point to the next increases; that is, the cluster means follow a random walk across time, grow farther and farther apart, and hence lack stationarity

without ϕ (i.e. $\phi = 1$). Lemma 2.1 and the following proof demonstrate this fact.

Lemma 2.1. *Given the dynamic latent space cluster model, the squared distance between cluster means achieves stationarity if the value of the first order auto-regressive coefficient is between zero and 1:* $0 \leq |\phi| < 1$. *We call* $d_0 = \mu_{1,t} - \mu_{2,t}$, *the distance between cluster means at time point* t.

Proof. For one dimension,

$$\begin{aligned}
\mathbb{E}\left[(\mu_{1,t+1} - \mu_{2,t+1})^2\right] &= \mathbb{E}\left[(\phi\mu_{1,t} + \varepsilon_{1,t}) - (\phi\mu_{2,t} + \varepsilon_{2,t})^2\right] \\
&= \mathbb{E}\left[(\phi\mu_{1,t} - \phi\mu_{2,t})^2 + (\varepsilon_{1,t} - \varepsilon_{2,t})^2\right] \\
&= \mathbb{E}\left[\phi^2(\mu_{1,t} - \mu_t)^2\right] + 2\mathbb{E}\left[\phi(\mu_{1,t} - \mu_{2,t})(\varepsilon_{1,t} - \varepsilon_{2,t})\right] + \mathbb{E}\left[(\varepsilon_{1,t} - \varepsilon_{2,t})^2\right] \\
&= \phi^2 d_0^2 + 2\phi\mathbb{E}\left[\mu_{1,t} - \mu_{2,t}\right]\mathbb{E}\left[\varepsilon_{1,t} - \varepsilon_{2,t}\right] + \mathbb{E}\left[\varepsilon_{1,t}^2 - 2\varepsilon_{1,t}\varepsilon_{2,t} + \varepsilon_{2,t}^2\right] \\
&= \phi^2 d_0^2 + 0 + 2\mathbb{E}\left[\varepsilon_{1,t}\right]\mathbb{E}\left[\varepsilon_{2,t}\right] + \mathbb{E}\left[(\varepsilon_{1,t} - 0)^2\right] + \mathbb{E}\left[(\varepsilon_{2,t} - 0)^2\right] \\
&= \phi^2 d_0^2 + 0 + \mathbb{V}(\varepsilon_{1,t}) + \mathbb{V}(\varepsilon_{2,t}) \\
&= \phi^2 d_0^2 + 2\omega^2
\end{aligned}$$

Thus, the expected squared distance depends on ϕ^2. For $|\phi| = 1$, the expected squared distance will always increase over time: $\mathbb{E}\left[(\mu_{1,t+1} - \mu_{2,t+1})^2\right] = d_0^2 + 2\omega^2$. With $0 \leq |\phi| < 1$, the expected squared distance at time t depends only on a fraction of the previous time's expected squared distance in addition to 2 times the variance of ε. This result generalizes to dimension $d > 1$, but we do not show those results here. □

To illustrate the effect of values of ϕ, we simulated the distance between 1000 pairs of cluster means over 100 time steps. In Figure 1 top left, the average distance increases over time with a value for ϕ of barely greater than 1. In the top right, $\phi = 1$ as was in the initial model, the distance still lacks stationarity as previously

described. For the values of ϕ that are less than 1 in the bottom left and right, there is no trend in the distances as time progresses, and stationarity is enforced.

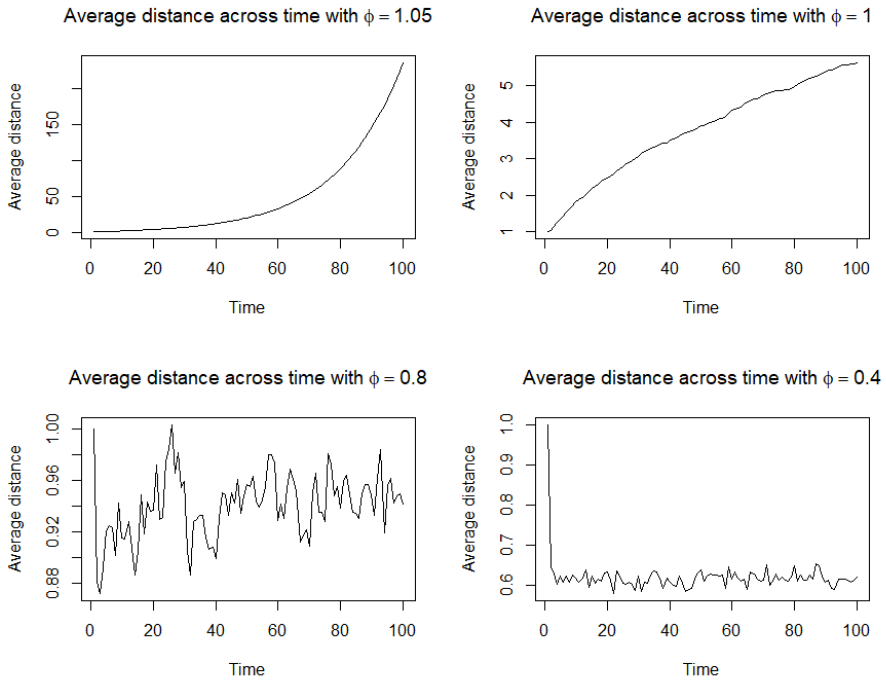

Figure 1: Average distance between cluster means across time for different values of ϕ.

3 Simulation Study

In order to test and study our model, we developed a simulation study in `R` (R Core Team, 2024) using `nimble` (de Valpine et al., 2017, 2024). The goal of our simulation study was to confirm that our model can recover the data generating values, and to examine the impact of our hierarchical time series structure on inference compared to the latent space cluster model of Handcock et al. (2007).

For the simulation, we first chose a sample size of $n = 100$, a latent space dimension of $d = 2$, number of clusters $G = 2$, and 9 time steps. First, we randomly generated values for the two cluster means and randomly chose values for the beta coefficients from $Unif(1, 5)$, $\omega = \sqrt{2}$, and $\phi = 0.5$. Second, we

randomly generated symmetric covariate matrices $X1$ and $X2$ whose elements are 1 or 0 for each time step. Finally, we randomly assigned cluster ids to the actors and simulated latent positions for all 100 actors from a multivariate normal with mean depending on the cluster mean according to cluster id for the first time point and covariance matrices as $0.05I_2$ regardless of cluster id. For the subsequent 8 time points, the cluster mean was that of the previous time step weighted by ϕ with some additional noise from a multivariate normal distribution with mean zero and variance ω. Then, we generated the response matrix (adjacency matrix) Y from $Y_{i,j,t} \sim binomial(N_{ij}, logit^{-1}(X1_{i,j,t}\beta_1 + X2_{i,j,t}\beta_2 - |Z_{i,t} - Z_{j,t}|))$.

Table 1 gives the true, simulated values and estimates calculated from the posterior mean for each parameter as well as the bounds generated from a 95% credible interval. On average, the model estimates the parameters accurately for β_1, β_2, ϕ, σ_1^2, σ_2^2, λ_1, and λ_2. There are two exceptions however; the posterior means for σ_1^2 and σ_2^2 at time point 1 are notably distant from the true, simulated value. Additionally, the posterior mean for ω, the variance parameter for cluster means is over estimated because we only simulated 2 clusters. As the number of clusters increases, so will the accuracy for the ω posterior mean keeping all other parameters constant. The traceplots for each of these parameters as well as the distances between cluster means and latent positions converge to the true simulated value (with the exception of ω and can be found in the appendix.

Latent positions and cluster mean location estimates are unidentifiable parameters, but the distances between these positions and means are. That is, the actual coordinate location for each of these parameters is inconsequential to the model since likelihood depends on the distance between the latent positions. Table 2 shows how well the simulation captured these distances. For this one simulation, the true latent position distances are captured approximately at or above 95% for each time step.

Because the coordinate locations of latent positions and cluster means are non-identifiable, visualizing the posterior means for these positions is challenging as the latent positions will almost certainly be oriented differently from their cluster means. To ameliorate this problem for the sake of visualization, we rotated the latent positions and cluster means using the same rotation matrix in a procrustes transformation (Hoff et al., 2002). Figure 3 shows the true, simulated latent positions and cluster means. Figure 3 shows the procrustes transformed posterior mean latent positions and cluster means. The locations on the axes are different between 2 and 3 as well as the general orientation of the points in particular for time points 6 and 8, but the distances between latent positions and cluster means are preserved.

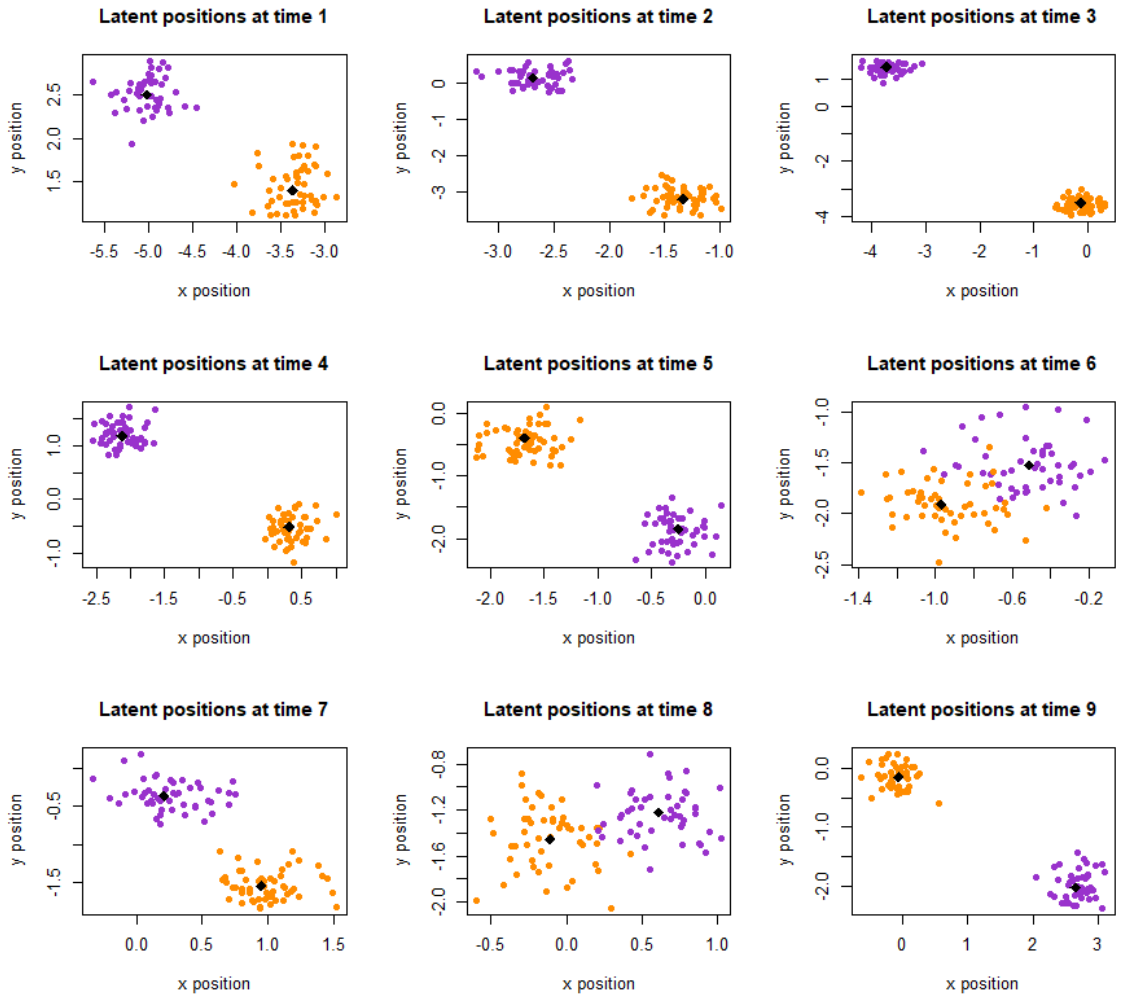

Figure 2: True, simulated latent positions colored by cluster id with cluster means as the black diamonds.

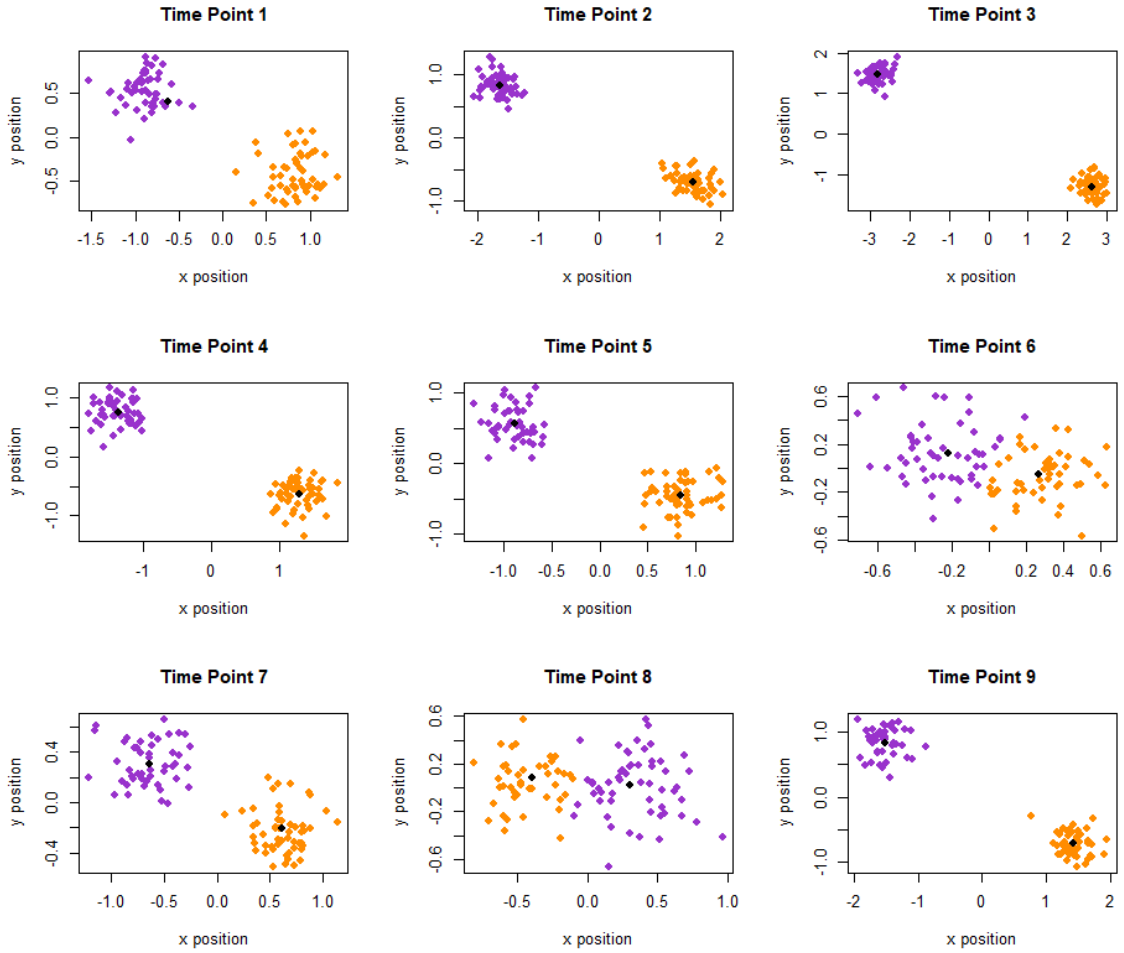

Figure 3: Procrustes transformed cluster means as the black diamonds and posterior mean latent positions colored by posterior mean cluster id.

β_1	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	4.348	4.331	4.342	4.355
	2	2.283	2.244	2.272	2.285
	3	3.721	3.621	3.712	3.730
	4	3.793	3.713	3.782	3.799
	5	2.828	2.823	2.832	2.842
	6	3.806	3.789	3.799	3.815
	7	2.663	2.645	2.653	2.663
	8	2.213	2.203	2.209	2.217
	9	4.506	4.378	4.493	4.519
β_2	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	1.476	1.468	1.475	1.484
	2	4.608	4.565	4.587	4.607
	3	4.812	4.663	4.791	4.818
	4	3.814	3.730	3.810	3.827
	5	3.063	3.055	3.064	3.075
	6	2.798	2.787	2.796	2.806
	7	4.369	4.349	4.360	4.379
	8	1.441	1.434	1.441	1.447
	9	4.453	4.321	4.442	4.467
ω	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	NA	1.414	1.737	2.997	4.87
ϕ	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	NA	0.5	0.153	0.489	0.818
σ_1^2	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	0.05	0.034	0.817	9.145
	2	0.05	0.037	0.051	0.067
	3	0.05	0.039	0.054	0.072
	4	0.05	0.040	0.055	0.072
	5	0.05	0.039	0.053	0.071
	6	0.05	0.023	0.064	0.093
	7	0.05	0.035	0.083	0.071
	8	0.05	0.028	0.477	5.082
	9	0.05	0.031	0.044	0.056

σ_2^2	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	0.05	0.033	0.100	0.589
	2	0.05	0.034	0.048	0.063
	3	0.05	0.034	0.048	0.064
	4	0.05	0.038	0.052	0.070
	5	0.05	0.035	0.048	0.064
	6	0.05	0.025	0.066	0.096
	7	0.05	0.034	0.051	0.071
	8	0.05	0.032	0.068	0.143
	9	0.05	0.040	0.056	0.076
λ_1	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	0.5	0.003	0.459	0.606
	2	0.5	0.423	0.520	0.617
	3	0.5	0.422	0.519	0.615
	4	0.5	0.422	0.519	0.615
	5	0.5	0.423	0.519	0.616
	6	0.5	0.241	0.499	0.759
	7	0.5	0.379	0.489	0.597
	8	0.5	0.003	0.423	0.664
	9	0.5	0.422	0.520	0.615
λ_2	**Time Step**	**Truth**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	0.5	0.394	0.541	0.997
	2	0.5	0.383	0.480	0.577
	3	0.5	0.385	0.481	0.578
	4	0.5	0.385	0.481	0.578
	5	0.5	0.384	0.481	0.577
	6	0.5	0.241	0.501	0.759
	7	0.5	0.403	0.511	0.621
	8	0.5	0.336	0.577	0.997
	9	0.5	0.385	0.577	0.578

Table 1: The true value, lower bound from a 95% credible interval, posterior mean (estimate), and upper bound from a 95% credible interval for each parameter.

Time Step	Latent Positions	Cluster Means
1	0.982	1.000
1	0.949	1.000
1	0.958	1.000
1	0.984	1.000
1	0.981	1.000
1	0.979	1.000
1	0.960	1.000
1	0.962	1.000
1	0.963	1.000

Table 2: Percentage of credible intervals of the distances between simulated latent positions and cluster means containing the true latent position and cluster mean distances from the single simulation. In the future, true coverage should be calculated for many simulations.

4 Data

Our motivating application is modeling political division within the United States Senate. Every two years, a subset of senators are replaced with new senators after the conclusion of their six year term. These two year periods are called Congresses, during which senators vote on hundreds of bills. If we treat each Congress as a time step, senators as actors, and ties between senators as when two senators both vote "yea" (affirmative) or "nay" (negative) on a bill as a tie, we can model the probability of ties between senators using the dynamic latent space cluster model developed here.

It is important to note that for each of the Congresses that we have inspected, each senator has voted the same as each of the other senators as least once, so we weight the ties by the number of times these pairs of senators voted in agreement. Hence, the $(i, j, t)^{th}$ cell from the three dimensional adjacency matrix, Y in this context must follow a binomial likelihood whose count parameter is the $(i, j, t)^{th}$ cell from the three dimensional array we call N. That is, position (i, j, t) from N is the number of times senators i and j participated in voting on the same bill during

Congress t, and position (i, j, t) from Y is the number of times senators i and j both voted either "yea" or "nay" during Congress t.

We acquired senator voting data from Senate.gov (United States Senate, 2023) for the 101st Congress (January 1989 - January 1991) through the 117th Congress (January 2021 - January 2023). This data contains the name and party of each senator for each Congress as well as their vote for each bill. For each Congress, we calculated the number of times each pair of senators voted the same on a bill and number of times they voted on the same bill to populate Y and N respectively. We also created two covariates, $X1$ and $X2$, in which cell (i, j, t) denotes senators i and j at Congress t are both Democrats for $X1$ and Republicans for $X2$.

We were only interested in investigating voting trends among Democrats and Republicans so we removed independent senators. Another complication is occasionally, during a given Congress, some senators pass away, resign, or are appointed to other government rolls, and in these cases, their tenures ends early. Special elections are held, and newly elected senators serve the rest of the term. This means that the dimension of Y, N, etc. vary from Congress to Congress and that must be incorporated in the analysis of the model.

5 Application

We input this data into our model, ran it in `R` (R Core Team, 2024) using `nimble` (de Valpine et al., 2017, 2024) with 10000 iterations with a initial burnin of 2000 iterations. Table 3 shows the posterior means for β_1 and β_2 as well as the 95% credible intervals. A notable result is that for the 17th time step which corresponds to the 117th congress (January 2021 - January 2023), the credible interval for β_1 associated with the covariate that both senators are either Democrat or not is noticeably greater than that for β_2 associated with the covariate that both senators are either Republican

or not. This seems to indicate, that for this Congress, there is greater impact towards whether or not two senators vote the same on a bill during the 117th Congress if both senators are Democrat than Republican. This can be seen by Figure 4 bottom right. The squares (Republican latent position) have a greater spread than the circular points (Democrats). Since the Democrats appear, on average, to have shorter distances between them, it is more likely that they will vote the same than Republicans who have greater distances.

As aforementioned, Figure 4 shows the procrustes transformed posterior mean latent positions and cluster means for a subset of Congresses: 102nd, 106th, 115th, and 117th Congresses. The most noticeable aspect of these plots is that there is a clear line on which the majority of points fall. This seems to indicated that for this application, that perhaps a latent space of dimension 1 will be a better fit. In the 106th Congress (January 1999 - January 2001), we noticed a senator whose latent position in noticeably distant from the rest in the bottom left corner. This latent position is that of senator John McCain (R-AZ). In the 115th Congress (January 2017 - January 2019), the very distant latent position on the far left also belongs to John McCain (R-AZ). This indicates that John McCain is relatively less likely to vote similarly with other senators. The other noticeable distant latent position in the top in the 115th Congress belongs to Jeff Sessions (R-AL). In the most recent Congress (January 2021 - January 2023), the top most latent position belongs to Diane Feinstein (D-CA), the right-most latent position belongs to Rand Paul (R-KY), and the left most latent position belongs to Mike Rounds (R-SD).

We expected to see that for at least a subset of Congresses, that cluster assignment and party affiliation would overlap closely. However, for these results, this is not the case. In some Congresses, such as the 102nd and 115th Congresses, there only appears to be a single cluster with only a very small subset of senators comprising the second cluster. There is some amount of error created in estimating cluster

assignment however. These assignments are made on a continuous scale between 1 and 2, and then are rounded to be either 1 or 2. There are some senators who fall "in between" clusters but are still assigned to a cluster.

A problem with this plot is that despite procrustes transforming, the cluster means of the 102nd and 106th Congresses are clearly oriented differently than the surrounding latent positions. This is likely an issue with the procrustes transformation code despite testing to ensure its functionality. This requires further attention in the future.

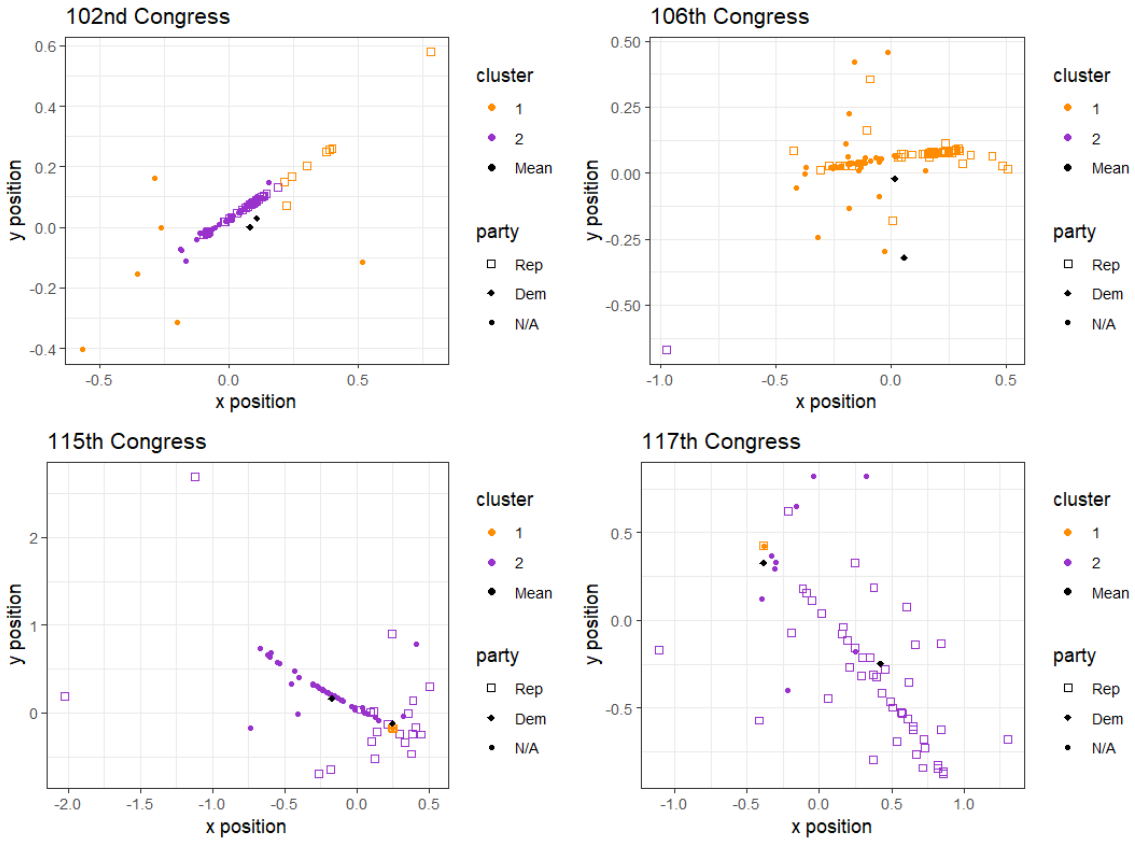

Figure 4: Selected procrustes transformed posterior mean cluster means in black and posterior mean latent position colored by poster mean cluster id. In addition, Democrats are represented by the circular points and Republicans by the squares.

β_1	**Time Step**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	1.093	1.098	1.103
	2	1.156	1.162	1.168
	3	1.458	1.464	1.470
	4	1.442	1.448	1.455
	5	1.501	1.508	1.515
	6	1.749	1.758	1.766
	7	1.665	1.672	1.679
	8	1.642	1.649	1.657
	9	1.650	1.658	1.666
	10	1.726	1.733	1.741
	11	2.131	2.139	2.146
	12	1.976	1.984	1.992
	13	2.587	2.596	2.606
	14	2.062	2.072	2.082
	15	2.016	2.025	2.033
	16	1.827	1.835	1.843
	17	3.350	3.366	3.383
β_2	**Time Step**	**Lower Bound**	**Estimate**	**Upper Bound**
	1	0.945	0.951	0.957
	2	1.077	1.084	1.091
	3	1.290	1.296	1.302
	4	1.778	1.784	1.790
	5	1.598	1.604	1.610
	6	1.742	1.748	1.755
	7	1.519	1.525	1.532
	8	2.087	2.095	2.103
	9	1.651	1.657	1.663
	10	1.327	1.335	1.343
	11	1.757	1.766	1.775
	12	1.375	1.382	1.390
	13	1.580	1.588	1.596
	14	1.697	1.706	1.715
	15	2.412	2.422	2.431
	16	2.150	2.158	2.166
	17	1.951	1.961	1.972

Table 3: Posterior mean (estimate) and 95% credible intervals for β_1 and β_2.

6 Conclusion

We developed a new dynamic latent space cluster model by extending the static latent position cluster model of Hoff et al. (2002). After simulating data from a dynamic network based on simulated latent positions informed by cluster, we were able to successfully recover nearly all parameter values with our model. With a greater number of clusters in the simulation, we expect that we will be able to recover ω, the cluster mean variance, as well. The development of our model was motivated by framing the US Senate as a dynamic network, and evaluating division within using clusters and seeing how they change over time. We acquired Senate data and were able to apply this model. At the current stage in development of the model, it does not appear that two clusters to the extent that they exist can be explained by party.

This project still has work and refinement that is required. In particular, this model has challenges being extended beyond 2 clusters. When we simulated data from 10 clusters, it became difficult to identify clusters due to label switching. In the future, we will need to find a solution for this. We also need to further inspect the results and resolve the problem from cluster means still having a different orientation than the latent positions after procrustes transformation. This should not be the case.

In the future, we hope to compare our new dynamic latent space cluster model with another proposed latent space cluster model applied to a dynamic network by (Jin, Z., Sosa, J., & Betancourt, B., 2002). A key difference with our proposed model is that at each time step, there are new latent positions.

We are also curious about applying this model to the US House of Representatives. If we are able to develop the model to extend beyond 2 clusters, this would be an interesting application. Since there are 435 representatives, there is greater potential for them to exhibit a greater number of clusters than 2.

References

de Valpine, P., Paciorek, C., Turek, D., Michaud, N., Anderson-Bergman, C., Obermeyer, F., Cortes, C. W., Rodriquez, A., Lang, D. T., and Paganin, S. (2024). *NIMBLE: MCMC, Particle Filtering, and Programmable Hierarchical Modeling*. R package version 1.1.0.

de Valpine, P., Turek, D., Paciorek, C., Anderson-Bergman, C., Lang, D. T., and Bodik, R. (2017). Programming with models: Writing statistical algorithms for general model structures with nimble. *Journal of Computational and Graphical Statistics*, 26:403–413.

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

Jin, Z., Sosa, J., & Betancourt, B. (2002). A robust Bayesian latent position approach for community detection in networks with continuous attributes. *arXic preprint arXiv:2301.00055*.

Martin, A. D., Quinn, K. M., and Park, J. H. (2011). MCMCpack: Markov chain monte carlo in R. *Journal of Statistical Software*, 42(9):22.

Plummer, M. (2023). *rjags: Bayesian Graphical Models using MCMC*. R package version 4-15.

Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). Coda: Convergence diagnosis and output analysis for mcmc. *R News*, 6(1):7–11.

R Core Team (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

Sarkar, P. and Moore, A. W. (2005). Dynamic social network analysis using latent space models. *Acm Sigkdd Explorations Newsletter*, 7(2):31–40.

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

United States Senate (2023). U.S. Senate: Bills, Acts, & Laws. `https://www.senate.gov/legislative/bills_acts_laws.htm` (accessed February 7, 2023).

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

7 Appendix

7.1 Simulation Study

Figures 5 and 6 show that the simulation study successfully captured the simulated values for β_1 and β_2.

Figure 7 shows that the simulation study over estimated the value for ω. This is because the simulation only included 2 clusters. The greater the number of clusters, the more accurate the estimate for ω will be.

Figure 8 shows that the simulation study successfully recovered the value for ϕ.

Figure 9 shows the traceplots for σ_1^2. The other traceplots associated with σ_2^2 are similar. They show that the simulation study successfully captured the true value for

σ_1^2 which was 0.05 for all time steps.

Figure 10 shows the traceplots for λ_1. The other traceplots associated with λ_2 are similar. These plots show that the simulation study successfully recovered the true value for λ_1 which was 0.5. Time point 8 for both clusters have chains that do not mix as thoroughly as the other time points.

Figure 11 shows that the simulation study successfully recovered the true distances between the two cluster means. This is especially important as cluster distances are identifiable and are vital for understanding the evolution of clusters over time as is motivation of the model.

Figure 12 shows that the model converged for the distance between the first and second latent positions. We inspected the traceplots between several other pairs of latent positions, and they similarly converge.

Figure 13 shows that for all distances between latent positions that the simulation study recovers the true distance.

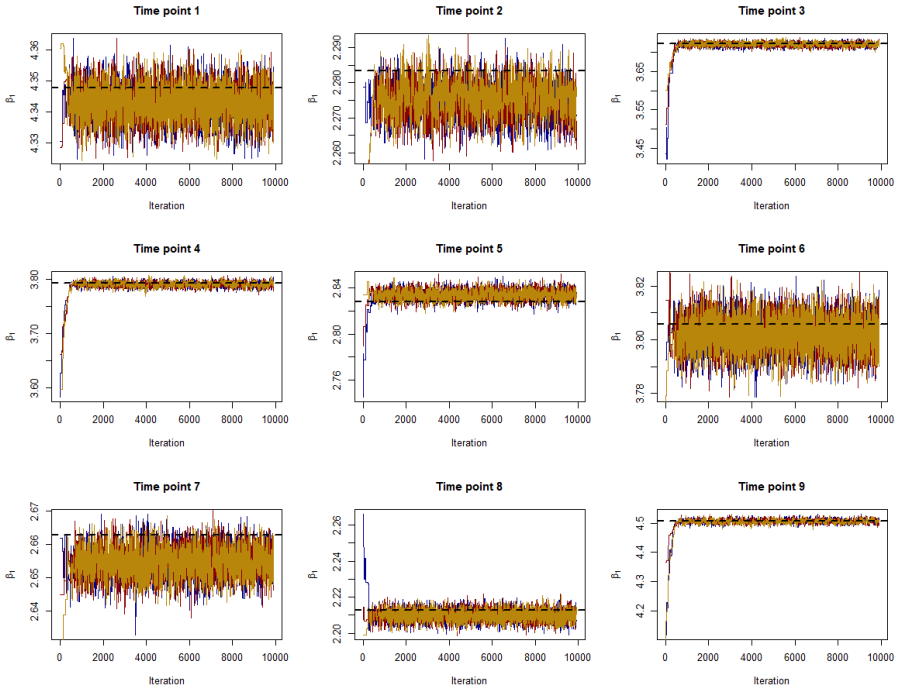

Figure 5: Traceplots for β_1 with the dashed black line at the true, simulated value.

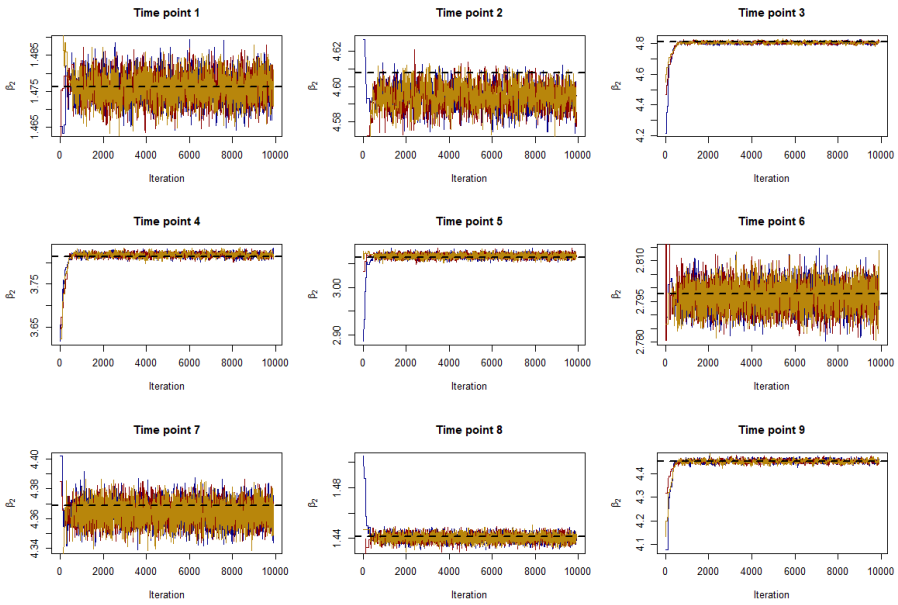

Figure 6: Traceplots for β_2 with the dashed black line at the true, simulated value.

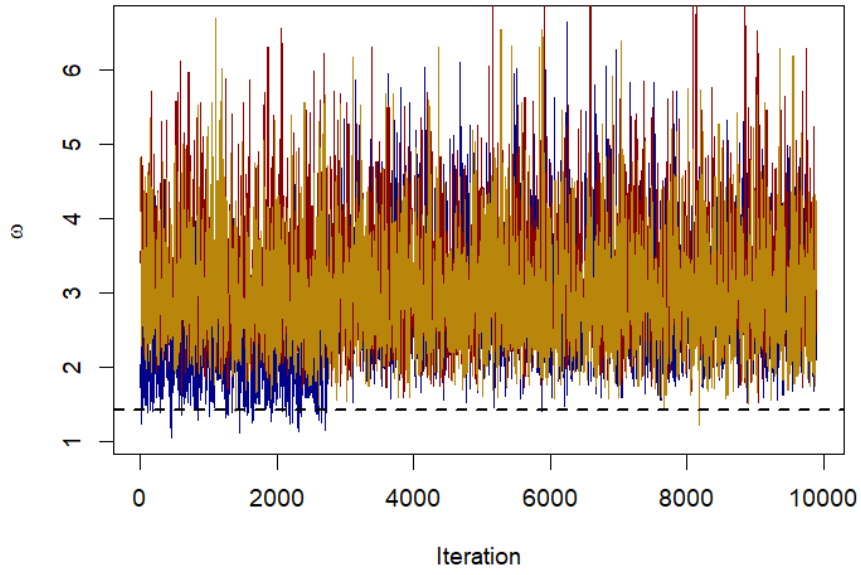

Figure 7: Traceplot for ω with the dashed black line at the true, simulated value.

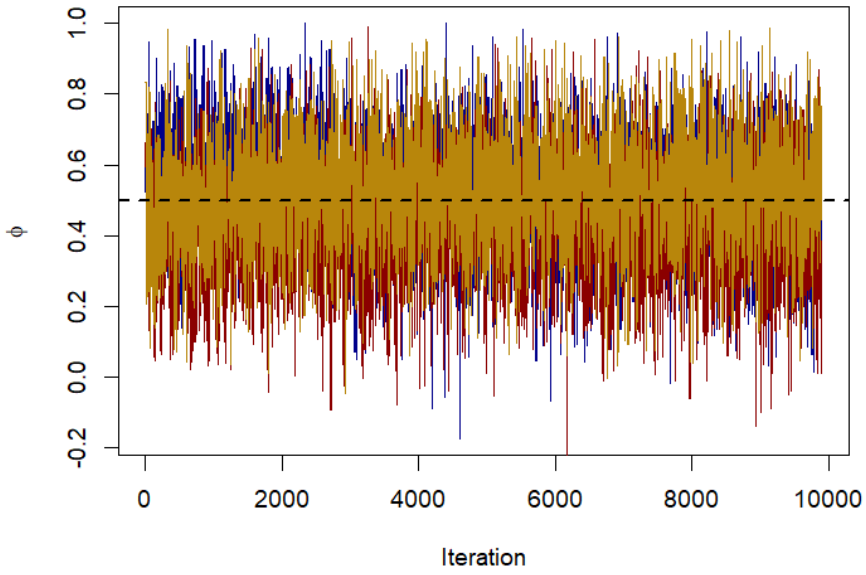

Figure 8: Traceplot for ϕ_1 with the dashed black line at the true, simulated value.

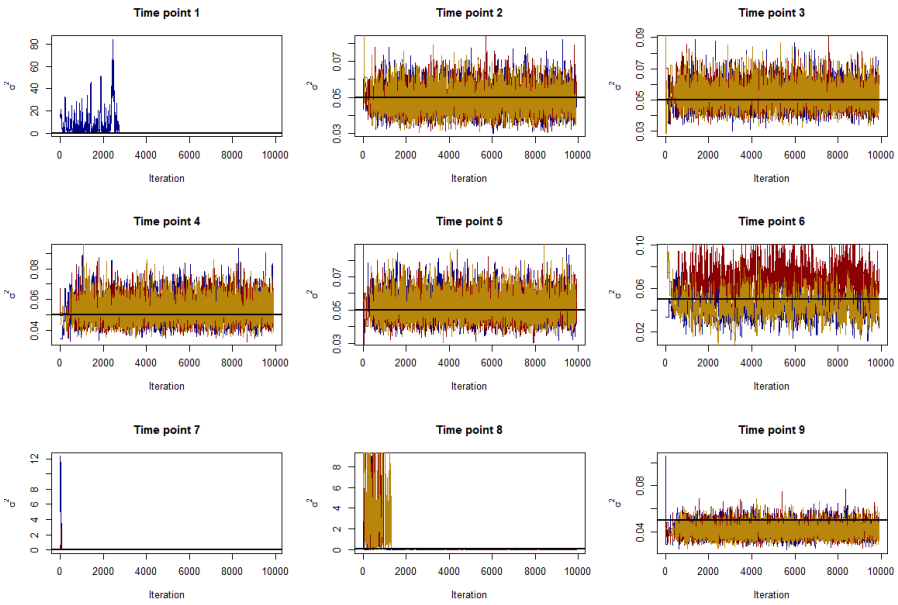

Figure 9: Traceplots for σ_1^2 (variance parameter for cluster 1) with the dashed black line at the true, simulated value.

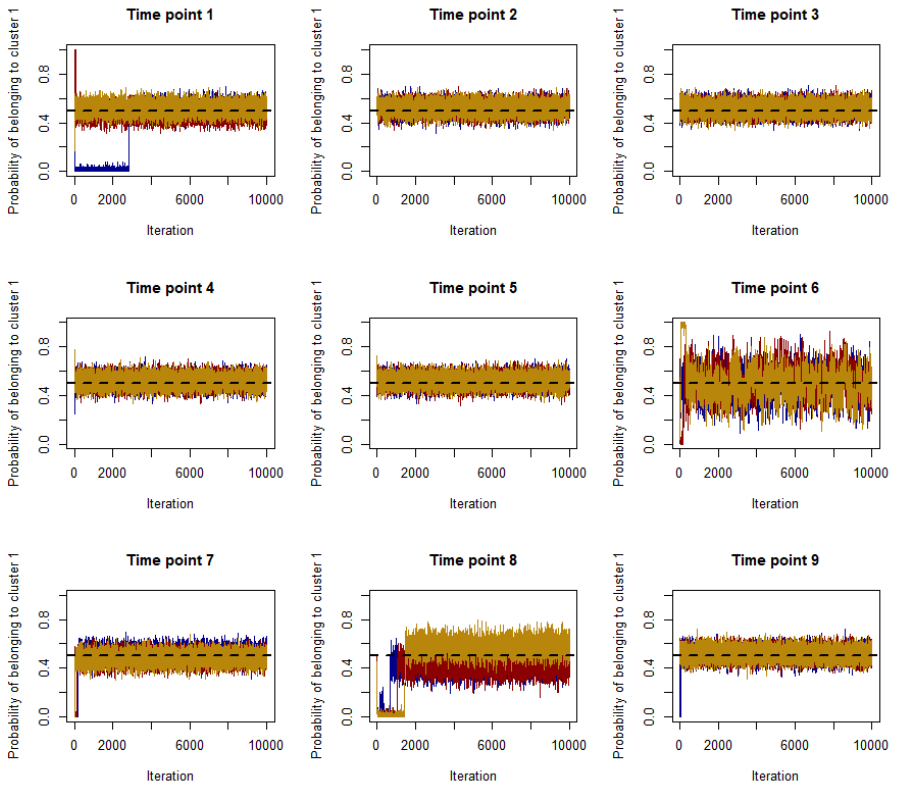

Figure 10: Traceplots for λ_1 (probability of belonging to cluster 1) with the dashed black line at the true, simulated value.

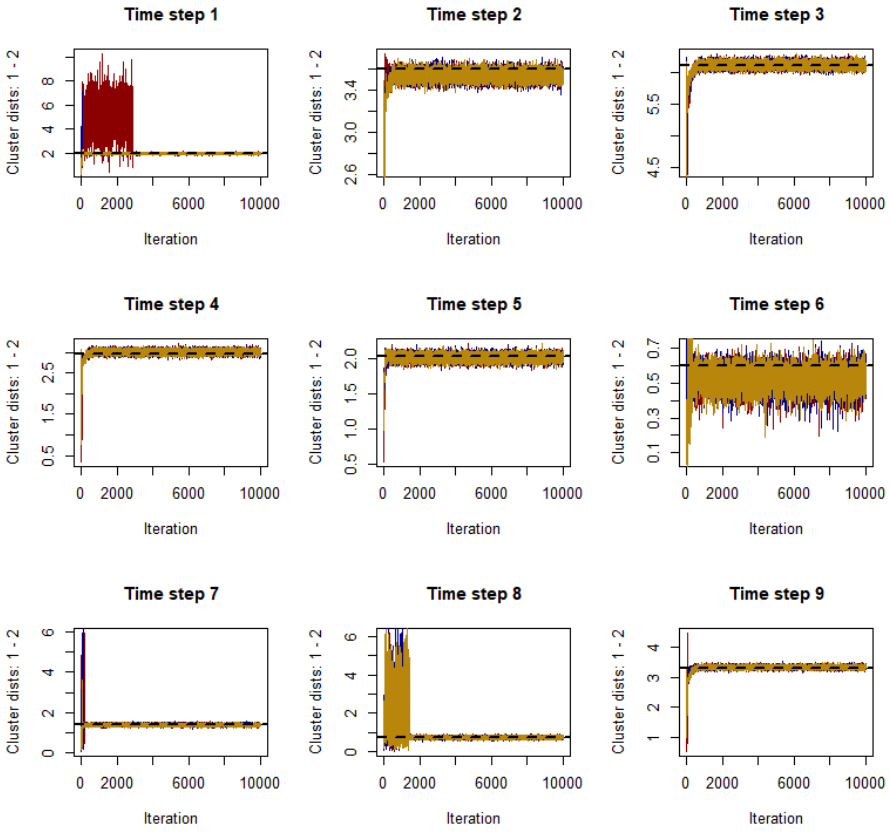

Figure 11: Traceplots for the distances between cluster means.

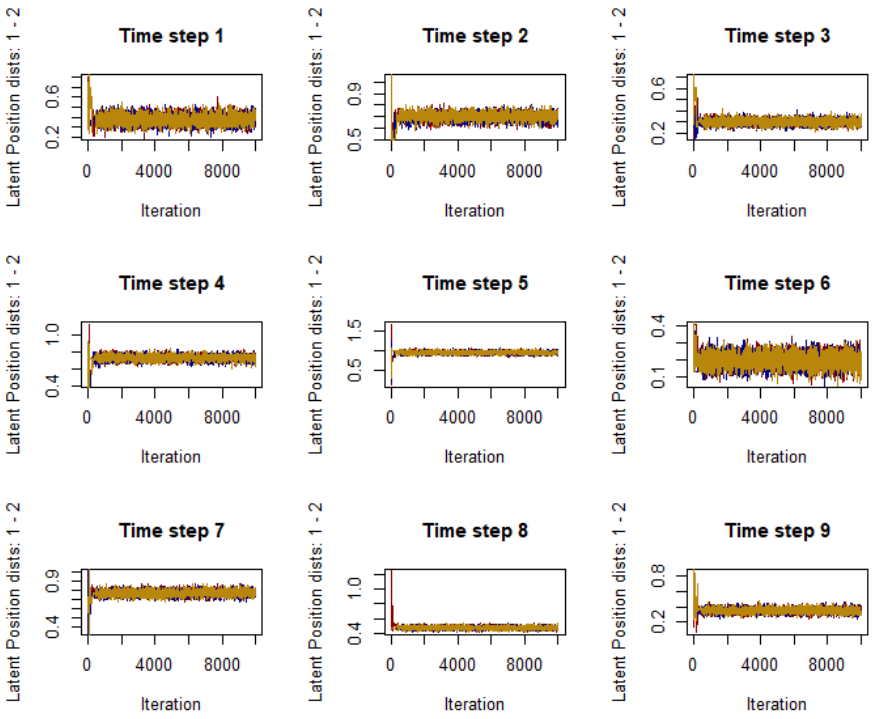

Figure 12: Traceplots for the distances between the first and second latent positions.

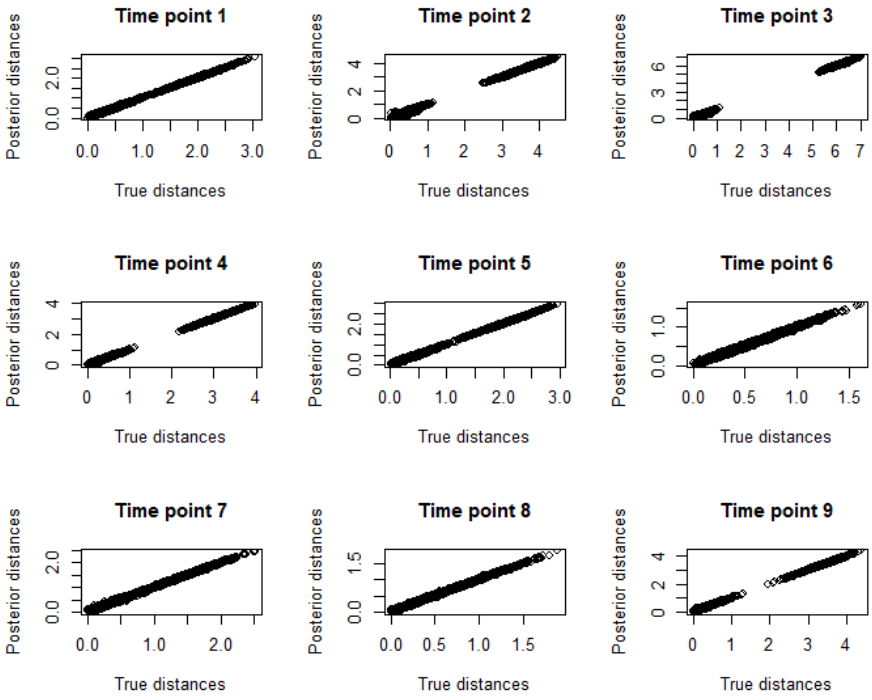

Figure 13: Plot of true distances between all latent positions against that of posterior latent positions distances.

7.2 Model Application

Figures 14 and 15 show that the model converged at every step for β_1 and β_2 with sufficient mixing for the 101st Congress through the 116th Congress. The traceplots for the 117th Congress are similar and were excluded for the sake of formatting.

Figure 16 shows that the model converged. This is likely an overestimate as in the simulation since we are only modeling two clusters.

Figure 17 shows that the model converged for ϕ.

Figure 18 shows the traceplots for σ_1^2. For many of the Congresses, the chains failed to mix indicating a failure in convergence. The variance parameter, σ_2^2 is similar for all Congresses. The 117th Congress was excluded for the sake of formatting. However, this is relatively less concerning since $\sigma_\cdot^2$ are not primary parameters.

Figure 19 shows the traceplots for λ_1. For many of the Congresses, the chains failed to mix indicating a failure in convergence. λ_2 is similar for all Congresses. The 117th Congress was excluded for the sake of formatting. However, this is relatively less concerning since $\lambda_\cdot$ are not primary parameters.

Figure 20 shows that the model converged for all Congresses for the distance between clusters. The 117th Congress was similar and was excluded for the sake of formatting.

Figure 21 shows that the model converged for the distance between the first and second latent positions. We inspected the traceplots between several other pairs of latent positions, and they similarly converge.

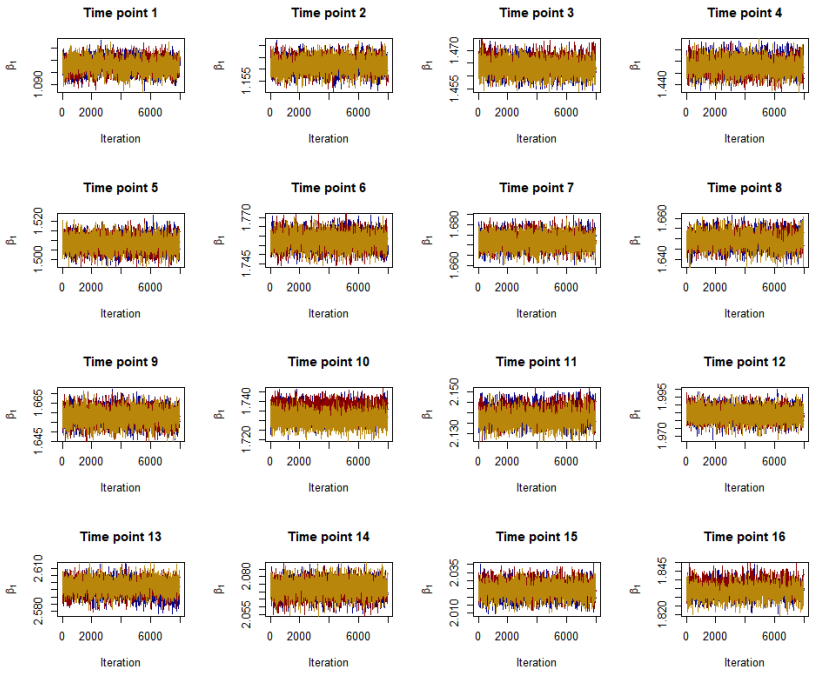

Figure 14: Traceplots for β_1 for the 101st Congress through the 116th Congress.

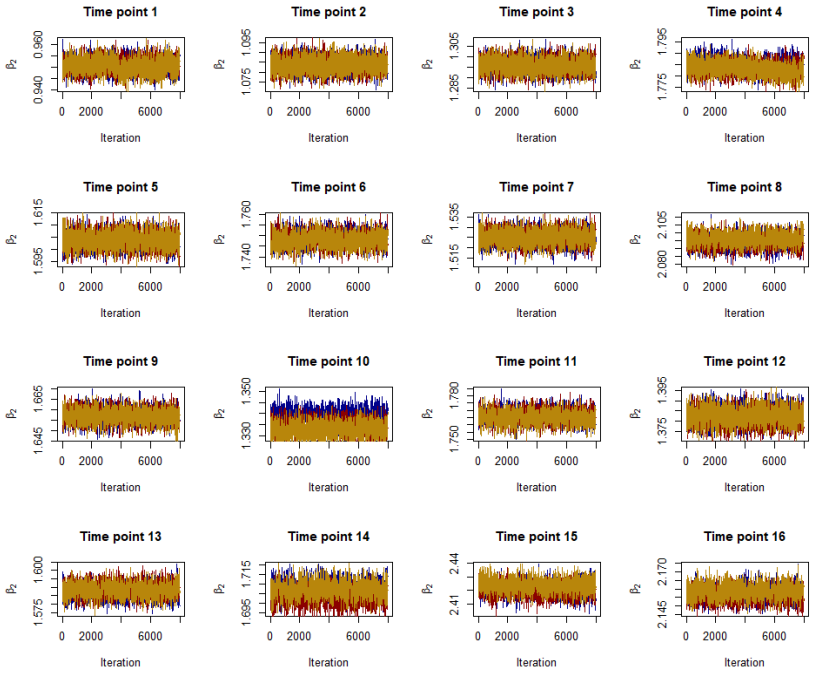

Figure 15: Traceplots for β_2 for the 101st Congress through the 116th Congress.

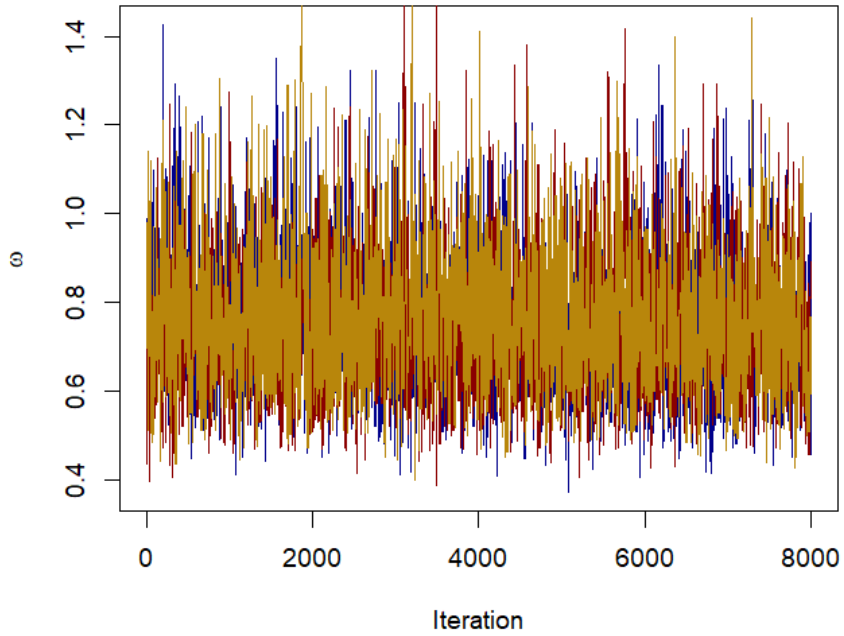

Figure 16: Traceplot for ω.

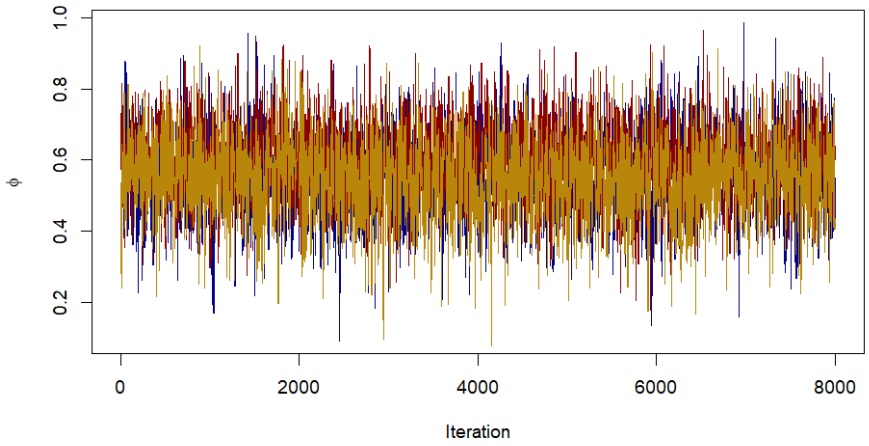

Figure 17: Traceplot for ϕ.

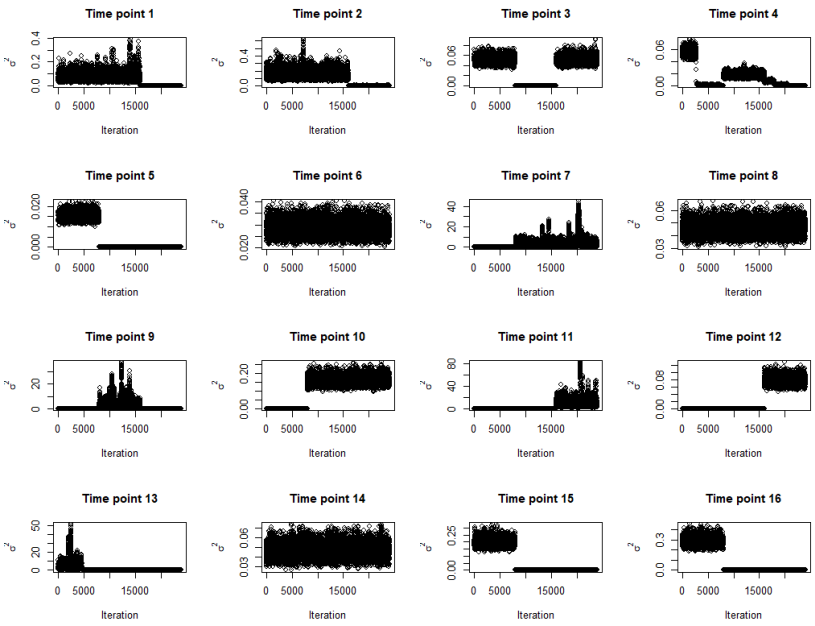

Figure 18: Traceplots for σ_1^2 (variance parameter for cluster 1) for the 101st Congress through the 116th Congress.

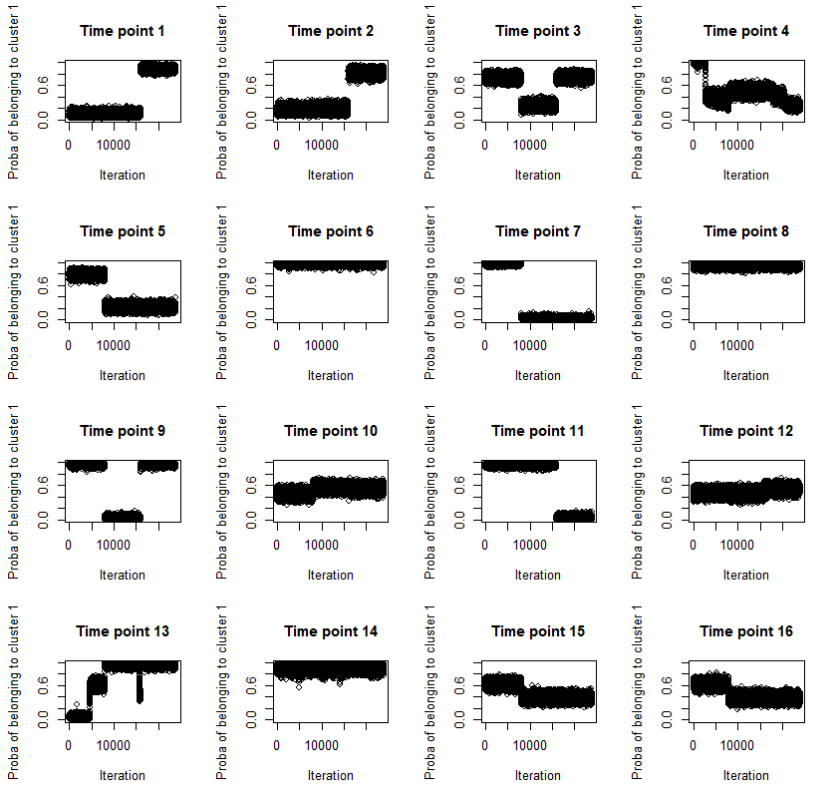

Figure 19: Traceplots for λ_1 (probability of belonging to cluster 1) for the 101st Congress through the 116th Congress.

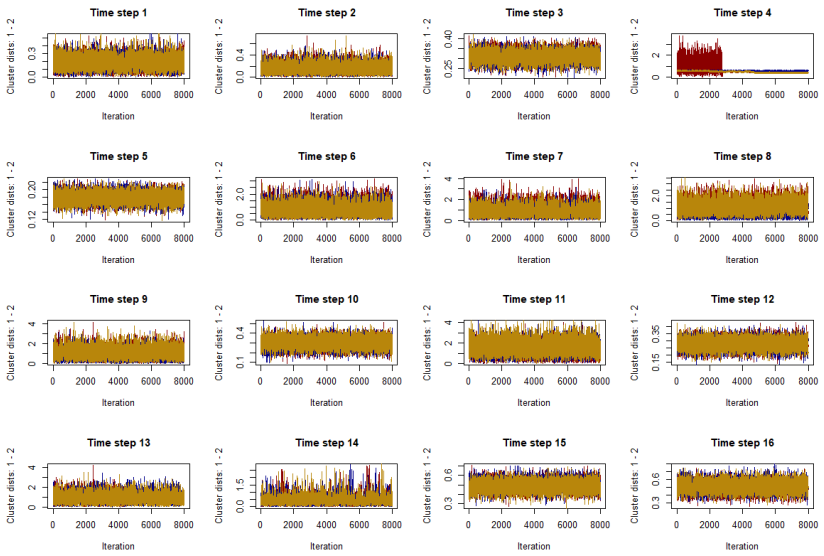

Figure 20: Traceplots for cluster distances for the 101st Congress through the 116th Congress.

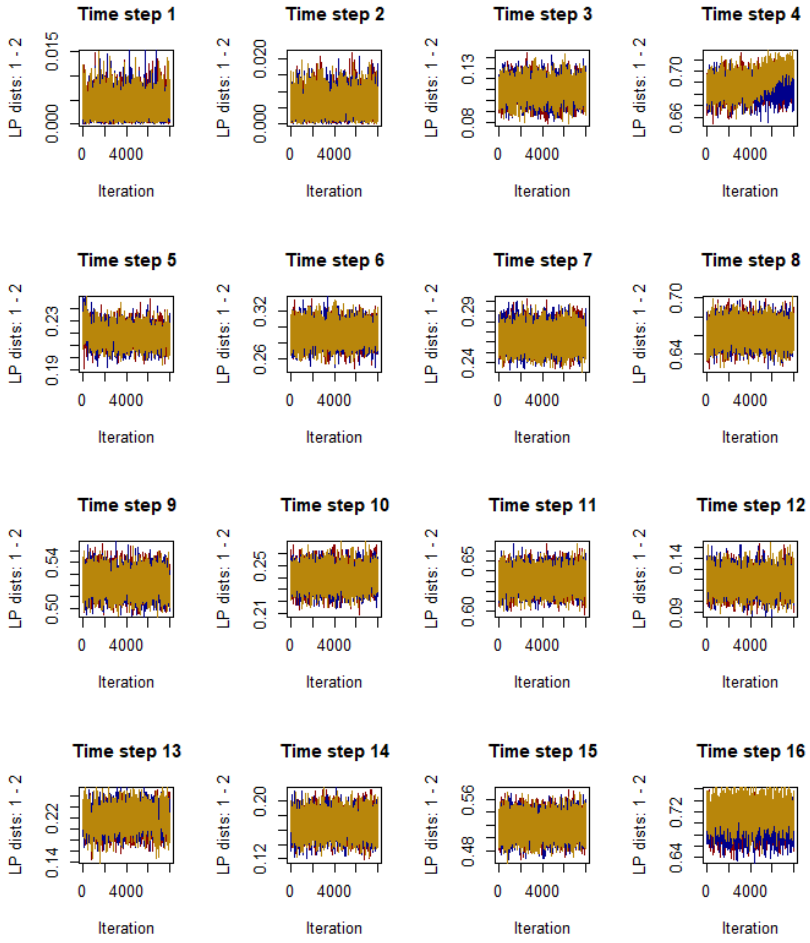

Figure 21: Traceplots for distances between the first and second senator for the 101st Congress through the 116th Congress.