Package ‘OptionPricing’

October 12, 2022

Type Package

Title Option Pricing with Efficient Simulation Algorithms

Version 0.1.1

Date 2022-04-16

Author Kemal Dingec, Wolfgang Hormann

Maintainer Wolfgang Hormann <hormannw@boun.edu.tr>

Description Efficient Monte Carlo Algorithms for the price and the sensitivities of Asian and European Options under Geometric Brownian Motion.

License GPL-2 | GPL-3

Copyright Wolfgang Hormann

NeedsCompilation no

Repository CRAN

Date/Publication 2022-04-15 15:42:32 UTC

R topics documented:

`OptionPricing-package` *Option Pricing and Greeks Estimation for Asian and European Options*

Description

The Price, Delta and Gamma of European and Asian Options under Geometric Brownian Motion are calculated using the Black-Scholes formula and Efficient Monte Carlo and Randomized Quasi Monte Carlo Algorithms.

Details

Package:	OptionPricing
Type:	Package
Version:	0.1
Date:	2014-11-07
License:	License: GPL-2 | GPL-3
Copyright:	Wolfgang Hormann

The OptionPricing package calculates the Price, Delta and Gamma for European options using the Black-Scholes formula (see BS_EC). The price, Delta and Gamma for Asian call options under geometric Brownian motion are calculated using a very efficient Monte Carlo and randomized quasi-Monte Carlo algorithm (see AsianCall). The function AsianCall_AppLord implements a high-quality approximation for the price of an Asian option.

Author(s)

Kemal Dingec, Wolfgang Hormann

Examples

```
# standard settings for an efficient simulation using QMC and variance reduction
AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method=c("best"),
  sampling=c("QMC"),metpar=list(maxiter=100,tol=1.e-14,cvmethod="splitting"),
  sampar=list(nout=50,n=2039,a=1487,baker=TRUE,genmethod="pca"))

# Calculation of the Price of an Asian option using a good approximation
  AsianCall_AppLord(T = 1, d = 12, K = 100, r = 0.05, sigma = 0.2, S0 = 100)


# standard settings for an efficient simulation using MC and variance reduction
AsianCall(T=1,d=12,K=170,r=0.05,sigma=0.2,S0=100,method="best",
          sampling="MC",metpar=list(maxiter=100,tol=1.e-14,np=1000),
  sampar=list(n=10^5))
# Calculation of the approximate price, a bit different to the above result
  AsianCall_AppLord(T = 1, d = 12, K = 170, r = 0.05, sigma = 0.2, S0 = 100)

# Calculation of the Price of an Asian option using a good approximation
  AsianCall_AppLord(T = 1, d = 12, K = 100, r = 0.05, sigma = 0.2, S0 = 100)

#Price, Delta and Gamma of European options using Black-Scholes
BS_EC(K=100, r = 0.05, sigma = 0.2, T = 0.25, S0 = 100)
BS_EP(K=100, r = 0.05, sigma = 0.2, T = 0.25, S0 = 100)
```

AsianCall *Calculates the Price, Delta and Gamma of an Asian Option*

Description

Prices arithmetic average Asian Call options under geometric Brownian motion. It also estimates the sensitivities Delta and Gamma.

Usage

```
AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method=c("best","naive"),
          sampling=c("QMC","MC"),
          metpar=list(maxiter=100,tol=1.e-14,cvmethod="splitting"),
          sampar=list(nout=50,seq.type="korobov",n=2039,a=1487,
                      baker=TRUE,genmethod="pca"))
```

Arguments

`T`	time to maturity (in years)
`d`	number of control points
`K`	strike price
`r`	risk free interest rate
`sigma`	volatility
`S0`	starting stockprice
`method`	selects the simulation method; method `"best"` uses a variance reduction method based on effective control variates and conditional Monte Carlo and is very effective. method `"naive"` is mainly provided for comparison purposes.
`sampling`	sampling `QMC` uses the Quasi Monte Carlo method Korobov lattice for the simulation. sampling `MC` uses standard Monte Carlo for the simulation.
`metpar`	list holding extra parameters related to the simulation method For `method="best"`: `maxiter=` maximal no of iterations for Newton method `tol=` error tolerance for Newton method for `sampling="QMC"`: `cvmethod=c("splitting","direct")` NOT necessary for method = "naive" "splitting" ... estimates CV coefficients using lm with bootstrap "direct" ... estimates CV coefficients using lm and the full sample for `sampling="MC"`: `np` ... sample size for pilot run for CV; NOT necessary for method = "naive"
`sampar`	list holding several parameters related to the sampling method; for `sampling="MC"` the list `sampar` only contains the total samplesize `n`; for `sampling="QMC"` the list `sampar` contains the elements: `nout` number of independent "randomized" copies of the Korobov lattice `n` number of points of the Korobov lattice

a important constant for the construction of the Korobov lattice

baker TRUE/FALSE, indicates if Baker transform should be used for making the integrand periodic

genmethod = c("pca", "std","pcamain","lt","ltpca"), note that for method=="naive" only genmethod=c("pca","std") can be used.

genmethod="pca" principal component analysis

genmethod="std" standard

genmethod="pcamain" use only first dirnum main directions of the PCA

genmethod="lt" uses a transform for the first dirnum

genmethod="ltpca" combination of lt with pca

dirnum number of main directions, only used for genmethod="pcamain" or "lt"

Details

Method best (see the reference Dingec and Hormann below) is a very efficient simulation algorithm using multiple Control Variates and conditional MonteCarlo to calculate the the price, delta and gamma of Asian call options under geometric Brownian motion. It is especially effective when QMC is selected as sampling method. As QMC method Korobov Lattice rules are used. For good parameter values see Table 1 of (L'Ecuyer, Lemieux).

Value

returns a matrix holding the price and greeks. The estimated Asian Call price and its estimated delta and gamma form the first column vector, the respective 95 percent error bounds are given in the second column.

Author(s)

Kemal Dingec, Wolfgang Hormann

References

K. D. Dingec and W. Hormann. Improved Monte Carlo and Quasi-Monte Carlo Methods for the Price and the Greeks of Asian Options, Proceedings of the 2014 Winter Simulation Conference A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

L'Ecuyer, P., and C. Lemieux. 2000. Variance Reduction via Lattice Rules. Management Science 46 (9): 1214-1235.

See Also

OptionPricing-package

Examples

```
# standard settings for an efficient simulation using QMC and variance reduction
AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method="best",
  sampling="QMC",metpar=list(maxiter=100,tol=1.e-14,cvmethod="splitting"),
  sampar=list(nout=50,n=2039,a=1487,baker=TRUE,genmethod="pca"))

# efficient Monte Carlo version of the above simulation
AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method="best",
  sampling="MC",metpar=list(maxiter=100,tol=1.e-14,np=1000),
  sampar=list(n=10^5))

# simple QMC version without variance reduction

AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method="naive",
  sampling="QMC",
  sampar=list(nout=50,n=2039,a=1487,baker=TRUE,genmethod="pca"))

# naive Monte Carlo version
AsianCall(T=1,d=12,K=100,r=0.05,sigma=0.2,S0=100,method="naive",
  sampling="MC",sampar=list(n=10^5))
```

AsianCall_AppLord *Asian Options - Approximation*

Description

The price of an arithmetic average Asian option is computed using the approximation method of Lord.

Usage

```
AsianCall_AppLord(T=1, d=12, K=100, r=0.05, sigma=0.1, S0=100, all=TRUE)
```

Arguments

T	T time to maturity (in years)
d	d number of controll points
K	K strike price
r	r risk free interest rate
sigma	sigma volatility (yearly)
S0	S0 starting stockprice
all	all TRUE means that the full Asian Call option price is approximated

Details

`AsianCall_AppLord()` uses a sophisticated approximation of Lord (2006).

Value

returns the approximate price.

Author(s)

Kemal Dingec, Wolfgang Hormann

References

Lord, R., Partially Exact and Bounded Approximations for Arithmetic Asian Options, Journal of Computational Finance, Vol. 10, No. 2, pp. 1-52, 2006

See Also

OptionPricing-package

Examples

```
AsianCall_AppLord(T = 1, d = 12, K = 100, r = 0.05, sigma = 0.25, S0 = 100, all = TRUE)
```

BS_EC *Black-Scholes Formula for European Call and Put*

Description

Calculates the Price, Delta and Gamma of an European Call or Put option using the Black-Scholes formula.

Usage

```
BS_EC( T = 0.25, K = 100, r = 0.05, sigma = 0.2, S0 = 100 )
BS_EP( T = 0.25, K = 100, r = 0.05, sigma = 0.2, S0 = 100 )
```

Arguments

T	time to maturity (in years)
K	Strike Price
r	risk-free interest rate
sigma	yearly volatility
S0	Starting Stock Price

Value

Returns a vector containing the option price, Delta and Gamma

Author(s)

Wolfgang Hormann

See Also

`OptionPricing-package`

Examples

```
BS_EC(K=100, r = 0.05, sigma = 0.2, T = 0.25, S0 = 100)
BS_EP(K=100, r = 0.05, sigma = 0.2, T = 0.25, S0 = 100)
```

Index