Graphical Models for Categorical Data

Camilla Caroni, Fabio Comazzi, Andrea Deretti, Francesco Rettore, Michele Russo, Luca Zerman

Politecnico di Milano

Tutors: Prof.ssa Lucia Paci, Prof. Federico Castelletti

February 15, 2022

POLITECNICO
MILANO 1863

Overview

1. Introduction
2. Models
   - Multinomial-Dirichlet Model
   - Latent Normal Inverse-Wishart Model
3. Clustering
4. Results
5. Future Developments

Introduction

Main Goal: Develop Bayesian methods for the analysis of multivariate categorical data. In particular, we are interested in inferring dependence relations between categorical variables, also accounting for possible heterogeneity related to latent clustering structures in the data.

Definition

A graphical model is a probabilistic model for a collection of random variables based on a graph structure.

In particular, we will work with undirected decomposable graphs.

Outline

1. **Inference on the graph:** Find the most suitable graph $\mathcal{G}$ representing the conditional independence relationships among the variables in the dataset. In particular, we consider two models:
   - *Multinomial Dirichlet model*;
   - *Latent Normal Inverse-Wishart model*;

   Note that the algorithms we developed allow us to approximate a posterior distribution on the space of undirected decomposable graphs;

2. **Clustering:** Cluster the observations of the dataset via a Dirichlet Process mixture of graphical models.

Inference on the Graph: General Framework

Ingredients for inference on the graph:

- **Data Model:** $\mathsf{X}_1, \ldots \mathsf{X}_\mathsf{q} \mid \underline{\theta}, \mathcal{G} \sim p(\underline{\mathsf{x}} \mid \underline{\theta}, \mathcal{G})$
- **Prior on graph-dependent parameter $\underline{\theta}$ (given the graph):** $p(\underline{\theta} \mid \mathcal{G})$
- **Prior on graph $\mathcal{G}$:** $p(\mathcal{G})$

To proceed, we need to compute the posterior probability of $\mathcal{G}$ given the data $\mathbf{X}$:

$$p(\mathcal{G} \mid \mathbf{X}) \propto p(\mathbf{X} \mid \mathcal{G}) p(\mathcal{G})$$

where $p(\mathbf{X} \mid \mathcal{G}) = \int p(\mathbf{X} \mid \underline{\theta}, \mathcal{G}) p(\underline{\theta} \mid \mathcal{G}) d\underline{\theta}$ is the marginal likelihood.

Prior on the Graph

We considered three different choices for $p(\mathcal{G})$:

- **Uniform prior:** Assigns equal probabilities to all the graphs.
- **Binomial prior:** Assumes $A_{u,v} \mid \pi \overset{iid}{\sim} Be(\pi), \pi \in (0,1)$ where $A_{u,v}$ is the (u,v)-element of the upper-triangular adjacency matrix of $\mathcal{G}$.
- **Beta-Binomial prior:** Assumes $A_{u,v} \mid \pi \overset{iid}{\sim} Be(\pi), \pi \sim Beta(a,b)$.

Remark: The last two choices are particularly convenient to include prior information about the sparsity of the graph (whenever available).

Multinomial-Dirichlet Model

The *Multinomial-Dirichlet model* is defined as:

$$\mathbb{X} \mid \underline{\theta}, \mathcal{G} \sim p(\underline{x}^{(1)}, \dots \underline{x}^{(n)} \mid \underline{\theta}, \mathcal{G}) = \frac{\prod_{C \in \mathcal{C}} \prod_{\underline{x}_C \in \mathcal{X}_C} \pi(\underline{x}_C)^{n(\underline{x}_C)}}{\prod_{S \in \mathcal{S}} \prod_{\underline{x}_S \in \mathcal{X}_S} \pi(\underline{x}_S)^{n(\underline{x}_S)}}$$

$$\underline{\theta} \mid \mathcal{G} \sim \mathit{Hyper-Dirichlet}(A) \text{ s.t. } \forall C \in \mathcal{C}, \forall S \in \mathcal{S} :$$

$$p(\theta_C \mid \mathcal{G}) = \frac{\Gamma(\sum_{\underline{x}_C \in \mathcal{X}_C} a(\underline{x}_C))}{\prod_{\underline{x}_C \in \mathcal{X}_C} \Gamma(a(\underline{x}_C))} \prod_{\underline{x}_C \in \mathcal{X}_C} \pi(\underline{x}_C)^{a(\underline{x}_C)-1}$$

$$p(\theta_S \mid \mathcal{G}) = \frac{\Gamma(\sum_{\underline{x}_S \in \mathcal{X}_S} a(\underline{x}_S))}{\prod_{\underline{x}_S \in \mathcal{X}_S} \Gamma(a(\underline{x}_S))} \prod_{\underline{x}_S \in \mathcal{X}_S} \pi(\underline{x}_S)^{a(\underline{x}_S)-1}$$

$$\mathcal{G} \sim p(\mathcal{G})$$

where $\mathcal{C}$ and $\mathcal{S}$ are the set of the cliques and the set of the separators of the graph respectively.

Multinomial-Dirichlet Model: Marginal Likelihood

In this case the marginal likelihood is available in closed form[1]:

$$m(N \mid \mathcal{G}) = \frac{\prod_{C \in \mathcal{C}} m(N_C \mid \mathcal{G})}{\prod_{S \in \mathcal{S}} m(N_S \mid \mathcal{G})}$$

$$m(N_C \mid \mathcal{G}) = \frac{\Gamma(\sum_{\underline{x}_C \in \mathcal{X}_C} a(\underline{x}_C))}{\Gamma(\sum_{\underline{x}_C \in \mathcal{X}_C} a(\underline{x}_C) + n(\underline{x}_C))} \prod_{\underline{x}_C \in \mathcal{X}_C} \frac{\Gamma(a(\underline{x}_C) + n(\underline{x}_C))}{\Gamma(a(\underline{x}_C))}$$

$$m(N_S \mid \mathcal{G}) = \frac{\Gamma(\sum_{\underline{x}_S \in \mathcal{X}_S} a(\underline{x}_S))}{\Gamma(\sum_{\underline{x}_S \in \mathcal{X}_S} a(\underline{x}_S) + n(\underline{x}_S))} \prod_{\underline{x}_S \in \mathcal{X}_S} \frac{\Gamma(a(\underline{x}_S) + n(\underline{x}_S))}{\Gamma(a(\underline{x}_S))}$$

where $n(\underline{x}) = \sum_{i=1}^{n} \mathbb{I}\{\underline{x}^{(i)} = \underline{x}\}$.

[1] S. L. Lauritzen et al.: Hyper Markov Laws in the Statistical Analysis of Decomposable Graphical Models, 1993.

Multinomial-Dirichlet Model: MH Algorithm

Algorithm MH algorithm for the Multinomial-Dirichlet Model

Input: $\mathcal{G}^{(0)}$ (the initial candidate graph), M (the number of MCMC iterations)

Output: An MCMC sample $\{\mathcal{G}^{(t)}\}_{t=1}^{M}$ from $p(\mathcal{G} \mid N)$

for $t \leftarrow 1$ **to** M **do**

set $\mathcal{G} \leftarrow \mathcal{G}^{(t-1)}$;

draw a new candidate $\mathcal{G}'$ from $q(\mathcal{G}' \mid \mathcal{G})$;

compute $\alpha(\mathcal{G}' \mid \mathcal{G}) = \min\left\{1, \frac{m(N|\mathcal{G}')}{m(N|\mathcal{G})} \cdot \frac{p(\mathcal{G}')}{p(\mathcal{G})} \cdot \frac{q(\mathcal{G}|\mathcal{G}')}{q(\mathcal{G}'|\mathcal{G})}\right\}$;

update $\mathcal{G}^{(t)} = \begin{cases} \mathcal{G}', & \text{with probability } \alpha \\ \mathcal{G}^{(t-1)}, & \text{with probability } 1-\alpha \end{cases}$

end

Proposal Distribution

Problem: We need to define a suitable proposal distribution $q(\mathcal{G}' \mid \mathcal{G})$ from which we sample a new proposal graph ($\mathcal{G}'$) starting from the current state of the chain.

In particular, we build $\mathcal{G}'$ by either adding (a) or removing (b) an edge from $\mathcal{G}$ according to the following scheme:

1. Construct the space $\mathcal{O}_{\mathcal{G}}$ of all possible *undirected* graphs obtained by (a) or (b) (starting from $\mathcal{G}$);
2. Uniformly draw a graph $\mathcal{G}'$ from $\mathcal{O}_{\mathcal{G}}$;
3. If $\mathcal{G}'$ is decomposable propose it, otherwise go back to step (2).

Remark: Such approach is computationally efficient as it guarantees that $\frac{q(\mathcal{G}|\mathcal{G}')}{q(\mathcal{G}'|\mathcal{G})} = 1$ so that we do not have to evaluate $q(\cdot)$ for the computation of the acceptance probability α in the MH algorithm.

Latent Normal Inverse-Wishart Model

In this case we introduce latent Gaussian random variables $(Z_1, ..., Z_q)$ in order to define a conjugate model and to be able to sample from the graph posterior distribution in an easy way.

Therefore, the implementation of the method will be based on two steps:

1. **Sampling of the latent Gaussian data**;
2. **Inference on the graph given the Gaussian data**.

Latent Normal Inv-Wish Model: Inference on the Graph

The *Normal-Inverse-Wishart model* is defined as:

$$\mathbb{Z} \mid \Sigma, \mathcal{G} \sim p(\underline{z}^{(1)}, \dots \underline{z}^{(n)} \mid \Sigma, \mathcal{G}) \propto \frac{\prod_{C \in \mathcal{C}} |\Sigma_C|^{-\frac{n}{2}} \cdot e^{-\frac{1}{2} \sum_{i=1}^{n} \underline{z}_C^{(i)^T} \Sigma_C^{-1} \underline{z}_C^{(i)}}}{\prod_{S \in \mathcal{S}} |\Sigma_S|^{-\frac{n}{2}} \cdot e^{-\frac{1}{2} \sum_{i=1}^{n} \underline{z}_S^{(i)^T} \Sigma_S^{-1} \underline{z}_S^{(i)}}}$$

$$\Sigma \mid \mathcal{G} \sim HIW(b, D) \text{ s.t. } \forall C \in \mathcal{C}, \forall S \in \mathcal{S} :$$

$$p(\Sigma_C \mid \mathcal{G}) \propto |\Sigma_C|^{-(\frac{b}{2}+|C|)} \cdot e^{-\frac{1}{2} tr(\Sigma_C^{-1} D_C)}$$

$$p(\Sigma_S \mid \mathcal{G}) \propto |\Sigma_S|^{-(\frac{b}{2}+|S|)} \cdot e^{-\frac{1}{2} tr(\Sigma_S^{-1} D_S)}$$

$$\mathcal{G} \sim p(\mathcal{G})$$

where $\mathcal{C}$ and $\mathcal{S}$ are the set of the cliques and the set of the separators of the graph respectively.

Latent Normal Inv-Wish Model: Marginal Likelihood

In this case the marginal likelihood is available in closed form[1]:

$$m(\underline{z}^{(1)}, \dots \underline{z}^{(n)} \mid \mathcal{G}) = (2\pi)^{-nq/2} \frac{h(\mathcal{G}, b, D)}{h(\mathcal{G}, b^*, D^*)}$$

$$b^* = b + n$$

$$D^* = D + \sum_{i=1}^{n} \underline{z}^{(i)} \underline{z}^{(i)^T}$$

where $h(\mathcal{G}, b, D) = \frac{\prod_{C \in \mathcal{C}} |\frac{1}{2} D_C|^{\frac{b+|C|-1}{2}} \Gamma_{|C|}\left(\frac{b+|C|-1}{2}\right)^{-1}}{\prod_{S \in \mathcal{S}} |\frac{1}{2} D_S|^{\frac{b+|S|-1}{2}} \Gamma_{|S|}\left(\frac{b+|S|-1}{2}\right)^{-1}}$ and $\Gamma_p(\cdot)$ denotes the *multivariate gamma function*.

[1] C. M. Carvalho, J. G. Scott: Objective Bayesian Model Selection in Gaussian Graphical Models, 2009

Gibbs Sampler: Sampling of the Latent Gaussian Data

For each $j = 1, \dots q$ we establish a link between binary variable $X_j \in \{0, 1\}$ and its latent counterpart Z_j as:

$$X_j = \begin{cases} 1 & \text{if } Z_j < \theta_0^{(j)} \\ 0 & \text{if } Z_j \geq \theta_0^{(j)} \end{cases}$$

where the Z_j's are the latent Gaussian random variables and the $\theta_0^{(j)}$'s represent unknown cut-offs.
The posterior distribution for our model is:

$$p(\Sigma, \Theta, \mathcal{G}, \mathbb{Z} \mid \mathbb{X}) = p(\mathbb{X}, \mathbb{Z} \mid \Sigma, \Theta, \mathcal{G}) p(\Sigma \mid \mathcal{G}) p(\mathcal{G}) \prod_{j=1}^{q} p(\theta_0^{(j)})$$

where:

- $p(\mathbb{X}, \mathbb{Z} \mid \Sigma, \Theta, \mathcal{G})$ is the *augmented likelihood*;
- $\theta_0^{(j)} \overset{iid}{\sim} \mathcal{N}(0, \tau^2)$.

Gibbs Sampler: Full-Conditionals

Direct sampling from $p(\Sigma, \Theta, \mathcal{G}, \mathbb{Z} \mid \mathbb{X})$ is not possible. So, we implement a Gibbs Sampler with Metropolis' Hastings steps.
We consider the following full-conditional distributions:

- $p(\Sigma, \mathcal{G} \mid \Theta, \mathbb{Z}, \mathbb{X})$
- $p(\mathbb{Z}, \Theta \mid \Sigma, \mathcal{G}, \mathbb{X})$

For the full-conditional of $(\Sigma, \mathcal{G})$ we have:

$$p(\Sigma, \mathcal{G} \mid \Theta, \mathbb{Z}, \mathbb{X}) \propto p(\mathbb{Z} \mid \Sigma) p(\Sigma \mid \mathcal{G}) p(\mathcal{G})$$

where the three terms are the likelihood and priors of the Normal Inverse-Wishart Model.

Gibbs Sampler: Full-Conditionals

For the full-conditional of $(\mathbb{Z}, \Theta)$ we have:

$$p(\mathbb{Z}, \Theta \mid \Sigma, \mathcal{G}, \mathbb{X}) = p(\mathbb{Z} \mid \Theta, \Sigma, \mathcal{G}, \mathbb{X}) p(\Theta \mid \Sigma, \mathcal{G}, \mathbb{X})$$

The first factor is given by:

$$p(\mathbb{Z} \mid \Theta, \Sigma, \mathcal{G}, \mathbb{X}) \propto \prod_{i=1}^{n} dN_q(\underline{z}_i \mid \underline{0}, \Sigma) \mathbb{I}(\underline{z}_i \in C(\underline{x}_i, \Theta))$$

so that we can sample the $\mathbf{Z}^{(i)}, i = 1, \ldots, n$ independently from a *Multivariate Truncated Normal* distribution of support $C(\underline{x}_i, \Theta)$.

The second factor, on the other hand, is sampled in a sequential Random-Walk Metropolis-Hastings scheme.

Latent Normal Inv-Wish Model: MH Algorithm

Algorithm MH algorithm for the Normal-Inverse-Wishart Model

Input: $\mathcal{G}^{(0)}$, M (the number of MCMC iterations), $\Theta^{(0)}$, $\mathbb{Z}^{(0)}$, $\Sigma^{(0)}$

Output: An MCMC sample $\{\mathcal{G}^{(t)}\}_{t=1}^{M}$ from the graph posterior distribution

for $t \leftarrow 1$ **to** M **do**

$\Theta^{(t)} \leftarrow$ Metropolis-Hastings step$(\Sigma^{(t-1)}, \mathbb{X}, \tau^2, \Theta^{(t-1)})$;

$\Sigma^{-1^{(t)}} \sim HIW(b+n, D + \mathbb{Z}^{(t-1)^T}\mathbb{Z}^{(t-1)}, \mathcal{G}^{(t-1)})$;

$\mathbb{Z}^{(t)} \sim t\mathcal{N}(\Sigma^{(t)}, \Theta^{(t)}, \mathbb{X})$;

$\mathcal{G} \leftarrow \mathcal{G}^{(t-1)}$;

$\mathcal{G}' \sim q(\mathcal{G}' \mid \mathcal{G})$;

compute $\alpha(\mathcal{G}' \mid \mathcal{G}) = \min\left\{1, \frac{m(\mathbb{Z}^{(t)}|\mathcal{G}')}{m(\mathbb{Z}^{(t)}|\mathcal{G})} \cdot \frac{p(\mathcal{G}')}{p(\mathcal{G})} \cdot \frac{q(\mathcal{G}|\mathcal{G}')}{q(\mathcal{G}'|\mathcal{G})}\right\}$;

update $\mathcal{G}^{(t)} = \begin{cases} \mathcal{G}', & \text{with probability } \alpha \\ \mathcal{G}^{(t-1)}, & \text{with probability } 1-\alpha \end{cases}$

end

Clustering: Dirichlet Process Mixture of Graphical Models

The Dirichlet Process mixture model can be written in the following hierarchical structure:

$$
\begin{aligned}
y_i \mid \theta_i &\sim F(\theta_i) \\
\theta_i \mid M &\sim M \\
M &\sim DP(M_0, \alpha)\,,
\end{aligned}
$$

where $F(\theta)$ is a generic component of the mixture model from which our data y_i are drawn and M is the mixing distribution over θ with prior the Dirichlet Process, with concentration parameter α and base distribution M_0.

If two units share the same parameter, then they are assigned to the same cluster.

Clustering: Dirichlet Process Mixture of Graphical Models

An equivalent representation of the Dirichlet Process mixture model is obtained as the limit of a **finite mixture model** with K component:

$$
\begin{aligned}
y_i \mid c_i, \boldsymbol{\phi} &\sim F(\phi_{c_i}) \\
\phi_k &\sim M_0 \\
c_i \mid \boldsymbol{p} &\sim \mathit{Discrete}(p_1, ..., p_K) \\
\boldsymbol{p} &\sim \mathit{Dirichlet}(\alpha/K, ..., \alpha/K)\,.
\end{aligned}
$$

where $c_i \in \{1, ..., K\}$ is a random variable indexing the cluster to which the i-th observation belongs and $\boldsymbol{\phi}$ is the collection of the cluster-specific parameters.

Clustering Algorithm

1. **Update of the indicator variables**: the c_i's and the number of clusters K are updated with a Gibbs sampler;

2. **Update of the cluster-specific graphs**: the $\mathcal{G}_k, k = 1, \ldots, K$ are updated by performing a step of the Metropolis-Hastings algorithm introduced for the Multinomial-Hyper-Dirichlet model.

Reference: Radford M. Neal: Markov chain sampling methods for dirichlet process mixture models, 2000.

First Step: Update of the c_i's

The cluster indicator variables are updated sequentially by sampling from their full-conditional distribution, which is given by

If $c_i = c_j$ for some $j \neq i$:

$$\mathbb{P}(c_i = k \mid c_{-i}, y_i, \mathcal{G}) = b\frac{n_{-i,k}}{n-1+\alpha}\int F(y_i \mid \underline{\theta}, \mathcal{G}_k) dH_{-i,k}(\underline{\theta} \mid \mathcal{G}_k).$$

If $c_i \neq c_j \ \forall j \neq i$:

$$\mathbb{P}(c_i \neq c_j \, \forall j \neq i \mid c_{-i}, y_i, \mathcal{G}) = b\frac{\alpha}{n-1+\alpha}\int F(y_i \mid \underline{\theta}, \mathcal{G}^*) dM_0(\underline{\theta} \mid \mathcal{G}^*)\,.$$

where $\mathcal{G}^*$ is a graph associated to a (new) empty cluster which has to be randomly sampled from the baseline measure on the space of undirected decomposable graphs.

Second Step: Update of the $\mathcal{G}_k$'s

Given the number of clusters K, the update of the cluster-specific graphs $\{\mathcal{G}_k\}_{k=1,\dots,K}$ is performed by running a step of the previously introduced Metropolis-Hastings algorithm on each cluster $C_k := \{y_i : c_i = k\}$.

Algorithm MH algorithm for the Multinomial-Dirichlet Model

Input: $\mathcal{G}^{(0)}$ (the initial candidate graph), M (the number of MCMC iterations)

Output: An MCMC sample $\{\mathcal{G}^{(t)}\}_{t=1}^{M}$ from $p(\mathcal{G} \mid N)$

for $t \leftarrow 1$ **to** M **do**

set $\mathcal{G} \leftarrow \mathcal{G}^{(t-1)}$;

draw a new candidate $\mathcal{G}'$ from $q(\mathcal{G}' \mid \mathcal{G})$;

compute $\alpha(\mathcal{G}' \mid \mathcal{G}) = \min\left\{1, \frac{m(N|\mathcal{G}')}{m(N|\mathcal{G})} \cdot \frac{p(\mathcal{G}')}{p(\mathcal{G})} \cdot \frac{q(\mathcal{G}|\mathcal{G}')}{q(\mathcal{G}'|\mathcal{G})}\right\}$;

update $\mathcal{G}^{(t)} = \begin{cases} \mathcal{G}', & \text{with probability } \alpha \\ \mathcal{G}^{(t-1)}, & \text{with probability } 1-\alpha \end{cases}$

end

Assessment of the Performances

Methodology: We created 20 categorical datasets starting from 20 randomly generated decomposable graphs (with 6 nodes) and we ran the MH algorithms on such datasets. After that, we computed the *Structural Hamming Distance* between the original graph and the one estimated from the chain.

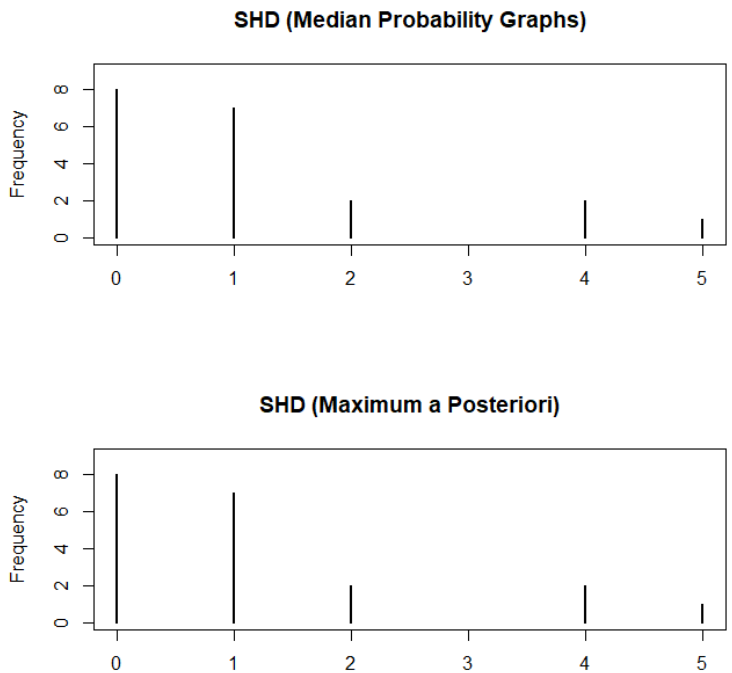


Figure: Multinomial-Dirichlet.

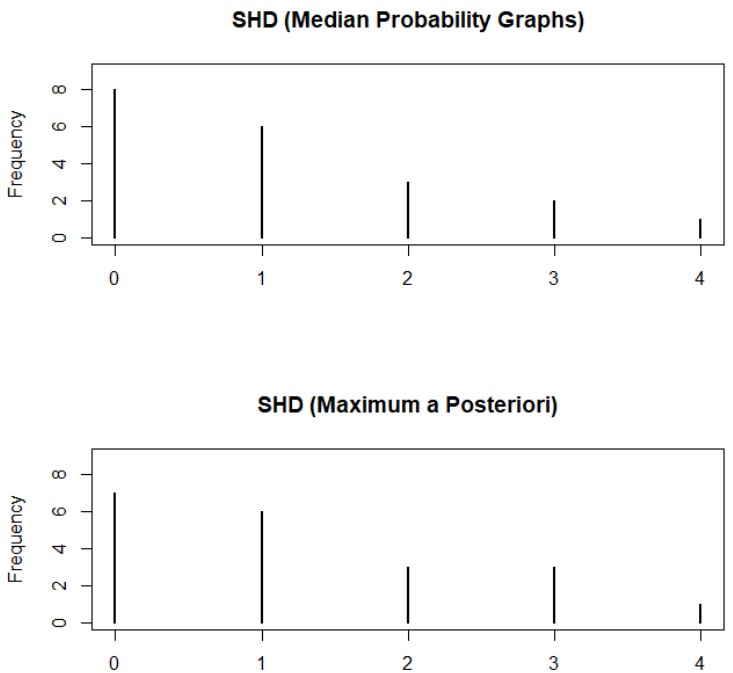
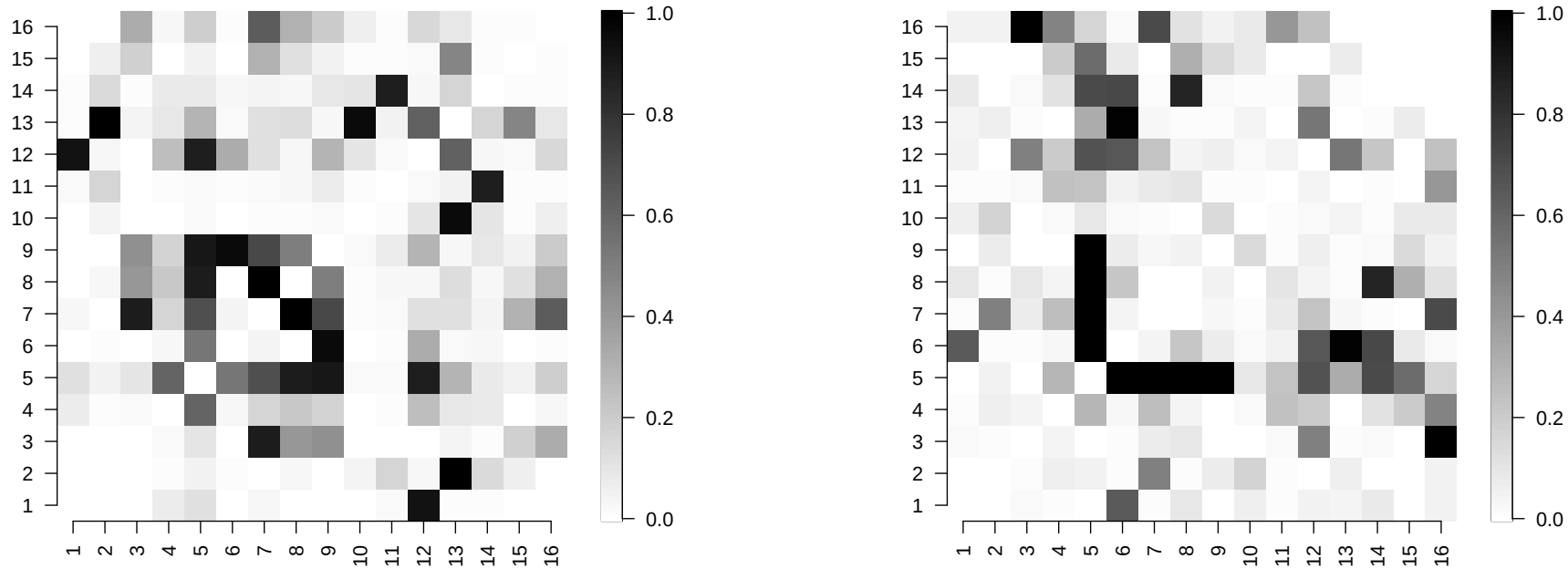


Figure: Latent Normal I-W.

Inference on the Congressional Voting Records dataset

The Congressional Voting Records dataset includes votes for each of the U.S. House of Representatives Congressmen on 16 questions.

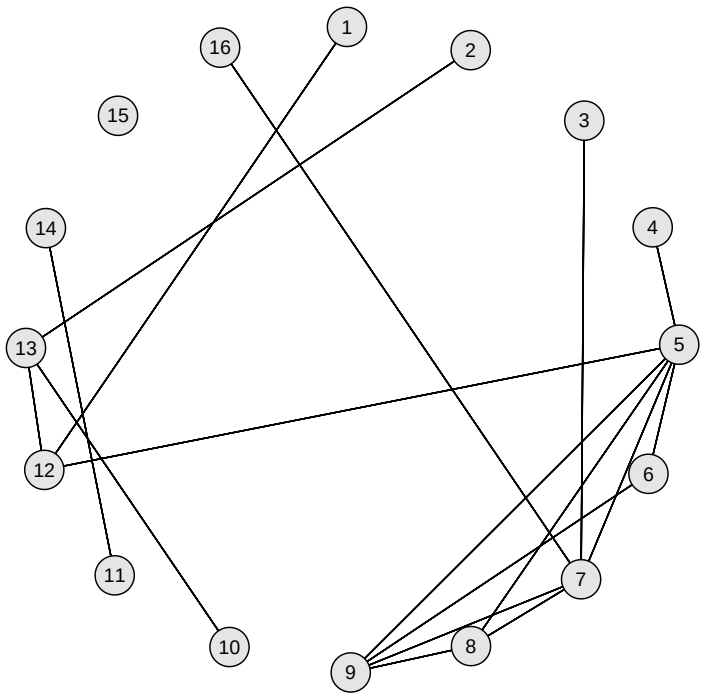


(a) Republican

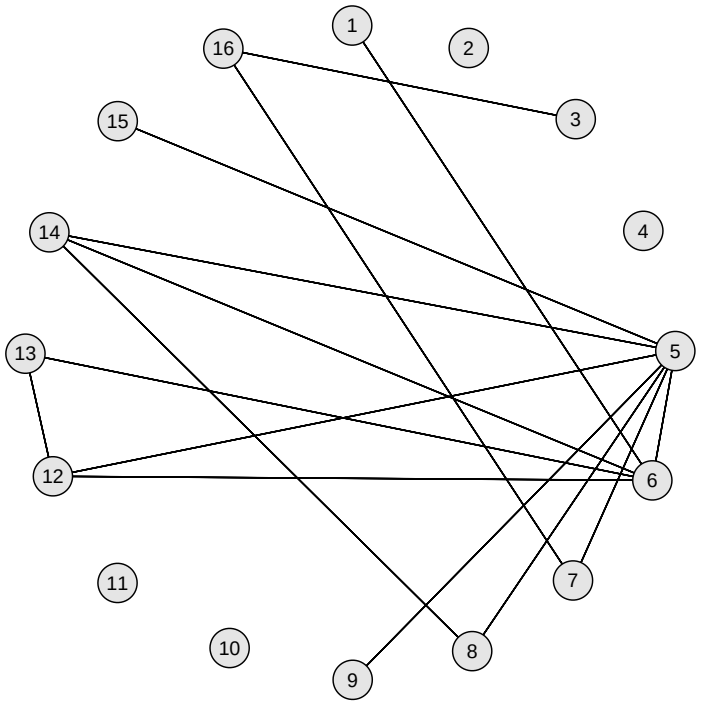
(b) Democrat

Figure: Heatmaps of the edge-inclusion probabilities.

Inference on the Congressional Voting Records dataset



(a) Republican



(b) Democrat

Figure: Median Probability Graphs estimated on the data regarding the two groups of congressmen.

Future Developments

Some possible future developments of the project are the following:

- **Improve the computational performance of the Latent Normal Inverse-Wishart model**;
- **Extend the Latent Normal Inverse-Wishart model to the case of non-binary ordinal variables**;
- **Implement the Dirichlet Process mixture of Latent Normal Inverse-Wishart models**;
- **Improve of the computational performance of the Dirichlet Process mixture of Multinomial-Dirichlet models**.