Package ‘ESGtoolkit’

February 19, 2015

Type Package

Title Toolkit for the simulation of financial assets and interest rates models.

Version 0.1

Date 2014-06-13

Author Jean-Charles Croix, Thierry Moudiki, Frederic Planchet, Wassim Youssef

Maintainer Thierry Moudiki <thierry.moudiki@gmail.com>

Description Toolkit for Monte Carlo simulations of financial assets and interest rates models, involved in an Economic Scenario Generator (ESG). The underlying simulation loops have been implemented in C++.

License GPL-2 | GPL-3

Depends CDVine, ggplot2, gridExtra, reshape2, ycircinterextra

Imports Rcpp(>= 0.11.0)

Suggests knitr, fOptions

LinkingTo Rcpp

VignetteBuilder knitr

Collate ’RcppExports.R’ ’code.R’

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-06-13 06:17:14

R topics documented:

ESGtoolkit-package ................................................. 2
esgcortest .......................................................... 3
esgdiscountfactor ................................................... 5
esgfwdrates ......................................................... 6
esgmartingaletest ................................................... 7
esgmccv .............................................................. 8
esgmcpprices ......................................................... 9
ESGtoolkit-package

Toolkit for financial assets and interest rates simulation.

Description

Toolkit for Monte Carlo simulation of financial assets and interest rates, involved in an Economic Scenario Generator (ESG).

Details

Package: ESGtoolkit
Type: Package
Version: 0.1
Date: 2014-06-13
License: GPL-2 | GPL-3

The main functions of the package are:

- `simdiff` for the simulation of diffusion processes.
- `simshocks` for the custom simulation of the gaussian shocks embedded into the diffusion processes.

There are also several functions for statistical tests on the simulations, and for visualization.

Author(s)

Jean-Charles Croix, Thierry Moudiki, Frederic Planchet, Wassim Youssef

Maintainer: Thierry Moudiki <thierry.moudiki@gmail.com>

References


---

esgcortest

**Correlation tests for the shocks**

**Description**

This function performs correlation tests for the shocks generated by `simshocks`.

**Usage**

```r
esgcortest(x, alternative = c("two.sided", "less", "greater"), method = c("pearson", "kendall", "spearman"), conf.level = 0.95)
```
Arguments

x            gaussian (bivariate) shocks, with correlation, generated by simshocks.
alternative indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less".
method      which correlation coefficient is to be used for the test: "pearson", "kendall", or "spearman".
conf.level  confidence level.

Value

a list with 2 components: estimated correlation coefficients, and confidence intervals for the estimated correlations.

Author(s)

Thierry Moudiki + stats package

References


See Also

esgplotbands

Examples

nb <- 500
s0.par1 <- simshocks(n = nb, horizon = 3, frequency = "semi", family = 1, par = 0.2)
s0.par2 <- simshocks(n = nb, horizon = 3, frequency = "semi", family = 1, par = 0.8)
(test1 <- esgcortest(s0.par1))
(test2 <- esgcortest(s0.par2))
par(mfrow=c(2, 1))
esgplotbands(test1)
esgplotbands(test2)
**esgdiscountfactor**  

*Stochastic discount factors or discounted values*

**Description**

This function provides calculation of stochastic discount factors or discounted values.

**Usage**

\[ \text{esgdiscountfactor}(r, X) \]

**Arguments**

- \( r \) the short rate, a numeric (constant rate) or a time series object
- \( X \) the asset's price, a numeric (constant payoff or asset price) or a time series object

**Details**

The function result is:

\[ X_t \exp\left( -\int_0^t r_s \, ds \right) \]

where \( X_t \) is an asset value at a given maturity \( t \), and \( (r_s)_s \) is the risk-free rate.

**Author(s)**

Thierry Moudiki

**See Also**

`esgmcpprices`, `esgmccv`

**Examples**

```r
kappa <- 1.5
V0 <- theta <- 0.04
sigma_v <- 0.2
theta1 <- kappa*theta
ttheta2 <- kappa
ttheta3 <- sigma_v

# OU
r <- simdiff(n = 10, horizon = 5,
             frequency = "quart",
             model = "OU",
x0 = V0, theta1 = theta1, theta2 = theta2, theta3 = theta3)
```
esgfwdrates

Instantaneous forward rates

Description

This function provides instantaneous forward rates. They can be used in no-arbitrage short rate models, to fit the yield curve exactly.

Usage

esgfwdrates(in.maturities, in.zerorates, n, horizon, out.frequency = c("annual", "semi-annual", "quarterly", "monthly", "weekly", "daily"), ...)

Arguments

in.maturities input maturities
in.zerorates input zero rates
n number of independent observations
horizon horizon of projection
out.frequency either "annual", "semi-annual", "quarterly", "monthly", "weekly", or "daily" (1, 1/2, 1/4, 1/12, 1/52, 1/252)
... additional parameters provided to ycinter

Author(s)

Thierry Moudiki

References


Examples

# Yield to maturities
txZC <- c(0.01422, 0.01309, 0.01380, 0.01549, 0.01747, 0.01940, 0.02104, 0.02236, 0.02348, 0.02446, 0.02535, 0.02614, 0.02679, 0.02727, 0.02760, 0.02779, 0.02787, 0.02786, 0.02776, 0.02762, 0.02745, 0.02727, 0.02707, 0.02686, 0.02663, 0.02640, 0.02618, 0.02597, 0.02578, 0.02563)

# Observed maturities
u <- 1:30
esgmartingaletest

Martingale and market consistency tests

Description

This function performs martingale and market consistency (t-)tests.

Usage

\[
esgmartingaletest(r, X, p0, alpha = 0.05)
\]

Arguments

- \( r \): a numeric or a time series object, the risk-free rate(s).
- \( X \): a time series object, containing payoffs or projected asset values.
- \( p0 \): a numeric or a vector or a univariate time series containing initial price(s) of an asset.
- \( alpha \): 1 - confidence level for the test. Default value is 0.05.

Value

The function result can be just displayed. Otherwise, you can get a list by an assignment, containing (for each maturity):

- the Student t values.
The function esgmccv computes and plots confidence intervals around the estimated average price, as functions of the number of simulations. It takes the following arguments:

- `r`: a numeric or a time series object, the risk-free rate(s).
- `X`: asset prices obtained with `simdiff`
- `maturity`: the corresponding maturity (optional). If missing, all the maturities available in `X` are used.
- `plot`: if TRUE (default), a plot of the convergence is displayed.
- `...`: additional parameters provided to `matplot`

```r
r0 <- 0.03
S0 <- 100

set.seed(10)
eps0 <- simshocks(n = 100, horizon = 3, frequency = "quart")
sim.GBM <- simdiff(n = 100, horizon = 3, frequency = "quart",
                   model = "GBM",
                   x0 = S0, theta1 = r0, theta2 = 0.1,
                   eps = eps0)

mc.test <- esgmartingaletest(r = r0, X = sim.GBM, p0 = S0, alpha = 0.05)
esgplotbands(mc.test)
```

**Description**

This function computes and plots confidence intervals around the estimated average price, as functions of the number of simulations.

**Usage**

```
esgmccv(r, X, maturity, plot = TRUE, ...)
```

**Arguments**

- `r`: a numeric or a time series object, the risk-free rate(s).
- `X`: asset prices obtained with `simdiff`
- `maturity`: the corresponding maturity (optional). If missing, all the maturities available in `X` are used.
- `plot`: if TRUE (default), a plot of the convergence is displayed.
- `...`: additional parameters provided to `matplot`
Details

Studying the convergence of the sample mean of:

\[ E[X_T \exp(-\int_0^T r_s ds)] \]

towards its true value.

Value

a list with estimated average prices and the confidence intervals around them.

Author(s)

Thierry Moudiki

Examples

```r
r <- .03
data.frame
set.seed(1)
esgm0 <- simshocks(n = 100, horizon = 5, frequency = "quart")
sim.GBM <- simdiff(n = 100, horizon = 5, frequency = "quart", model = "GBM",
                  x0 = 100, theta1 = 0.03, theta2 = 0.1,
esm = eps0)
# monte carlo prices
esgmprimes(r, sim.GBM)

# convergence to a specific price
(esgmccv(r, sim.GBM, 2))
```

Description

This function computes estimators (sample mean) of

\[ E[X_T \exp(-\int_0^T r_s ds)] \]

where \( X_T \) is an asset value at given maturities \( T \), and \((r_s)_s\) is the risk-free rate.

Usage

esgmprimes(r, X, maturity = NULL)
Arguments

- `r`: a numeric or a time series object, the risk-free rate(s).
- `X`: asset prices obtained with `simdiff`
- `maturity`: the corresponding maturity (optional). If missing, all the maturities available in `X` are used.

Author(s)

Thierry Moudiki

See Also

`esgdiscountfactor`, `esgmccv`

Examples

```r
# GBM
r <- 0.03

eps0 <- simshocks(n = 100, horizon = 5, frequency = "quart")
sim.GBM <- simdiff(n = 100, horizon = 5, frequency = "quart", model = "GBM",
                   x0 = 100, theta1 = 0.03, theta2 = 0.1,
                   eps = eps0)

# monte carlo prices
esgmcpprices(r, sim.GBM)

# monte carlo price for a given maturity
esgmcpprices(r, sim.GBM, 2)
```

---

**esgplotbands**

*Plot time series percentiles and confidence intervals*

Description

This function plots colored bands for time series percentiles and confidence intervals. You can use it for outputs from `simdiff`, `esgmartingaletest`, `esgcortest`.

Usage

```r
esgplotbands(x, ...)
```

Arguments

- `x`: a times series object
- `...`: additional (optional) parameters provided to `plot`
esgplotbands

Author(s)
Thierry Moudiki

See Also
esgplotts

Examples

# Times series

kappa <- 1.5
V0 <- theta <- 0.04
sigma <- 0.2
theta1 <- kappa*theta
theta2 <- kappa
theta3 <- sigma
x <- simdiff(n = 100, horizon = 5,
frequency = "quart",
model = "OU",
x0 = V0, theta1 = theta1, theta2 = theta2, theta3 = theta3)

par(mfrow=c(2,1))
esgplotbands(x, xlab = "time", ylab = "values")
matplot(time(x), x, type = 'l', xlab = "time", ylab = "series values")

# Martingale test

r0 <- 0.03
S0 <- 100
sigma0 <- 0.1
nbScenarios <- 100
horizon0 <- 10
eps0 <- simshocks(n = nbScenarios, horizon = horizon0, frequency = "quart",
method = "anti")
sim.GBM <- simdiff(n = nbScenarios, horizon = horizon0, frequency = "quart",
model = "GBM",
x0 = S0, theta1 = r0, theta2 = sigma0,
eps = eps0)

mc.test <- esgmartingaletest(r = r0, X = sim.GBM, p0 = S0, alpha = 0.05)
esgplotbands(mc.test)

# Correlation test

nb <- 500

s0.par1 <- simshocks(n = nb, horizon = 3, frequency = "semi",
family = 1, par = 0.2)
s0.par2 <- simshocks(n = nb, horizon = 3, frequency = "semi",
family = 1, par = 0.8)
esgplotshocks

Visualize the dependence between 2 gaussian shocks

Description
This function helps you in visualizing the dependence between 2 gaussian shocks.

Usage
esgplotshocks(x, y = NULL)

Arguments
x an output from simshocks, a list with 2 components.
y an output from simshocks, a list with 2 components (Optional).

Author(s)
Thierry Moudiki + some nice blogs :)

References

See Also
simshocks

Examples
# Number of risk factors
d <- 2

# Number of possible combinations of the risk factors
dd <- d*(d-1)/2

# Family : Gaussian copula
fam1 <- rep(1,dd)
# Correlation coefficients between the risk factors (d*(d-1)/2)
par0.1 <- 0.1
par0.2 <- -0.9
# Family: Rotated Clayton (180 degrees)
fam2 <- 13
par0.3 <- 2

# Family: Rotated Clayton (90 degrees)
fam3 <- 23
par0.4 <- -2

# Number of simulations
nb <- 500

# Simulation of shocks for the d risk factors
s0.par1 <- simshocks(n = nb, horizon = 4, family = fam1, par = par0.1)
s0.par2 <- simshocks(n = nb, horizon = 4, family = fam1, par = par0.2)
s0.par3 <- simshocks(n = nb, horizon = 4, family = fam2, par = par0.3)
s0.par4 <- simshocks(n = nb, horizon = 4, family = fam3, par = par0.4)

## Not run:
esgplotshocks(s0.par1, s0.par2)
esgplotshocks(s0.par2, s0.par3)
esgplotshocks(s0.par2, s0.par4)
esgplotshocks(s0.par1, s0.par4)
## End(Not run)

---

**esgplotts**  
*Plot time series objects*

**Description**

This function plots outputs from `simdiff`.

**Usage**

`esgplotts(x)`

**Arguments**

- `x` a time series object, an output from `simdiff`.

**Details**

For a large number of simulations, it’s preferable to use `esgplotbands` for a synthetic view by percentiles.
**simdiff**

**Simulation of diffusion processes.**

**Description**

This function makes simulations of diffusion processes, that are building blocks for various risk factors' models.

**Usage**

```r
simdiff(n, horizon,
  frequency = c("annual", "semi-annual", "quarterly", "monthly", "weekly", "daily"),
  model = c("GBM", "CIR", "OU"), x0, theta1 = NULL,
  theta2 = NULL, theta3 = NULL, lambda = NULL,
  mu.z = NULL, sigma.z = NULL, p = NULL, eta_up = NULL,
  eta_down = NULL, eps = NULL)
```

**Arguments**

- `n` number of independent observations.
- `horizon` horizon of projection.
- `frequency` either "annual", "semi-annual", "quarterly", "monthly", "weekly", or "daily" (1, 1/2, 1/4, 1/12, 1/52, 1/252).

**Examples**

```r
kappa <- 1.5
V0 <- theta <- 0.04
sigma <- 0.2
theta1 <- kappa*theta
teta2 <- kappa
teta3 <- sigma
x <- simdiff(n = 10, horizon = 5, frequency = "quart",
  model = "OU",
  x0 = V0, theta1 = theta1, theta2 = theta2, theta3 = theta3)
esgplottsx(x)
```
either Geometric Brownian motion-like ("GBM"), Cox-Ingersoll-Ross ("CIR"), or Ornstein-Uhlenbeck ("OU").

**GBM-like (GBM, Merton, Kou, Heston, Bates)**

\[ dX_t = \theta_1(t)X_t dt + \theta_2(t)X_t dW_t + X_t JdN_t \]

**CIR**

\[ dX_t = (\theta_1 - \theta_2 X_t)dt + \theta_3 \sqrt{X_t}dW_t \]

**Ornstein-Uhlenbeck**

\[ dX_t = (\theta_1 - \theta_2 X_t)dt + \theta_3 dW_t \]

Where \((W_t)\) is a standard brownian motion:

\[ dW_t \sim \epsilon \sqrt{dt} \]

and

\[ \epsilon \sim N(0, 1) \]

The \(\epsilon\) is a gaussian increment that can be an output from `simshocks`.

For 'GBM-like', \(\theta_1\) and \(\theta_2\) can be held constant, and the jumps part \(JdN_t\) is optional. In case the jumps are used, they arise following a Poisson process \((N_t)\), with intensities \(J\) drawn either from lognormal or asymmetric double-exponential distribution.

**Value**

a time series object.
Author(s)

Thierry Moudiki

References


See Also

simshocks, esgplotts

Examples

kappa <- 1.5
V0 <- theta <- 0.04
sigma_v <- 0.2
theta1 <- kappa*theta
theta2 <- kappa
theta3 <- sigma_v

# OU

sim.OU <- simdiff(n = 10, horizon = 5,
                   frequency = "quart",
                   model = "OU",
                   x0 = V0, theta1 = theta1, theta2 = theta2, theta3 = theta3)
head(sim.OU)
par(mfrow=c(2,1))
esgplotbands(sim.OU, xlab = "time", ylab = "values", main = "with esgplotbands")
matplot(time(sim.OU), sim.OU, type = 'l', main = "with matplot")

# OU with simulated shocks (check the dimensions)
simshocks

Underlying gaussian shocks for risk factors' simulation.

Description

This function makes simulations of correlated or dependent gaussian shocks for risk factors.
Usage

`simshocks(n, horizon, frequency = c("annual", "semi-annual", "quarterly", "monthly", "weekly", "daily"), method = c("classic", "antithetic", "mm", "hybridantimm", "TAG"), family = NULL, par = NULL, par2 = NULL, type = c("CVine", "DVine"))`

Arguments

- `n` number of independent observations for each risk factor.
- `horizon` horizon of projection.
- `frequency` either "annual", "semi-annual", "quarterly", "monthly", "weekly", or "daily" (1, 1/2, 1/4, 1/12, 1/52, 1/252).
- `method` either classic monte carlo, antithetic variates, moment matching, hybrid antithetic variates + moment matching or  "TAG" (see the 4th reference for the latter).
- `family` the same as `cdvinesim` from package `cdvine`. A d*(d-1)/2 integer vector of C-/D-vine pair-copula families with values 0 = independence copula, 1 = Gaussian copula, 2 = Student t copula (t-copula), 3 = Clayton copula, 4 = Gumbel copula, 5 = Frank copula, 6 = Joe copula, 7 = BB1 copula, 8 = BB6 copula, 9 = BB7 copula, 10 = BB8 copula, 13 = rotated Clayton copula (180 degrees; "survival Clayton"), 14 = rotated Gumbel copula (180 degrees; "survival Gumbel"), 16 = rotated Joe copula (180 degrees; "survival Joe"), 17 = rotated BB1 copula (180 degrees; "survival BB1"), 18 = rotated BB6 copula (180 degrees; "survival BB6"), 19 = rotated BB7 copula (180 degrees; "survival BB7"), 20 = rotated BB8 copula (180 degrees; "survival BB8"), 23 = rotated Clayton copula (90 degrees), 24 = rotated Gumbel copula (90 degrees), 26 = rotated Joe copula (90 degrees), 27 = rotated BB1 copula (90 degrees), 28 = rotated BB6 copula (90 degrees), 29 = rotated BB7 copula (90 degrees), 30 = rotated BB8 copula (90 degrees), 33 = rotated Clayton copula (270 degrees), 34 = rotated Gumbel copula (270 degrees), 36 = rotated Joe copula (270 degrees), 37 = rotated BB1 copula (270 degrees), 38 = rotated BB6 copula (270 degrees), 39 = rotated BB7 copula (270 degrees), 40 = rotated BB8 copula (270 degrees).
- `par` the same as `cdvinesim` from package `cdvine`. A d*(d-1)/2 vector of pair-copula parameters.
- `par2` the same as `cdvinesim` from package `cdvine`. A d*(d-1)/2 vector of second parameters for pair-copula families with two parameters (t, BB1, BB6, BB7, BB8; no default).
- `type` type of the vine model: 1 : C-vine 2 : D-vine

Details

The function shall be used along with `simdiff`, in order to embed correlated or dependent random gaussian shocks into simulated diffusions. `esgplotshocks` can help in visualizing the type of dependence between the shocks.
Value

If family and par are not provided, a univariate time series object with simulated gaussian shocks for one risk factor. Otherwise, a list of time series objects, containing gaussian shocks for each risk factor.

Author(s)

Thierry Moudiki

References


See Also

simdiff, esgplotshocks

Examples

```r
# Number of risk factors
d <- 6

# Number of possible combinations of the risk factors
dd <- d*(d-1)/2

# Family : Gaussian copula for all
fam1 <- rep(1,dd)

# Correlation coefficients between the risk factors (d*(d-1)/2)
par1 <- c(0.2,0.69,0.73,0.22,-0.09,0.51,0.32,0.01,0.82,0.01,
-0.2,-0.32,-0.19,-0.17,-0.06)

# Simulation of shocks for the 6 risk factors
simshocks(n = 10, horizon = 5, family = fam1, par = par1)

# Simulation of shocks for the 6 risk factors
# on a quarterly basis
simshocks(n = 10, frequency = "quarterly", horizon = 2, family = fam1,
```
par = par1)

# Simulation of shocks for the 6 risk factors simulation
# on a quarterly basis, with antithetic variates and moment matching.
s0 <- simshocks(n = 10, method = "hyb", horizon = 4,
family = fam1, par = par1)

colMeans(s0[[1]])
colMeans(s0[[2]])
apply(s0[[3]], 2, sd)
apply(s0[[4]], 2, sd)
Index

*Topic ESG, Economic Scenario Generator, Finance, Insurance, Risk Management

ESGtoolkit-package, 2

CDVineSim, 18

esgcortest, 3
esgdiscountfactor, 5, 10
esgfwdrates, 6
esgmartingaletest, 7
esgmccv, 5, 8, 10
esgmcprices, 5, 9
esgplotbands, 4, 8, 10, 13, 14
esgplotshocks, 12, 18, 19
esgplotts, 11, 13, 16
ESGtoolkit (ESGtoolkit-package), 2
ESGtoolkit-package, 2

matplot, 8

simdiff, 2, 8, 10, 13, 14, 18, 19
simshocks, 2–4, 12, 15, 16, 17

ycinter, 6