

Package ‘ctmcd’

September 13, 2016

Type Package

Title Estimating the Parameters of a Continuous-Time Markov Chain from Discrete-Time Data

Version 1.0

Date 2016-09-13

Author Marius Pfeuffer

Maintainer Marius Pfeuffer <marius.pfeuffer@fau.de>

Description Functions for estimating Markov generator matrices from discrete-time observations. The implemented approaches comprise diagonal adjustment, weighted adjustment and quasi-optimization of matrix logarithm based candidate solutions, an expectation-maximization algorithm as well as a Gibbs sampler.

License GPL-3

Imports Rcpp (>= 0.12.6), expm, numDeriv

Suggests knitr, rmarkdown

LinkingTo Rcpp, RcppArmadillo

VignetteBuilder knitr

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-09-13 21:19:15

R topics documented:

ctmcd-package	2
ciEM	3
ciGS	4
gm	5
gm.default	5
gmci	7
gmci.default	8
gmDA	9
gmEM	10

gmGS	11
gmQO	12
gmWA	13
plot.gm	14
plot.gmci	15
plotM	17
print.gm	18
print.gmci	18
rNijTRiT_ModRej	19
rNijTRiT_Unif	20
summary.gm	21
tm_abs	21

Index 23

ctmcd-package	<i>Estimating the Parameters of a Continuous-Time Markov Chain from Discrete-Time Data</i>
---------------	--

Description

Functions for estimating Markov generator matrices from discrete-time observations.

Author(s)

Marius Pfeuffer

Maintainer: Marius Pfeuffer <marius.pfeuffer@fau.de>

References

Y. Inamura: Estimating Continuous Time Transition Matrices from Discretely Observed Data. Bank of Japan Working Paper Series, 2006

R. B. Israel et al.: Finding Generators for Markov Chains via Empirical Transition Matrices, with Applications to Credit Ratings. *Mathematical Finance* 11(2):245-265, 2001

E. Kreinin and M. Sidelnikova: Regularization Algorithms for Transition Matrices. *Algo Research Quarterly* 4(1):23-40, 2001

M. Bladt and M. Soerensen: Statistical Inference for Discretely Observed Markov Jump Processes. *Journal of the Royal Statistical Society B* 67(3):395-410, 2005

Examples

```
## Not run:
data(tm_abs)

## Maximum Likelihood Generator Matrix Estimate
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
```

```

gm0[8,]=0

gmem=gm(tm_abs,te=1,method="EM",gmguess=gm0)
plot(gmem)

## Confidence Interval
ciem=gmci(gmem,alpha=0.05)
plot(ciem)

## End(Not run)

```

ciEM

Wald Confidence Interval

Description

Function to derive a Wald confidence interval for expectation-maximization based maximum likelihood generator matrix estimate. The Fisher information estimation method of Oakes, 1999 is employed.

Usage

```
ciEM(x, alpha, eps = 1e-04, expmethod = "PadeRBS")
```

Arguments

x	expectation-maximization output object
alpha	significance level
eps	threshold for which generator matrix parameters are assumed to be fixed at zero
expmethod	method for computation of matrix exponential, by default "PadeRBS" is chosen (see ?expm from expm package for more information)

Details

Computation of confidence interval on the basis of the last iteration of the expectation-maximization algorithm.

Author(s)

Marius Pfeuffer

References

M. Bladt and M. Soerensen. Efficient Estimation of Transition Rates Between Credit Ratings from Observations at Discrete Time Points. *Quantitative Finance* 9(2):147-160, 2009

D. Oakes. Direct calculation of the information matrix via the EM algorithm. *Journal of the Royal Statistical Society B* 61(2):479-482, 1999

Examples

```

## Not run:
data(tm_abs)
## Initial guess for generator matrix (absorbing default state)
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

## Derive expectation-maximization algorithm generator matrix estimate
gmem=gm(tm=tm_abs,te=1,method="EM",gmguess=gm0,verbose=TRUE)

## derive confidence interval
ciem=ciEM(gmem, alpha=0.05)
ciem

## End(Not run)

```

ciGS

Equal Tailed Credibility Interval

Description

Function to derive an equal-tailed credibility interval from Gibbs sampling draws as described by Bladt and Soerensen, 2009.

Usage

```
ciGS(x, alpha)
```

Arguments

x	Gibbs sampling output object
alpha	significance level

Details

Computation of credibility interval es empirical quantiles of Gibbs sampling draws.

Author(s)

Marius Pfeuffer

References

M. Bladt and M. Soerensen. Efficient Estimation of Transition Rates Between Credit Ratings from Observations at Discrete Time Points. *Quantitative Finance* 9(2):147-160, 2009

Examples

```
data(tm_abs)

## Prior
pr=list()
pr[[1]]=matrix(1,8,8)
pr[[1]][8,]=0

pr[[2]]=c(rep(5,7),Inf)

## Derive Gibbs sampling generator matrix estimate
gmgs=gm(tm=tm_abs,te=1,method="GS",sampl_method="Unif",prior=pr,burnin=10,niter=100)

cigs=ciGS(gmgs,alpha=.05)
cigs
```

gm

Generic Generator Matrix Estimation

Description

Generic function for generator matrix estimation

Usage

```
gm(tm, ...)
```

Arguments

tm	matrix of either absolute or relative transition frequencies
...	additional arguments

See Also

[gm.default](#)

gm.default

Generator Matrix Estimation

Description

Function to estimate the parameters of a continuous Markov chain

Usage

```
## Default S3 method:
gm(tm, te, method, gmguess = NULL, prior = NULL, burnin = NULL,
eps = 1e-06, niter = 10000, sampl_func = NULL, combmat = NULL, sampl_method = "Unif",
logmethod = "Eigen", expmethod = "PadeRBS", verbose = FALSE, ...)
```

Arguments

tm	matrix of either absolute transition frequencies (if method is "EM" or "GS") or relative transition frequencies (if method is "DA", "WA" or "QO")
te	time elapsed in transition process
method	method to derive generator matrix: "DA" - Diagonal Adjustment, "WA" - Weighted Adjustment, "QO" - Quasi-Optimization, "EM" - Expectation-Maximization Algorithm, "GS" - Gibbs Sampler
gmguess	initial guess for generator matrix estimation procedure (if method is "EM")
prior	prior parametrization (if method is "GS")
burnin	burn-in period (if method is "GS")
eps	convergence criterion (if method is "EM" or "GS")
niter	maximum number of iterations (if method is "EM" or "GS")
sampl_func	optional self-written path sampling function for endpoint-conditioned Markov processes (if method is "GS")
combmat	matrix stating combined use of modified rejection sampling / uniformization sampling algorithms (if method is "GS")
sampl_method	sampling method for deriving endpoint-conditioned Markov process path: "Unif" - Uniformization Sampling, "ModRej" - Modified Rejection Sampling (if method is "GS")
logmethod	method to compute matrix logarithm (if method is "DA", "WA" or "QO", see ?logm from expm package for more information)
expmethod	method to compute matrix exponential (if method is "EM" or "GS", see ?expm from expm package for more information)
verbose	verbose mode (if method is "EM" or "GS")
...	additional arguments

Details

The methods "DA", "WA" and "QO" provide adjustments of a matrix logarithm based candidate solution, "EM" gives the maximum likelihood estimate and "GS" a posterior mean estimate in a Bayesian setting with conjugate Gamma priors.

Author(s)

Marius Pfeuffer

References

- M. Pfeuffer: Generator Matrix Approximation Based on Discrete Time Rating Migration Data. Master Thesis, University of Munich, 2016
- Y. Inamura: Estimating Continuous Time Transition Matrices from Discretely Observed Data. Bank of Japan Working Paper Series, 2006
- R. B. Israel et al.: Finding Generators for Markov Chains via Empirical Transition Matrices, with Applications to Credit Ratings. *Mathematical Finance* 11(2):245-265, 2001
- E. Kreinin and M. Sidelnikova: Regularization Algorithms for Transition Matrices. *Algo Research Quarterly* 4(1):23-40, 2001
- M. Bladt and M. Soerensen: Statistical Inference for Discretely Observed Markov Jump Processes. *Journal of the Royal Statistical Society B* 67(3):395-410, 2005

See Also

[gmDA](#), [gmWA](#), [gmQO](#), [gmEM](#), [gmGS](#)

Examples

```
data(tm_abs)

## Maximum Likelihood Generator Matrix Estimate
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

gmem=gm(tm_abs,te=1,method="EM",gmguess=gm0)
gmem

## Quasi Optimization Estimate
tm_rel=rbind((tm_abs/rowSums(tm_abs))[1:7,],c(rep(0,7),1))

gmqo=gm(tm_rel,te=1,method="QO")
gmqo
```

gmci

Generic Generator Matrix Confidence / Credibility Interval

Description

Generic function for generator matrix confidence / credibility intervals

Usage

```
gmci(gm, ...)
```

Arguments

gm	a generator matrix estimation object
...	additional arguments

See Also

[gmci.default](#)

gmci.default

Confidence / Credibility Intervals for Generator Matrix Objects

Description

Function to derive confidence / credibility intervals for "EM" or "GS" based generator matrix objects

Usage

```
## Default S3 method:
gmci(gm, alpha, eps = 1e-04, expmethod = "PadeRBS", ...)
```

Arguments

gm	a "EM" or "GS" generator matrix object
alpha	significance level
eps	threshold for which generator matrix parameters are assumed to be fixed at zero (if "EM" object)
expmethod	method to compute matrix exponentials (see ?expm from expm package for more information)
...	additional arguments

Details

If gm is based on the "EM" method (expectation-maximization algorithm), the function computes a Wald confidence interval based on the method of Oakes, 1999. If gm is based on the "GS" method (Gibbs sampler), the function computes an equal-tailed credibility interval.

Author(s)

Marius Pfeuffer

References

D. Oakes. Direct calculation of the information matrix via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(2):479-482, 1999

M. Bladt and M. Soerensen. Efficient Estimation of Transition Rates Between Credit Ratings from Observations at Discrete Time Points. *Quantitative Finance*, 9(2):147-160, 2009

See Also[ciEM](#), [ciGS](#)**Examples**

```
## Not run:
data(tm_abs)

## Maximum Likelihood Generator Matrix Estimate
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

gmem=gm(tm_abs,te=1,method="EM",gmguess=gm0)

## Oakes Confidence Interval
ciem=gmci(gmem,alpha=0.05)
ciem

## End(Not run)
```

gmDA

Diagonal Adjustment

Description

Function for deriving a Markov generator matrix estimate based on the diagonal adjustment method of Israel et al., 2001

Usage

```
gmDA(tmrel, te, logmethod = "Eigen")
```

Arguments

tmrel	matrix of relative transition frequencies
te	time elapsed in transition process
logmethod	method for computation of matrix logarithm, by default eigendecomposition is chosen (see ?logm from expm package for more information)

Details

A candidate solution is derived by the matrix logarithm and then adjusted in order to fulfil the properties of a Markov generator matrix.

Author(s)

Marius Pfeuffer

References

R. B. Israel et al.: Finding Generators for Markov Chains via Empirical Transition Matrices, with Applications to Credit Ratings. *Mathematical Finance* 11(2):245-265, 2001

Examples

```
## Derive matrix of relative transition frequencies
data(tm_abs)
tm_rel=rbind((tm_abs/rowSums(tm_abs))[1:7,],c(rep(0,7),1))

## Derive diagonal adjustment generator matrix estimate
gmda=gmdA(tm_rel,1)
gmda
```

gmEM

Expectation-Maximization Algorithm

Description

Function for deriving a Markov generator matrix estimate by an instance of the expectation-maximization algorithm (described by Bladt and Soerensen, 2005)

Usage

```
gmEM(tmabs, te, gmguess, eps = 1e-06, niter = 10000, expmethod = "PadeRBS",
      verbose = FALSE)
```

Arguments

tmabs	matrix of absolute transition frequencies
te	time elapsed in transition process
gmguess	initial guess (for generator matrix)
eps	stop criterion: stop, if relative change in log-likelihood is smaller than eps
niter	stop criterion: maximum number of iterations
expmethod	method for computation of matrix exponential, by default "PadeRBS" is chosen (see ?expm from expm package for more information)
verbose	verbose mode

Details

A maximum likelihood generator matrix estimate is derived by an instance of the expectation-maximization algorithm.

Author(s)

Marius Pfeuffer

References

M. Bladt and M. Soerensen: Statistical Inference for Discretely Observed Markov Jump Processes. Journal of the Royal Statistical Society B 67(3):395-410, 2005

Examples

```
data(tm_abs)

## Initial guess for generator matrix (absorbing default state)
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

## Derive expectation-maximization algorithm generator matrix estimate
gmem=gmEM(tmabs=tm_abs,1,gmguess=gm0,verbose=TRUE)
gmem
```

gmGS

Gibbs Sampler

Description

Function for deriving a Markov generator matrix estimate by Gibbs sampling (described by Bladt and Soerensen, 2005)

Usage

```
gmGS(tmabs, te, prior, burnin, eps = 1e-06, niter = 10000, sampl_method = "Unif",
expmethod = "PadeRBS", verbose = FALSE, combmat=NULL, sampl_func = NULL)
```

Arguments

tmabs	matrix of absolute transition frequencies
te	time elapsed in transition process
prior	list of prior parameters (Gamma prior)
burnin	number of burn-in iterations
eps	stop criterion: stop, if L1 metric between parameter estimates of subsequent iteration steps is below eps at least twice
niter	stop criterion: stop, if maximum number of iterations is exceeded
sampl_method	method for sampling paths from endpoint-conditioned Markov processes. options: "Unif" - Uniformization sampling, "ModRej" - Modified Rejection Sampling
expmethod	method for computation of matrix exponential, by default "PadeRBS" is chosen (see ?expm from expm package for more information)

verbose	verbose mode
combat	matrix specifying the combined use of sampling methods: "U" - uniformization sampling, "M" - modified rejection sampling
sampl_func	interface for own endpoint-conditioned Markov process sampling function

Details

A posterior mean generator matrix estimate is derived by Gibbs Sampling. The gamma distribution is used as prior.

Author(s)

Marius Pfeuffer

References

M. Bladt and M. Soerensen: Statistical Inference for Discretely Observed Markov Jump Processes. Journal of the Royal Statistical Society B 67(3):395-410, 2005

See Also

[rNijTRiT_ModRej](#), [rNijTRiT_Unif](#)

Examples

```
data(tm_abs)

## Example prior parametrization (absorbing default state)
pr=list()
pr[[1]]=matrix(1,8,8)
pr[[1]][8,]=0

pr[[2]]=c(rep(5,7),Inf)

## Derive Gibbs sampling generator matrix estimate
gmgs=gmGS(tmabs=tm_abs,te=1,sampl_method="Unif",prior=pr,burnin=10,niter=100,verbose=TRUE)
gmgs
```

gmQQ

Quasi-Optimization

Description

Function for deriving a Markov generator matrix estimate based on the quasi-optimization procedure of Kreinin and Sidelnikova, 2001

Usage

```
gmQQ(tmrel, te, logmethod = "Eigen")
```

Arguments

tmrel	matrix of relative transition frequencies
te	time elapsed in transition process
logmethod	method for computation of matrix logarithm, by default eigendecomposition is chosen (see ?logm from expm package for more information)

Details

From the set of possible Markov generator matrices, the one is chosen which is closest to a matrix logarithm based candidate solution in terms of sum of squared deviations.

Author(s)

Marius Pfeuffer

References

E. Kreinin and M. Sidelnikova: Regularization Algorithms for Transition Matrices. Algo Research Quarterly 4(1):23-40, 2001

Examples

```
data(tm_abs)
## Derive matrix of relative transition frequencies
data(tm_abs)
tm_rel=rbind((tm_abs/rowSums(tm_abs))[1:7,],c(rep(0,7),1))

## Derive quasi optimization generator matrix estimate
gmqo=gmQO(tm_rel,1)
gmqo
```

gmWA

Weighted Adjustment

Description

Function for deriving a Markov generator matrix estimate based on the weighted adjustment method of Israel et al., 2001

Usage

```
gmWA(tmrel, te, logmethod = "Eigen")
```

Arguments

tmrel	matrix of relative transition frequencies
te	time elapsed in transition process
logmethod	method for computation of matrix logarithm, by default eigendecomposition is chosen (see ?logm from expm package for more information)

Details

A candidate solution is derived by the matrix logarithm and then adjusted in order to fulfil the properties of a Markov generator matrix.

Author(s)

Marius Pfeuffer

References

R. B. Israel et al.: Finding Generators for Markov Chains via Empirical Transition Matrices, with Applications to Credit Ratings. *Mathematical Finance* 11(2):245-265, 2001

Examples

```
## Derive matrix of relative transition frequencies
data(tm_abs)
tm_rel=rbind((tm_abs/rowSums(tm_abs))[1:7,],c(rep(0,7),1))

## Derive weighted adjustment generator matrix estimate
gmwa=gmWA(tm_rel,1)
gmwa
```

plot.gm

Plot Function for Generator Matrix Estimation Objects

Description

Function for visualizing the output of a generator matrix estimation procedure.

Usage

```
## S3 method for class 'gm'
plot(x, mattext, col = c("grey", "red"), main = x$method, las = 1,
xlab = "To", ylab = "From", xnames, ynames, cex = 1, fig = 3, opacity_factor, ...)
```

Arguments

x	a generator matrix estimation object
mattext	optional: matrix of strings replacing the parameter estimates
col	two element vector of basis colors for positive and negative parameter estimate entries
main	optional: plot title
las	orientation of x and y axis elements
xlab	x axis name
ylab	y axis name

xnames	description of x axis elements
ynames	description of y axis elements
cex	font size
fig	number of significant figure to be plotted
opacity_factor	two element vector for specification of opacity for positive and negative parameter entry highlighting (must be greater than zero)
...	additional arguments

Author(s)

Marius Pfeuffer

See Also

[print.gm](#), [summary.gm](#), [plotM](#)

Examples

```
data(tm_abs)

## Maximum Likelihood Generator Matrix Estimate
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

gmem=gm(tm_abs,te=1,method="EM",gmguess=gm0)
plot(gmem)
```

plot.gmci	<i>Plot Function for Generator Matrix Confidence / Credibility Interval Objects</i>
-----------	---

Description

Function for visualizing the boundaries of generator matrix confidence / credibility intervals

Usage

```
## S3 method for class 'gmci'
plot(x, mattext, col = c("grey", "red"), main, las = 1, xlab = "To",
ylab = "From", xnames, ynames, cex = 1, fig = 2, opacity_factor, ...)
```

Arguments

x	a generator matrix confidence / credibility interval object
mattext	optional: matrix of strings replacing the parameter estimates
col	two element vector of basis colors for positive and negative parameter estimate entries
main	optional: plot title
las	orientation of x and y axis elements
xlab	x axis name
ylab	y axis name
xnames	description of x axis elements
yname	description of y axis elements
cex	font size
fig	number of significant figures to be plotted
opacity_factor	two element vector for specification of opacity for positive and negative parameter entry highlighting (must be greater than zero)
...	additional arguments

Author(s)

Marius Pfeuffer

See Also

[print.gmci](#), [plotM](#)

Examples

```
## Not run:
data(tm_abs)

## Maximum Likelihood Generator Matrix Estimate
gm0=matrix(1,8,8)
diag(gm0)=0
diag(gm0)=-rowSums(gm0)
gm0[8,]=0

gmem=gm(tm_abs,te=1,method="EM",gmguess=gm0)
plot(gmem)

## Confidence Interval
ciem=gmci(gmem,alpha=0.05)
plot(ciem)

## End(Not run)
```

plotM	<i>Matrix Plot Function</i>
-------	-----------------------------

Description

Function to visualize matrices

Usage

```
plotM(mat, mattext, col = c("grey", "red"), main, las = 1, xlab = "To",  
ylab = "From", xnames, ynames, cex = min(1, nrow(mat)/8), fig = 3, opacity_factor)
```

Arguments

mat	a matrix
mattext	optional: matrix of strings replacing the original matrix entries
col	two element vector of basis colors for positive and negative matrix entries
main	optional: plot title
las	orientation of x and y axis elements
xlab	x axis name
ylab	y axis name
xnames	description of x axis elements
ynames	description of y axis elements
cex	font size
fig	number of significant figures to be plotted
opacity_factor	two element vector for specification of opacity for positive and negative parameter entry highlighting (must be greater than zero)

Author(s)

Marius Pfeuffer

See Also

[plot.gm](#), [plot.gmci](#)

Examples

```
gm0=matrix(1,8,8)  
diag(gm0)=0  
diag(gm0)=-rowSums(gm0)  
gm0[8,]=0  
  
plotM(gm0)
```

`print.gm`*Print Method for Generator Matrix Estimation Objects*

Description

Function for printing the results of a generator matrix estimation

Usage

```
## S3 method for class 'gm'  
print(x, ...)
```

Arguments

`x` a generator matrix estimation object
`...` additional arguments

See Also

[summary.gm](#), [plot.gm](#)

`print.gmci`*Print Method for Generator Matrix Confidence / Credibility Interval Objects*

Description

Function for printing the boundaries of a generator matrix confidence / credibility interval

Usage

```
## S3 method for class 'gmci'  
print(x, ...)
```

Arguments

`x` a generator matrix confidence / credibility interval
`...` additional arguments

See Also

[plot.gmci](#)

rNijTRiT_ModRej *C++ Based Modified Rejection Sampling*

Description

Function for generating initial and endpoint-conditioned Markov process sampling paths for a given discrete-time transition matrix

Usage

```
rNijTRiT_ModRej(tmabs, te, gm)
```

Arguments

tmabs	matrix of absolute transition frequencies
te	time elapsed in transition process
gm	generator matrix

Details

Function for the simulation of paths from an endpoint-conditioned Markov process. Returns number of transitions NijT and cumulative holding times RiT.

Author(s)

Jon Fintzi, Marius Pfeuffer

References

J. Fintzi: R Package ECctmc, 2016.

A. Hobolth and E. A. Stone: Simulation from Endpoint-Conditioned, Continuous-Time Markov Chains on a Finite State Space, with Applications to Molecular Evolution. *Annals of Applied Statistics* 3(3):1204-1231, 2009

Examples

```
data(tm_abs)

## Initial guess for generator matrix (absorbing default state)
data(tm_abs)

gm=matrix(1,8,8)
diag(gm)=0
diag(gm)=-rowSums(gm)
gm[8,]=0

rNijTRiT_ModRej(tm_abs,1,gm)
```

`rNijTRiT_Unif`*C++ Based Uniformization Sampling*

Description

Function for generating initial and endpoint-conditioned Markov process sampling paths for a given discrete-time transition matrix

Usage

```
rNijTRiT_Unif(tmabs, te, gm, tpm)
```

Arguments

<code>tmabs</code>	matrix of absolute transition frequencies
<code>te</code>	time elapsed in transition process
<code>gm</code>	generator matrix
<code>tpm</code>	discrete-time transition probability matrix, matrix exponential of gm

Details

Function for the simulation of paths from an endpoint-conditioned Markov process. Returns number of transitions `NijT` and cumulative holding times `RiT`.

Author(s)

Jon Fintzi, Marius Pfeuffer

References

J. Fintzi: R Package ECctmc, 2016.

A. Hobolth and E. A. Stone: Simulation from Endpoint-Conditioned, Continuous-Time Markov Chains on a Finite State Space, with Applications to Molecular Evolution. *Annals of Applied Statistics* 3(3):1204-1231, 2009

Examples

```
data(tm_abs)

## Generator Matrix
gm=matrix(1,8,8)
diag(gm)=0
diag(gm)=-rowSums(gm)
gm[8,]=0

## Transition Probability Matrix
library(expm)
```

```

te=1
tpm=expm(gm*te)

rNijTRiT_Unif(tm_abs, te, gm, tpm)

```

summary.gm

Extended Output for Generator Matrix Estimate Objects

Description

Function for providing results and extended output of a generator matrix estimation procedure.

Usage

```

## S3 method for class 'gm'
summary(object, ...)

```

Arguments

```

object      a generator matrix estimation object
...         additional arguments

```

See Also

[print.gm](#), [plot.gm](#)

tm_abs

Single Year Corporate Credit Rating Transitions

Description

Matrix of Standard and Poor's Global Corporate Rating Transition Frequencies 2000 (NR Removed)

Usage

```

data("tm_abs")

```

Format

The format is: num [1:8, 1:8] 17 2 0 0 0 0 0 1 455 ... - attr(*, "dimnames")=List of 2 ..\$: chr [1:8] "AAA" "AA" "A" "BBB"\$: chr [1:8] "AAA" "AA" "A" "BBB" ...

References

European Securities and Markets Authority, 2016
<https://cerep.esma.europa.eu/cerep-web/statistics/transitionMatrice.xhtml>

Examples

```
data(tm_abs)

## Matrix of relative transition frequencies
tm_rel=rbind((tm_abs/rowSums(tm_abs))[1:7,],c(rep(0,7),1))
tm_rel
```

Index

*Topic **datasets**

tm_abs, [21](#)

*Topic **package**

ctmcd-package, [2](#)

ciEM, [3](#), [9](#)

ciGS, [4](#), [9](#)

ctmcd (ctmcd-package), [2](#)

ctmcd-package, [2](#)

gm, [5](#)

gm.default, [5](#), [5](#)

gmci, [7](#)

gmci.default, [8](#), [8](#)

gmDA, [7](#), [9](#)

gmEM, [7](#), [10](#)

gmGS, [7](#), [11](#)

gmQO, [7](#), [12](#)

gmWA, [7](#), [13](#)

plot.gm, [14](#), [17](#), [18](#), [21](#)

plot.gmci, [15](#), [17](#), [18](#)

plotM, [15](#), [16](#), [17](#)

print.gm, [15](#), [18](#), [21](#)

print.gmci, [16](#), [18](#)

rNijTRiT_ModRej, [12](#), [19](#)

rNijTRiT_Unif, [12](#), [20](#)

summary.gm, [15](#), [18](#), [21](#)

tm_abs, [21](#)