Package ‘greeks’

August 26, 2022

Title Sensitivities of Prices of Financial Options and Implied Volatilites

Version 0.7.0

Description Methods to calculate sensitivities of financial option prices for European, Asian, American and Digital Options options in the Black Scholes model, and in more general jump diffusion models. Furthermore, methods to compute implied volatilities are provided for a wide range of option types and custom payoff functions. Classical formulas are implemented for European options in the Black Scholes Model, as is presented in Hull, J. C. (2017). Options, Futures, and Other Derivatives, Global Edition (9th Edition). Pearson. In the case of Asian options, Malliavin Monte Carlo Greeks are implemented, see Hudde, A. & Rüschendorf, L. (2016). European and Asian Greeks for exponential Lévy processes. <arXiv:1603.00920>. For American options, the Binomial Tree Method is implemented, as is presented in Hull, J. C. (2017).

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.0

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Imports magrittr, dqrng, Rcpp

LinkingTo Rcpp

NeedsCompilation yes

Author Anselm Hudde [aut, cre] (<https://orcid.org/0000-0002-5652-2815>)

Maintainer Anselm Hudde <anselmhudde@gmx.de>

Repository CRAN

Date/Publication 2022-08-26 19:12:09 UTC

R topics documented:

`Binomial_American_Greeks` *Computes the Greeks of an American call- or put-option with the Binomial options pricing model*

Description

Computes the Greeks of an American call- or put-option with the Binomial options pricing model

Usage

```
Binomial_American_Greeks(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "gamma"),
  steps = 1000,
  eps = 1/10000
)
```

Arguments

`initial_price`	• initial price of the underlying asset.
`exercise_price`	• strike price of the option.
`r`	• risk-free interest rate.
`time_to_maturity`	• time to maturity.
`volatility`	• volatility of the underlying asset.
`dividend_yield`	• dividend yield.
`payoff`	• the payoff function, a string in ("call", "put").
`greek`	• the Greek to be calculated.
`steps`	• the number of integration steps.
`eps`	• the step size for the finite difference method to calculate theta, vega, rho and epsilon

Value

Named vector containing the values of the Greeks specified in the parameter greek.

Examples

```
Binomial_American_Greeks(initial_price = 100, exercise_price = 100,
r = 0, time_to_maturity = 1, volatility = 0.3, dividend_yield = 0,
payoff = "call", greek = c("fair_value", "delta", "vega", "theta", "rho",
"epsilon", "gamma"), steps = 20)
```

BS_European_Greeks — *Computes the Greeks of an European call- or put-option, or of digital options in the Black Scholes model*

Description

Computes the Greeks of an European call- or put-option, or of digital options in the Black Scholes model

Usage

```
BS_European_Greeks(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "lambda",
    "gamma", "vanna", "charm", "vomma", "veta", "speed")
)
```

Arguments

initial_price	• initial price of the underlying asset
exercise_price	• strike price of the option
r	• risk-free interest rate
time_to_maturity	• time to maturity in years
volatility	• volatility of the underlying asset
dividend_yield	• dividend yield
payoff	• in c("call", "put", "cash_or_nothing_call", "cash_or_nothing_put", "asset_or_nothing_call", "asset_or_nothing_put")
greek	• Greeks to be calculated in c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "lambda", "gamma", "vanna", "charm", "vomma", "veta", "vera", "speed", "zomma", "color", "ultima")

Value

Named vector containing the values of the Greeks specified in the parameter greek.

Examples

```
BS_European_Greeks(initial_price = 120, exercise_price = 100,
r = 0.02, time_to_maturity = 4.5, dividend_yield = 0.015, volatility = 0.22,
greek = c("fair_value", "delta", "gamma"), payoff = "put")
```

BS_European_Greeks_test   *Computes the Greeks of an European call- or put-option in the Black Scholes model*

Description

Computes the Greeks of an European call- or put-option in the Black Scholes model

Usage

```
BS_European_Greeks_test(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "lambda",
    "gamma", "vanna", "charm", "vomma", "veta", "speed")
)
```

Arguments

initial_price
- initial price of the underlying asset

exercise_price
- strike price of the option

r
- risk-free interest rate

time_to_maturity
- time to maturity in years

volatility
- volatility of the underlying asset

dividend_yield
- dividend yield

payoff
- in "call", "put", "cash_or_nothing_call", "cash_or_nothing_put", "asset_or_nothing_call", "asset_or_nothing_put"

greek
- Greeks to be calculated in c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "lambda", "gamma", "vanna", "charm", "vomma", "veta", "speed")

Value

Named vector containing the values of the greeks specified in the parameter `greek`.

Examples

```
BS_European_Greeks(initial_price = 120, exercise_price = 100,
r = 0.02, time_to_maturity = 4.5, dividend_yield = 0.015, volatility = 0.22,
greek = c("fair_value", "delta", "gamma"), payoff = "put")
```

`BS_Implied_Volatility` *Computes the implied volatility for European-, American- and Asian options.*

Description

Computes the implied volatility for European-, American- and Asian options.

Usage

```
BS_Implied_Volatility(
  option_price,
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  dividend_yield = 0,
  payoff = "call",
  start_volatility = 0.3,
  precision = 1e-09
)
```

Arguments

`option_price`	• current price of the option
`initial_price`	• initial price of the underlying asset.
`exercise_price`	• strike price of the option.
`r`	• risk-free interest rate.
`time_to_maturity`	• time to maturity.
`dividend_yield`	• dividend yield.
`payoff`	• the payoff function, a string in ("call", "put").
`start_volatility`	• the volatility value to start the approximation
`precision`	• precision of the result

Value

Named vector containing the values of the Greeks specified in the parameter greek.

Examples

```
BS_Implied_Volatility(option_price = 27, initial_price = 100,
exercise_price = 100, r = 0.03, time_to_maturity = 5, dividend_yield = 0.015,
payoff = "call")
```

Greeks *Computes the Greeks of various options*

Description

Computes the Greeks of various options

Usage

```
Greeks(
  initial_price,
  exercise_price,
  r,
  time_to_maturity,
  volatility,
  dividend_yield = 0,
  model = "Black_Scholes",
  option_type = "European",
  payoff = "call",
  greek = c("fair_value", "delta", "vega", "theta", "rho", "gamma")
)
```

Arguments

initial_price	• initial price of the underlying asset
exercise_price	• strike price of the option
r	• risk-free interest rate
time_to_maturity	• time to maturity in years
volatility	• volatility of the underlying asset
dividend_yield	• dividend yield
model	• the model to be chosen
option_type	in c("European", "American", "Asian", "Digital", "Binomial) - the type of option to be considered
payoff	• in c("call", "put", "cash_or_nothing_call", "cash_or_nothing_put", "asset_or_nothing_call", "asset_or_nothing_put")

`greek`
- Greeks to be calculated in c("fair_value", "delta", "vega", "theta", "rho", "epsilon", "lambda", "gamma", "vanna", "charm", "vomma", "veta", "vera", "speed", "zomma", "color", "ultima")

Value

Named vector containing the values of the Greeks specified in the parameter `greek`.

`Implied_Volatility` *Computes the implied volatility for various options via Newton's method*

Description

Computes the implied volatility for various options via Newton's method

Usage

```
Implied_Volatility(
  option_price,
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  dividend_yield = 0,
  model = "Black_Scholes",
  option_type = "European",
  payoff = "call",
  start_volatility = 0.3,
  precision = 1e-06,
  max_iter = 30
)
```

Arguments

`option_price`
- current price of the option

`initial_price`
- initial price of the underlying asset

`exercise_price`
- strike price of the option

`r`
- risk-free interest rate

`time_to_maturity`
- time to maturity in years

`dividend_yield`
- dividend yield

`model`
- the model to be chosen

`option_type` in c("European", "American", "Asian", "Digital") - the type of option to be considered

`payoff`
: • in c("call", "put")

`start_volatility`
: initial guess

`precision`
: precision of the computation

`max_iter`
: maximal number of iterations of the approximation

Value

Named vector containing the values of the greeks specified in the parameter `greek`.

Examples

```
Implied_Volatility(15, r = 0.05, option_type = "Asian",
payoff = "call")
```

Malliavin_Asian_Greeks

Computes the Greeks of an Asian option with the Malliavin Monte Carlo Method in the Black Scholes model

---

Description

Computes the Greeks of an Asian option with the Malliavin Monte Carlo Method in the Black Scholes model

Usage

```
Malliavin_Asian_Greeks(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "rho", "vega", "theta", "gamma"),
  model = "black_scholes",
  lambda = 0.2,
  alpha = 0.3,
  jump_distribution = function(n) stats::rt(n, df = 3),
  steps = round(time_to_maturity * 252),
  paths = 10000,
  seed = 1,
  antithetic = FALSE
)
```

Arguments

`initial_price`	• initial price of the underlying asset
`exercise_price`	• strike price of the option
`r`	• risk-free interest rate
`time_to_maturity`	• time to maturity in years
`volatility`	• volatility of the underlying asset
`dividend_yield`	• dividend yield
`payoff`	• the payoff function, either a string in ("call", "put", "digital_call", "digital_put"), or a function
`greek`	• the Greek to be calculated
`model`	• the model to be chosen in ("black_scholes", "jump_diffusion")
`lambda`	• the lambda of the Poisson process in the jump-diffusion model
`alpha`	• the alpha in the jump-diffusion model influences the jump size
`jump_distribution`	• the distribution of the jumps, choose a function which generates random numbers with the desired distribution
`steps`	• the number of integration steps
`paths`	• the number of simulated paths
`seed`	• the seed of the random number generator
`antithetic`	• if TRUE, antithetic random numbers will be chosen to decrease variance

Value

Named vector containing the values of the Greeks specified in the parameter `greek`.

Examples

```
Malliavin_Asian_Greeks(initial_price = 110, exercise_price = 100,
r = 0.02, time_to_maturity = 4.5, dividend_yield = 0.015, volatility = 0.22,
greek = c("fair_value", "delta", "rho"), payoff = "put")
```

`Malliavin_Asian_Greeks_Black_Scholes`

Computes the Greeks of an Asian option with the Malliavin Monte Carlo Method in the Black Scholes model

---

Description

Computes the Greeks of an Asian option with the Malliavin Monte Carlo Method in the Black Scholes model

Usage

```
Malliavin_Asian_Greeks_Black_Scholes(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "rho", "vega", "theta", "gamma"),
  steps = round(time_to_maturity * 252),
  paths = 10000,
  seed = 1,
  antithetic = FALSE
)
```

Arguments

`initial_price`	• initial price of the underlying asset
`exercise_price`	• strike price of the option
`r`	• risk-free interest rate
`time_to_maturity`	• time to maturity in years
`volatility`	• volatility of the underlying asset
`dividend_yield`	• dividend yield
`payoff`	• the payoff function, either a string in ("call", "put", "digital_call", "digital_put"), or a function
`greek`	• the Greek to be calculated
`steps`	• the number of integration steps
`paths`	• the number of simulated paths
`seed`	• the seed of the random number generator
`antithetic`	• if TRUE, antithetic random numbers will be chosen to decrease variance

Value

Named vector containing the values of the Greeks specified in the parameter `greek`.

Examples

```
Malliavin_Asian_Greeks(initial_price = 110, exercise_price = 100,
r = 0.02, time_to_maturity = 4.5, dividend_yield = 0.015, volatility = 0.22,
greek = c("fair_value", "delta", "rho"), payoff = "put")
```

Malliavin_European_Greeks
Computes the Greeks of an European option with the Malliavin Monte Carlo Method in the Black Scholes model

Description

Computes the Greeks of an European option with the Malliavin Monte Carlo Method in the Black Scholes model

Usage

```
Malliavin_European_Greeks(
  initial_price = 100,
  exercise_price = 100,
  r = 0,
  time_to_maturity = 1,
  volatility = 0.3,
  dividend_yield = 0,
  payoff = "call",
  greek = c("fair_value", "delta", "vega", "theta", "rho", "gamma"),
  model = "Black Scholes",
  paths = 10000,
  seed = 1,
  antithetic = FALSE
)
```

Arguments

`initial_price`	• initial price of the underlying asset
`exercise_price`	• strike price of the option
`r`	• risk-free interest rate
`time_to_maturity`	• time to maturity in years
`volatility`	• volatility of the underlying asset
`dividend_yield`	• dividend yield
`payoff`	• the payoff function, either a string in ("call", "put", "digital_call", "digital_put"), or a function
`greek`	• the greek to be calculated
`model`	• the model to be chosen
`paths`	• the number of simulated paths
`seed`	• the seed of the random number generator
`antithetic`	• if TRUE, antithetic random numbers will be chosen to decrease variance

Value

Named vector containing the values of the Greeks specified in the parameter `greek`

Examples

```
Malliavin_European_Greeks(initial_price = 110, exercise_price = 100,
r = 0.02, time_to_maturity = 4.5, dividend_yield = 0.015, volatility = 0.22,
greek = c("fair_value", "delta", "rho"), payoff = "put")
```

Index