Package 'LSMRealOptions'

Title Value American and Real Options Through LSM Simulation

Version 0.2.1

Description The least-squares Monte Carlo (LSM) simulation method is a popular method for the approximation of the value of early and multiple exercise options. 'LSMRealOptions' provides implementations of the LSM simulation method to value American option products and capital investment projects through real options analysis. 'LSMRealOptions' values capital investment projects with cash flows dependent upon underlying state variables that are stochastically evolving, providing analysis into the timing and critical values at which investment is optimal. 'LSMRealOptions' provides flexibility in the stochastic processes followed by underlying assets, the number of state variables, basis functions and underlying asset characteristics to allow a broad range of assets to be valued through the LSM simulation method. Real options projects are further able to be valued whilst considering construction periods, time-varying initial capital expenditures and path-dependent operational flexibility including the ability to temporarily shutdown or permanently abandon projects after initial investment has occurred. The LSM simulation method was first presented in the prolific work of Longstaff and Schwartz (2001) <doi:10.1093/rfs/14.1.113>.

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1.9001

Suggests knitr, rmarkdown, ggplot2, dplyr, scales, NFCP

Imports stats

VignetteBuilder knitr

NeedsCompilation no

Author Thomas Aspinall [aut, cre] (<https://orcid.org/0000-0002-6968-1989>),
Adrian Gepp [aut] (<https://orcid.org/0000-0003-1666-5501>),
Geoff Harris [aut] (<https://orcid.org/0000-0003-4284-8619>),
Simone Kelly [aut] (<https://orcid.org/0000-0002-6528-8557>),
Colette Southam [aut] (<https://orcid.org/0000-0001-7263-2347>),
Bruce Vanstone [aut] (<https://orcid.org/0000-0002-3977-2468>)

Maintainer Thomas Aspinall <tomaspinall2512@gmail.com>

Repository CRAN

Date/Publication 2021-06-26 10:30:02 UTC

R **topics documented:**

GBM_simulate   *Simulate the geometric Brownian motion (GBM) stochastic process*
                                    through Monte Carlo simulation

---

Description

GBM is a commonly used stochastic process to simulate the price paths of stock prices and other
assets, in which the log of the asset follows a random walk process with drift. The GBM_simulate
function utilizes antithetic variates as a simple variance reduction technique.

Usage

```
GBM_simulate(n, t, mu, sigma, S0, dt)
```

Arguments

n	The total number of price paths to simulate
t	The forecasting period, in years
mu	The drift term of the GBM process
sigma	The volatility term of the GBM process
S0	The initial value of the underlying asset
dt	The discrete time step of observations, in years

Details

A stochastic process S(t) is a geometric brownian motion that follows the following continuous-time
stochastic differential equation:

$$\frac{dS(t)}{S(t)} = \mu dt + \sigma dW(t)$$

Where μ is the drift term, σ the volatility term and W_t is defined as a Weiner process.

The GBM is log-normally distributed.

Value

A matrix of simulated price paths of the GBM process. Each column corresponds to a simulated price path, and each row corresponds to a simulated observed price of the simulated price paths at each discrete time period.

Examples

```
## 100 simulations of 1 year of monthly price paths:
Simulated <- GBM_simulate(n = 100,
                          t = 1,
                          mu = 0.05,
                          sigma = 0.2,
                          S0 = 100,
                          dt = 1/12)
```

GOU_simulate	*Simulate the geometric Ornstein-Uhlenbeck (GOU) stochastic process through Monte Carlo simulation*

Description

The geometric Ornstein-Uhlenbeck process is a member of the general affine class of stochastic process. The Ornstein-Uhlenbeck process is a Gaussian process, a Markov process, is temporally homogeneous and exhibits mean-reverting behaviour. The `IGBM_simulate` function utilizes antithetic variates as a simple variance reduction technique.

Usage

```
GOU_simulate(n, t, reversion_rate, sigma, equilibrium, risk_premium, S0, dt)
```

Arguments

n	The total number of price paths to simulate
t	The forecasting period, in years
reversion_rate	The reversion rate term of the GOU process
sigma	The volatility term of the GOU process
equilibrium	The equilibrium term of the GOU process
risk_premium	The risk premium of the GOU process
S0	The initial value of the underlying asset
dt	The discrete time step of observations, in years
	A stochastic process S(t) is an IGBM that follows the following continuous-time stochastic differential equation:

$$dS(t)/ = reversion_rate(equilibrium - S(t))dt + \sigma dW(t)$$

$$\frac{dS(t)}{S(t)} = equilibrium - (reversion_rate * S(t) + risk_premium)dt + \sigma dW(t)$$

Where 'reversion_rate' is the rate of reversion term, 'equilibrium' is the equilibrium value the process reverts towards, 'risk_premium' is the risk premium of the process, σ the volatility term and W_t is defined as a Weiner process.

Value

A matrix of simulated price paths of the GOU process. Each column corresponds to a simulated price path, and each row corresponds to a simulated observed price of the simulated price paths at each discrete time period.

Examples

```
## 100 simulations of 1 year of monthly price paths:
Simulated <- GOU_simulate(n = 100,
                          t = 1,
                          reversion_rate = 1,
                          sigma = 0.2,
                          equilibrium = 100,
                          risk_premium = 0.05,
                          S0 = 100,
                          dt = 1/12)
```

IGBM_simulate   *Simulate the inhomogeneous geometric Brownian motion (IGBM)*
                             stochastic process through Monte Carlo simulation

---

Description

The inhomogeneous geometric Brownian motion, also known as the integrated GBM process, is a member of the general affine class of stochastic process that has been reported to be well suited for modelling energy prices. The IGBM_simulate function utilizes antithetic variates as a simple variance reduction technique.

Usage

```
IGBM_simulate(n, t, reversion_rate, sigma, equilibrium, S0, dt)
```

Arguments

n	The total number of price paths to simulate
t	The forecasting period, in years
reversion_rate	The reversion rate term of the IGBM process
sigma	The volatility term of the IGBM process

equilibrium	The equilibrium term of the IGBM process
S0	The initial value of the underlying asset
dt	The discrete time step of observations, in years
	A stochastic process S(t) is an IGBM that follows the following continuous-time stochastic differential equation:

$$dS(t) = reversion_rate(equilibrium - S(t))dt + \sigma dW(t)$$

Where 'reversion_rate' is the rate of reversion term, 'equilibrium' is the equilibrium value the process reverts towards, σ the volatility term and W_t is defined as a Weiner process.

Value

A matrix of simulated price paths of the IGBM process. Each column corresponds to a simulated price path, and each row corresponds to a simulated observed price of the simulated price paths at each discrete time period.

Examples

```
## 100 simulations of 1 year of monthly price paths:
Simulated <- IGBM_simulate(n = 100,
                           t = 1,
                           reversion_rate = 1,
                           sigma = 0.2,
                           equilibrium = 100,
                           S0 = 100,
                           dt = 1/12)
```

LSM_american_option	*Value American-style options through least-squares Monte Carlo (LSM) simulation*

Description

Given a set of state variables and associated payoffs simulated through Monte Carlo simulation, solve for the value of an American-style call or put option through the least-squares Monte Carlo simulation method.

Usage

```
LSM_american_option(
  state_variables,
  payoff,
  K,
  dt,
  rf,
  call = FALSE,
```

```
    orthogonal = "Power",
    degree = 2,
    cross_product = TRUE,
    verbose = FALSE
)
```

Arguments

state_variables

> `matrix` or `array`. The simulated states of the underlying stochastic variables. The first dimension corresponds to the simulated values of the state variables at each discrete observation point. The second dimension corresponds to each individual simulation of the state variable. The third dimension corresponds to each state variable considered.

payoff

> `matrix` The payoff at each observation point resulting from exercise into the underlying asset. The first dimension corresponds to the simulated values of the state variables at each discrete observation point. The second dimension corresponds to each individual simulation of the state variable.

K

> the exercise price of the American-style option

dt

> Constant, discrete time step of simulated observations

rf

> The annual risk-free interest rate

call

> `logical` Is the American-style option a call or put option?

orthogonal

> `character`. The orthogonal polynomial used to develop basis functions that estimate the continuation value in the LSM simulation method. `orthogonal` arguments available are: "Power", "Laguerre", "Jacobi", "Legendre", "Chebyshev", "Hermite". See **details**.

degree

> The number of orthogonal polynomials used in the least-squares fit. See **details**.

cross_product

> `logical`. Should a cross product of state variables be considered? Relevant only when the number of state variables is greater than one.

verbose

> `logical`. Should additional information be output? See **values**.

Details

The `LSM_american_option` function provides an implementation of the least-squares Monte Carlo (LSM) simulation approach to numerically approximate the value of American-style options (options with early exercise opportunities). The function provides flexibility in the stochastic process followed by the underlying asset, with simulated values of stochastic processes provided within the `state_variables` argument. It also provides flexibility in the payoffs of the option, allowing for vanilla as well as more exotic options to be considered. `LSM_american_option` also provides analysis into the exercise timing and probability of early exercise of the option.

Least-Squares Monte Carlo Simulation:

The least-squares Monte Carlo (LSM) simulation method is a numeric approach first presented by Longstaff and Schwartz (2001) that approximates the value of options with early exercise opportunities. The LSM simulation method is considered one of the most efficient methods of valuing

American-style options due to its flexibility and computational efficiency. The approach can feature multiple stochastically evolving underlying uncertainties, following both standard and exotic stochastic processes.

The LSM method first approximates stochastic variables through a stochastic process to develop cross-sectional information, then directly estimates the continuation value of in-the-money simulation paths by "(regressing) the ex-post realized payoffs from continuation on functions of the values of the state variables" (Longstaff and Schwartz, 2001).

The 'LSM_american_option' function at each discrete time period, for each simulated price path, compares the payoff that results from immediate exercise of the option with the expected value of continuing to hold the option for subsequent periods. The payoff of immediate exercise is provided in the `payoff` argument and could take several different meanings depending upon the type of American-style option being valued (e.g. the current stock price, the maximum price between multiple assets, etc.).

The immediate profit resulting from exercise of the option is dependent upon the type of option being calculated. The profit of price path i and time t is given by:

When `call = TRUE`:
$$profit_{(t,i)} = max(payoff_{(t,i)} - K, 0)$$

When `call = FALSE`:
$$profit_{(t,i)} = max(K - payoff_{(t,i)}, 0)$$

Orthogonal Polynomials:

To improve the accuracy of estimation of continuation values, the economic values in each period are regressed on a linear combination of a set of basis functions of the stochastic variables. These estimated regression parameters and the simulated stochastic variables are then used to calculate the estimator for the expected economic values.

Longstaff and Schwartz (2001) state that as the conditional expectation of the continuation value belongs to a Hilbert space, it can be represented by a combination of orthogonal basis functions. Increasing the number of stochastic state variables therefore increases the number of required basis functions exponentially. The orthogonal polynomials available in the `LSMRealOptions` package are: Laguerre, Jacobi, Legendre (spherical), Hermite (probabilistic), Chebyshev (of the first kind). The simple powers of state variables is further available. Explicit expressions of each of these orthogonal polynomials are available within the textbook of Abramowitz and Stegun (1965).

Value

The 'LSM_american_option' function by default returns a `numeric` object corresponding to the calculated value of the American-style option.

When `verbose = T`, 6 objects are returned within a `list` class object. The objects returned are:

Value	The calculated option value.
Standard Error	The standard error of the option value.
Expected Timing	The expected time of early exercise.
Expected Timing SE	The standard error of the expected time of early exercise.
Exercise Probability	The probability of early exercise of the option being exercised.
Cumulative Exercise Probability	vector. The cumulative probability of option exercise at each discrete observation po

References

Abramowitz, M., and I. A. Stegun, (1965). Handbook of mathematical functions with formulas, graphs, and mathematical tables. Courier Corporation.

Longstaff, F. A., and E. S. Schwartz, (2001). "Valuing American options by simulation: a simple least-squares approach." The review of financial studies, 14(1), 113-147.

Examples

```
# Price a vanilla American put option on an asset that follows
# Geometric Brownian Motion

## Step 1 - simulate stock prices:
stock_prices <- GBM_simulate(n = 100, t = 1, mu = 0.05,
                             sigma = 0.2, S0 = 100, dt = 1/2)

## Step 2 - Value the American put option:
option_value <- LSM_american_option(state_variables = stock_prices,
                                     payoff = stock_prices,
                                     K = 100,
                                     dt = 1/2,
                                     rf = 0.05)
```

LSM_real_option   *Value capital investment projects through least-squares Monte Carlo (LSM) simulation:*

Description

Given a set of state variables and associated net cash flows for an investment project simulated through Monte Carlo simulation, solve for the real option value of a capital investment project through the least-squares Monte Carlo simulation method.

Usage

```
LSM_real_option(
  state_variables,
  NCF,
  CAPEX,
  dt,
  rf,
  construction = 0,
  orthogonal = "Laguerre",
  degree = 9,
  cross_product = TRUE,
  verbose = FALSE,
  debugging = FALSE
)
```

Arguments

state_variables

> `matrix` or `array`. The simulated states of the underlying stochastic variables. The first dimension corresponds to the simulated values of the state variables at each discrete observation point. The second dimension corresponds to each individual simulation of the state variable. The third dimension corresponds to each state variable considered.

NCF

> The net cash flows resulting from operating the underlying capital investment project for one discrete time step at the current simulated values of the state variables. Each column corresponds to a simulated price path of underlying stochastic variables, and each row the net cash flows at a discrete time point for each simulated path. The dimensions of object 'NCF' must match the first two dimensions of the object passed to the 'state_variables' argument.

CAPEX

> `numeric` or `vector` object. The initial capital investment of the project. This value can be either constant or time dependent. When the 'CAPEX' argument is time dependent, it must be a vector object of length equal to the number of discrete observations of the simulations (i.e. the number of rows of both 'state_variables' and 'NCF')

dt

> Constant, discrete time step of simulated observations

rf

> The annual risk-free interest rate

construction

> An `integer` corresponding to the number of periods of construction of the underlying asset. The construction time represents the time between the initial capital expenditure and when net cash flows are accrued, representing the construction time required of the investment project.

orthogonal

> `character`. The orthogonal polynomial used to develop basis functions that estimate the continuation value in the LSM simulation method. `orthogonal` arguments available are: "Power", "Laguerre", "Jacobi", "Legendre", "Chebyshev", "Hermite". See **details**.

degree

> The number of orthogonal polynomials used in the least-squares fit. See **details**.

cross_product

> `logical`. Should a cross product of state variables be considered? Relevant only when the number of state variables is greater than one

verbose

> `logical`. Should additional information be output? See **values**.

debugging

> `logical` Should additional simulation information be output? See **values**.

Details

The `LSM_real_option` function provides an implementation of the least-squares Monte Carlo (LSM) simulation approach to numerically approximate the value of capital investment projects considering the flexibility of timing of investment under stochastically evolving uncertainty. The function provides flexibility in the stochastic behavior of the underlying uncertainty, with simulated values of state variables provided within the `state_variables` argument. The `LSM_real_option` function also provides analysis into the expected investment timing, probability, and the expected payback period of the project.

Least-Squares Monte Carlo Simulation:

The least-squares Monte Carlo (LSM) simulation method is a numeric approach first presented by Longstaff and Schwartz (2001) that approximates the value of options with early exercise opportunities. The LSM simulation method is considered one of the most efficient methods of valuing American-style options due to its flexibility and computational efficiency. The approach can feature multiple stochastically evolving underlying uncertainties, following both standard and exotic stochastic processes.

The LSM method first approximates stochastic variables through a stochastic process to develop cross-sectional information, then directly estimates the continuation value of in-the-money simulation paths by "(regressing) the ex-post realized payoffs from continuation on functions of the values of the state variables" (Longstaff and Schwartz, 2001).

Real Options Analysis

Real options analysis of investment projects considers the value of the option to delay investment in a project until underlying, stochastically evolving uncertainty has revealed itself. Real options analysis treats investment into capital investment projects as an optimal stopping problem, optimizing the timing of investment that maximizes the payoffs of investment under underlying stochastic uncertainty. Real options analysis is also capable of evaluating the critical value of underlying state variables at which immediate investment into a project is optimal. See Dixit and Pindyck (1994) for more details of real options analysis.

The `LSM_real_option` function considers the option to invest into a capital investment project within a finite forecasting horizon. Investment into the project results in accruing all future net cash flows (NCF) until the end of the forecasting horizon at the cost of the capital expenditure (CAPEX). Immediate investment into a capital investment project is optimal when the waiting option value (WOV) is zero. Critical values of state variables at which immediate investment in optimal can therefore be obtained through finding the root of the WOV.

The primary difference between the `LSM_real_option` and `LSM.AmericanOption` function is the way in which they evaluate the payoff of exercise of the American-style option. The `LSM.AmericanOption` function considers the payoff of exercise to be a one time payoff (i.e. buying or selling the security in a vanilla call or put option) corresponding to the respective payoff argument. The `LSM_real_option` function instead, at each discrete time period, for each simulated price path, compares the sum of all remaining discounted cash flows that are accrued following immediate investment into a project to the end of the forecasting horizon with the expected value of delaying investment. This is is known as the 'running present value' (RPV) of the project, which is the discretised present value of all future cash flows of the project. The RPV of a project increases as the size of the discrete time step decreases, highlighting the need for small discrete time steps to accurately value investment projects. This is due to the discounting effect of discounting larger cash flows over greater time periods compared to smaller cash flows being discounted more frequently.

Orthogonal Polynomials:

To improve the accuracy of estimation of continuation values, the economic values in each period are regressed on a linear combination of a set of basis functions of the stochastic variables. These estimated regression parameters and the simulated stochastic variables are then used to calculate the estimator for the expected economic values.

Longstaff and Schwartz (2001) state that as the conditional expectation of the continuation value belongs to a Hilbert space, it can be represented by a combination of orthogonal basis functions. Increasing the number of stochastic state variables therefore increases the number of required basis functions exponentially. The orthogonal polynomials available in the `LSM_real_options` package are: Laguerre, Jacobi, Legendre (spherical), Hermite (probabilistic), Chebyshev (of the first kind).

The simple powers of state variables is further available. Explicit expressions of each of these orthogonal polynomials are available within the textbook of Abramowitz and Stegun (1965).

Value

The `LSM_real_option` function returns a `list` object. The number of objects returned in the list is dependent upon the `logical` values of arguments verbose and debugging.

`LSM_real_option` by default returns 3 objects within the `list` class object:

ROV   'Real Option value': The value of the capital investment project considering flexibility in the timing of investment.
NPV   'Net Present Value': The value of the capital investment project considering immediate investment.
WOV   'Waiting Option Value': The value of the option to delay initial investment, equal to the difference between the ROV an

When verbose = T, an additional 9 objects are returned within the `list` class object, providing further analysis into the capital investment project:

ROV SE	The standard error of 'ROV'.
NPV SE	The standard error of 'NPV'.
WOV SE	The standard error of 'WOV'.
Expected Timing	The expected timing of investment, in years.
Expected Timing SE	The standard error of the expected timing of investment.
Investment Prob	The probability of investment within the forecasting horizon.
Cumulative Investment Prob	The cumulative probability of investment at each discrete time point over the forecasting ho
PB	The expected payback time of initial capital investment, in years.
PB SE	The standard error of the expected payback time.

When debugging = T, an additional 4 objects are returned within the `list` class object. These objects provide information about the values of individual simulated price paths:

Investment Period	The time of investment of invested price paths. Price paths that did not trigger investment are repre
Project Value	The calculated project value at time zero for each simulated price path. The 'ROV' is equal to the
Immediate Profit	The profit resulting from immediate investment for each discrete time period and for all simulated
Running Present Value	The present value of all future cash flows of an investment project for each discrete time period an

References

Abramowitz, M., and I. A. Stegun, (1965). Handbook of mathematical functions with formulas, graphs, and mathematical tables. Courier Corporation.

Dixit, A. K., and R. S. Pindyck, (1994). Investment under uncertainty. Princeton university press.

Longstaff, F. A., and E. S. Schwartz, (2001). Valuing American options by simulation: a simple least-squares approach. The review of financial studies, 14(1), 113-147.

Examples

```
# Example: Value a capital investment project where the revenues follow a
```

```
# Geometric Brownian Motion stochastic process:

## Step 1 - Simulate asset prices:
asset_prices <- GBM_simulate(n = 100, t = 10, mu = 0.05,
                             sigma = 0.2, S0 = 100, dt = 1/2)

## Step 2 - Perform Real Option Analysis (ROA):
ROA <- LSM_real_option(state_variables = asset_prices,
                       NCF = asset_prices - 100,
                       CAPEX = 1000,
                       dt = 1/2,
                       rf = 0.05)
```

LSM_real_option_OF   *Value operationally flexible capital investment projects through least-squares Monte Carlo (LSM) simulation:*

Description

Given a set of state variables and associated net cash flows for an investment project simulated through Monte Carlo simulation, solve for the real option value of a capital investment project that has the flexibility to temporarily suspend or permanently abandon operations through the least-squares Monte Carlo simulation method.

Usage

```
LSM_real_option_OF(
  state_variables,
  NCF,
  CAPEX,
  dt,
  rf,
  construction = 0,
  orthogonal = "Laguerre",
  degree = 9,
  cross_product = TRUE,
  suspend_CAPEX = NULL,
  suspend_OPEX = NULL,
  resume_CAPEX = NULL,
  abandon_CAPEX = NULL,
  save_states = FALSE,
  verbose = FALSE,
  debugging = FALSE
)
```

Arguments

state_variables

        `matrix` or `array`. The simulated states of the underlying stochastic variables. The first dimension corresponds to the simulated values of the state variables at each discrete observation point. The second dimension corresponds to each individual simulation of the state variable. The third dimension corresponds to each state variable considered.

NCF   The net cash flows resulting from operating the underlying capital investment project for one discrete time step at the current simulated values of the state variables. Each column corresponds to a simulated price path of underlying stochastic variables, and each row the net cash flows at a discrete time point for each simulated path. The dimensions of object 'NCF' must match the first two dimensions of the object passed to the 'state_variables' argument.

CAPEX   `numeric` or `vector` object. The initial capital investment of the project. This value can be either constant or time dependent. When the 'CAPEX' argument is time dependent, it must be a vector object of length equal to the number of discrete observations of the simulations (i.e. the number of rows of both 'state_variables' and 'NCF')

dt   Constant, discrete time step of simulated observations

rf   The annual risk-free interest rate

construction   An integer corresponding to the number of periods of construction of the underlying asset. The construction time represents the time between the initial capital expenditure and when net cash flows are accrued, representing the construction time required of the investment project.

orthogonal   `character`. The orthogonal polynomial used to develop basis functions that estimate the continuation value in the LSM simulation method. `orthogonal` arguments available are: "Power", "Laguerre", "Jacobi", "Legendre", "Chebyshev", "Hermite". See **details**.

degree   The number of orthogonal polynomials used in the least-squares fit. See **details**.

cross_product   `logical`. Should a cross product of state variables be considered? Relevant only when the number of state variables is greater than one

suspend_CAPEX   The cash flow resulting from suspending an operational project

suspend_OPEX   the ongoing cash flow resulting from being suspended for one discrete time period

resume_CAPEX   the cash flow resulting from resuming a suspended project

abandon_CAPEX   the terminal cash flow resulting from permanently abandoning a project

save_states   `logical`. Should the operational state of each simulated price path be returned? See **values**.

verbose   `logical`. Should additional information be output? See **values**.

debugging   `logical` Should additional simulation information be output? See **values**.

Details

The `LSM_real_option_OF` function provides an implementation of the least-squares Monte Carlo (LSM) simulation approach to numerically approximate the value of capital investment projects considering the flexibility of timing of investment and operational states under stochastically evolving uncertainty. The function provides flexibility in the stochastic behavior of the underlying uncertainty, with simulated values of state variables provided within the `state_variables` argument. Multiple stochastically evolving uncertainties can also be considered. The `LSM_real_option_OF` function also provides analysis into the expected investment timing, probability, and operating modes of the project.

Least-Squares Monte Carlo Simulation:

The least-squares Monte Carlo (LSM) simulation method is a numeric approach first presented by Longstaff and Schwartz (2001) that approximates the value of options with early exercise opportunities. The LSM simulation method is considered one of the most efficient methods of valuing American-style options due to its flexibility and computational efficiency. The approach can feature multiple stochastically evolving underlying uncertainties, following both standard and exotic stochastic processes.

The LSM method first approximates stochastic variables through a stochastic process to develop cross-sectional information, then directly estimates the continuation value of in-the-money simulation paths by "(regressing) the ex-post realized payoffs from continuation on functions of the values of the state variables" (Longstaff and Schwartz, 2001).

Real Options Analysis

Real options analysis of investment projects considers the value of the option to delay investment in a project until underlying, stochastically evolving uncertainty has revealed itself. Real options analysis treats investment into capital investment projects as an optimal stopping problem, optimizing the timing of investment that maximizes the payoffs of investment under underlying stochastic uncertainty. Real options analysis is also capable of evaluating the critical value of underlying state variables at which immediate investment into a project is optimal. See Dixit and Pindyck (1994) for more details of real options analysis.

The `LSM_real_option_OF` function considers the option to invest into a capital investment project within a finite forecasting horizon. Investment into the project results in accruing all future net cash flows (NCF) until the end of the forecasting horizon at the cost of the capital expenditure (CAPEX).

The primary difference between the `LSM_real_option` and `LSM_american_option` function is the way in which they evaluate the payoff of exercise of the American-style option. The `LSM_american_option` function considers the payoff of exercise to be a one time payoff (i.e. buying or selling the security in a vanilla call or put option) corresponding to the respective payoff argument. The `LSM_real_option_OF` function instead, at each discrete time period, for each simulated price path, compares the sum of all remaining discounted cash flows that are accrued following immediate investment into a project to the end of the forecasting horizon with the expected value of delaying investment. This is is known as the 'running present value' (RPV) of the project, which is the discretised present value of all future cash flows of the project. The RPV of a project increases as the size of the discrete time step decreases, highlighting the need for small discrete time steps to accurately value investment projects. This is due to the discounting effect of discounting larger cash flows over greater time periods compared to smaller cash flows being discounted more frequently.

Orthogonal Polynomials:

To improve the accuracy of estimation of continuation values, the economic values in each period are regressed on a linear combination of a set of basis functions of the stochastic variables. These

estimated regression parameters and the simulated stochastic variables are then used to calculate the estimator for the expected economic values.

Longstaff and Schwartz (2001) state that as the conditional expectation of the continuation value belongs to a Hilbert space, it can be represented by a combination of orthogonal basis functions. Increasing the number of stochastic state variables therefore increases the number of required basis functions exponentially. The orthogonal polynomials available in the `LSM_real_options` package are: Laguerre, Jacobi, Legendre (spherical), Hermite (probabilistic), Chebyshev (of the first kind). The simple powers of state variables is further available. Explicit expressions of each of these orthogonal polynomials are available within the textbook of Abramowitz and Stegun (1965).

Operational Flexibility:

Operational flexibility is adjustment made to a project in response to, or anticipation of, positive or negative economic outcomes that occur after a project has begun. Operational flexibility has the ability to mitigate the effect of poor timing of initial investment by maximizing gains and minimizing losses. Operational flexibility that reduces losses in response to negative economic conditions (e.g. output prices falling to levels where operations are no longer profitable) reduce the irreversibility of initial investment into a project. This is particularly valuable for investment projects in highly competitive business operating conditions, high initial investment costs, long residual lifetimes of operations, high levels of volatility in underlying assets and net present values near zero.

The operational flexibility supported by the `LSM_real_option_OF` function is the ability to temporarily suspend and permanently abandon an operational investment project. The ability to temporarily suspend and permanently abandon operations was first presented in the prolific work of Brennan and Schwartz (1985), whom solved for the trigger values of suspension, resumption and abandonment of a finite natural resource investment under natural resource price uncertainty. The work of Cortazar, Gravet and Urzua (2008) solved this optimal switching problem using the LSM simulation method.

Operational flexibilities are path dependent, compound and mutually exclusive real options. Under consideration of operational flexibility, there are several different "states" or operating modes available for an investment project. These are:

* Not yet invested * Operational * Suspended * Abandoned

With the "Under construction" state further considered when construction time is also considered.

Generally, there is a cost of switching between operating modes (for example, switching from the "not yet invested" state into the "operational" state incurs the cost of the initial capital investment, 'CAPEX'). The costs incurred in temporarily suspending, resuming, or abandoning a project could be attributed to firing/hiring project employees, loss in perishable products and general cost of moving a project into a 'suspended' state. Operational expenditures of a suspended project could include maintenance to inactive machinery, preventing weathering of project sites, ongoing rent or utility payments, etc. Finally, the abandonment expenditure could be negative, representing a 'scrap' value of the project that is returned as project assets are sold on abandonment of the project.

Solving for the value of an investment project with operational flexibility is classified as an optimal switching or optimal control model. The objective is to optimize the value of the project by taking the optimal set of decisions for the project operating mode at each time point under uncertainty.

The 'LSM_real_option_OF' function solves for the optimal switching problem at each discrete time point throughout the backward induction process. For each operating mode, the immediate payoff of switching to a different operating mode is compared against the discounted 'continuation' value of staying in the current operating mode. These continuation values are directly estimated through

least-squares regression of all simulated price paths (rather than just 'in-the-money' price paths, as is considered in the value of the option to delay the initial investment decision).

The 'ROV' value of the investment project is then calculated by considering the option to invest into the 'operational' state. Investment into the project is optimal when the value of the 'operational' state is greater than the expected continuation value.

The introduction of operational flexibility to the LSM simulation method greatly increases the level of computational complexity and thus the LSM_real_option_OF function is considerably slower than the LSM_real_option function.

Value

The LSM_real_option_OF function by default returns a list object containing 3 objects, with further objects returned within this list when the save_states, verbose and debugging arguments are set to TRUE.

The objects returned are:

ROV	Real option value. The value of an investment project when actively managing the investment and appropriately ex
NPV + OF	Net present Value + operational flexibility. The value immediate investment into an investment project when active
NPV	Net present value. The value immediate investment into an investment project
WOV	Waiting option value. The value of the option to delay initial investment into an investment project

When save_states = TRUE, an additional object is returned:

Project States	A data.frame object detailing the proportion of simulated price paths in each operating state at each discre

When verbose = T, an additional 7 objects are returned within the list class object, providing further analysis into the capital investment project:

ROV SE	The standard error of 'ROV'.
NPV SE	The standard error of 'NPV'.
WOV SE	The standard error of 'WOV'.
Expected Timing	The expected timing of investment, in years.
Expected Timing SE	The standard error of the expected timing of investment.
Investment Prob	The probability of investment within the forecasting horizon.
Cumulative Investment Prob	The cumulative probability of investment at each discrete time point over the forecasting hor

When debugging = T, an additional 5 objects are returned within the list class object. These objects provide information about the values of individual simulated price paths:

Investment Period	The time of investment of invested price paths. Price paths that did not trigger investment are repre
Project Value	The calculated project value at time zero for each simulated price path. The 'ROV' is equal to the
Immediate Profit	The profit resulting from immediate investment for each discrete time period and for all simulated
Running Present Value	The present value of all future cash flows of an investment project for each discrete time period an
Path Project States	The current project state for each discrete time period and for all simulated price paths

References

Abramowitz, M., and I. A. Stegun, (1965). Handbook of mathematical functions with formulas, graphs, and mathematical tables. Courier Corporation.

Brennan, M. J., and E. S. Schwartz, (1985). Evaluating natural resource investments. Journal of business, 135-157.

Dixit, A. K., and R. S. Pindyck, (1994). Investment under uncertainty. Princeton university press.

Longstaff, F. A., and E. S. Schwartz, (2001). Valuing American options by simulation: a simple least-squares approach. The review of financial studies, 14(1), 113-147.

Cortazar, G., M. Gravet, and J. Urzua, (2008). The valuation of multidimensional American real options using the LSM simulation method. Computers & Operations Research, 35(1), 113-129.

Examples

```
# Example: Value a capital investment project where the revenues follow a
# Geometric Brownian Motion stochastic process:

## Step 1 - Simulate asset prices:
asset_prices <- GBM_simulate(n = 100, t = 10, mu = 0.05,
                             sigma = 0.2, S0 = 100, dt = 1/2)

## Step 2 - Perform Real Option Analysis (ROA):
ROA <- LSM_real_option_OF(state_variables = asset_prices,
                      NCF = asset_prices - 100,
                      CAPEX = 1000,
                      dt = 1/2,
                      rf = 0.05,
                      suspend_CAPEX = 100,
                      suspend_OPEX = 10,
                      resume_CAPEX = 100,
                      abandon_CAPEX = 0
                      )
```

Index