Package 'Jdmbs'

October 12, 2022

Type Package

Version 1.4

Title Monte Carlo Option Pricing Algorithms for Jump Diffusion Models with Correlational Companies

Description Option is a one of the financial derivatives and its pricing is an important problem in practice. The process of stock prices are represented as Geometric Brownian motion [Black (1973) <doi:10.1086/260062>] or jump diffusion processes [Kou (2002) <doi:10.1287/mnsc.48.8.1086.166>]. In this package, algorithms and visualizations are implemented by Monte Carlo method in order to calculate European option price for three equations by Geometric Brownian motion and jump diffusion processes and furthermore a model that presents jumps among companies affect each other.

Date 2020-07-23

Author Masashi Okada [aut, cre]

Maintainer Masashi Okada <okadaalgorithm@gmail.com>

Depends R (>= 3.6.0)

License GPL (>= 2)

Imports igraph, graphics, stats, utils, png, ggplot2

Suggests R.rsp

VignetteBuilder R.rsp

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

NeedsCompilation no

Repository CRAN

Date/Publication 2020-07-24 13:22:10 UTC

R topics documented:

`data` *correlation coefficients between all pair companies*

Description

A dataset containing a matrix of correlation coefficients between all pair companies. 6 row and 6 col.

Usage

```
data
```

Format

An object of class `function` of length 1.

`jdm_bs` *A Monte Carlo Option Pricing Algorithm for Jump Diffusion Model*

Description

A Monte Carlo Option Pricing Algorithm for Jump Diffusion Model

Usage

```
jdm_bs(
  day = 180,
  monte_carlo = 1000,
  start_price = start_price,
  mu = mu,
  sigma = sigma,
  lambda = lambda,
  K = K,
  plot = TRUE
)
```

Arguments

`day`	: an integer of a time duration of simulation.
`monte_carlo`	: an integer of an iteration number for monte carlo.
`start_price`	: a vector of company's initial stock prices.
`mu`	: a vector of drift parameters of geometric Brownian motion.
`sigma`	: a vector of volatility parameters of geometric Brownian motion.
`lambda`	: an integer of how many times jump in unit time.
`K`	: a vector of option strike prices.
`plot`	: a logical type of whether plot a result or not.

Value

option prices : a list of (call_price, put_price)

Examples

```
jdm_bs(100,10,c(5500,6500,8000),c(0.1,0.2,0.05),c(0.11,0.115,0.1),2,c(6000,7000,12000),plot=TRUE)
```

`jdm_new_bs` *A Monte Carlo Option Pricing Algorithm for Jump Diffusion Model with Correlational Companies*

Description

A Monte Carlo Option Pricing Algorithm for Jump Diffusion Model with Correlational Companies

Usage

```
jdm_new_bs(
  correlation_matrix,
  day = 180,
  monte_carlo = 1000,
  start_price = start_price,
  mu = mu,
  sigma = sigma,
  lambda = lambda,
  K = K,
  plot = TRUE
)
```

Arguments

`correlation_matrix`	: a matrix of a correlation coefficient of companies
`day`	: an integer of a time duration of simulation.
`monte_carlo`	: an integer of an iteration number for monte carlo.
`start_price`	: a vector of company's initial stock prices.
`mu`	: a vector of drift parameters of geometric Brownian motion.
`sigma`	: a vector of volatility parameters of geometric Brownian motion.
`lambda`	: an integer of how many times jump in unit time.
`K`	: a vector of option strike prices.
`plot`	: a logical type of whether plot a result or not.

Value

option prices : a list of (call_price, put_price)

Examples

```
price <- jdm_new_bs(matrix(c(0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9),nrow=3, ncol=3),
                    day=100,monte_carlo=20, c(1000,500,500),
                    c(0.002, 0.012, 0.005),c(0.05,0.05,0.06), 3,
                    c(1500,1000,700),plot=TRUE
                   )
```

`normal_bs` *A Normal Monte Carlo Option Pricing Algorithm*

Description

A Normal Monte Carlo Option Pricing Algorithm

Usage

```
normal_bs(
  day = 100,
  monte_carlo = 1000,
  start_price = start_price,
  mu = mu,
  sigma = sigma,
  K = K,
  plot = TRUE
)
```

Arguments

`day`	: an integer of a time duration of simulation.
`monte_carlo`	: an integer of an iteration number for monte carlo.
`start_price`	: a vector of company's initial stock prices.
`mu`	: a vector of drift parameters of geometric Brownian motion.
`sigma`	: a vector of volatility parameters of geometric Brownian motion.
`K`	: a vector of option strike prices.
`plot`	: a logical type of whether plot a result or not.

Value

option prices : a list of (call_price, put_price)

Examples

```
price <- normal_bs(100,10,c(300,500,850),c(0.1,0.2,0.05),c(0.05,0.1,0.09),c(600,700,1200),plot=TRUE)
```

Index