

Package ‘MEclustnet’

October 13, 2016

Type Package

Title Fits the Mixture of Experts Latent Position Cluster Model to Network Data

Version 1.0

Date 2016-10-11

Author Isobel Claire Gormley

Maintainer Isobel Claire Gormley <claire.gormley@ucd.ie>

Description

Fits the mixture of experts latent position cluster model to network data to cluster nodes into sub-groups, while incorporating covariate information, in a mixture of experts model setting.

Imports MASS, mvtnorm, mclust, nnet, ellipse, latentnet, vegan, e1071

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2016-10-13 11:30:32

R topics documented:

MEclustnet-package	2
formatting.covars	4
lawyers.adjacency.advice	5
lawyers.adjacency.coworkers	6
lawyers.adjacency.friends	7
lawyers.covariates	7
MEclustnet	8
plotMEclustnet	11
summaryMEclustnet	13
us.twitter.adjacency	15
us.twitter.covariates	15

Index	17
--------------	-----------

MEclustnet-package *Fits the Mixture of Experts Latent Position Cluster Model to Network Data*

Description

Fits the mixture of experts latent position cluster model to network data to cluster nodes into subgroups, while incorporating covariate information, in a mixture of experts model setting.

Details

The DESCRIPTION file:

```
Package:      MEclustnet
Type:        Package
Title:       Fits the Mixture of Experts Latent Position Cluster Model to Network Data
Version:     1.0
Date:       2016-10-11
Author:      Isobel Claire Gormley
Maintainer:  Isobel Claire Gormley <claire.gormley@ucd.ie>
Description: Fits the mixture of experts latent position cluster model to network data to cluster nodes into subgroups, while
Imports:     MASS, mvtnorm, mclust, nnet, ellipse, latentnet, vegan, e1071
License:     GPL-2
```

Index of help topics:

```
MEclustnet          Fit a mixture of experts latent position
                   cluster model to network data.
MEclustnet-package  Fits the Mixture of Experts Latent Position
                   Cluster Model to Network Data
formatting.covars   Reformat matrix of covariates.
lawyers.adjacency.advice
                   Adjacency matrix detailing the presence or
                   absence of advice links between the 'Lazega
                   Lawyers'.
lawyers.adjacency.coworkers
                   Adjacency matrix detailing the presence or
                   absence of coworker links between the 'Lazega
                   Lawyers'.
lawyers.adjacency.friends
                   Adjacency matrix detailing the presence or
                   absence of friendship links between the 'Lazega
                   Lawyers'.
lawyers.covariates  A matrix of covariates of the 'Lazega Lawyers'.
plotMEclustnet     Plot latent position network.
summaryMEclustnet   Summary of MEclustnet object.
```

```

us.twitter.adjacency   Directed adjacency matrix detailing the
                       presence or absence of Twitter friend/follower
                       links between US politicians.
us.twitter.covariates  A matrix of covariates of the US politicians.

```

The main function of interest is MEclustnet which will fit a mixture of experts latent position cluster model to a binary network.

Author(s)

Isobel Claire Gormley

Maintainer: Isobel Claire Gormley <claire.gormley@ucd.ie>

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```

# An example from the Gormley and Murphy (2010) paper, using the Lazega lawyers friendship network.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(lawyers.adjacency.friends)
data(lawyers.covariates)
link.vars = c(1)
mix.vars = c(1,4,5)
## Not run: fit = MEclustnet(lawyers.adjacency.friends, lawyers.covariates,
link.vars, mix.vars, G=2, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$mustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, lawyers.adjacency.friends)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, lawyers.adjacency.friends)
## End(Not run)

# An example analysing a 2016 Twitter network of US politicians.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(us.twitter.adjacency)
data(us.twitter.covariates)
link.vars = c(1)
mix.vars = c(1)
## Not run: fit = MEclustnet(us.twitter.adjacency, us.twitter.covariates,
link.vars, mix.vars, G=4, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$mustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, us.twitter.adjacency)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)

```

```

# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, us.twitter.adjacency)
# Examine which politicians are in which clusters...
clusters = list()
for(g in 1:fit$G)
{
  clusters[[g]] = us.twitter.covariates[summ$Kmode==g,c("name", "party")]
}
clusters
## End(Not run)

```

formatting.covars *Reformat matrix of covariates.*

Description

This function reformats the matrix of input covariates into the required format for the link probabilities and for the mixing proportions.

Usage

```
formatting.covars(covars, link.vars, mix.vars, n)
```

Arguments

<code>covars</code>	The $n \times p$ data frame of node specific covariates passed in to the overall <code>MEclustnet</code> function. The first column should be a column of 1's and categorical variables should be factors.
<code>link.vars</code>	A vector detailing the column numbers of the matrix <code>covars</code> that should be included in the link probabilities model.
<code>mix.vars</code>	A vector detailing the column numbers of the matrix <code>covars</code> that should be included in the mixing proportions probabilities model.
<code>n</code>	The number of nodes in the network.

Details

For the link regression model, the difference in the `link.vars` covariates, for all pairs of nodes is calculated.

For the mixing proportions model, the required representation of the `mix.vars` required is formed, where for categorical/factor variables a dummy value representation is used.

Value

A list with

<code>x.link</code>	A matrix with n^2 rows and <code>length(link.vars)</code> columns, detailing the differences in covariates for all pairs of nodes.
---------------------	--

`x.mix` A matrix with `n` rows and number of columns equal to the number of variables detailed in `mix.vars`, where dummy variable representations will be used for categorical factor covariates.

Author(s)

Isobel Claire Gormley <claire.gormley@ucd.ie>

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

See Also

[MEclustnet](#)

Examples

```
data(us.twitter.covariates)
link.vars = c(1,5,7,8)
mix.vars = c(1,5,7,8)
res = formatting.covars(us.twitter.covariates, link.vars, mix.vars, nrow(us.twitter.covariates))
dim(res$x.link)
dim(res$x.mix)
```

lawyers.adjacency.advice

Adjacency matrix detailing the presence or absence of advice links between the 'Lazega Lawyers'.

Description

Data on whether or not 71 lawyers in a northeastern American law firm asked each other for advice.

Usage

```
data("lawyers.adjacency.advice")
```

Format

A 71 x 71 binary matrix, with 0 down the diagonal.

Source

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

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```
data(lawyers.adjacency.advice)
str(lawyers.adjacency.advice)
```

```
lawyers.adjacency.coworkers
```

Adjacency matrix detailing the presence or absence of coworker links between the 'Lazega Lawyers'.

Description

Data on whether or not 71 lawyers in a northeastern American law firm work with each other.

Usage

```
data("lawyers.adjacency.coworkers")
```

Format

A 71 x 71 binary matrix, with 0 down the diagonal.

Source

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

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```
data(lawyers.adjacency.coworkers)
str(lawyers.adjacency.coworkers)
```

```
lawyers.adjacency.friends
```

Adjacency matrix detailing the presence or absence of friendship links between the 'Lazega Lawyers'.

Description

Data on whether or not 71 lawyers in a northeastern American law firm are friends outside of work.

Usage

```
data("lawyers.adjacency.friends")
```

Format

A 71 x 71 binary matrix, with 0 down the diagonal.

Source

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

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```
data(lawyers.adjacency.friends)  
str(lawyers.adjacency.friends)
```

```
lawyers.covariates
```

A matrix of covariates of the 'Lazega Lawyers'.

Description

Covariates on each of 71 lawyers in a northeastern American law firm. Note the first column is a column of 1's.

Usage

```
data("lawyers.covariates")
```

Format

A data frame with 71 observations on the following 8 variables.

Intercept a column of 1s should always be the first column.

Seniority a factor with levels 1 = partner, 2 = associate.

Gender a factor with 1 = male, 2 = female.

Office a factor with levels 1 = Boston, 2 = Hartford and 3 = Providence

Years a numeric vector detailing years with the firm.

Age a numeric vector detailing the age of each lawyer.

Practice a factor with levels 1 = litigation and 2 = corporate.

School a factor with levels 1 = Harvard or Yale, 2 = University of Connecticut and 3 = Other.

Source

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

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```
data(lawyers.covariates)
head(lawyers.covariates)
```

MEclustnet

Fit a mixture of experts latent position cluster model to network data.

Description

MEclustnet will fit a mixture of experts latent position cluster model to a binary network.

Usage

```
MEclustnet(Y, covars, link.vars = c(1:ncol(covars)), mix.vars = c(1:ncol(covars)),
G = 2, d = 2, itermax = 10000, uphill = 100, burnin = 1000, thin = 10,
rho.input = 1, verbose = TRUE, ...)
```


Arguments

Y	An $n \times n$ binary matrix of links between n nodes, with 0 on the diagonal and 1 indicating a link.
covars	An $n \times p$ data frame of node specific covariates. Categorical variables should be factors. First column should be a column of 1s, and should always be passed in.
link.vars	A vector of the column numbers of the data frame <code>covars</code> to be included in link probability model. If none are to be included, this argument should be 1.
mix.vars	A vector of the column numbers of the data frame <code>covars</code> to be included in mixing proportions model. If none are to be included, argument should be 1.
G	The number of clusters in the model to be fitted.
d	The dimension of the latent space.
itermax	Maximum number of iterations in the MCMC chain.
uphill	Number of iterations for which uphill only steps in the MCMC chain should be run to find <i>maximum a posteriori</i> estimates.
burnin	Number of burnin iterations in the MCMC chain.
thin	The degree of thinning to be applied to the MCMC chain.
rho.input	Scaling factor to achieve desirable acceptance rates in Metropolis-Hastings steps.
verbose	Print progress updates to screen? Recommended as the models are slow to run.
...	Additional arguments.

Details

This function fits the mixture of experts latent position cluster model to a binary network via a Metropolis-within-Gibbs sampler. Covariates can influence either the link probabilities between nodes and/or the cluster memberships of nodes.

Value

An object of class `MEclustnet`, which is a list containing:

zstore	An $n \times d \times \text{store.dim}$ array of sampled latent location matrices, where <code>store.dim</code> is the number of post burnin thinned iterations.
betastore	A $\text{store.dim} \times p$ matrix of sampled beta vectors, the logistic regression parameters of the link probabilities model.
Kstore	A $\text{store.dim} \times n$ matrix of sampled cluster membership vectors.
mustore	A $G \times d \times \text{store.dim}$ array of sampled cluster mean latent location matrices.
sigma2store	A $\text{store.dim} \times G$ matrix of sampled cluster variances.
lambdastore	An $n \times G \times \text{store.dim}$ array of sampled mixing proportion matrices.
taustore	A $G \times s \times \text{store.dim}$ array of sampled tau vectors, the logistic regression parameters of the mixing proportions model, where s is the length of tau.
LLstore	A vector of length <code>store.dim</code> storing the loglikelihood from each stored iteration.
G	The number of clusters fitted

d	The dimension of the latent space
countbeta	Count of accepted beta values
counttau	Count of accepted tau values

Author(s)

Isobel Claire Gormley <claire.gormley@ucd.ie>

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

See Also

[summaryMEclustnet](#), [plotMEclustnet](#)

Examples

```
# An example from the Gormley and Murphy (2010) paper, using the Lazega lawyers friendship network.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(lawyers.adjacency.friends)
data(lawyers.covariates)
link.vars = c(1)
mix.vars = c(1,4,5)
## Not run: fit = MEclustnet(lawyers.adjacency.friends, lawyers.covariates,
link.vars, mix.vars, G=2, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$mustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, lawyers.adjacency.friends)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, lawyers.adjacency.friends)
## End(Not run)

# An example analysing a 2016 Twitter network of US politicians.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(us.twitter.adjacency)
data(us.twitter.covariates)
link.vars = c(1)
mix.vars = c(1)
## Not run: fit = MEclustnet(us.twitter.adjacency, us.twitter.covariates,
link.vars, mix.vars, G=4, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$mustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, us.twitter.adjacency)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
```

```
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, us.twitter.adjacency)
# Examine which politicians are in which clusters...
clusters = list()
for(g in 1:fit$G)
{
  clusters[[g]] = us.twitter.covariates[summ$Kmode==g,c("name", "party")]
}
clusters
## End(Not run)
```

plotMEclustnet *Plot latent position network.*

Description

Function to plot the resulting fitted network, using first two dimensions only.

Usage

```
plotMEclustnet(fit, Y, link.vars, mix.vars)
```

Arguments

fit	An object storing the output of the function MEclustnet .
Y	The $n \times n$ binary adjacency matrix, with 0 down the diagonal, that was passed to MEclustnet .
link.vars	A vector detailing the columns of the covariate matrix which are included in the link probabilities regression model.
mix.vars	A vector detailing the columns of the covariate matrix which are included in the mixing proportions regression model.

Details

This function will plot the posterior mean latent location for each node in the network. The colour of each node reflects the posterior modal cluster membership, and the ellipses are 50% posterior sets illustrating the uncertainty in the latent locations. The grey lines illustrate the observed links between the nodes.

Author(s)

Isobel Claire Gormley <claire.gormley@ucd.ie>

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

See Also[MEclustnet](#)**Examples**

```

# An example from the Gormley and Murphy (2010) paper, using the Lazega lawyers friendship network.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(lawyers.adjacency.friends)
data(lawyers.covariates)
link.vars = c(1)
mix.vars = c(1,4,5)
## Not run: fit = MEclustnet(lawyers.adjacency.friends, lawyers.covariates,
link.vars, mix.vars, G=2, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$ustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, lawyers.adjacency.friends)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, lawyers.adjacency.friends)
## End(Not run)

# An example analysing a 2016 Twitter network of US politicians.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(us.twitter.adjacency)
data(us.twitter.covariates)
link.vars = c(1)
mix.vars = c(1)
## Not run: fit = MEclustnet(us.twitter.adjacency, us.twitter.covariates,
link.vars, mix.vars, G=4, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$ustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, us.twitter.adjacency)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, us.twitter.adjacency)
# Examine which politicians are in which clusters...
clusters = list()
for(g in 1:fit$G)
{
  clusters[[g]] = us.twitter.covariates[summ$Kmode==g,c("name", "party")]
}
clusters

## End(Not run)

```

summaryMEclustnet *Summary of MEclustnet object.*

Description

Summary of the output of the function MEclustnet which fits a mixture of experts latent position cluster model.

Usage

```
summaryMEclustnet(fit, Y)
```

Arguments

fit	An object storing the output of the function <code>MEclustnet</code> .
Y	The $n \times n$ binary adjacency matrix, with 0 down the diagonal, that was passed to <code>MEclustnet</code> .

Value

A list with:

AICM	The value of the AICM criterion for the fitted model.
BICM	The value of the BICM criterion for the fitted model.
BICMCMC	The value of the BICMCMC criterion for the fitted model.
betamean	The posterior mean vector of the regression coefficients for the link probabilities model.
betasd	The standard deviation of the posterior distribution of beta.
taumean	A matrix with G rows, detailing the posterior mean of the regression coefficients for the mixing proportions model.
tausd	The standard deviation of the posterior distribution of tau.
mumean	A $G \times d$ matrix containing the posterior mean of the latent locations' mean.
meansd	The standard deviation of the posterior distribution of mu.
sigma2mean	A vector of length G containing the posterior mean of the latent locations' covariance.
sigma2sd	The standard deviation of the posterior distribution of the latent locations' covariance.
Knode	A vector of length n detailing the posterior modal cluster membership for each node.
zmean	An $n \times d$ matrix containing the posterior mean latent location for each node.

Author(s)

Isobel Claire Gormley <claire.gormley@ucd.ie>

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

See Also

[MEclustnet](#)

Examples

```
# An example from the Gormley and Murphy (2010) paper, using the Lazega lawyers friendship network.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(lawyers.adjacency.friends)
data(lawyers.covariates)
link.vars = c(1)
mix.vars = c(1,4,5)
## Not run: fit = MEclustnet(lawyers.adjacency.friends, lawyers.covariates,
link.vars, mix.vars, G=2, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$ustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, lawyers.adjacency.friends)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, lawyers.adjacency.friends)
## End(Not run)

# An example analysing a 2016 Twitter network of US politicians.
# Iteration etc. are set to low values for illustrative purposes.
# Longer run times are likely to be required to achieve sufficient mixing.
data(us.twitter.adjacency)
data(us.twitter.covariates)
link.vars = c(1)
mix.vars = c(1)
## Not run: fit = MEclustnet(us.twitter.adjacency, us.twitter.covariates,
link.vars, mix.vars, G=4, d=2, itermax = 500, burnin = 50, uphill = 1, thin=10)
# Plot the trace plot of the mean of dimension 1 for each cluster.
matplot(t(fit$ustore[,1,]), type="l")
# Compute posterior summaries
summ = summaryMEclustnet(fit, us.twitter.adjacency)
plot(summ$zmean, col=summ$Kmode, xlab="Dimension 1", ylab="Dimension 2", pch=summ$Kmode)
# Plot the resulting latent space, with uncertainties
plotMEclustnet(fit, us.twitter.adjacency)
# Examine which politicians are in which clusters...
clusters = list()
for(g in 1:fit$G)
{
  clusters[[g]] = us.twitter.covariates[summ$Kmode==g,c("name", "party")]
}
clusters
## End(Not run)
```

us.twitter.adjacency *Directed adjacency matrix detailing the presence or absence of Twitter friend/follower links between US politicians.*

Description

Network data on whether or not 69 US politicians are friends/followers on Twitter.

Usage

```
data("us.twitter.adjacency")
```

Format

A 69 x 69 binary matrix, with 0 down the diagonal.

Source

With thanks to Dr. Derek Greene. School of Computer Science, University College Dublin.

Examples

```
data(us.twitter.adjacency)
str(us.twitter.adjacency)
```

us.twitter.covariates *A matrix of covariates of the US politicians.*

Description

Covariates on each of 69 US politicians. Note the first column is a column of 1's.

Usage

```
data("us.twitter.covariates")
```

Format

A data frame with 69 observations on the following 8 variables.

'1' a column of 1s should always be the first column.

twitter_id Twitter number.

twitter_name Twitter name.

name Actual name.

party a factor with levels Democrat Republican

location a factor with levels detailing location.

role a factor with levels Candidate, Representative and Senator

gender a factor with levels Female and Male

Source

With thanks to Dr. Derek Greene. School of Computer Science, University College Dublin.

References

Isobel Claire Gormley and Thomas Brendan Murphy. (2010) A Mixture of Experts Latent Position Cluster Model for Social Network Data. *Statistical Methodology*, 7 (3), pp.385-405.

Examples

```
data(us.twitter.covariates)  
str(us.twitter.covariates)
```


Index

*Topic **cluster**

formatting.covars, 4
MEclustnet, 8
MEclustnet-package, 2

*Topic **datasets**

lawyers.adjacency.advice, 5
lawyers.adjacency.coworkers, 6
lawyers.adjacency.friends, 7
lawyers.covariates, 7
us.twitter.adjacency, 15
us.twitter.covariates, 15

*Topic **hplot**

plotMEclustnet, 11

*Topic **methods**

formatting.covars, 4
MEclustnet, 8
summaryMEclustnet, 13

*Topic **multivariate**

formatting.covars, 4

*Topic **package**

MEclustnet-package, 2

formatting.covars, 4

lawyers.adjacency.advice, 5
lawyers.adjacency.coworkers, 6
lawyers.adjacency.friends, 7
lawyers.covariates, 7

MEclustnet, 4, 5, 8, 11–14
MEclustnet-package, 2

plotMEclustnet, 10, 11

summaryMEclustnet, 10, 13

us.twitter.adjacency, 15
us.twitter.covariates, 15