# Package ‘hergm’

February 20, 2010

**Version** 1.0

**Date** 2010-19-02

**Title** Hierarchical Exponential-Family Random Graph Models

**Author** Michael Schweinberger `<michael.schweinberger@stat.psu.edu>`

**Maintainer** Michael Schweinberger `<michael.schweinberger@stat.psu.edu>`

**Depends** ergm, gtools, snow

**Suggests**

**Description** The R package ’hergm’ implements Hierarchical Exponential-Family Random Graph Models (HERGMs), which can be used to model a wide range of relational data (networks). ’hergm’ implements both simulation and Bayesian inference.

**License** GPL-3

## R topics documented:

`example` *Example data set*

### Description

Example data set: synthetic, undirected network with 15 nodes.

### Usage

```
data(example)
```

### Value

Undirected network.

### See Also

network, hergm, ergm.terms, hergm.terms, hergm.postprocess

### Examples

```
## Not run: # Load undirected network with 15 nodes
data(example)

# p_1 model for undirected network with Dirichlet process prior
hergm(d ~ edges_i)

# Hierarchical exponential-family model with stick-breaking prior
hergm(d ~ edges + triangle_ijk)

## End(Not run)
```

`hergm` *Hierarchical Exponential-Family Random Graph Models: Simulation and Bayesian inference*

### Description

The package `hergm` implements exponential-family random graph models with Dirichlet process / stick-breaking priors, including

- the p_1 model for directed networks of Holland and Leinhardt (1981) and its extension to undirected random graph models with Dirichlet process priors (see edges_i). While the p_1 model for undirected and directed networks with parametric priors contains O(n) parameters (n = number of nodes) and therefore is not parsimonious, the non-parametric Dirichlet process prior encourages a small number of unique parameters and therefore represents an elegant alternative to parametric priors.

- the stochastic block model of Snijders and Nowicki (1997) and Nowicki and Snijders (2001) with natural parameterization (restricted between-block parameters) and Dirichlet process priors (see edges_ij).

- the hierarchical exponential-family models of Schweinberger and Handcock (2009) with stick-breaking priors (see mutual_ij, twostar_ijk, triangle_ijk, ttriple_ijk, ctriple_ijk). Hierarchical exponential-family models replace the strong dependence of simple exponential-family models by weak dependence with an eye to solving the near-degeneracy problem of simple exponential-family model.

The package `hergm` implements simulation and Bayesian inference for the mentioned models.

## Usage

```
hergm(formula,
      alpha = NULL,
      alpha_shape = NULL,
      alpha_rate = NULL,
      eta = NULL,
      eta_mean = NULL,
      eta_sd = NULL,
      parallel = 1,
      simulate = FALSE,
      seeds = NULL,
      samplesize = 1e+5,
      burnin = 1e+4,
      interval = 1e+2,
      output = FALSE,
      verbose = -1,
      name = NULL,
      ...)
```

## Arguments

`formula` formula of the form `network ~ terms`. Networks can be created by calling the function `network`. Possible terms can be found in `ergm.terms` and `hergm.terms`.

`alpha` clustering parameter of truncated Dirichlet process / stick-breaking prior of natural parameters of exponential-family model.

`alpha_shape, alpha_rate`
shape and rate (inverse scale) parameter of Gamma prior of clustering parameter.

`eta` natural parameters of exponential-family model.

`eta_mean, eta_sd`
means and standard deviations of Gaussian baseline distribution of Dirichlet process / stick-breaking prior of natural parameters.

`parallel` number of computing nodes; if more than one, computing is parallel.

| | |
|---|---|
| `simulate` | if TRUE, simulation of networks, otherwise Bayesian inference. |
| `seeds` | seed of pseudo-random number generator; if computing is parallel, number of seeds must equal number of computing nodes. |
| `samplesize` | if simulate = TRUE, number of networks to be sampled, otherwise number of draws from posterior; if computing is parallel, number of draws per computing node. |
| `burnin` | if simulate = TRUE, number of burn-in iterations. |
| `interval` | if simulate = TRUE, number of proposals between sampled networks. |
| `output` | if TRUE, full output, otherwise limited output. |
| `name` | name of project; if output = TRUE, name of project is used to name output files. |
| `verbose` | console output: -1: no output; 0: short output; +1: long output. |
| `...` | additional arguments, to be passed to lower-level functions in the future. |

## Value

If called with the option `simulate = TRUE`, the function `hergm` returns a sample of networks, otherwise a raw MCMC sample of parameters from the posterior in the form of a vector; to post-process the raw MCMC sample, call the function `hergm.postprocess`.

## References

Holland, P. W. and S. Leinhardt (1981). An exponential family of probability distributions for directed graphs. Journal of the American Statistical Association 76 (373), 33–65.

Nowicki, K. and T. A. B. Snijders (2001). Estimation and prediction for stochastic blockstructures. Journal of the American Statistical Association 96 (455), 1077–1087.

Schweinberger, M. and M. S. Handcock (2009). Hierarchical exponential-family random graph models. Technical report, Pennsylvania State University. Submitted.

Snijders, T. A. B. and K. Nowicki (1997). Estimation and prediction for stochastic blockmodels for graphs with latent block structure. Journal of Classification 14, 75–100.

## See Also

network, ergm.terms, hergm.terms, hergm.postprocess

## Examples

```
## Not run: # Load undirected network with 15 nodes (see ?example)
data(example)

# p_1 model for undirected network with Dirichlet process prior
hergm(d ~ edges_i, alpha_shape = 1, alpha_rate = 1, eta_mean = -1, eta_sd = 2)

# Stochastic block model for undirected network
# with natural parameterization and Dirichlet process prior
hergm(d ~ edges_ij)

# Hierarchical exponential-family model with stick-breaking prior
```

```
hergm(d ~ edges + triangle_ijk, alpha_shape = 1, alpha_rate = 1,
eta_mean = c(-1, 0), eta_sd = c(2, 1))

## End(Not run)
```

---

`hergm-postprocess` *Hierarchical Exponential-Family Random Graph Models: Postprocessing of Bayesian MCMC samples*

---

## Description

If called with the option `simulate = FALSE`, the function `hergm` returns a raw MCMC sample of parameters from the posterior in the form of a vector. The function `hergm.process` postprocesses the raw MCMC sample: if called with the `output = FALSE`, `hergm.postprocess` extracts information of interest, otherwise it solves, in addition, the so-called label-switching problem. The label-switching problem is rooted in the invariance of the likelihood function to permutations of the labels of blocks, and implies that the raw MCMC sample cannot be used to infer to block-dependent entities. The label-switching problem can be solved in a Bayesian decision-theoretic framework: by defining a loss function and minimizing the posterior expected loss. Calling `hergm.process` minimizes the posterior expected loss using a simple and convenient loss function. The required computations can be time-consuming when the number of blocks `k` is large.

## Usage

```
hergm.postprocess(n = NULL,
                  k = NULL,
                  d1 = 0,
                  d2 = 0,
                  burnin = NULL,
                  samplesize = NULL,
                  mcmc = NULL,
                  output = TRUE,
                  name = NULL,
                  ...)
```

## Arguments

| | |
|---|---|
| `n` | number of nodes. |
| `k` | number of blocks. |
| `d1` | number of ergm terms. |
| `d2` | number of hergm terms. |
| `burnin` | number of burn-in iterations. |
| `samplesize` | total sample size, including number of burn-in iterations and, if parallel > 1, summed across computing nodes. |

| | |
|---|---|
| `mcmc` | MCMC sample in the form of vector. |
| `output` | if TRUE, full output, including relabeled MCMC sample; otherwise limited output. |
| `name` | name of project; if output = TRUE, name of project is used to name output files. |
| `...` | additional arguments, to be passed to lower-level functions in the future. |

### Value

Postprocessed MCMC sample.

### See Also

network, hergm.example, hergm, ergm.terms, hergm.terms

### Examples

```
## Not run: # Load undirected network with 15 nodes (see ?example)
data(example)

# Number of blocks: truncation value of Dirichlet process prior
k <- 10

# Generate MCMC sample: total samplesize is 1 * 1,000 = 1,000
mcmc <- hergm(d ~ edges_i, alpha_shape = 1, alpha_rate = 1,
eta_mean = -1, eta_sd = 2, seed = 2010, parallel = 1, samplesize = 1000)
# Postprocess MCMC sample of size 1,000
processed_mcmc <- hergm.postprocess(n = 15, k = k, d1 = 0, d2 = 1,
burnin = 0, samplesize = 1000, mcmc = mcmc)

# Generate MCMC sample: total samplesize is 20 * 100 = 2,000
mcmc <- hergm(d ~ edges + triangle_ijk, alpha_shape = 1, alpha_rate = 1,
eta_mean = c(-1, 0), eta_sd = c(2, 1), seed = 2010, parallel = 20, samplesize = 100)
# Postprocess MCMC sample of size 2,000
processed_mcmc <- hergm.postprocess(n = 15, k = k, d1 = 0, d2 = 1,
burnin = 0, samplesize = 2000, mcmc = mcmc)

## End(Not run)
```

---

## `hergm-terms` *Hierarchical Exponential-Family Random Graph Models: Terms*

### Description

Hierarchical Exponential-Family Random Graph Models can be specified by calling the function `hergm(formula)`, where formula is a formula of the form `network ~ terms`.

By using suitable terms, it is possible to specify

- the p_1 model for directed networks of Holland and Leinhardt (1981) and its extension to undirected random graph models with Dirichlet process priors (see edges_i). While the p_1 model for undirected and directed networks with parametric priors contains O(n) parameters (n = number of nodes) and therefore is not parsimonious, the non-parametric Dirichlet process prior encourages a small number of unique parameters and therefore represents an elegant alternative to parametric priors.

- the stochastic block model of Snijders and Nowicki (1997) and Nowicki and Snijders (2001) with natural parameterization (restricted between-block parameters) and Dirichlet process priors (see edges_ij).

- the hierarchical exponential-family models of Schweinberger and Handcock (2009) with stick-breaking priors (see mutual_ij, twostar_ijk, triangle_ijk, ttriple_ijk, ctriple_ijk). Hierarchical exponential-family models replace the strong dependence of simple exponential-family models by weak dependence with an eye to solving the near-degeneracy problem of simple exponential-family model.

`hergm.terms` can be found here. Additional terms, e.g. covariate-dependent terms, can be found in `ergm.terms`.

## Arguments

`edges_i (undirected, directed network)`
: adding the term `edges_i` to the model adds node-dependent edge terms to the model.

`edges_ij (undirected, directed network)`
: adding the term `edges_ij` to the model adds block-dependent edge terms to the model.

`mutual_ij (directed network)`
: adding the term `mutual_ij` to the model adds block-dependent mutual edge terms to the model.

`twostar_ijk (undirected network)`
: adding the term `twostar_ijk` to the model adds block-dependent two-star terms to the model.

`triangle_ijk (undirected, directed network)`
: adding the term `triangle_ijk` to the model adds block-dependent triangle terms to the model.

`ttriple_ijk (directed network)`
: adding the term `ttriple_ijk` to the model adds block-dependent transitive triple terms to the model.

`ctriple_ijk (directed network)`
: adding the term `ctriple_ijk` to the model adds block-dependent cyclic triple terms to the model.

## References

Holland, P. W. and S. Leinhardt (1981). An exponential family of probability distributions for directed graphs. Journal of the American Statistical Association 76 (373), 33–65.

Nowicki, K. and T. A. B. Snijders (2001). Estimation and prediction for stochastic blockstructures. Journal of the American Statistical Association 96 (455), 1077–1087.

Schweinberger, M. and M. S. Handcock (2009). Hierarchical exponential-family random graph models. Technical report, Pennsylvania State University. Submitted.

Snijders, T. A. B. and K. Nowicki (1997). Estimation and prediction for stochastic blockmodels for graphs with latent block structure. Journal of Classification 14, 75–100.

### See Also

network, hergm, ergm.terms, hergm.postprocess

### Examples

```
## Not run: # Load undirected network with 15 nodes (see ?example)
data(example)
# p_1 model for undirected network with Dirichlet process prior
hergm(d ~ edges_i)

# Load directed network with 18 nodes (see ?sampson)
data(sampson)
# p_1 model for directed network with Dirichlet process prior
hergm(samplike ~ edges_i + mutual)

# Load undirected network with 15 nodes (see ?example)
data(example)
# Stochastic block model for undirected network
# with natural parameterization and Dirichlet process prior
hergm(d ~ edges_ij)

# Load directed network with 18 nodes (see ?sampson)
data(sampson)
# Stochastic block model for directed network
# with natural parameterization and Dirichlet process prior
hergm(samplike ~ edges_ij + mutual)

# Load undirected network with 15 nodes (see ?example)
data(example)
# Hierarchical exponential-family model with stick-breaking prior
hergm(d ~ edges + mutual + ttriple_ijk)

# Load directed network with 18 nodes (see ?sampson)
data(sampson)
# Hierarchical exponential-family model with stick-breaking prior
hergm(samplike ~ edges + mutual + ttriple_ijk)

## End(Not run)
```

# Index