Package ‘CreditMetrics’

February 19, 2015

Version 0.0-2

Date 2007-01-02

Title Functions for calculating the CreditMetrics risk model

Author Andreas Wittmann <andreas_wittmann@gmx.de>

Maintainer Andreas Wittmann <andreas_wittmann@gmx.de>

Depends R (>= 2.2.0)

Description A set of functions for computing the CreditMetrics risk model

License Unlimited

NeedsCompilation no

Repository CRAN

Date/Publication 2009-02-01 13:46:28

R topics documented:

cm.cs *Computation of credit spreads*

Description

`cm.cs` computes the credit spreads for each rating of a one year empirical migration matrix. The failure limit is the quantile of the failure probability.

Usage

```
cm.cs(M, lgd)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.
`lgd`	loss given default

Details

This function computes the credit spreads for each rating of a given one year empirical migration matrix with a default class in the last row. The credit spread is the risk premium demanded by the market.

According migration the nominal is differently calculated

$$V_0 = V_t e^{-(r_t + CS_t)t}$$

where t is the time. Under a riskless probability measure the value of a credit position at time t is computed as

$$V_0 = E[V_t] e^{-r_t t}$$

The default event is bernoulli distributed, so the expected value is

$$E[V_t] = V_t(1 - PD_t) + V_t(1 - LGD)PD_t$$

By using the above equations and following transformation, we get the formula for the credit spread

$$CS_t = -(\ln(1 - LGDPD_t))/t$$

This function computes the credit spread for t = 1, this is the credit spread for one year is calculated.

Value

Return value is the credit spread for time t = 1 of each rating in the migration matrix.

Author(s)

Andreas Wittmann `<andreas\_wittmann@gmx.de>`

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix

Examples

```
lgd <- 0.45

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.cs(M, lgd)
```

---

cm.CVaR *Computation of the Credit Value at Risk (CVaR)*

Description

cm.CVaR computes the credit value at risk for the simulated profits and losses.

Usage

```
cm.CVaR(M, lgd, ead, N, n, r, rho, alpha, rating)
```

Arguments

M	one year empirical migration matrix, where the last row gives the default class.
lgd	loss given default
ead	exposure at default
N	number of companies
n	number of simulated random numbers
r	riskless interest rate
rho	correlation matrix
alpha	confidence level
rating	rating of companies

Details

With function cm.gain one gets the profit and loss distribution of the credit positions. By building the quantile at confidence level α the credit value at risk can be reached.

Value

Return value is the credit value at risk at confidence level α.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, cm.gain, quantile

Examples

```
N <- 3
n <- 50000
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
lgd <- 0.45
rating <- c("BBB", "AA", "B")
firmnames <- c("firm 1", "firm 2", "firm 3")
alpha <- 0.99

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.CVaR(M, lgd, ead, N, n, r, rho, alpha, rating)
```

`cm.gain` *Computation of simulated profits and losses*

Description

`cm.gain` computes profits or losses, this is done by building the difference from the reference value and the simulated portfolio values of the credit positions.

Usage

```
cm.gain(M, lgd, ead, N, n, r, rho, rating)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.
`lgd`	loss given default
`ead`	exposure at default
`N`	number of companies
`n`	number of simulated random numbers
`r`	riskless interest rate
`rho`	correlation matrix
`rating`	rating of companies

Details

This function uses `cm.portfolio` and `cm.ref`. By building the difference of these functions, one gets the profits, if the difference is positive, or the losses, if the difference is negative.

Value

This functions returns simulated profits or losses.

Author(s)

Andreas Wittmann `<andreas\_wittmann@gmx.de>`

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

`cm.matrix`, `cm.ref`, `cm.portfolio`

Examples

```
N <- 3
n <- 50000
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
lgd <- 0.45
rating <- c("BBB", "AA", "B")
firmnames <- c("firm 1", "firm 2", "firm 3")

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.gain(M, lgd, ead, N, n, r, rho, rating)
```

---

cm.hist *Profit / Loss Distribution histogram*

Description

`cm.hist` plots a histogram for the simulated profit / loss distribution.

Usage

```
cm.hist(M, lgd, ead, N, n, r, rho, rating,
        col = "steelblue4", main = "Profit / Loss Distribution",
        xlab = "profit / loss", ylab = "frequency")
```

Arguments

M	one year empirical migration matrix, where the last row gives the default class.
lgd	loss given default
ead	exposure at default
N	number of companies

n	number of simulated random numbers
r	riskless interest rate
rho	correlation matrix
rating	rating of companies
col	a colour to be used to fill the bars, the default is 'steelblue4'.
main	an overall title for the plot, the default is 'Profit / Loss Distribution'.
xlab	a title for the x axis, the default is 'profit / loss'.
ylab	a title for the y axis, the defualt is 'frequency'.

Details

This function gives a histogram of the simulated profits and losses. The 'breaks' of the histogram are obtained through the minimum and the maximum of the simulated values and the number of simulated random numbers. This is

$$\text{breaks} = (\max(\text{SimGV}) - \min(\text{SimGV}))/2n$$

Value

A histogram of the the simulated profit and loss distribution.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, cm.gain, hist

Examples

```
N <- 3
n <- 50000
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
lgd <- 0.45
rating <- c("BBB", "AA", "B")
firmnames <- c("firm 1", "firm 2", "firm 3")

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)
```

```
# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.hist(M, lgd, ead, N, n, r, rho, rating,
        col = "steelblue4", main = "Profit / Loss Distribution",
        xlab = "profit / loss", ylab = "frequency")
```

`cm.matrix` *Testing for migration matrix*

Description

`cm.matrix` tests if the given matrix `M` is a migration matrix. So the dimensions of the migration matrix should be at least 2 times 2 and the row and column dimensions must be equal. Further the values in the migration matrix should be between 0 and 1. And the sum of each row should be 1.

Usage

```
cm.matrix(M)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.

Value

There is no return value if the given migration matrix `M` fullfills the above attributes.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

`is.matrix`

Examples

```
# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.matrix(M)
```

---

cm.portfolio — *Computation of simulated portfolio values*

Description

`cm.portfolio` computes simulated portfolio values by using function `cm.val`.

Usage

```
cm.portfolio(M, lgd, ead, N, n, r, rho, rating)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.
`lgd`	loss given default
`ead`	exposure at default
`N`	number of companies
`n`	number of simulated random numbers
`r`	riskless interest rate
`rho`	correlation matrix
`rating`	rating of companies

Details

The simulated portfolio values are computed by using the function `cm.val` and summing up each column.

Value

This functions returns the simulated portfolio values for each scenario.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, cm.val, colSums

Examples

```
N <- 3
n <- 50000
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
lgd <- 0.45
rating <- c("BBB", "AA", "B")
firmnames <- c("firm 1", "firm 2", "firm 3")

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.portfolio(M, lgd, ead, N, n, r, rho, rating)
```

---

cm.quantile *Computation of migration quantils*

Description

cm.quantile computes the empirical migration quantils for each rating of a one year empirical migration matrix. The failure limit is the quantile of the failure probability.

Usage

```
cm.quantile(M)
```

Arguments

M	one year empirical migration matrix, where the last row gives the default class.

Details

This function computes the empirical migration threshold value of a given one year empirical migration matrix with a default class in the last row. So the migration threshold can be computed with the migration probabilities. Migration quantiles have to be computed for each output rating.

The default threshold value S of the standard normal distribution with expectation 0 and standard deviation 1 gives

$$S = N^{-1}(PD)$$

where N^{-1} is the inverse function of the standard normal distribution and PD is the probability of default.

Thus an example for an BBB rated company is

$$S = N^{-1}(PD_{BBB})$$

So for each rating class thresholds can be computed.

Value

Return value is the quantile of each rating in the migration matrix.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, qnorm

Examples

```
# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
```

```
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.quantile(M)
```

---

cm.ref *Computation of reference value*

Description

`cm.ref` computes the value of a credit in one year for each rating, this is the return value `constVal`. Further the portfolio value at time t = 1 is computed, this is `constPV`.

Usage

```
cm.ref(M, lgd, ead, r, rating)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.
`lgd`	loss given default
`ead`	exposure at default
`r`	riskless interest rate
`rating`	rating of companies

Details

This function computes the value of the credit in one year, this is

$$V_t = EAD_t e^{-(r_t + CS_t)t}$$

where t = 1.

Value

a list containing following components:

`constVal`	credit value in one year
`constPV`	portfolio of all credit values in one year

Author(s)

Andreas Wittmann `<andreas\_wittmann@gmx.de>`

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, cm.cs

Examples

```
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
rating <- c("BBB", "AA", "B")
lgd <- 0.45

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.ref(M, lgd, ead, r, rating)
```

---

cm.rnorm — *Computation of standard normal distributed random numbers*

Description

cm.rnorm simulates standard normal distributed random numbers while using antithetic sampling.

Usage

```
cm.rnorm(N, n)
```

Arguments

N	number of simulations
n	number of simulated random numbers

Details

This function computes standard normal distributed random numbers with antithetic sampling. Here one has a sequence of standard normal distributed random numbers $(X_1, ..., X_{n/2})$. Reflected random numbers are computed with

$$X_i' = (-1)X_i$$

So the sequence $X_1', ..., X_{n/2}'$ is also standard normal distributed

Value

The function returns N simulations with n simulated random numbers each.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

matrix, rnorm

Examples

```
N <- 3
n <- 50000

cm.rnorm(N, n)
```

`cm.rnorm.cor` *Computation of correlated standard normal distributed random numbers*

Description

cm.rnorm.cor computes correlated standard normal distributed random numbers. This function uses a correlation matrix rho and later the cholesky decompositon in order to get correlated random numbers.

Usage

```
cm.rnorm.cor(N, n, rho)
```

Arguments

N	number of simulations
n	number of simulated random numbers
rho	correlation matrix

Details

This function computes standard normal distributed random numbers, which include the correlation matrix rho. One has a random matrix Y which is $N(0, 1)$ distributed. With the linear transformation $X = \mu + AY$ one gets X, which is $N(\mu, AA^T)$ distributed. If X should have the correlation matrix Σ. By using the cholesky decomposition the matrix A can be computed from Σ.

Value

The function returns N simulations with n simulated random numbers each, which include the correlation matrix rho.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

eigen, chol, cm.rnorm

Examples

```
N <- 3
n <- 50000
firmnames <- c("firm 1", "firm 2", "firm 3")

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)

cm.rnorm.cor(N, n, rho)
```

---

cm.state *Computation of state space*

Description

cm.state computes a state space, this is at time t = 1 the credit positions of all companies for all migrations is calculated. This state space is needed for the later valuation for the credit positions of each scenario.

Usage

```
cm.state(M, lgd, ead, N, r)
```

Arguments

M	one year empirical migration matrix, where the last row gives the default class.
lgd	loss given default
ead	exposure at default
N	number of companies
r	riskless interest rate

Details

This function computes the value of the credits of each firm in one year, this is

$$V_t = EAD_t e^{-(r_t + CS_t)t}$$

where t = 1. Also the value for the default class is calculated, that is

$$V_t = EAD(1 - LGD)$$

Value

Return value is the matrix V for time t = 1 of each rating in the migration matrix including the credit values for all companies. The last column in the matrix V is the value for the default event of each company.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, cm.cs, matrix

Examples

```
N <- 3
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
lgd <- 0.45

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
```

```
                 0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                    0,     0,     0,     0,     0,     0,     0, 100
               )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.state(M, lgd, ead, N, r)
```

`cm.val` *Valuation for the credit positions of each scenario*

Description

`cm.val` performs a valuation for the credit positions of each scenario. This is an allocation in rating classes identification of the credit position values.

Usage

```
cm.val(M, lgd, ead, N, n, r, rho, rating)
```

Arguments

`M`	one year empirical migration matrix, where the last row gives the default class.
`lgd`	loss given default
`ead`	exposure at default
`N`	number of companies
`n`	number of simulated random numbers
`r`	riskless interest rate
`rho`	correlation matrix
`rating`	rating of companies

Details

According to the value V_t the company is located in an other rating class. This location is performed with the migration matrix by determining the thresholds. In order to implement a valuation at time t, the credit spreads must be computed. With these the nominal is risk adjusted calculated. For a portfolio with many credits correlations are included by simulating correlated company yield returns. So the simulated ratings for each firm at time t = 1 can be computed.

Value

Simulated values of the firms for each rating of each scenario.

Author(s)

Andreas Wittmann <andreas_wittmann@gmx.de>

References

Glasserman, Paul, Monte Carlo Methods in Financial Engineering, Springer 2004

See Also

cm.matrix, eigen, cm.state, cm.quantile, cm.rnorm.cor

Examples

```
N <- 3
n <- 50000
r <- 0.03
ead <- c(4000000, 1000000, 10000000)
lgd <- 0.45
rating <- c("BBB", "AA", "B")
firmnames <- c("firm 1", "firm 2", "firm 3")

# correlation matrix
rho <- matrix(c(  1, 0.4, 0.6,
                0.4,   1, 0.5,
                0.6, 0.5,   1), 3, 3, dimnames = list(firmnames, firmnames),
                byrow = TRUE)

# one year empirical migration matrix from standard&poors website
rc <- c("AAA", "AA", "A", "BBB", "BB", "B", "CCC", "D")
M <- matrix(c(90.81,  8.33,  0.68,  0.06,  0.08,  0.02,  0.01,   0.01,
               0.70, 90.65,  7.79,  0.64,  0.06,  0.13,  0.02,   0.01,
               0.09,  2.27, 91.05,  5.52,  0.74,  0.26,  0.01,   0.06,
               0.02,  0.33,  5.95, 85.93,  5.30,  1.17,  1.12,   0.18,
               0.03,  0.14,  0.67,  7.73, 80.53,  8.84,  1.00,   1.06,
               0.01,  0.11,  0.24,  0.43,  6.48, 83.46,  4.07,   5.20,
               0.21,     0,  0.22,  1.30,  2.38, 11.24, 64.86,  19.79,
                  0,     0,     0,     0,     0,     0,     0, 100
            )/100, 8, 8, dimnames = list(rc, rc), byrow = TRUE)

cm.val(M, lgd, ead, N, n, r, rho, rating)
```

Index