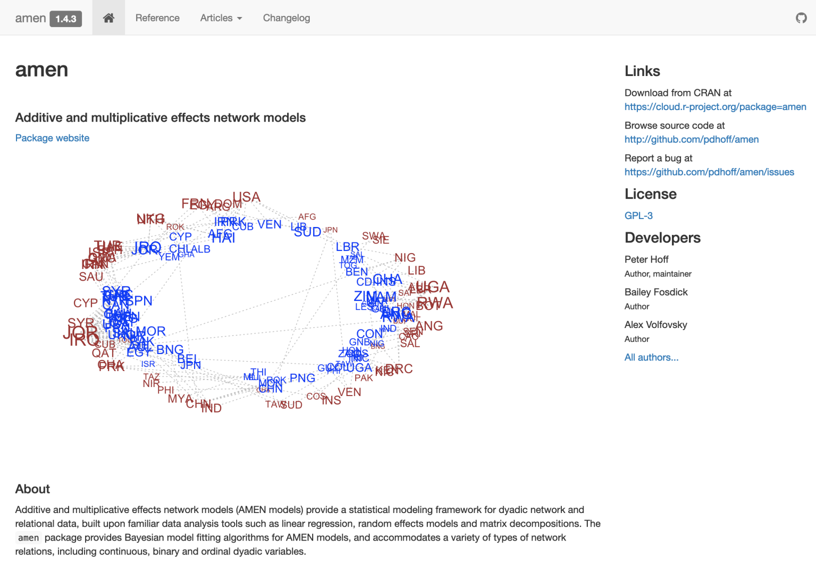
AMEN for network modeling

Peter Hoff
Duke University



amen

Additive and multiplicative effects network models

Package website

Links

Download from CRAN at
https://cloud.r-project.org/package=amen

Browse source code at
http://github.com/pdhoff/amen

Report a bug at
https://github.com/pdhoff/amen/issues

License

GPL-3

Developers

Peter Hoff
Author, maintainer

Bailey Fosdick
Author

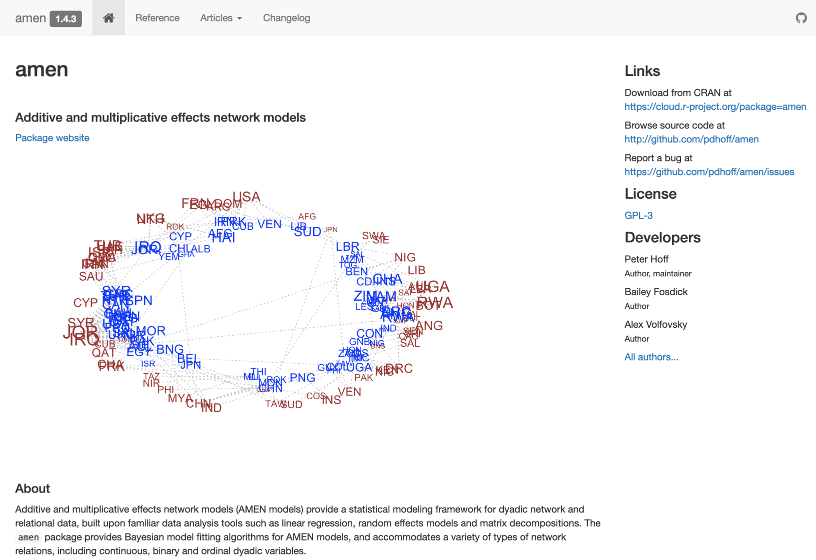
Alex Volfovsky
Author

All authors...

About

Additive and multiplicative effects network models (AMEN models) provide a statistical modeling framework for dyadic network and relational data, built upon familiar data analysis tools such as linear regression, random effects models and matrix decompositions. The `amen` package provides Bayesian model fitting algorithms for AMEN models, and accommodates a variety of types of network relations, including continuous, binary and ordinal dyadic variables.

- "Dyadic data analysis with amen" (50 page tutorial and user manual)
- "Additive and multiplicative effects network models" (review, in press)



amen

Additive and multiplicative effects network models

Package website

Links

Download from CRAN at
https://cloud.r-project.org/package=amen

Browse source code at
http://github.com/pdhoff/amen

Report a bug at
https://github.com/pdhoff/amen/issues

License

GPL-3

Developers

Peter Hoff
Author, maintainer

Bailey Fosdick
Author

Alex Volfovsky
Author

All authors...

About

Additive and multiplicative effects network models (AMEN models) provide a statistical modeling framework for dyadic network and relational data, built upon familiar data analysis tools such as linear regression, random effects models and matrix decompositions. The `amen` package provides Bayesian model fitting algorithms for AMEN models, and accommodates a variety of types of network relations, including continuous, binary and ordinal dyadic variables.

- "Dyadic data analysis with amen" (50 page tutorial and user manual)
- "Additive and multiplicative effects network models" (review, in press)

Goals of statistical network analysis

$y_{i,j}$ measures $i \rightarrow j$, $\mathbf{x}_{i,j}$ is a vector of explanatory variables.

$$\mathbf{Y} = \begin{pmatrix} y_{1,1} & y_{1,2} & y_{1,3} & \text{NA} & y_{1,5} & \cdots \\ y_{2,1} & y_{2,2} & y_{2,3} & y_{2,4} & y_{2,5} & \cdots \\ y_{3,1} & \text{NA} & y_{3,3} & y_{3,4} & \text{NA} & \cdots \\ y_{4,1} & y_{4,2} & y_{4,3} & y_{4,4} & y_{4,5} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{1,2} & \mathbf{x}_{1,3} & \mathbf{x}_{1,4} & \mathbf{x}_{1,5} & \cdots \\ \mathbf{x}_{2,1} & \mathbf{x}_{2,2} & \mathbf{x}_{2,3} & \mathbf{x}_{2,4} & \mathbf{x}_{2,5} & \cdots \\ \mathbf{x}_{3,1} & \mathbf{x}_{3,2} & \mathbf{x}_{3,3} & \mathbf{x}_{3,4} & \mathbf{x}_{3,5} & \cdots \\ \mathbf{x}_{4,1} & \mathbf{x}_{4,2} & \mathbf{x}_{4,3} & \mathbf{x}_{4,4} & \mathbf{x}_{4,5} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}$$

Data analysis goals:

- uncover "hidden structure"
- impute missing values
- evaulate the association between $y_{i,j}$ and $\mathbf{x}_{i,j}$.

Goals of statistical network analysis

$y_{i,j}$ measures $i \rightarrow j$, $\mathbf{x}_{i,j}$ is a vector of explanatory variables.

$$\mathbf{Y} = \begin{pmatrix} y_{1,1} & y_{1,2} & y_{1,3} & \text{NA} & y_{1,5} & \cdots \\ y_{2,1} & y_{2,2} & y_{2,3} & y_{2,4} & y_{2,5} & \cdots \\ y_{3,1} & \text{NA} & y_{3,3} & y_{3,4} & \text{NA} & \cdots \\ y_{4,1} & y_{4,2} & y_{4,3} & y_{4,4} & y_{4,5} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{1,2} & \mathbf{x}_{1,3} & \mathbf{x}_{1,4} & \mathbf{x}_{1,5} & \cdots \\ \mathbf{x}_{2,1} & \mathbf{x}_{2,2} & \mathbf{x}_{2,3} & \mathbf{x}_{2,4} & \mathbf{x}_{2,5} & \cdots \\ \mathbf{x}_{3,1} & \mathbf{x}_{3,2} & \mathbf{x}_{3,3} & \mathbf{x}_{3,4} & \mathbf{x}_{3,5} & \cdots \\ \mathbf{x}_{4,1} & \mathbf{x}_{4,2} & \mathbf{x}_{4,3} & \mathbf{x}_{4,4} & \mathbf{x}_{4,5} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}$$

Data analysis goals:

- uncover "hidden structure"
- impute missing values
- evaulate the association between $y_{i,j}$ and $\mathbf{x}_{i,j}$.

ANOVA and SRM

ANOVA for networks: an additive matrix decomposition.

$$\mathbf{Y} \sim \mu \mathbf{1}\mathbf{1}^\top + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mu + a_i + b_j + e_{i,j}$$

Social relations covariance model: an additive random effects model

$$\text{Var}[\left(\begin{smallmatrix} a_i \\ b_i \end{smallmatrix}\right)] = \Sigma = \begin{pmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{pmatrix} \quad \text{Var}[\left(\begin{smallmatrix} \epsilon_{i,j} \\ \epsilon_{j,i} \end{smallmatrix}\right)] = \sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}, \qquad (1)$$

This implies:

$$\text{Cov}[y_{i,j}, y_{i,k}] = \sigma_a^2 \qquad \text{(within-row covariance)}$$

$$\text{Cov}[y_{i,j}, y_{k,j}] = \sigma_b^2 \qquad \text{(within-column covariance)}$$

$$\text{Cov}[y_{i,j}, y_{j,k}] = \sigma_{ab} \qquad \text{(row-column covariance)}$$

$$\text{Cov}[y_{i,j}, y_{j,i}] = 2\sigma_{ab} + \rho\sigma^2 \qquad \text{(row-column covariance plus reciprocity)}$$

- “Degree corrected” - since 1979 [WKS], 1981 [HL]
- Uncovers hidden structure via $\{(a_i, b_i) : i = 1, \ldots, n\}$ - in a limited way.
- Does not evaluate associations with nodal or dyadic covariates.

ANOVA and SRM

ANOVA for networks: an additive matrix decomposition.

$$\mathbf{Y} \sim \mu \mathbf{1}\mathbf{1}^\top + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mu + a_i + b_j + e_{i,j}$$

Social relations covariance model: an additive random effects model

$$\text{Var}[(\begin{smallmatrix} a_i \\ b_i \end{smallmatrix})] = \Sigma = \begin{pmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{pmatrix} \quad \text{Var}[(\begin{smallmatrix} \epsilon_{i,j} \\ \epsilon_{j,i} \end{smallmatrix})] = \sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}, \tag{1}$$

This implies:

$$\text{Cov}[y_{i,j}, y_{i,k}] = \sigma_a^2 \qquad \text{(within-row covariance)}$$

$$\text{Cov}[y_{i,j}, y_{k,j}] = \sigma_b^2 \qquad \text{(within-column covariance)}$$

$$\text{Cov}[y_{i,j}, y_{j,k}] = \sigma_{ab} \qquad \text{(row-column covariance)}$$

$$\text{Cov}[y_{i,j}, y_{j,i}] = 2\sigma_{ab} + \rho\sigma^2 \qquad \text{(row-column covariance plus reciprocity)}$$

- "Degree corrected" - since 1979 [WKS], 1981 [HL]
- Uncovers hidden structure via $\{(a_i, b_i) : i = 1, \ldots, n\}$ - in a limited way.
- Does not evaluate associations with nodal or dyadic covariates.

ANOVA and SRM

ANOVA for networks: an additive matrix decomposition.

$$\mathbf{Y} \sim \mu\mathbf{1}\mathbf{1}^\top + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$
$$y_{i,j} \sim \mu + a_i + b_j + e_{i,j}$$

Social relations covariance model: an additive random effects model

$$\text{Var}[(\begin{smallmatrix} a_i \\ b_i \end{smallmatrix})] = \Sigma = \begin{pmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{pmatrix} \quad \text{Var}[(\begin{smallmatrix} \epsilon_{i,j} \\ \epsilon_{j,i} \end{smallmatrix})] = \sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}, \tag{1}$$

This implies:

$$\begin{aligned}
\text{Cov}[y_{i,j}, y_{i,k}] &= \sigma_a^2 && \text{(within-row covariance)} \\
\text{Cov}[y_{i,j}, y_{k,j}] &= \sigma_b^2 && \text{(within-column covariance)} \\
\text{Cov}[y_{i,j}, y_{j,k}] &= \sigma_{ab} && \text{(row-column covariance)} \\
\text{Cov}[y_{i,j}, y_{j,i}] &= 2\sigma_{ab} + \rho\sigma^2 && \text{(row-column covariance plus reciprocity)}
\end{aligned}$$

- "Degree corrected" - since 1979 [WKS], 1981 [HL]
- Uncovers hidden structure via $\{(a_i, b_i) : i = 1, \ldots, n\}$ - in a limited way.
- Does not evaluate associations with nodal or dyadic covariates.

ANOVA and SRM

ANOVA for networks: an additive matrix decomposition.

$$\mathbf{Y} \sim \mu \mathbf{1}\mathbf{1}^\top + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mu + a_i + b_j + e_{i,j}$$

Social relations covariance model: an additive random effects model

$$\mathrm{Var}[\left(\begin{smallmatrix} a_i \\ b_i \end{smallmatrix} \right)] = \Sigma = \begin{pmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{pmatrix} \quad \mathrm{Var}[\left(\begin{smallmatrix} \epsilon_{i,j} \\ \epsilon_{j,i} \end{smallmatrix} \right)] = \sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}, \tag{1}$$

This implies:

$$\begin{aligned}
\mathrm{Cov}[y_{i,j}, y_{i,k}] &= \sigma_a^2 && \text{(within-row covariance)} \\
\mathrm{Cov}[y_{i,j}, y_{k,j}] &= \sigma_b^2 && \text{(within-column covariance)} \\
\mathrm{Cov}[y_{i,j}, y_{j,k}] &= \sigma_{ab} && \text{(row-column covariance)} \\
\mathrm{Cov}[y_{i,j}, y_{j,i}] &= 2\sigma_{ab} + \rho\sigma^2 && \text{(row-column covariance plus reciprocity)}
\end{aligned}$$

- "Degree corrected" - since 1979 [WKS], 1981 [HL]
- Uncovers hidden structure via $\{(a_i, b_i) : i = 1, \ldots, n\}$ - in a limited way.
- Does not evaluate associations with nodal or dyadic covariates.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$
$$y_{i,j} \sim \mathbf{x}_{i,j}^{\top} \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$
$$y_{i,j} \sim \mathbf{x}_{i,j}^{\top} \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$
$$y_{i,j} \sim \mathbf{x}_{i,j}^{\top} \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^{\top} \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$
$$y_{i,j} \sim \mathbf{x}_{i,j}^{\top} \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

Linear and generalized linear models

(generalized) Linear model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{E}$$
$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + e_{i,j}$$

Networks typically shows evidence AGAINST independence of $\{e_{i,j} : i \neq j\}$.

Not accounting for dependence can lead to

- biased effect estimation;
- uncalibrated confidence intervals;
- poor predictive performance;
- inaccurate description of network phenomenon.

SRRM

Social relations regression model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \epsilon_{i,j}$$

SRRM represents certain features of network data:

- linear associations with predictor variables;
- row heterogeneity;
- column heterogeneity;
- within-dyad similarity.

i.e. means, variances and covariances - first and second order moments.

SRRM

Social relations regression model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \epsilon_{i,j}$$

SRRM represents certain features of network data:

- linear associations with predictor variables;
- row heterogeneity;
- column heterogeneity;
- within-dyad similarity.

i.e. means, variances and covariances - first and second order moments.

SRRM

Social relations regression model:

$$\mathbf{Y} \sim \langle \mathbf{X}, \boldsymbol{\beta} \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \epsilon_{i,j}$$

SRRM represents certain features of network data:

- linear associations with predictor variables;
- row heterogeneity;
- column heterogeneity;
- within-dyad similarity.

i.e. means, variances and covariances - first and second order moments.

```
Y[1:5,1:5]

##           ARG        AUL       BEL        BNG        BRA
## ARG        NA 0.05826891 0.2468601 0.03922071 1.76473080
## AUL 0.0861777         NA 0.3784364 0.10436002 0.21511138
## BEL 0.2700271 0.35065687        NA 0.01980263 0.39877612
## BNG 0.0000000 0.01980263 0.1222176         NA 0.01980263
## BRA 1.6937791 0.23901690 0.6205765 0.03922071         NA
```

```
dim(Xd)

## [1] 30 30  4

dimnames(Xd)

## [[1]]
##  [1] "ARG" "AUL" "BEL" "BNG" "BRA" "CAN" "CHN" "COL" "EGY" "FRN" "IND"
## [12] "INS" "IRN" "ITA" "JPN" "MEX" "NTH" "PAK" "PHI" "POL" "ROK" "SAF"
## [23] "SAU" "SPN" "SWD" "TAW" "THI" "TUR" "UKG" "USA"
##
## [[2]]
##  [1] "ARG" "AUL" "BEL" "BNG" "BRA" "CAN" "CHN" "COL" "EGY" "FRN" "IND"
## [12] "INS" "IRN" "ITA" "JPN" "MEX" "NTH" "PAK" "PHI" "POL" "ROK" "SAF"
## [23] "SAU" "SPN" "SWD" "TAW" "THI" "TUR" "UKG" "USA"
##
## [[3]]
## [1] "conflicts"   "logdist"     "shared_igos" "polity_int"
```

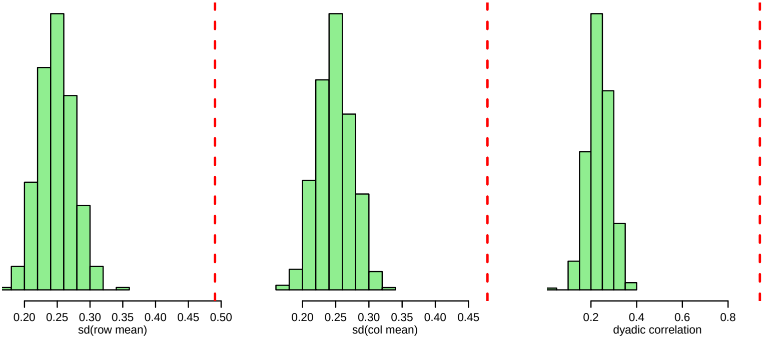
```
fit<-ame(Y, Xd, rvar=FALSE, cvar=FALSE, dcor=FALSE)
```

```
summary(fit)

##
## Regression coefficients:
##                   pmean   psd z-stat p-val
## intercept         0.283 0.146  1.937 0.053
## conflicts.dyad    0.507 0.093  5.423 0.000
## logdist.dyad     -0.204 0.047 -4.366 0.000
## shared_igos.dyad  0.017 0.002  9.931 0.000
## polity_int.dyad   0.001 0.000  2.296 0.022
##
## Variance parameters:
##     pmean   psd
## va  0.000 0.000
## cab 0.000 0.000
## vb  0.000 0.000
## rho 0.000 0.000
## ve  0.525 0.026
```

Posterior predictive checks

For a given summary statistic $g()$

1. simulate $\mathbf{Y}_{sim} \sim p(\mathbf{Y}_{sim}|\mathbf{Y}_{obs}) = \int p(\mathbf{Y}_{sim}|\theta)p(d\theta|\mathbf{Y}_{obs})$
2. compare $g(\mathbf{Y}_{sim})$ to $g(\mathbf{Y}_{obs})$.



```
fit_srm<-ame(Y, Xd)
```

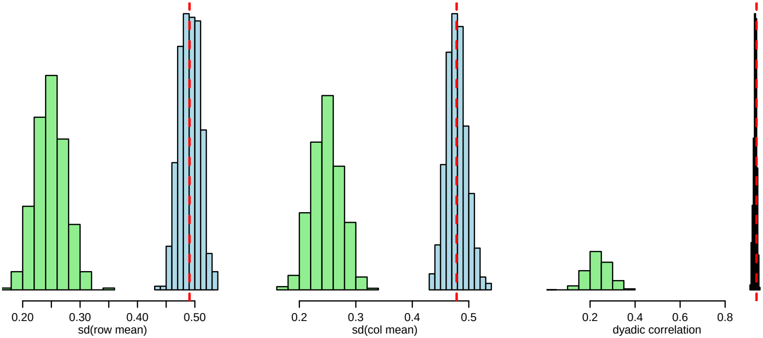
```
summary(fit_srm)

##
## Regression coefficients:
##                     pmean   psd z-stat p-val
## intercept           0.425 0.239  1.779 0.075
## conflicts.dyad      0.064 0.041  1.549 0.121
## logdist.dyad       -0.318 0.044 -7.195 0.000
## shared_igos.dyad    0.021 0.003  8.038 0.000
## polity_int.dyad    -0.002 0.001 -2.662 0.008
##
## Variance parameters:
##     pmean   psd
## va  0.267 0.070
## cab 0.217 0.063
## vb  0.247 0.070
## rho 0.749 0.022
## ve  0.134 0.009
```

`conflicts.dyad` is no longer significant.

Posterior predictive checks

For a given summary statistic $g()$,

1. simulate $\mathbf{Y}_{sim} \sim p(\mathbf{Y}_{sim}|\mathbf{Y}_{obs}) = \int p(\mathbf{Y}_{sim}|\theta)p(d\theta|\mathbf{Y}_{obs})$
2. compare $g(\mathbf{Y}_{sim})$ to $g(\mathbf{Y}_{obs})$.



Limitations of additive effects models

SRM represents certain features of network data

- row heterogeneity
- column heterogeneity
- within-dyad similarity

i.e. variances and covariances - second order effects.

What about other often-observed features of networks?

- clustering
- transitivity
- communities

$$t(\mathbf{Y}) = \sum_{i<j<k} y_{i,j} y_{j,k} y_{k,i}$$

Limitations of additive effects models

SRM represents certain features of network data

- row heterogeneity
- column heterogeneity
- within-dyad similarity

i.e. variances and covariances - second order effects.

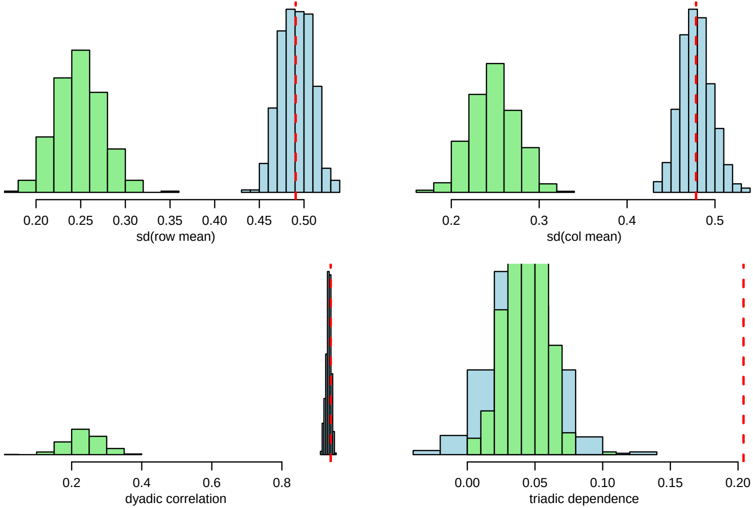
What about other often-observed features of networks?

- clustering
- transitivity
- communities

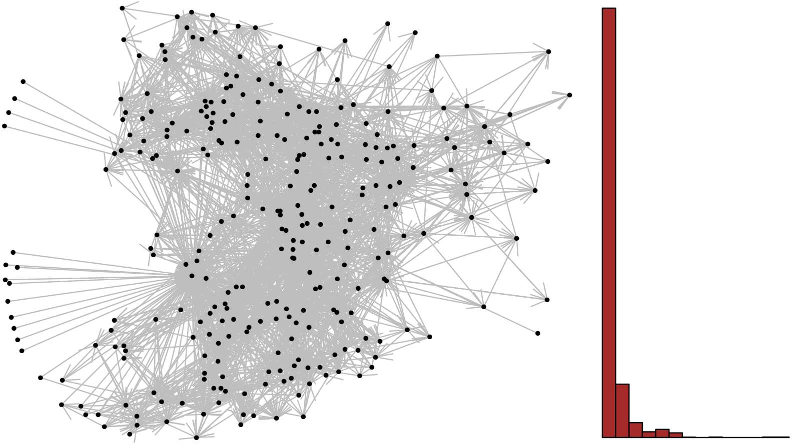
$$t(\mathbf{Y}) = \sum_{i<j<k} y_{i,j} y_{j,k} y_{k,i}$$

Trade data - fit statistics

```
fit_glm<-ame(Y,Xd,rvar=FALSE,cvar=FALSE,dcor=FALSE)

fit_srm<-ame(Y,Xd)
```

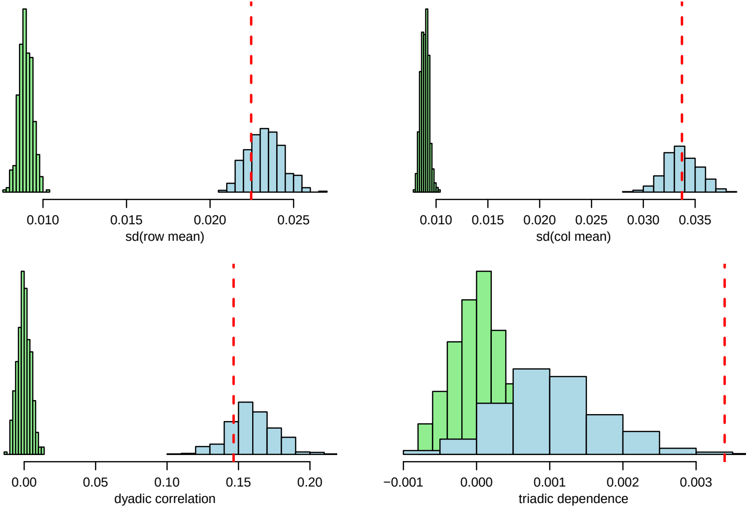


Elegans neural network

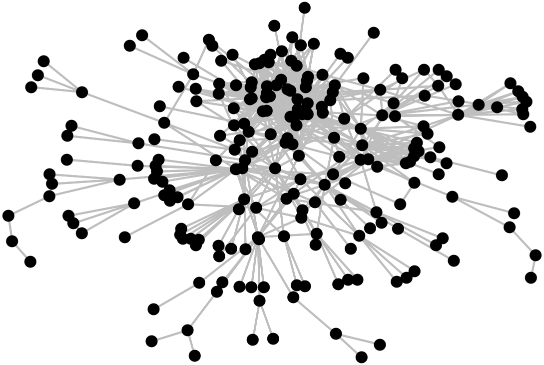


Elegans data - fit statistics

```
fit_glm<-ame(Y_elegans,model="ord",rvar=FALSE,cvar=FALSE,dcor=FALSE)

fit_srm<-ame(Y_elegans,model="ord")
```

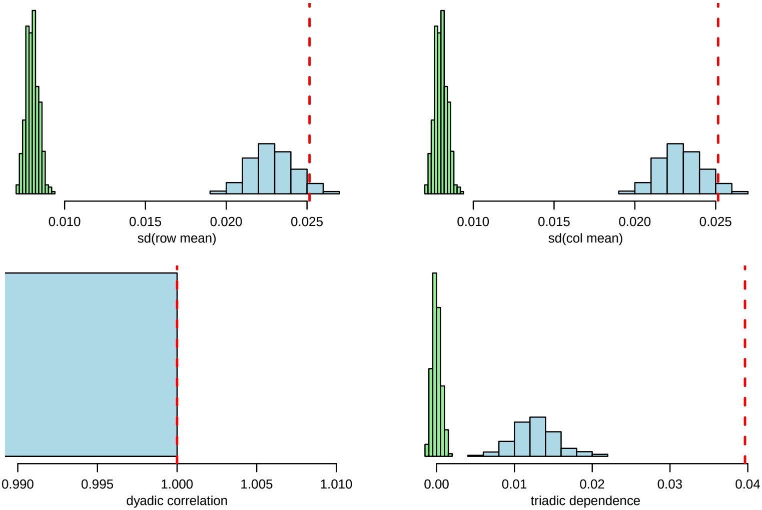


Protein binding data



Protein binding data - fit statistics

```
fit_glm<-ame(Y,model="bin",sym=TRUE,nvar=FALSE)

fit_srm<-ame(Y,model="bin",sym=TRUE)
```



AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds "hidden structure" via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the "degree-corrected" blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds “hidden structure” via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the “degree-corrected” blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds "hidden structure" via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the "degree-corrected" blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds "hidden structure" via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the "degree-corrected" blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds “hidden structure” via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the “degree-corrected” blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds “hidden structure” via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the “degree-corrected” blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

AMEN models

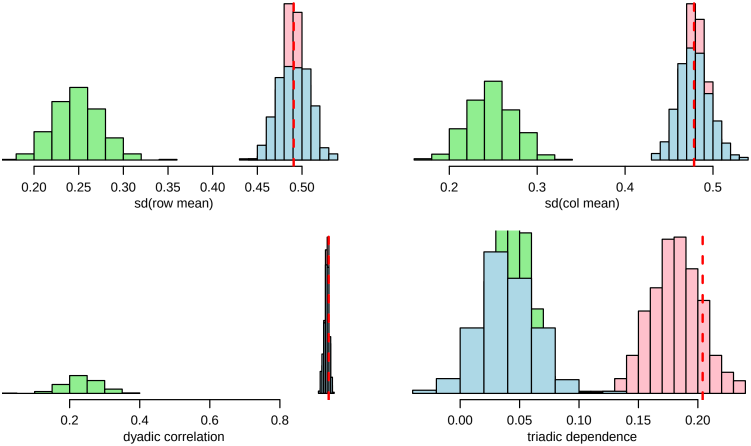
$$\mathbf{Y} \sim \langle \mathbf{X}, \beta \rangle + \mathbf{a}\mathbf{1}^\top + \mathbf{1}\mathbf{b}^\top + \mathbf{U}\mathbf{V}^\top + \mathbf{E}$$

$$y_{i,j} \sim \mathbf{x}_{i,j}^\top \boldsymbol{\beta} + a_i + b_j + \mathbf{u}_i^\top \mathbf{v}_j + \epsilon_{i,j}$$

- Finds "hidden structure" via (a_i, b_i) and $(\mathbf{u}_i, \mathbf{v}_j)$;
- Generalizes the "degree-corrected" blockmodel and distance model;
- Estimates association between network outcomes and predictor variables;
- Accounts for network correlations;
- Permits imputation of missing values;
- Computationally intensive, especially for non-Gaussian outcomes.

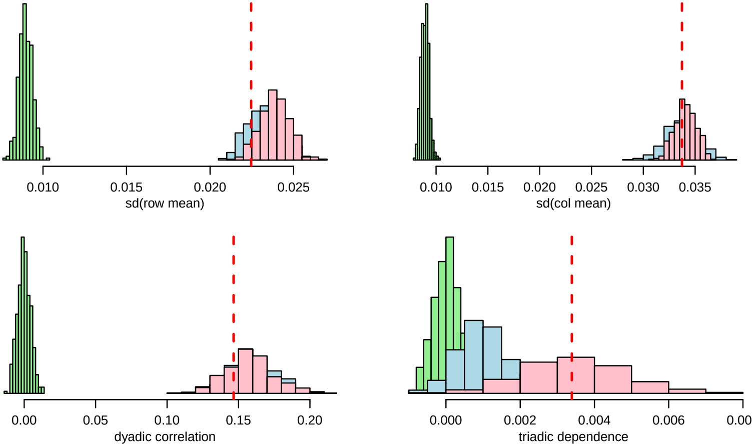
Do AMEN models fit? Trade data fit statistics

```
fit_glm<-ame(Y,Xd,rvar=FALSE,cvar=FALSE,dcor=FALSE)

fit_srm<-ame(Y,Xd)

fit_ame<-ame(Y,Xd,R=2)
```



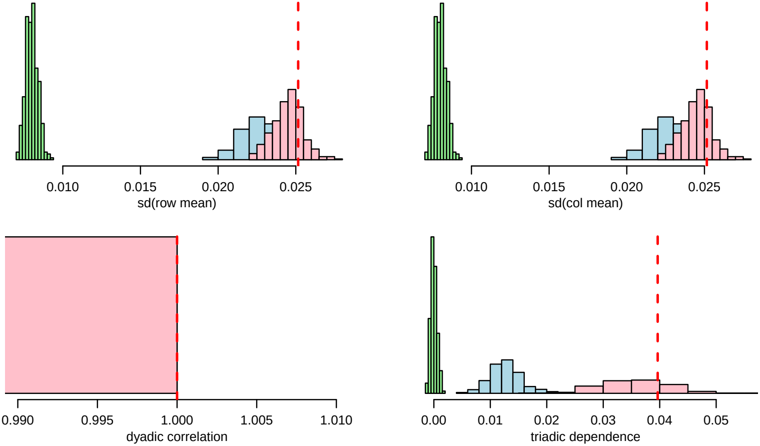
Do AMEN models fit? Elegans data fit statistics

```
fit_glm<-ame(Y_elegans,family="ord",rvar=FALSE,cvar=FALSE,dcor=FALSE)

fit_srm<-ame(Y_elegans,family="ord")

fit_ame<-ame(Y_elegans,family="ord",R=2)
```

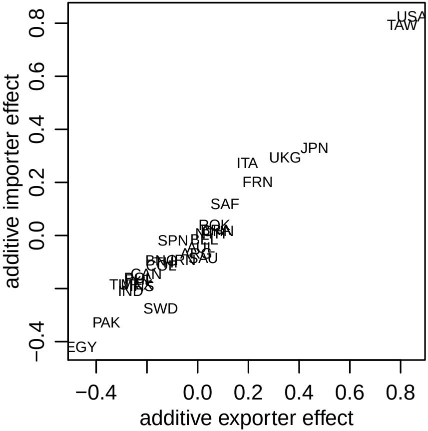


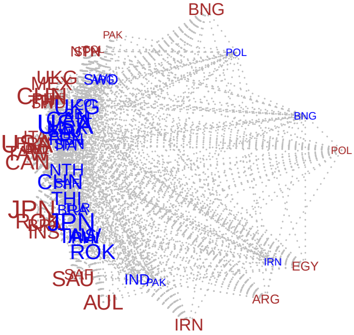
Do AMEN models fit? Protein data fit statistics

```
fit_glm<-ame(Y_ppn,model="bin",sym=TRUE,nvar=FALSE)

fit_srm<-ame(Y_ppn,model="bin",sym=TRUE)

fit_ame<-ame(Y_ppn,model="bin",sym=TRUE,R=2)
```

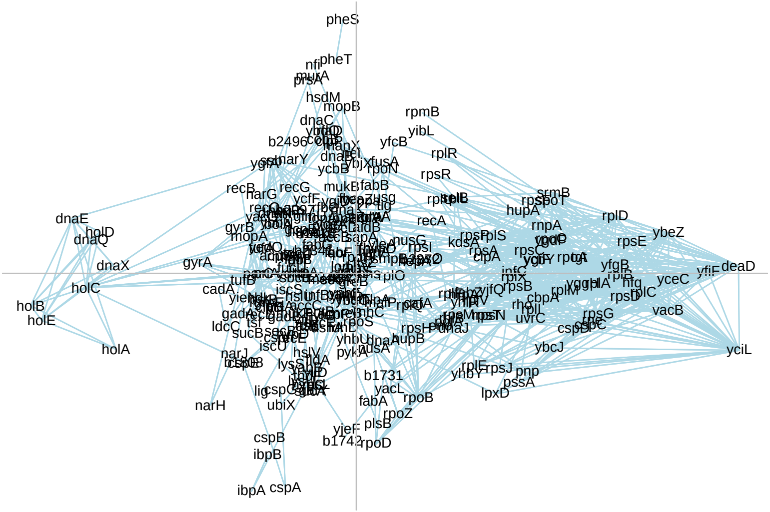


Uncovering structure - trade data





Uncovering structure - protein interaction data



```
AME model fitting routine

Description:

     An MCMC routine providing a fit to an additive and multiplicative
     effects (AME) regression model to relational data of various types

Usage:

     ame(Y, Xdyad=NULL, Xrow=NULL, Xcol=NULL, family, R=0, rvar = !(family=="rrl"),
     cvar=TRUE, dcor=!symmetric, nvar=TRUE, intercept=!is.element(family,c("rrl","ord")),
     symmetric=FALSE,odmax=rep(max(apply(Y>0,1,sum,na.rm=TRUE)),nrow(Y)),
     seed=1, nscan=10000, burn=500, odens=25, plot=TRUE, print=TRUE, gof=TRUE,
     prior=list())

Arguments:

       Y: an n x n square relational matrix of relations. See family
          below for various data types.

   Xdyad: an n x n x pd array of covariates

    Xrow: an n x pr matrix of nodal row covariates

    Xcol: an n x pc matrix of nodal column covariates

  family: character: one of "nrm","bin","ord","cbin","frn","rrl" - see
          the details below

       R: integer: dimension of the multiplicative effects (can be
          zero)
```

Model fitting

Bilinear regression for Gaussian outcomes:

$$y_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$
$$y_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$

- Iterative estimation (Gibbs sampling).
- Correlation between $e_{i,j}$ and $e_{j,i}$ is the hard part.

Transformation models for non-Gaussian outcomes:

$$z_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$
$$y_{i,j} = g(z_{i,j}).$$

- binary, ordinal, censored binary, ranks.
- overdispersed Poisson, "tobit" (sparse+valued).

Model fitting

Bilinear regression for Gaussian outcomes:

$$y_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$
$$y_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$

- Iterative estimation (Gibbs sampling).
- Correlation between $e_{i,j}$ and $e_{j,i}$ is the hard part.

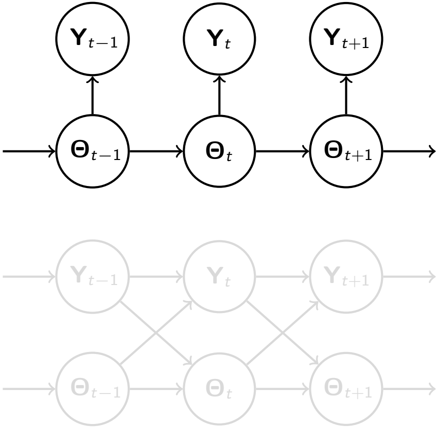
Transformation models for non-Gaussian outcomes:

$$z_{i,j} = \beta^\top x_{i,j} + a_i + b_j + u_i{}^\top v_j + e_{i,j}$$
$$y_{i,j} = g(z_{i,j}).$$

- binary, ordinal, censored binary, ranks.
- overdispersed Poisson, "tobit" (sparse+valued).

Extensibility - dynamic networks

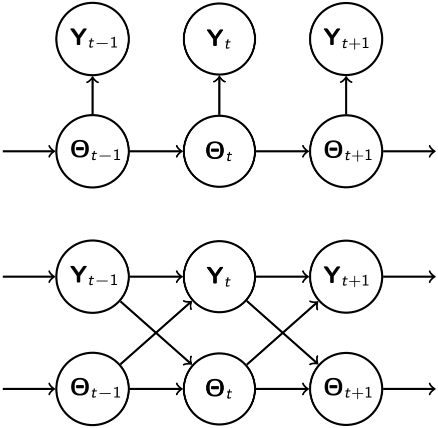
Let $\mathbf{\Theta}$ include one or more of $\boldsymbol{\beta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}$.



(He and Hoff [2019])

Extensibility - dynamic networks

Let $\mathbf{\Theta}$ include one or more of $\boldsymbol{\beta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}$.

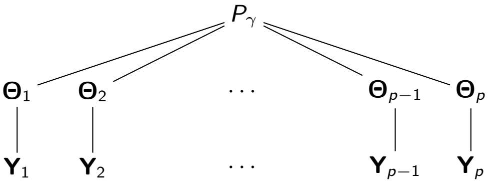


(He and Hoff [2019])

Extensibility - multilevel networks

Share information/describe heterogeneity across

- social networks of multiple schools;
- brain networks of multple patients.



$$\mathbf{Y}_j \sim AMEN(\mathbf{Y}|\mathbf{\Theta}_j)$$

$$\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_p \sim \text{i.i.d. } P_\gamma, \text{ some } \gamma \in \Gamma$$

Summary

Many network patterns can be described by node-specific latent variables:

- additive random effects for second-order dependencies;
- multiplicative factors for higher-order patterns.

Multiplicative effects generalize blockmodels and distance models.

AMEN framework:

- covariate effects + SRM + LFM/low-rank interaction;
- implemented in **amen** software package (CRAN, github);
- 50 page tutorial availalble (arXiv, github);
- review article in press/on arXiv with technical details.

Work to do:

- Study alternatives to MCMC;
- Extend modular model fitting routine to other network scenarios.

Summary

Many network patterns can be described by node-specific latent variables:

- additive random effects for second-order dependencies;
- multiplicative factors for higher-order patterns.

Multiplicative effects generalize blockmodels and distance models.

AMEN framework:

- covariate effects + SRM + LFM/low-rank interaction;
- implemented in **amen** software package (CRAN, github);
- 50 page tutorial availalble (arXiv, github);
- review article in press/on arXiv with technical details.

Work to do:

- Study alternatives to MCMC;
- Extend modular model fitting routine to other network scenarios.

Summary

Many network patterns can be described by node-specific latent variables:

- additive random effects for second-order dependencies;
- multiplicative factors for higher-order patterns.

Multiplicative effects generalize blockmodels and distance models.

AMEN framework:

- covariate effects + SRM + LFM/low-rank interaction;
- implemented in **amen** software package (CRAN, github);
- 50 page tutorial availalble (arXiv, github);
- review article in press/on arXiv with technical details.

Work to do:

- Study alternatives to MCMC;
- Extend modular model fitting routine to other network scenarios.