

Exploring networks using latent variable models in R with lvm4net

Isabella Gollini isabella.gollini@ucd.ie

Sunbelt 2019 - Montreal



Setup before we start:

Slides and material available at:

<https://github.com/igollini/lvm4net-Workshop>

Check you have these packages installed:

```
install.packages(c("lvm4net", "statnet", "latentnet", "RSiena",  
  "sna", "manet", "sand", "colorspace"))
```

If you want you can join the workshop on RStudio cloud by clicking here.



Overview

Latent Variable Models

Models for one-mode networks:

- ▶ Latent Space Models (LSM) for a single network

Models for multiplex one-mode networks:

- ▶ Latent Space Joint Models (LSJM) for multiple network views

Models for two-mode networks:

- ▶ Latent Class Analysis (LCA)
- ▶ Latent Trait Analysis (LTA)
- ▶ Mixture of Latent Trait Analysers (MLTA)



Latent Variable Models

- ▶ Useful approach to analyze and visualize multivariate and network data.
- ▶ They assume that the behaviour of the observed data depends on the presence of unobserved latent variables.

Abstract LV

- ▶ Intelligence
- ▶ Power
- ▶ Social Class

Measurable LV

- ▶ Income
- ▶ Political party
- ▶ Education



Latent Variable Models

LATENT VARIABLE	DATA		
	Continuous	Categorical Two-Mode Net.	One-mode Network
Categorical	Latent Profile A.	Latent Class A.	Stochastic Blockmodels
Continuous	Factor A.	Latent Trait A.	Latent Space M.
Cat + Cont	Mixture FA	Mixture LTA	Latent Position Cluster M.

- ▶ Categorical LV are assumed to be *multinomial*.
- ▶ Continuous LV are assumed to be *multivariate normal*.
- ▶ Cat + Cont LV assume that the data come from different *groups* within which the variability is explained by *multivariate normal* latent variables.



Latent Space Models (LSM) for a single network



Notation

- ▶ N number of nodes of an observed one-mode network;
- ▶ \mathbf{Y} ($N \times N$) adjacency matrix

$$y_{ij} = \begin{cases} 1 & \text{if there is an edge between node } i \text{ and } j \\ 0 & \text{otherwise} \end{cases}$$

- ▶ $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ D -dim latent variable



Latent Space Model (LSM) for one-mode Networks

We assume that each node i has an unknown position \mathbf{z}_i in a D -dim *Euclidean latent space*.

$$p(\mathbf{Y}|\mathbf{Z}, \alpha) = \prod_{i \neq j}^N \frac{\exp(\alpha - |\mathbf{z}_i - \mathbf{z}_j|^2)^{y_{ij}}}{1 + \exp(\alpha - |\mathbf{z}_i - \mathbf{z}_j|^2)}$$

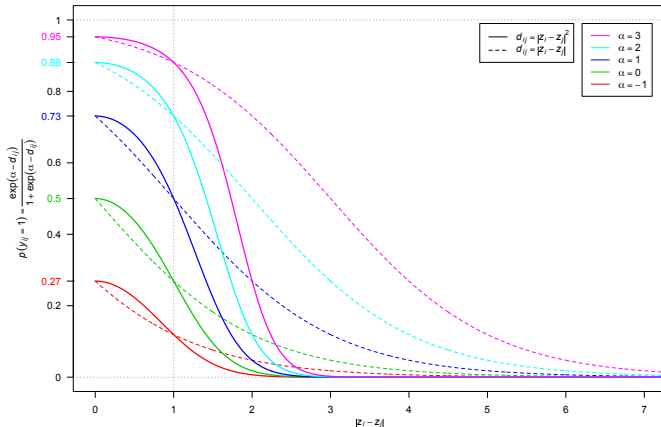
with $p(\alpha) = \mathcal{N}(\xi, \psi^2)$, $p(\mathbf{z}_i) \stackrel{iid}{=} \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_D)$ and σ^2, ξ, ψ^2 are fixed parameters.

- ▶ The posterior distribution *cannot* be calculated analytically.
- ▶ We propose to use the **Squared Euclidean Distance**



LSM – Squared Euclidean Distance vs Euclidean Distance

- ▶ SED requires less approximation in the estimation procedure we propose.
- ▶ It allows to visualize more clearly the presence of potential clusters, giving a higher probability of a link between two close nodes in the latent space and lower probability to two nodes lying far away from each other.



LSM – Variational Approach

The posterior distribution *cannot* be calculated analytically.

We use a **Variational inference approach** that is considerably quicker but less accurate than MCMC.

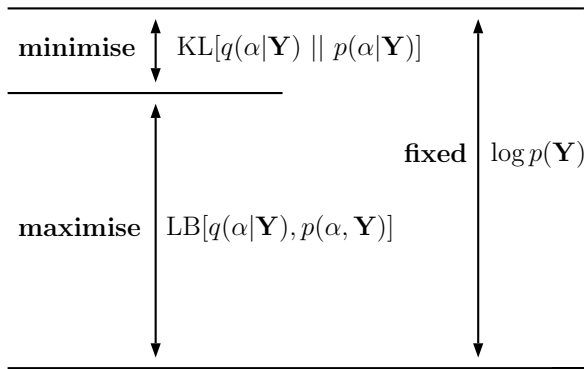
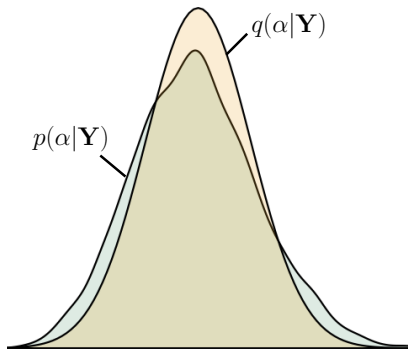
- ▶ We find a lower bound of the log marginal likelihood $\log p(\mathbf{Y})$ by introducing the variational posterior distribution $q(\mathbf{Z}, \alpha | \mathbf{Y})$:

$$q(\mathbf{Z}, \alpha | \mathbf{Y}) = q(\alpha) \prod_{i=1}^N q(\mathbf{z}_i)$$

where $q(\alpha) = \mathcal{N}(\tilde{\xi}, \tilde{\psi}^2)$ and $q(\mathbf{z}_i) = \mathcal{N}(\tilde{\mathbf{z}}_i, \tilde{\Sigma})$.



Variational Inference



Variational Inference VS MCMC

- ▶ Closed form posteriors
- ▶ Far faster than MCMC based methods
- ▶ In the absence of posterior dependence, the lower bound would match the log likelihood.
- ▶ As long as the posterior dependence is weak, the VA may be useful:
 - ▶ Larger networks
 - ▶ For starting point of MCMC algorithms
 - ▶ To explore the model space.

Main cons:

- ▶ Underestimates variances
- ▶ Difficult to assess how tight the lower bound is.
- ▶ Sensitive to starting values (local minima)



Girls Networks

Excerpt of 50 girls from the “Teenage Friends and Lifestyle Study” data (Pearson and Michell 2000, and Pearson and West 2003).

The dataset contains three directed networks about friendship relations between students in a school in Glasgow, Scotland.

Each student was asked to name up to six best friends in the cohort.

The data comes from three yearly waves, from 1995 to 1997.

This dataset is available from at the SIENA software website and in the RSiena package.



Girls Networks

Load the data about the first wave:

```
Y <- RSiena::s501  
alcohol <- RSiena::s50a[,1]  
smoking <- RSiena::s50s[,1]
```

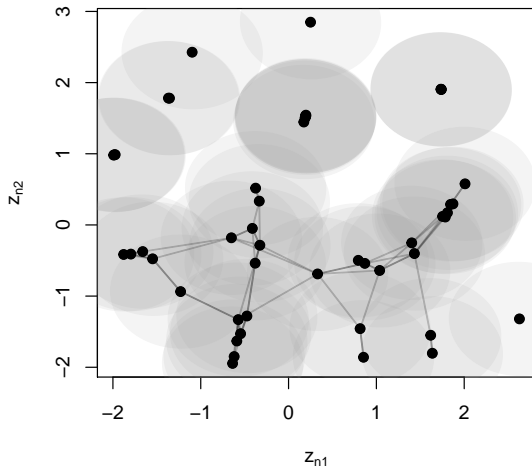
Fit the model:

```
library(lvm4net)  
D <- 2  
modLSM <- lsm(Y, D, nstart = 3)
```



Plot the latent positions:

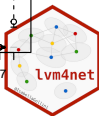
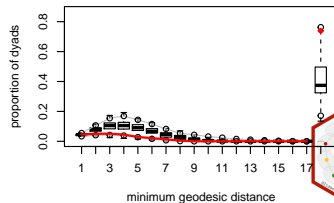
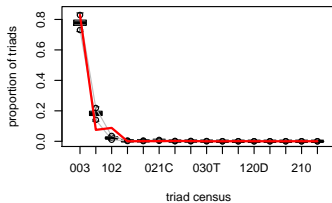
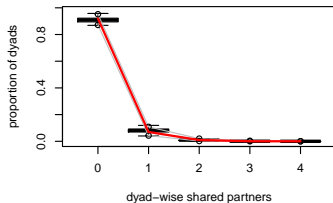
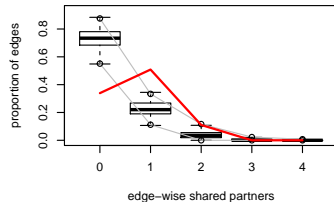
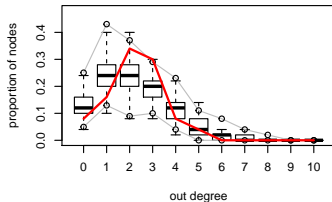
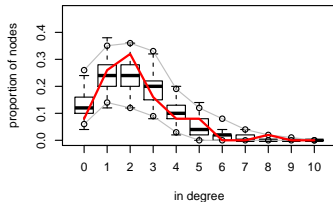
```
plot(modLSM, Y, drawCB = TRUE)
```



Check the Goodness of Fit:

```
gof_s501 <- goflsm(modLSM, Y, doplot = TRUE)
```

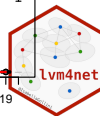
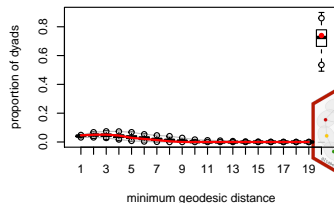
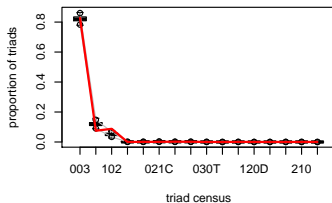
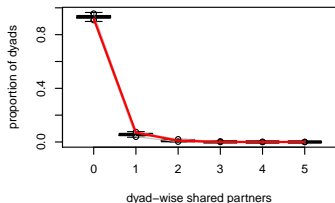
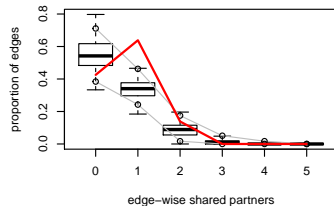
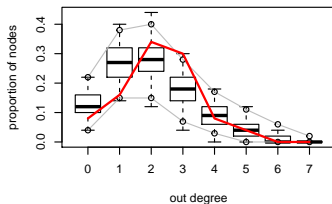
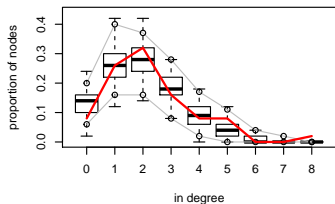
Goodness-of-fit diagnostics



Changing sigma and the prior for alpha to :

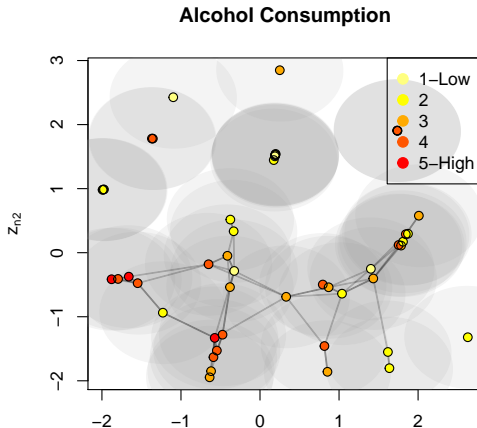
```
modLSM2 <- lsm(Y, D, sigma = 2.5, xi = -1, psi2 = 1, nstart = 3)
gof_s501_2 <- goflsm(modLSM2, Y, doplot = TRUE)
```

Goodness-of-fit diagnostics



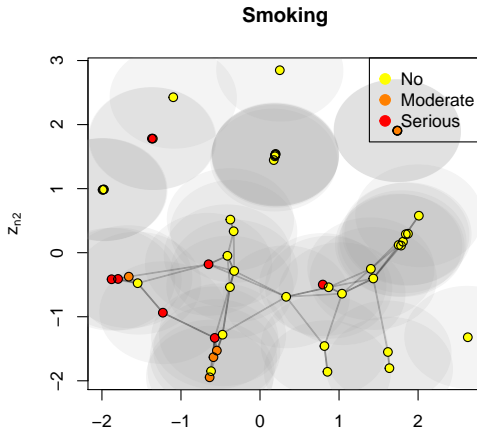
Latent positions – Alcohol Consumption:

```
plot(modLSM, Y, drawCB = TRUE,  
     colPl = rev(heat.colors(5))[alcohol], main = "Alcohol Consumption")  
legend("topright", legend = c("1-Low", 2:4, "5-High"),  
      col = rev(heat.colors(5)), pch = 20, pt.cex = 2)
```



Latent positions – Smoking:

```
plot(modLSM, Y, drawCB = TRUE,  
     colPl = rev(heat.colors(3))[smoking], main = "Smoking")  
legend("topright", legend = c("No", "Moderate", "Serious"),  
      col = rev(heat.colors(3)), pch = 20, pt.cex = 2)
```



Comparison between R packages

Package	Eucl.Dist.	(Eucl.Dist.) ²	MCMC	Variational	Clusters
latentnet	✓	✗	✓	✗	✓
VBLPCM	✓	✗	✗	✓	✓
lvm4net	✗	✓	✗	✓	✗



Your turn!!

- ▶ Reproduce the analysis in the slides,
- ▶ Now use the Lazega lawyers network data, see `?sand::lazega` for more information.

```
data(lazega, package = "sand")  
Y <- as.matrix(igraph::get.adjacency(lazega))  
A <- igraph::get.data.frame(lazega, what = "vertices")
```

- ▶ Remember to check the goodness of fit, and use the vertex attributes to interpret your results.



Models for multiplex one-mode networks:

Latent Space Joint Models (LSJM) for multiple network views



Joint Modelling of Multiple Networks

- ▶ In many applications the behaviour of the nodes is strongly shaped by the complex relation of many interactions.
- ▶ We have K networks on the same N nodes.
- ▶ We assume that different network views positively depend on each other
- ▶ A latent variable $\mathbf{z}_i \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ identifies the position of each node in a D -dim latent space.



Joint Modelling of Multiple Network Views – Model

The probability of a link depends on the distance between two nodes in the latent space.

$$p(\mathbf{Y}_1, \dots, \mathbf{Y}_K | \mathbf{Z}, \alpha) = \prod_{k=1}^K \prod_{i \neq j}^N \frac{\exp(\alpha_k - |\mathbf{z}_i - \mathbf{z}_j|^2)^{y_{ijk}}}{1 + \exp(\alpha_k - |\mathbf{z}_i - \mathbf{z}_j|^2)}$$

- ▶ Variational Approach $k = 1, \dots, K$: $p(\mathbf{z}_i | \mathbf{Y}_k) \sim \mathcal{N}(\tilde{\mathbf{z}}_{ik}, \tilde{\Sigma}_k)$.
- ▶ Joining the two models:

$$\begin{aligned} p(\mathbf{z}_i | \mathbf{Y}_1, \dots, \mathbf{Y}_K; \Theta_1, \dots, \Theta_K) &\propto \frac{\prod_{k=1}^K p(\mathbf{z}_i | \mathbf{Y}_k; \Theta_k)}{p(\mathbf{z}_i)^{K-1}} \\ &\propto \mathcal{N}(\bar{\mathbf{z}}_i, \bar{\Sigma}) \end{aligned}$$

$$\text{where } \bar{\Sigma} = \left[\sum_{k=1}^K \tilde{\Sigma}_k^{-1} - \frac{K-1}{\sigma^2} \mathbf{I}_D \right]^{-1} \text{ and } \bar{\mathbf{z}}_n = \bar{\Sigma} \left[\sum_{k=1}^K \tilde{\Sigma}_k^{-1} \tilde{\mathbf{z}}_{ik} \right]$$



Joint Modelling of Multiple Network Views – EM Algorithm

E-Step Estimate the parameters of the joint posterior distribution $\tilde{\Sigma}^{(i+1)}$ and $\tilde{\mathbf{z}}_i^{(i+1)}$:

$$\begin{aligned} Q(\Theta_1, \dots, \Theta_K; \Theta_1^{(i)}, \dots, \Theta_K^{(i)}) &= \\ &= \sum_{k=1}^K \mathbb{E}_{p(\mathbf{Z}|\mathbf{Y}_1, \dots, \mathbf{Y}_K; \Theta_1^{(i)}, \dots, \Theta_K^{(i)})} [\log(p(\mathbf{Y}_k, \mathbf{Z}|\Theta_k))] \end{aligned}$$

- ▶ We estimate the parameters $\tilde{\mathbf{z}}_{nk}, \tilde{\Sigma}_k$ of the posterior distribution $p(\mathbf{z}_n|\mathbf{Y}_k; \Theta_k)$ given each network k separately.
- ▶ We merge these estimates to find joint posterior distribution of the latent positions $\mathcal{N}(\tilde{\mathbf{z}}_i, \tilde{\Sigma})$.

M-Step Update the variational model parameters $\tilde{\xi}_1, \dots, \tilde{\xi}_K, \tilde{\psi}_1^2, \dots, \tilde{\psi}_K^2$:

$$(\Theta_1^{(i+1)}, \dots, \Theta_K^{(i+1)}) = \operatorname{argmax} Q(\Theta_1, \dots, \Theta_K; \Theta_1^{(i)}, \dots, \Theta_K^{(i)}).$$



Back to the School Girls Example

```
Y1 <- RSiena::s501  
Y2 <- RSiena::s502  
Y3 <- RSiena::s503  
Y123 <- list(Y1 = Y1, Y2 = Y2, Y3 = Y3)
```

We fit LSM separately to each network view

```
modLSM1 <- lsm(Y1, D)  
modLSM2 <- lsm(Y2, D)  
modLSM3 <- lsm(Y3, D)
```

We fit LSJM

```
modLSJM123 <- lsjm(Y123, D)
```



Match rotations

```
Z123 <- modLSJM123$EZ  
  
Z1 <- rotXtoY(modLSM1$lsmeEZ, Z123)$Xrot  
Z2 <- rotXtoY(modLSM2$lsmeEZ, Z123)$Xrot  
Z3 <- rotXtoY(modLSM3$lsmeEZ, Z123)$Xrot
```

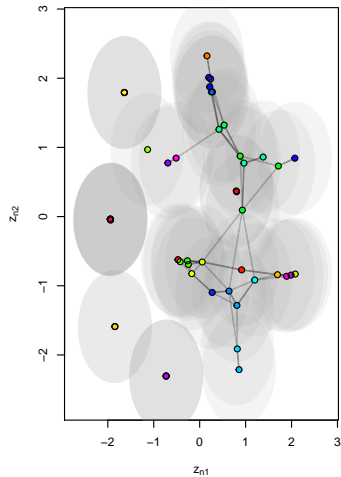
Some set up for the plots:

```
XYlimb <- range(Z123, Z1, Z2, Z3)  
namesb <- paste('Network ', 1:3, sep = ' ')  
colPl <- rainbow(50)
```

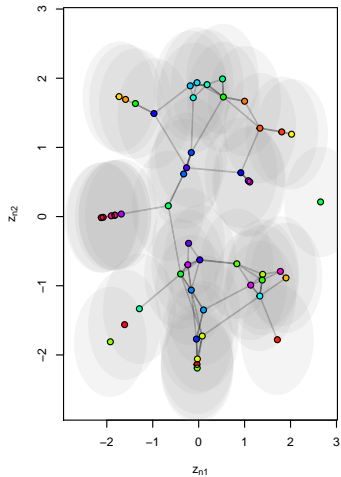


LSM PLOTS

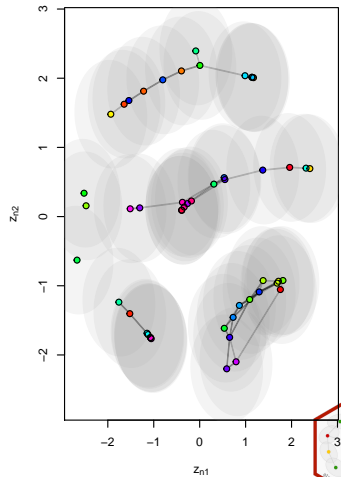
Network 1



Network 2



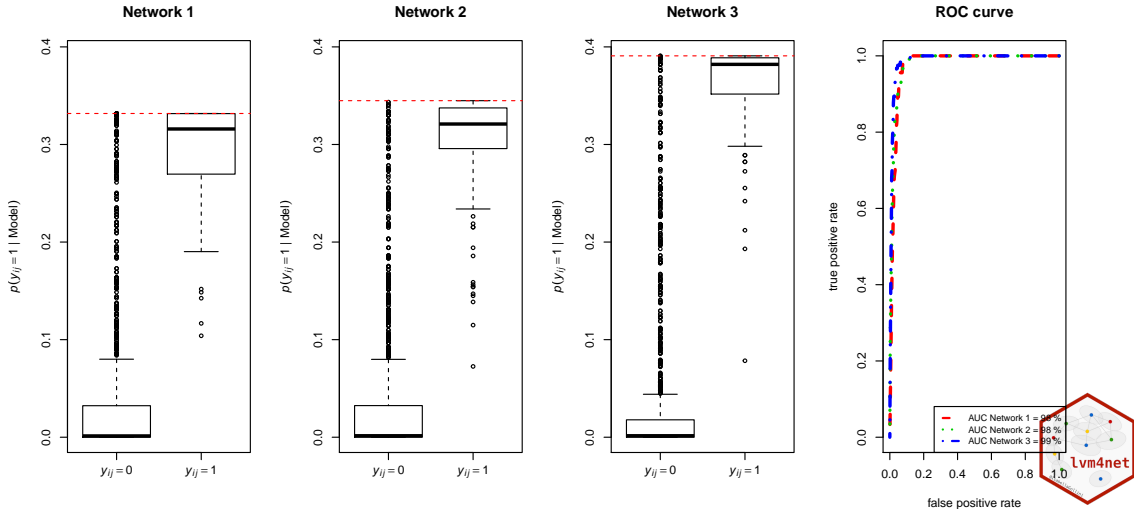
Network 3



Box-roc plots for three models fitted separately

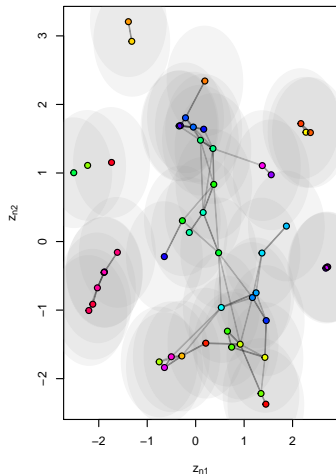
Network 1 Network 2 Network 3

0.9775391 0.9819276 0.9883845

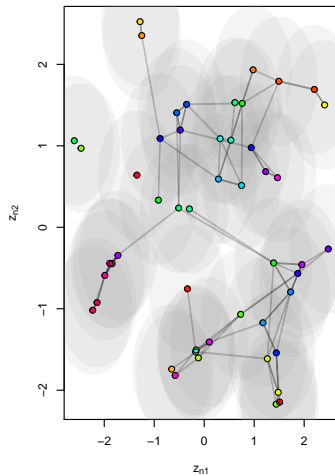


LSJM plot for each view

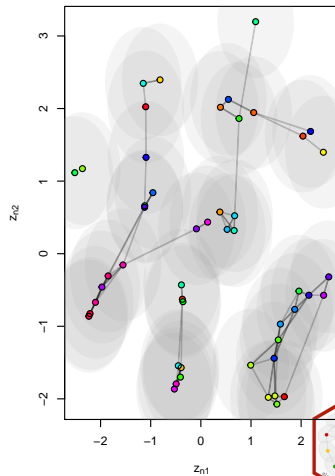
Network 1



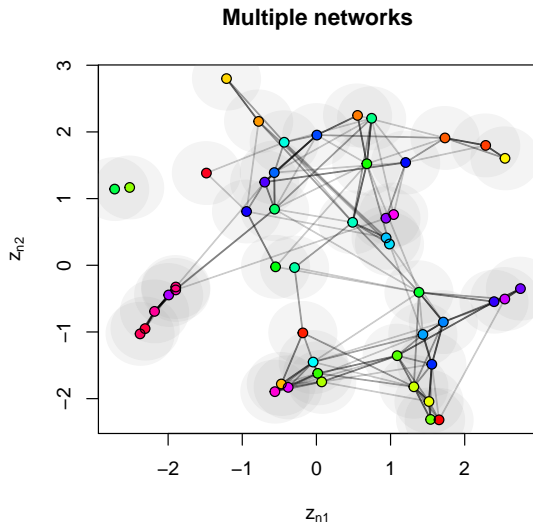
Network 2



Network 3



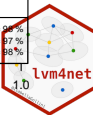
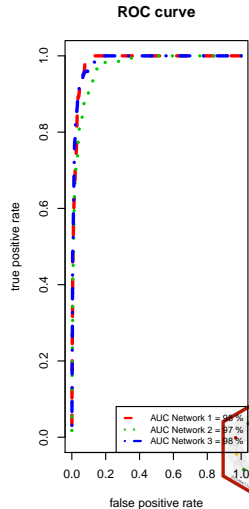
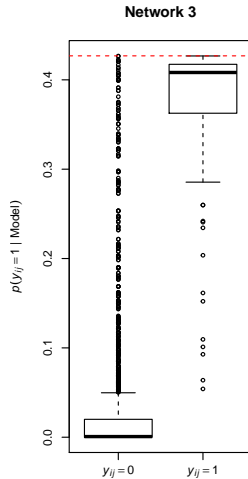
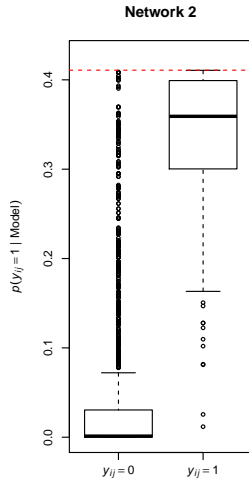
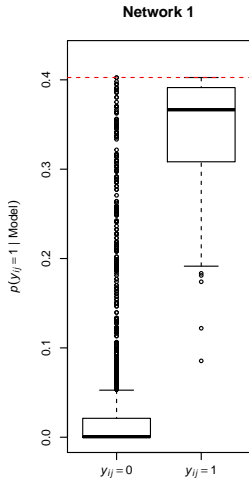
Overall position for the LSJM



Box-roc plots for the joint model

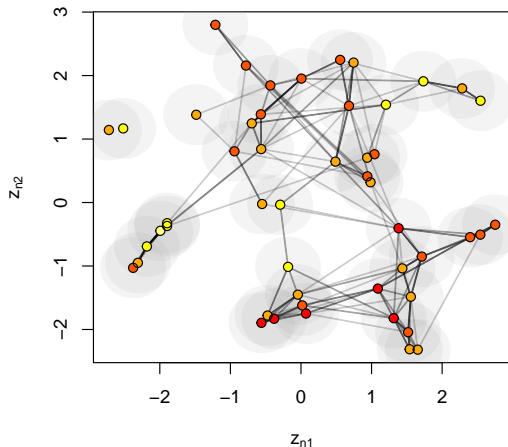
Network 1 Network 2 Network 3

0.9830223 0.9688229 0.9841734

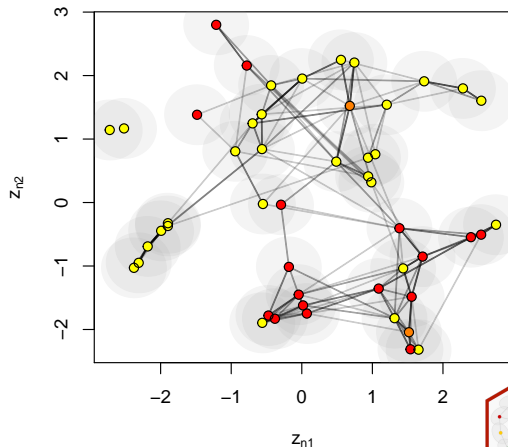


Use the attributes to interpret the latent space

Multiple networks – Alcohol



Multiple networks – Smoking



Your turn!!

- ▶ Reproduce the analysis in the slides,
- ▶ Now use the Sampson network data, (sampson dataset has been downloaded from the lda package)
 - ▶ Start by analysing: Esteem + Positive Influence + Praise

```
load("sampson.Rdata")  
Y1 <- (sampson$SAMPES > 0) * 1 # Esteem  
Y2 <- (sampson$SAMPIN > 0) * 1 # Positive Influence  
Y3 <- (sampson$SAMPPR > 0) * 1 # Praise
```



Models for two-mode networks:

- Latent Class Analysis (LCA)
- Latent Trait Analysis (LTA)
- Mixture of Latent Trait Analysers (MLTA)



Two-Mode Networks

- ▶ Two-mode networks: the sender and the receiver nodes belong to two separate sets.
- ▶ The incidence matrix is an $(N \times R)$ matrix, where N is the number of sender nodes and R the number of receiver nodes.

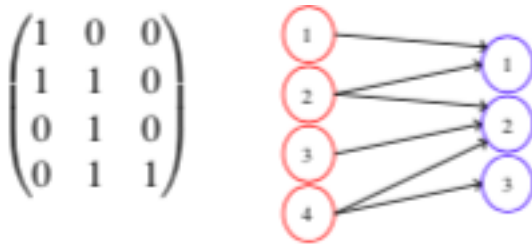


Figure 1: Two-mode Network with $N = 4$ sender and $R = 3$ receiver nodes

- ▶ Two-mode networks can be analyzed using multivariate models.
- ▶ One-mode network models can be applied projecting the two-mode network into a $(N \times N)$ or $(R \times R)$ one-mode network. This leads to a loss of information.

Notation

- ▶ N number of *sender* nodes of an observed two-mode network;
- ▶ R number of *receiver* nodes of an observed two-mode network;
- ▶ \mathbf{Y} ($N \times R$) incidence matrix

$$y_{nr} = \begin{cases} 1 & \text{if there is an edge between sender node } n \text{ and receiver node } r \\ 0 & \text{otherwise} \end{cases}$$

- ▶ G different latent groups or classes of observations
- ▶ η_g the prior probability that an individual is in the g -th class
- ▶ t_{ng} binary latent variable:
 - ▶ $t_{ng} = 1$ if obs n comes from group g
 - ▶ $t_{ng} = 0$ otherwise
- ▶ $\mathbf{z}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ D -dim continuous latent variable



Latent Class Analysis

LCA assumes that a set of N sender nodes comes from G different groups within which they are assumed to be conditionally independent.

The log-likelihood function is defined as:

$$\ell = \sum_{n=1}^N \log \left(\sum_{g=1}^G \eta_g \prod_{r=1}^R \pi_{gr}^{y_{nr}} (1 - \pi_{gr})^{(1-y_{nr})} \right).$$

LCA can be seen as a *finite mixture model* in which the component distributions are $Bernoulli(\pi_{g1}, \dots, \pi_{gR})$ with mixing proportions η_g . The parameters can be estimated via MLE using the EM algorithm.



Latent Trait Analysis

LTA assumes that there is a D -dimensional continuous latent variable \mathbf{z}_n underlying the behavior of the presence of links given by a sender node \mathbf{y}_n .

The conditional distribution of y_{nr} given \mathbf{z}_n is:

$$p(y_{nr}|\mathbf{z}_n) = (\pi_r(\mathbf{z}_n))^{y_{nr}} (1 - \pi_r(\mathbf{z}_n))^{1-y_{nr}},$$

And, the response function is:

$$\pi_r(\mathbf{z}_n) = \frac{1}{1 + \exp(-[b_r + \mathbf{w}_r^T \mathbf{z}_n])}, \quad 0 \leq \pi_r(\mathbf{z}_n) \leq 1,$$

Therefore, the log-likelihood is:

$$\ell = \sum_{n=1}^N \log \left(\int \prod_{r=1}^R p(x_{nr}|\mathbf{z}_n) p(\mathbf{z}_n) d\mathbf{z}_n \right)$$



Mixture of Latent Trait Analyzers

MLTA assumes that a set of observations comes from G different groups, and the presence of links given by the observation \mathbf{y}_n depends on both the group and the D dimensional continuous latent variable \mathbf{z}_n .

The conditional distribution given \mathbf{z}_n for each group is:

$$p(y_{nr} | \mathbf{z}_n, t_{ng} = 1) = (\pi_{rg}(\mathbf{z}_n))^{y_{nr}} (1 - \pi_{rg}(\mathbf{z}_n))^{1-y_{nr}},$$

And, the response function for each group is given by:

$$\pi_{rg}(\mathbf{z}_n) = \frac{1}{1 + \exp(-[b_{rg} + \mathbf{w}_{rg}^T \mathbf{z}_n])}, \quad 0 \leq \pi_{rg}(\mathbf{z}_n) \leq 1.$$

The log-likelihood can be written as:

$$\ell = \sum_{n=1}^N \log \left(\sum_{g=1}^G \eta_g \int \prod_{r=1}^R p(x_{nr} | \mathbf{z}_n, t_{ng} = 1) p(\mathbf{z}_n) d\mathbf{z}_n \right)$$



Model Fitting – MLTA

To obtain the approximation of ℓ it is necessary to use an EM algorithm:

- [1] *E-step*: Estimate t_{ng} as the posterior probability that an individual with response vector \mathbf{y}_n belongs to group g .
- [2] *M-step*: Estimate η_g as the proportion of observations in group g .
- [3] Estimate the model parameters \mathbf{b} , \mathbf{w} and ℓ :
 - Variational approach
- [4] Return to [1] until convergence is attained.



Variational Approach – MLTA

The main aim of this approach is to maximize a lower bound of the marginal likelihood function introducing the variational parameters ξ .

$$p(y_{nr}|\mathbf{z}_n, t_{ng} = 1) \leq \tilde{p}(y_{nr}|\mathbf{z}_n, t_{ng} = 1, \xi_{nrg}) \text{ with equality when } |\xi_{nrg}| = A_{nrg}$$

$$\tilde{p}(y_{nr}|\mathbf{z}_n, t_{ng} = 1, \xi_{nrg}) = \sigma(\xi_{nrg}) \exp\left(\frac{A_{nrg} - \xi_{nrg}}{2} + \lambda(\xi_{nrg})(A_{nrg}^2 - \xi_{nrg}^2)\right)$$

where $A_{nrg} = (2y_{nr} - 1)(b_{rg} + \mathbf{w}_{rg}^T \mathbf{z}_n)$, $\sigma(\xi_{nrg}) = [1 + \exp(-\xi_{nrg})]^{-1}$, and $\lambda(\xi_{nrg}) = \frac{0.5 - \sigma(\xi_{nrg})}{2\xi_{nrg}}$.



Variational Approach – MLTA

This gives the lower bound for the component distributions:

$$\mathcal{L}(\xi_{ng}) = \int \prod_{r=1}^R \tilde{p}(y_{nr} | \mathbf{z}_n, t_{ng} = 1, \xi_{nrg}) p(\mathbf{z}_n) d\mathbf{z}_n$$

The log-likelihood lower bound takes the form:

$$\ell \geq \sum_{n=1}^N \log \left(\sum_{g=1}^G \eta_g \mathcal{L}(\xi_{ng}) \right).$$

- ▶ The parameters \mathbf{b} , \mathbf{w} and ξ can be estimated by using an EM algorithm.
- ▶ This approximation of ℓ gets more accurate as dimensionality increases, as the likelihood becomes more Gaussian in form.
- ▶ It is useful to estimate ℓ at the last step of the algorithm by using the *Gauss-Hermite quadrature*.



Parsimonious Model – MLTA

When R , G or D are large, it can be useful to use a response function that involves less parameters.

One more parsimonious model considers the same correlation of the variables within each group. Assuming $\mathbf{w}_{rg} = \mathbf{w}_r$:

$$\pi_{rg}(\mathbf{z}_n) = \frac{1}{1 + \exp(-[b_{rg} + \mathbf{w}_r^T \mathbf{z}_n])}, \quad 0 \leq \pi_{rg}(\mathbf{z}_n) \leq 1.$$



Noordin dataset

Load the noordin dataset which is available in the manet package. For more info:
`?manet::noordin`

```
data(noordin, package = "manet")
```

Set the incidence matrix in the correct format.

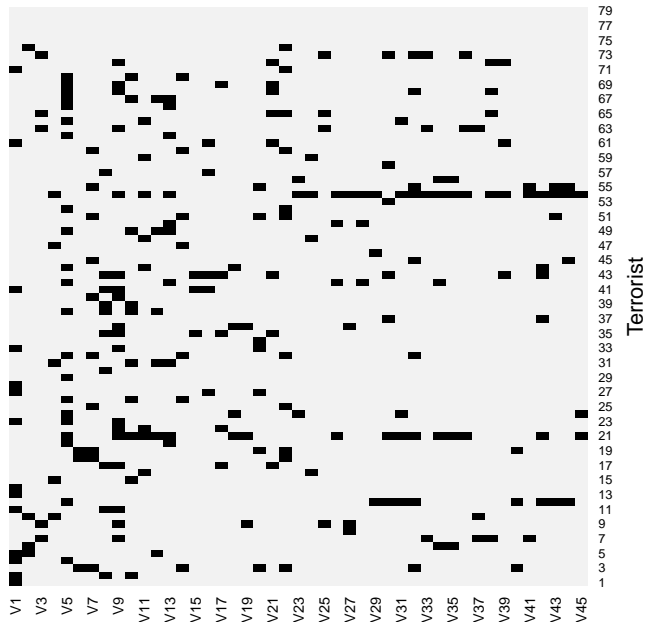
```
Y <- as.matrix(noordin)
```



Heatmap of the data:

```
heatmap(  
  Y,  
  Rowv = NA,  
  Colv = NA,  
  col = grey(c(0.95, 0.0)),  
  scale = "none",  
  margins = c(3, 3),  
  xlab = "Event",  
  ylab = "Terrorist"  
)
```

Heatmap of the data:



MLTA

We want to fit the model on a range of groups G , from 2 to 4 and the latent continuous latent variable takes value D from 0 to 3.

Fit the mlta

```
mod.mlta <- mlta(Y, G = G, D = D, wfix = FALSE)
mod.mlta$BIC$`Table of BIC Results`
```

##		G=2	G=3	G=4
##	dim y=0	2065.400	2162.161	2304.431
##	dim y=1	2142.905	2438.696	2749.910
##	dim y=2	2444.736	2870.997	3400.966
##	dim y=3	2780.816	3446.812	4152.085



MLTA

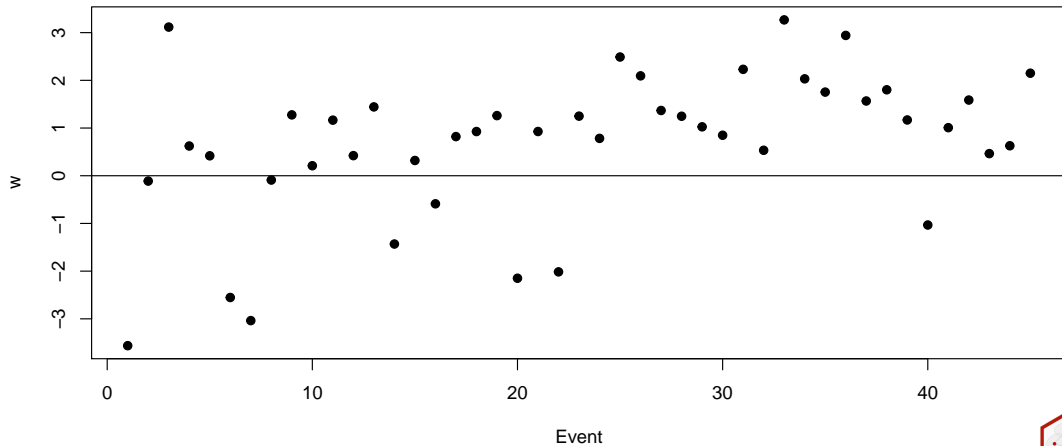
```
mod.mlta.wfix <- mlta(Y, G = G, D = 1:3, wfix = TRUE)
mod.mlta.wfix$BIC$`Table of BIC Results`
```

```
##           G=2      G=3      G=4
## dim y=1 2026.752 2099.796 2234.139
## dim y=2 2126.363 2232.359 2401.785
## dim y=3 2331.084 2432.718 2624.798
```

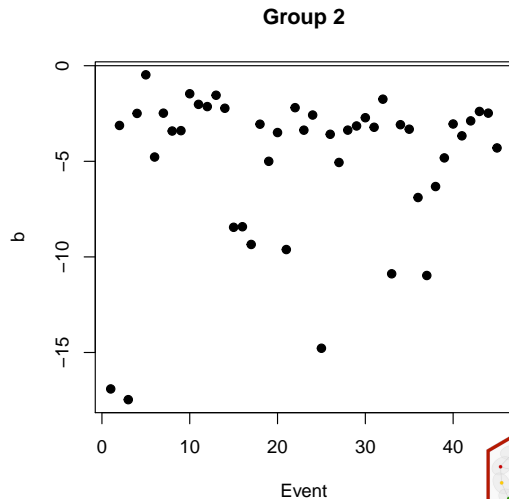
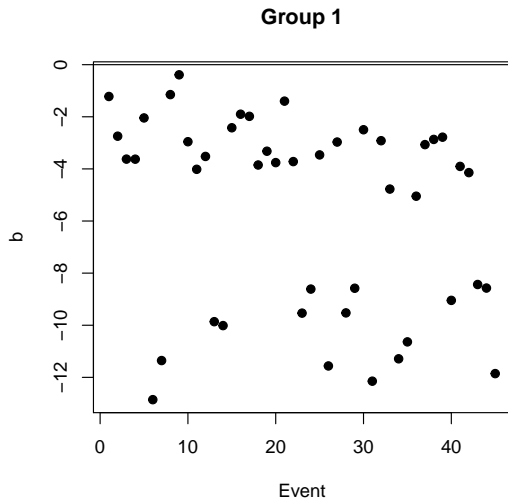
According to the BIC the best model selected is has two groups ($G = 2$) and a one dimensional continuous latent variable ($D = 1$) and common slope parameters across groups ($wfix = TRUE$).



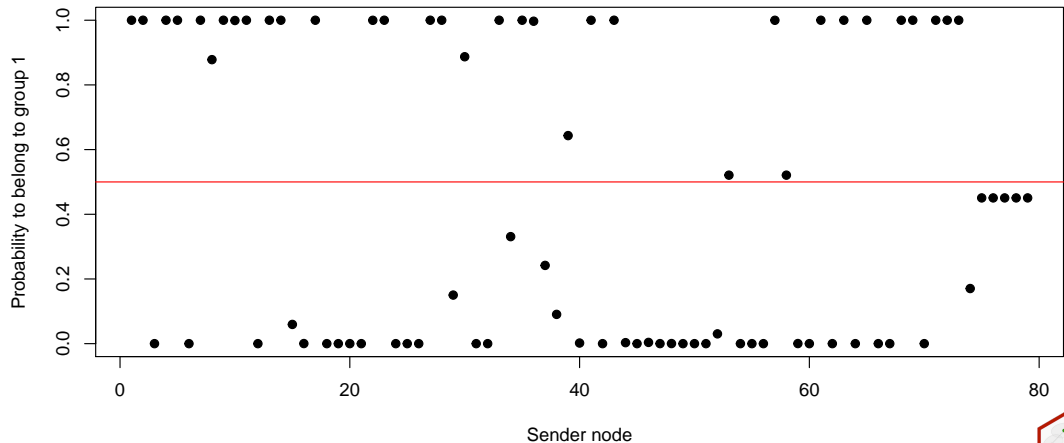
Best model



Plot b parameters

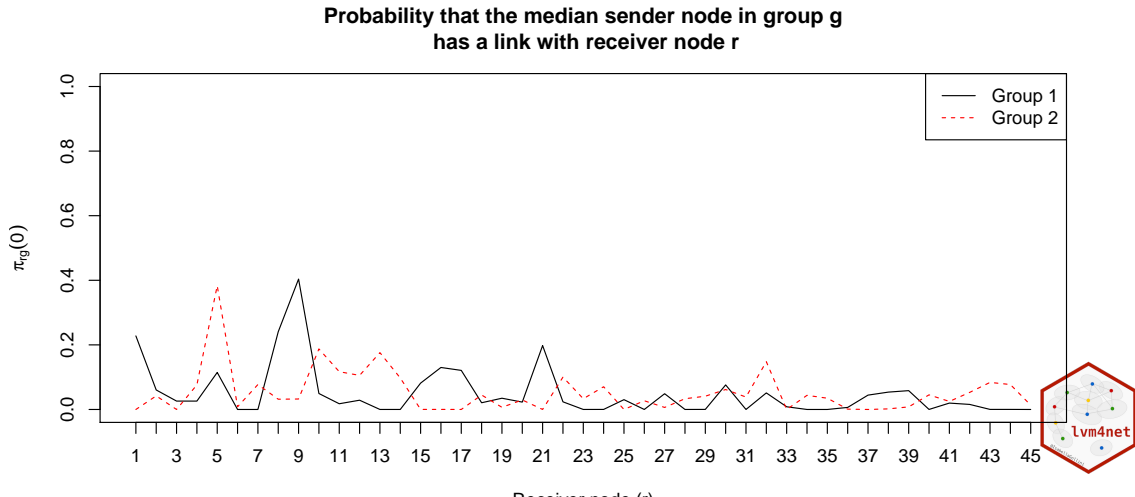


Plot the probability for each sender node to belong to group 1.



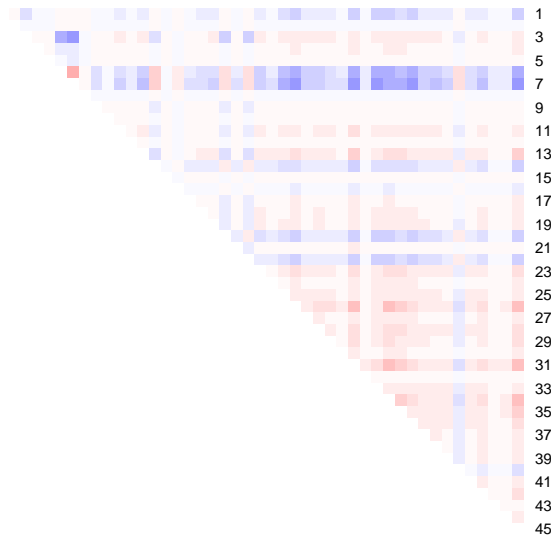
$$\pi_{gr}(0)$$

Find and plot the probability that the median sender node in group g has a link with receiver node r .

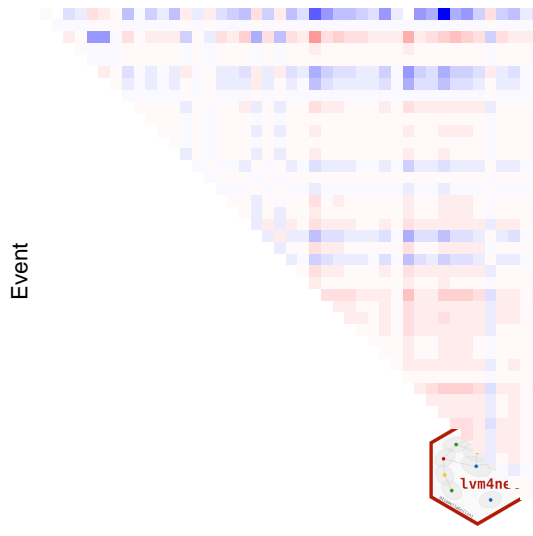


Log-lift for the best model selected.

Log-Lift for Group 1



Log-Lift for Group 2



LCA and LTA

When $D = 0$ the `mlta` reduces to the latent class analysis that can be fitted by using `lca`.

```
mod.lca <- lca(Y, G = 2:4)
```

When there are no groups ($G = 1$) and $D > 0$ the `mlta` reduces to the latent trait analysis that can be fitted by using `lta`

```
mod.lta <- lta(Y, D = 1:3)
```



Your turn!!

- ▶ Reproduce the analysis in the slides,
- ▶ Now use the davis Southern Women Data Set, see `?latentnet::davis` for more information.

```
data(davis, package = "latentnet")  
Y <- network::as.matrix.network(davis)
```



References

<http://igollini.github.io/lvm4net/>

- ▶ Latent variable Models for One-mode Networks:
 - ▶ Hoff, P., Raftery, A., and Handcock, M. (2002), “Latent Space Approaches to Social Network Analysis”, *Journal of the American Statistical Association*, 97, 1090–1098.
 - ▶ Gollini, I., and Murphy, T. B. (2016), “Joint Modelling of Multiple Network Views”, *Journal of Computational and Graphical Statistics*, arXiv:1301.3759.
- ▶ Latent variable Models for Two-mode Networks:
 - ▶ Aitkin, M., Vu, D., Francis, B.: “Statistical modelling of the group structure of social networks” *Social Networks* 38, 74–87 (2014)
 - ▶ Gollini, I. (in press) “A mixture model approach for clustering bipartite networks”, *Challenges in Social Network Research Volume in the Lecture Notes in Social Networks* (LNSN - Series of Springer). Preprint: arXiv:1905.02659.
 - ▶ Gollini, I., and Murphy, T. B. (2014), “Mixture of Latent Trait Analyzers for Model-Based Clustering of Categorical Data”, *Statistics and Computing*, 24(4), 569-588, arXiv:1301.2167.

