Package ‘binomlogit’

February 19, 2015

Type Package

Title Efficient MCMC for Binomial Logit Models

Version 1.2

Date 2014-03-12

Author Agnes Fussl

Maintainer Agnes Fussl <avf@gmx.at>

Description The R package contains different MCMC schemes to estimate the regression coefficients of a binomial (or binary) logit model within a Bayesian framework: a data-augmented independence MH-sampler, an auxiliary mixture sampler and a hybrid auxiliary mixture (HAM) sampler. All sampling procedures are based on algorithms using data augmentation, where the regression coefficients are estimated by rewriting the logit model as a latent variable model called difference random utility model (dRUM).

License GPL-3

NeedsCompilation no

Repository CRAN

Date/Publication 2014-03-12 18:11:36

R topics documented:

`binomlogit-package` *Efficient MCMC for Binomial Logit Models*

Description

The R package contains different MCMC schemes to estimate the regression coefficients of a binomial (or binary) logit model within a Bayesian framework: a data-augmented independence MH-sampler, an auxiliary mixture sampler and a hybrid auxiliary mixture (HAM) sampler. All sampling procedures are based on algorithms using data augmentation, where the regression coefficients are estimated by rewriting the logit model as a latent variable model called difference random utility model (dRUM).

Details

Package:	binomlogit
Type:	Package
Version:	1.1
Date:	2014-03-12
License:	GPL-3

The main functions are `dRUMIndMH` for independence Metropolis-Hastings sampling, `dRUMHAM` for hybrid auxiliary mixture sampling and `dRUMAuxMix` for auxiliary mixture sampling in the dRUM representation of a binomial logit model. The function `IndivdRUMIndMH` is designed to work with binary instead of binomial observations to estimate the regression coefficients of a logit model. All four functions simulate the posterior distribution of the regression coefficients of the logit model and return the MCMC draws. For more details about the models and the estimation procedures see **References**. The results are returned in a list of class "binomlogit"" and can be displayed by using the functions `print` and `summary`. The `plot` method returns a plot of the MCMC draws and the acf. The Caesarean birth data set (`caesarean`) is provided in three different formats to serve as exemplary input for the different MCMC schemes.

Author(s)

Agnes Fussl

Maintainer: Agnes Fussl <avf@gmx.at>

References

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2010), "Data augmentation and MCMC for binary and multinomial logit models." In *Statistical Modelling and Regression Structures - Festschrift in Honour of Ludwig Fahrmeir*, T. Kneib and G. Tutz, Eds. Physica-Verlag, Heidelberg, pp. 111-132.

See Also

dRUMIndMH, dRUMAuxMix, dRUMHAM, IndivdRUMIndMH

To evaluate the efficiency of the different MCMC samplers as in the paper by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013), use e.g. the initial sequence estimator initseq by Geyer (package **mcmc**), the numEff function by Rossi (package **bayesm**) or the appropriate functions of the **coda** package by Plummer, Best, Cowles and Vines.

Examples

```
# please run the examples in:
# dRUMIndMH, dRUMAuxMix, dRUMHAM and IndivdRUMIndMH
```

---

caesarean *Caesarean Birth Data*

Description

The data set contains information on infection from births by Caesarean section, originating from a 3-way contingency table. 251 mothers were categorized by the variables "Caesarean planned" (yes/no), "antibiotics given" (yes/no) and "risk factors present" (yes/no). To obtain data for a binary regression model the originally observed two types of infection are ignored and just the binary event "infection" or "no infection" is considered. For the binomial logit regression model all binary observations with the same covariate pattern are aggregated to a binomial observation.

Usage

```
data(caesarean)

data(caesarean_aux)

data(caesarean_binary)
```

Format

The binomial data set `caesarean` consists of 8 binomial observations (= aggregated binary observations) and the following 6 variables:

- `yi` number of women with infection
- `Ni` number of observed women in each group
- `intercept` column consisting of ones
- `planned` Caesarean birth planned (1 = yes, 0 = no)
- `riskfactors` risk factors present (1 = yes, 0 = no)
- `antibiotics` antibiotics given (1 = yes, 0 = no)

The binary data set `caesarean_binary` consists of 251 binary observations and the following 5 variables:

`y` infection (1 = yes, 0 = no)

`planned` Caesarean birth planned (1 = yes, 0 = no)

`riskfactors` risk factors present (1 = yes, 0 = no)

`antibiotics` antibiotics given (1 = yes, 0 = no)

`intercept` column consisting of ones

To run auxiliary mixture sampling in the individual dRUM representation of the binomial logit model the binary data set should have the same form as the binomial data set. For this purpose a column consisting of ones is added to the binary data set. The data set `caesarean_aux` then consists of 251 observations and the following 6 variables:

`yi` number of women with infection

`Ni` number of observed women in each group, which is equal to 1 for all observations

`planned` Caesarean birth planned (1 = yes, 0 = no)

`riskfactors` risk factors present (1 = yes, 0 = no)

`antibiotics` antibiotics given (1 = yes, 0 = no)

`intercept` column consisting of ones

Source

Fahrmeir, L. and Tutz, G. (2001) *Multivariate Statistical Modelling based on Generalized Linear Models*, 2nd Ed. Springer Series in Statistics. Springer, New York/Berlin/Heidelberg.

See Also

`dRUMIndMH`, `dRUMAuxMix`, `dRUMHAM`, `IndivdRUMIndMH`

Examples

```
data(caesarean)
data(caesarean_binary)
data(caesarean_aux)
## see dRUMIndMH, dRUMAuxMix, dRUMHAM and IndivdRUMIndMH documentation
## for examples using these data
```

`dRUMAuxMix` *Auxiliary mixture sampling for the binomial logit model*

Description

`dRUMAuxMix` simulates the posterior distribution of the regression coefficients of a binomial logit model and returns the MCMC draws. The sampling procedure is based on an algorithm using data augmentation, where the regression coefficients are estimated by rewriting the binomial logit model as a latent variable model called difference random utility model (dRUM). The Type III generalized logistic distribution of the error in this dRUM representation is approximated by a scale mixture of normal distributions.

Usage

```
dRUMAuxMix(yi, Ni, X, sim = 12000, burn = 2000, b0, B0,
           start, verbose = 500)
```

Arguments

`yi`	an integer vector of counts for binomial data.
`Ni`	an integer vector containing the number of trials for binomial data.
`X`	a design matrix of predictors.
`sim`	number of MCMC draws including burn-in. The default value is 12000 draws.
`burn`	number of MCMC draws discarded as burn-in. Default is a burn-in of 2000 draws.
`b0`	an optional vector of length `dims = ncol(X)` containing the prior mean. Otherwise a vector of zeros is used.
`B0`	an optional `dims x dims` prior variance-covariance matrix. Otherwise a diagonal matrix with all diagonal elements equal to 10 is used.
`start`	an optional vector of length `dims = ncol(X)` containing the starting values for the regression parameters. Otherwise a vector of zeros is used.
`verbose`	an optional non-negative integer indicating that in each `verbose`-th iteration step a status report is printed (default: `verbose = 500`). If 0, no output is generated during MCMC sampling.

Details

For details concerning the algorithm see the paper by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013).

Value

The output is a list object of class "binomlogit" containing

`beta`	a `dims x sim` matrix of sampled regression coefficients from the posterior distribution
`sim`	the argument `sim`
`burn`	the argument `burn`
`dims`	number of covariates (`dims = ncol(X)`)
`t`	number of binomial observations/covariate patterns (`t = length(yi)`); covariate patterns where `Ni = 0` are not included
`b0`	the argument `b0`
`B0`	the argument `B0`
`duration`	a numeric value indicating the total time (in secs) used for the function call
`duration_wBI`	a numeric value indicating the time (in secs) used for the `sim-burn` MCMC draws after burn-in

To display the output use `print`, `summary` and `plot`. The `print` method prints the number of observations and covariates entered in the routine, the total number of MCMC draws (including burn-in), the number of draws discarded as burn-in and the runtime used for the whole algorithm and for the `sim-burn` MCMC draws after burn-in. The `summary` method additionally returns the used prior parameters `b0` and `B0` and the posterior mean for the regression coefficients without burn-in. The `plot` method plots the MCMC draws and their acf for each regression coefficient, both without burn-in.

Note

`dRUMAuxMix` can also be used to estimate the regression coefficients in the individual dRUM representation of the binomial logit model, i.e. using binary observations as input. For this purpose, add a column in the data set next to the binary response variable, containing the repetition parameter `Ni = 1` for each binary observation.

Author(s)

Agnes Fussl <avf@gmx.at>

References

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

See Also

`dRUMIndMH`, `dRUMHAM`, `IndivdRUMIndMH`

Examples

```
## Auxiliary mixture sampling in the aggregated dRUM representation of a
## binomial logit model

## load caesarean birth data
data(caesarean)
yi <- as.numeric(caesarean[,1])
Ni <- as.numeric(caesarean[,2])
X <- as.matrix(caesarean[,-(1:2)])

## start auxiliary mixture sampler
aux1=dRUMAuxMix(yi,Ni,X,verbose=0)
## Not run:
aux2=dRUMAuxMix(yi,Ni,X)

## End(Not run)

print(aux1)
summary(aux1)
plot(aux1)
```

```
## Not run:
## Auxiliary mixture sampling in the individual dRUM representation of a
## binomial logit model

## load caesarean birth data
data(caesarean_aux)
yi <- as.numeric(caesarean_aux[,1])
Ni <- as.numeric(caesarean_aux[,2])
X <- as.matrix(caesarean_aux[,-(1:2)])

## start auxiliary mixture sampler
aux3=dRUMAuxMix(yi,Ni,X)

print(aux3)
summary(aux3)
plot(aux3)

## End(Not run)
```

---

dRUMHAM *Hybrid auxiliary mixture sampling for the binomial logit model*

Description

dRUMHAM simulates the posterior distribution of the regression coefficients of a binomial logit model and returns the MCMC draws. The sampling procedure is based on an algorithm using data augmentation, where the regression coefficients are estimated by rewriting the binomial logit model as a latent variable model called difference random utility model (dRUM). For binomial observations where the success rate yi/Ni is neither close to 0 nor close to 1 we use the normal distribution as for the dRUMIndMH sampler to approximate the Type III generalized logistic distributed error in the dRUM representation. For extreme ratios yi/Ni <= low and yi/Ni >= up the error is approximated by the precise scale mixture of normal distributions as used for the dRUMAuxMix sampler. The resulting posterior of this regression model is then used as proposal density for the regression coefficients.

Usage

```
dRUMHAM(yi, Ni, X, sim = 12000, burn = 2000, b0, B0, start,
        low = 0.05, up = 0.95, verbose = 500)
```

Arguments

yi	an integer vector of counts for binomial data.
Ni	an integer vector containing the number of trials for binomial data.
X	a design matrix of predictors.
sim	number of MCMC draws including burn-in. The default value is 12000 draws.
burn	number of MCMC draws discarded as burn-in. Default is a burn-in of 2000 draws.

`b0`	an optional vector of length `dims = ncol(X)` containing the prior mean. Otherwise a vector of zeros is used.
`B0`	an optional `dims x dims` prior variance-covariance matrix. Otherwise a diagonal matrix with all diagonal elements equal to 10 is used.
`start`	an optional vector of length `dims = ncol(X)` containing the starting values for the regression parameters. Otherwise a vector of zeros is used.
`low`	a numeric value between 0 and 1 indicating that for all observations where the ratio `yi/Ni <= low` the precise mixture approximation is used instead of the simpler normal approximation. The default value is 0.05.
`up`	a numeric value between 0 and 1 indicating that for all observations where the ratio `yi/Ni >= up` the precise mixture approximation is used instead of the simpler normal approximation. The default value is 0.95.
`verbose`	an optional non-negative integer indicating that in each `verbose`-th iteration step a status report is printed (default: `verbose = 500`). If 0, no output is generated during MCMC sampling.

Details

For details concerning the algorithm see the paper by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013).

Value

The output is a list object of class `c("binomlogitHAM","binomlogit")` containing

`beta`	a `dims x sim` matrix of sampled regression coefficients from the posterior distribution
`sim`	the argument `sim`
`burn`	the argument `burn`
`dims`	number of covariates (`dims = ncol(X)`)
`t`	number of binomial observations/covariate patterns (`t = length(yi)`); covariate patterns where `Ni = 0` are not included
`b0`	the argument `b0`
`B0`	the argument `B0`
`low`	the argument `low`
`up`	the argument `up`
`duration`	a numeric value indicating the total time (in secs) used for the function call
`duration_wBI`	a numeric value indicating the time (in secs) used for the `sim-burn` MCMC draws after burn-in
`rate`	acceptance rate based on the `sim-burn` MCMC draws after burn-in

To display the output use `print`, `summary` and `plot`. The `print` method prints the number of observations and covariates entered in the routine, the total number of MCMC draws (including burn-in), the number of draws discarded as burn-in, the runtime used for the whole algorithm and for the `sim-burn` MCMC draws after burn-in and the acceptance rate. The `summary` method additionally returns the boundaries `low` and `up` used for HAM sampling, the prior parameters `b0` and `B0` and the posterior mean for the regression coefficients without burn-in. The `plot` method plots the MCMC draws and their acf for each regression coefficient, both without burn-in.

Author(s)

Agnes Fussl <avf@gmx.at>

References

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

See Also

`dRUMIndMH`, `dRUMAuxMix`, `IndivdRUMIndMH`

Examples

```
## Hybrid auxiliary mixture sampling in the aggregated dRUM representation
## of a binomial logit model

## load caesarean birth data
data(caesarean)
yi <- as.numeric(caesarean[,1])
Ni <- as.numeric(caesarean[,2])
X <- as.matrix(caesarean[,-(1:2)])

## start HAM sampler
ham1 <- dRUMHAM(yi,Ni,X)
## Not run:
ham2 <- dRUMHAM(yi,Ni,X,low=0.01,up=0.99)

## End(Not run)

print(ham1)
summary(ham1)
plot(ham1)
```

`dRUMIndMH` *Data-augmented independence Metropolis-Hastings sampling for the binomial logit model*

Description

dRUMIndMH simulates the posterior distribution of the regression coefficients of a binomial logit model and returns the MCMC draws. The sampling procedure is based on an algorithm using data augmentation, where the regression coefficients are estimated by rewriting the binomial logit model as a latent variable model called difference random utility model (dRUM). The Type III generalized logistic distribution of the error in the dRUM representation is approximated by a normal distribution with same mean and variance. The posterior of this approximate regression model is then used as independence proposal density for the regression coefficients.

Usage

```
dRUMIndMH(yi, Ni, X, sim = 12000, burn = 2000, acc = 0, b0, B0,
          start, verbose = 500)
```

Arguments

yi	an integer vector of counts for binomial data.
Ni	an integer vector containing the number of trials for binomial data.
X	a design matrix of predictors.
sim	number of MCMC draws including burn-in. The default value is 12000 draws.
burn	number of MCMC draws discarded as burn-in. Default is a burn-in of 2000 draws.
acc	number of MCMC draws at the beginning of the burn-in phase where each proposed parameter vector is accepted with probability 1 rather than according to the MH acceptance rule. Choose a small number `acc > 0` if the sampler gets stuck at the starting values, otherwise `acc` is set to 0.
b0	an optional vector of length `dims = ncol(X)` containing the prior mean. Otherwise a vector of zeros is used.
B0	an optional `dims x dims` prior variance-covariance matrix. Otherwise a diagonal matrix with all diagonal elements equal to 10 is used.
start	an optional vector of length `dims = ncol(X)` containing the starting values for the regression parameters. Otherwise a vector of zeros is used.
verbose	an optional non-negative integer indicating that in each `verbose`-th iteration step a status report is printed (default: `verbose = 500`). If 0, no output is generated during MCMC sampling.

Details

For details concerning the algorithm see the paper by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013).

Value

The output is a list object of class `c("binomlogitMH","binomlogit")` containing

beta	a `dims x sim` matrix of sampled regression coefficients from the posterior distribution

sim	the argument sim
burn	the argument burn
acc	the argument acc
dims	number of covariates (dims = ncol(X))
t	number of binomial observations/covariate patterns (t = length(yi)); covariate patterns where Ni = 0 are not included
b0	the argument b0
B0	the argument B0
duration	a numeric value indicating the total time (in secs) used for the function call
duration_wBI	a numeric value indicating the time (in secs) used for the sim-burn MCMC draws after burn-in
rate	acceptance rate based on the sim-burn MCMC draws after burn-in

To display the output use print, summary and plot. The print method prints the number of observations and covariates entered in the routine, the total number of MCMC draws (including burn-in), the number of draws discarded as burn-in, the runtime used for the whole algorithm and for the sim-burn MCMC draws after burn-in and the acceptance rate. The summary method additionally returns the length of the acceptance phase during burn-in, the used prior parameters b0 and B0 and the posterior mean for the regression coefficients without burn-in. The plot method plots the MCMC draws and their acf for each regression coefficient, both without burn-in.

Note

dRUMIndMH could also be used to estimate the regression coefficients in the individual dRUM representation of the binomial logit model (analogous to dRUMAuxMix). However, it is more straightforward to use IndivdRUMIndMH, where binary observations can directly be used as input.

Author(s)

Agnes Fussl <avf@gmx.at>

References

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

See Also

dRUMAuxMix, dRUMHAM, IndivdRUMIndMH

Examples

```
## Independence MH sampling in the aggregated dRUM representation of a
## binomial logit model

## load caesarean birth data
data(caesarean)
```

```
yi <- as.numeric(caesarean[,1])
Ni <- as.numeric(caesarean[,2])
X <- as.matrix(caesarean[,-(1:2)])

## start independence MH sampler
mh1 <- dRUMIndMH(yi,Ni,X)

print(mh1)
summary(mh1)
plot(mh1)

## Not run:
## load simulated data set
data(simul)
yi <- as.numeric(simul[,1])
Ni <- as.numeric(simul[,2])
X <- as.matrix(simul[,-(1:2)])

## use a small acc>0 (e.g. acc=50), otherwise the sampler gets stuck at
## the starting values
mh2 <- dRUMIndMH(yi,Ni,X,acc=50)

print(mh2)
summary(mh2)
plot(mh2)

## End(Not run)
```

IndivdRUMIndMH *Data-augmented independence Metropolis-Hastings sampling for the binary logit model*

Description

IndivdRUMIndMH simulates the posterior distribution of the regression coefficients of a binary logit model and returns the MCMC draws. The sampling procedure is based on an algorithm using data augmentation, where the regression coefficients are estimated by rewriting the binary logit model as a latent variable model called difference random utility model (dRUM). This dRUM representation of the binary logit model is equivalent to the individual dRUM representation of the binomial logit model. The logistic distribution of the error in the dRUM representation is approximated by a normal distribution with same mean and variance. The posterior of this approximate regression model is then used as independence proposal density for the regression coefficients.

Usage

```
IndivdRUMIndMH(y, X, sim = 12000, burn = 2000, acc = 0, b0, B0,
               start, verbose = 500)
```

Arguments

y	logical or coercible to integer with values 0 and 1 (Bernoulli data).
X	a design matrix of predictors.
sim	number of MCMC draws including burn-in. The default value is 12000 draws.
burn	number of MCMC draws discarded as burn-in. Default is a burn-in of 2000 draws.
acc	number of MCMC draws at the beginning of the burn-in phase where each proposed parameter vector is accepted with probability 1 rather than according to the MH acceptance rule. Choose a small number `acc > 0` if the sampler gets stuck at the starting values, otherwise `acc` is set to 0.
b0	an optional vector of length `dims = ncol(X)` containing the prior mean. Otherwise a vector of zeros is used.
B0	an optional `dims x dims` prior variance-covariance matrix. Otherwise a diagonal matrix with all diagonal elements equal to 10 is used.
start	an optional vector of length `dims = ncol(X)` containing the starting values for the regression parameters. Otherwise a vector of zeros is used.
verbose	an optional non-negative integer indicating that in each `verbose`-th iteration step a status report is printed (default: `verbose = 500`). If 0, no output is generated during MCMC sampling.

Details

For details concerning the algorithm see the papers by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013) and Fruehwirth-Schnatter and Fruehwirth (2010).

Value

The output is a list object of class `c("binomlogitIndiv","binomlogitMH", "binomlogit")` containing

beta	a `dims x sim` matrix of sampled regression coefficients from the posterior distribution
sim	the argument `sim`
burn	the argument `burn`
acc	the argument `acc`
dims	number of covariates (`dims = ncol(X)`)
N	number of binary observations
b0	the argument `b0`
B0	the argument `B0`
duration	a numeric value indicating the total time (in secs) used for the function call
duration_wBI	a numeric value indicating the time (in secs) used for the `sim-burn` MCMC draws after burn-in
rate	acceptance rate based on the `sim-burn` MCMC draws after burn-in

To display the output use `print`, `summary` and `plot`. The `print` method prints the number of observations and covariates entered in the routine, the total number of MCMC draws (including burn-in), the number of draws discarded as burn-in, the runtime used for the whole algorithm and for the `sim-burn` MCMC draws after burn-in and the acceptance rate. The `summary` method additionally returns the length of the acceptance phase during burn-in, the used prior parameters `b0` and `B0` and the posterior mean for the regression coefficients without burn-in. The `plot` method plots the MCMC draws and their acf for each regression coefficient, both without burn-in.

Note

To perform auxiliary mixture sampling for the binary logit model use `dRUMAuxMix` and follow the instructions given there.

Author(s)

Agnes Fussl <avf@gmx.at>

References

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2010), "Data augmentation and MCMC for binary and multinomial logit models." In *Statistical Modelling and Regression Structures - Festschrift in Honour of Ludwig Fahrmeir*, T. Kneib and G. Tutz, Eds. Physica-Verlag, Heidelberg, pp. 111-132.

See Also

`dRUMIndMH`, `dRUMAuxMix`, `dRUMHAM`

Examples

```
## Data-augmented independence Metropolis-Hastings sampling for the
## binary logit model

## load caesarean birth data
data(caesarean_binary)
y <- as.numeric(caesarean_binary[,1])
X <- as.matrix(caesarean_binary[,-1])

## start independence MH sampler
indivMH1 <- IndivdRUMIndMH(y,X)

print(indivMH1)
summary(indivMH1)
plot(indivMH1)

## Not run:
## load simulated data set
data(simul_binary)
```

```
y <- as.numeric(simul_binary[,1])
X <- as.matrix(simul_binary[,-1])

## use a small acc>0 (e.g. acc=50), otherwise the sampler gets stuck at
## the starting values
indivMH2 <- IndivdRUMIndMH(y,X,acc=50)

print(indivMH2)
summary(indivMH2)
plot(indivMH2)

## End(Not run)
```

---

simul *Simulated data set*

Description

For testing purposes we constructed the very extreme and unbalanced simulated binomial data set `simul`. The pattern of this data set is typical of models for rare events, e.g. rare diseases or financial defaults. Based on a fixed number of `dims = 10` covariates consisting of nine binary variables and the intercept, the design matrix `X` is built by computing all 2^9 possible 0/1 combinations. The true parameter vector is `beta={0.05,2,1.5,-3,-0.01,-1.3,2.9,-2.1,0.5,-0.2}`. For details concerning the simulation of the data set see the paper by Fussl, Fruehwirth-Schnatter and Fruehwirth (2013). To use the data set with the function `IndivdRUMIndMH`, binary outcomes are reconstructed from the binomial observations and saved as `simul_binary`.

Usage

```
data(simul)

data(simul_binary)
```

Format

The binomial data set `simul` consists of 512 binomial observations and the following 12 variables:

`yi` number of successes for each covariate pattern

`Ni` group size for each covariate pattern

`X,X.1,...,X.8` binary covariates

`X.9` intercept

Only 490 covariate patterns have a group size `Ni > 0` and will be included when using the functions `dRUMIndMH`, `dRUMAuxMix` and `dRUMHAM`.

The binary data set `simul_binary` consists of 25803 binary observations and the following 11 variables:

`y` binary response variable

`X,X.1,...,X.8` binary covariates

`X.9` intercept

Source

Agnes Fussl, Sylvia Fruehwirth-Schnatter and Rudolf Fruehwirth (2013), "Efficient MCMC for Binomial Logit Models". *ACM Transactions on Modeling and Computer Simulation 23*, 1, Article 3, 21 pages.

See Also

dRUMIndMH, IndivdRUMIndMH

Examples

```
data(simul)
data(simul_binary)
## see dRUMIndMH and IndivdRUMIndMH documentation for examples using
## these data
```

Index