# Package 'ruin'

October 14, 2022

**Type** Package

**Title** Simulation of Various Risk Processes

**Version** 0.1.1

**Date** 2018-07-20

**Description** A (not yet exhaustive) collection of common models of risk processes in actuarial science, represented as formal S4 classes. Each class (risk model) has a simulator of its path, and a plotting function. Further, a Monte-Carlo estimator of a ruin probability for a finite time is implemented, using a parallel computation. Currently, the package extends two classical risk models Cramer-Lundberg and Sparre Andersen models by including capital injections, that are positive jumps (see Breuer L. and Badescu A.L. (2014) <doi:10.1080/03461238.2011.636969>). The intent of the package is to provide a user-friendly interface for ruin processes' simulators, as well as a solid and extensible structure for future extensions.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**URL** http://github.com/irudnyts/ruin

**BugReports** http://github.com/irudnyts/ruin/issues

**Depends** R (>= 3.5.0)

**Imports** methods, parallel, ggplot2 (>= 2.2.1)

**Collate** 'AllClass.R' 'AllGeneric.R' 'methods-CramerLundberg.R' 'methods-CramerLundbergCapitalInjections.R' 'methods-SparreAndersen.R' 'methods-SparreAndersenCapitalInjections.R' 'plot_path.R' 'ruin_probability.R' 'zzz.R'

**RoxygenNote** 6.0.1

**Suggests** testthat, actuar (>= 2.3.0), knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Iegor Rudnytskyi [aut, cre]

**Maintainer** Iegor Rudnytskyi <iegor.rudnytskyi@gmail.com>

**Repository** CRAN

**Date/Publication** 2018-07-30 12:56:18 UTC

# R topics documented:



---

`CramerLundberg` *Constructs an object of CramerLundberg S4 class*

---

## Description

`CramerLundberg()` constructs an object of `CramerLundberg` S4 class.

## Usage

```
CramerLundberg(initial_capital = NULL, premium_rate = NULL,
  claim_poisson_arrival_rate = NULL, claim_size_generator = NULL,
  claim_size_parameters = NULL)
```

## Arguments

`initial_capital`
: a length one numeric non-negative vector specifying an initial capital. Default: `0`.

`premium_rate`
: a length one numeric non-negative vector specifying a premium rate. Default: `1`.

`claim_poisson_arrival_rate`
: a length one numeric positive vector specifying the rate of the Poisson process of claims' arrivals. Default: `1`.

`claim_size_generator`
: a function indicating the random generator of claims' sizes. Default: `rexp`.

`claim_size_parameters`
: a named list containing parameters for the random generator of claims' sizes. Default: `list(rate = 1)`.

## Details

The function constructs an object of a formal S4 class `CramerLundberg`, a representation of classical risk process defined as follows:

$$X(t) = u + ct - \sum_{i=1}^{N(t)} Y_i,$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N(t)$ is the Poisson process with intensity $\lambda$ (`claim_poisson_arrival_rate`), $Y_i$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters` ).

## Value

An object of CramerLundberg class.

## References

Albrecher H., Asmussen A. *Ruin Probabilities*. World Scientific, 2010.

## See Also

`CramerLundbergCapitalInjections`, `SparreAndersen`, `link{SparreAndersenCapitalInjections}`.

## Examples

```
model <- CramerLundberg(initial_capital = 10,
                        premium_rate = 1,
                        claim_poisson_arrival_rate = 1,
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))
```

`CramerLundberg-class` *A formal S4 class CramerLundberg*

## Description

A formal S4 class representation of classical Cramer-Lundberg model.

## Details

The model is defined as follows:

$$X(t) = u + ct - \sum_{i=1}^{N(t)} Y_i,$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N(t)$ is the Poisson process with intensity $\lambda$ (`claim_poisson_arrival_rate`), $Y_i$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters` ).

Objects of class can be created only by using the constructor `CramerLundberg`.

## Slots

`initial_capital` a length one numeric non-negative vector specifying an initial capital.

`premium_rate` a length one numeric non-negative vector specifying a premium rate.

`claim_poisson_arrival_rate` a length one numeric positive vector specifying the rate of the Poisson process of claims' arrivals.

`claim_size_generator` a function indicating the random generator of claims' sizes.

`claim_size_parameters` a named list containing parameters for the random generator of claims' sizes.

## References

Albrecher H., Asmussen A. *Ruin Probabilities*. World Scientific, 2010.

## See Also

`CramerLundberg`

`CramerLundbergCapitalInjections` *Constructs an object of CramerLundbergCapitalInjections S4 class*

## Description

`CramerLundbergCapitalInjections()` constructs an object of `CramerLundbergCapitalInjections` S4 class.

## Usage

```
CramerLundbergCapitalInjections(initial_capital = NULL, premium_rate = NULL,
  claim_poisson_arrival_rate = NULL, claim_size_generator = NULL,
  claim_size_parameters = NULL, capital_injection_poisson_rate = NULL,
  capital_injection_size_generator = NULL,
  capital_injection_size_parameters = NULL)
```

## Arguments

`initial_capital`
: a length one numeric non-negative vector specifying an initial capital. Default: `0`.

`premium_rate`
: a length one numeric non-negative vector specifying a premium rate. Default: `1`.

`claim_poisson_arrival_rate`
: a length one numeric positive vector specifying the rate of the Poisson process of claims' arrivals. Default: `1`.

`claim_size_generator`
: a function indicating the random generator of claims' sizes. Default: `rexp`.

`claim_size_parameters`
: a named list containing parameters for the random generator of claims' sizes. Default: `list(rate = 1)`.

`capital_injection_poisson_rate`
: a length one numeric positive vector specifying the rate of the Poisson process of capital injections' arrivals. Default: `1`.

`capital_injection_size_generator`
: a function indicating the random generator of capital injections' sizes. Default: `rexp`.

`capital_injection_size_parameters`
: a named list containing parameters for the random generator of capital injections' sizes. Default: `list(rate = 1)`.

## Details

The function constructs an object of a formal S4 class `CramerLundbergCapitalInjections`, a representation of an extension of Cramer-Lundberg model that allows for positive jumps and defined as follows:

$$X_(t) = u + ct + \sum_{k=1}^{N^{(+)}(t)} Y_k^{(+)} - \sum_{i=1}^{N^{(-)}(t)} Y_i^{(-)}$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N^{(+)}(t)$ is the Poisson process of positive jumps (capital injections) with intensity $\lambda^{(+)}$ (`capital_injection_poisson_rate`), $Y_k^{(+)}$ are iid capital injections' sizes (`capital_injection_size_generator` and `capital_injection_size_parameters`), $N^{(-)}(t)$ is the Poisson process of negative jumps (claims) with intensity $\lambda^{(-)}$ (`claim_poisson_arrival_rate`), $Y_i^{(-)}$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

## Value

An object of CramerLundbergCapitalInjections class.

## References

Breuera L., Badescu A. L. *A generalised Gerber Shiu measure for Markov-additive risk processes with phase-type claims and capital injections*. Scandinavian Actuarial Journal, 2014(2): 93-115, 2014.

## See Also

`CramerLundberg`, `SparreAndersen`, `link{SparreAndersenCapitalInjections}`.

## Examples

```
model <- CramerLundbergCapitalInjections(
  initial_capital = 10,
  premium_rate = 1,
  claim_poisson_arrival_rate = 1,
  claim_size_generator = rexp,
  claim_size_parameters = list(rate = 1),
  capital_injection_poisson_rate = 1,
  capital_injection_size_generator = rexp,
  capital_injection_size_parameters = list(rate = 1)
)
```

---

# CramerLundbergCapitalInjections-class

*A formal S4 class CramerLundbergCapitalInjections*

---

## Description

A formal S4 class representation of Cramer-Lundberg's extension that includes capital injections.

## Details

The model is defined as follows:

$$X_(t) = u + ct + \sum_{k=1}^{N^{(+)}(t)} Y_k^{(+)} - \sum_{i=1}^{N^{(-)}(t)} Y_i^{(-)}$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N^{(+)}(t)$ is the Poisson process of positive jumps (capital injections) with intensity $\lambda^{(+)}$ (`capital_injection_poisson_rate`), $Y_k^{(+)}$ are iid capital injections' sizes (`capital_injection_size_generator` and `capital_injection_size_parameters`), $N^{(-)}(t)$ is the Poisson process of negative jumps (claims) with intensity $\lambda^{(-)}$ (`claim_poisson_arrival_rate`), $Y_i^{(-)}$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

Objects of class can be created only by using the constructor `CramerLundbergCapitalInjections`.

## Slots

`initial_capital` a length one numeric non-negative vector specifying an initial capital.

`premium_rate` a length one numeric non-negative vector specifying a premium rate.

`claim_poisson_arrival_rate` a length one numeric positive vector specifying the rate of the Poisson process of claims' arrivals.

`claim_size_generator` a function indicating the random generator of claims' sizes.

`claim_size_parameters` a named list containing parameters for the random generator of claims' sizes.

`capital_injection_poisson_rate` a length one numeric positive vector specifying the rate of the Poisson process of capital injections' arrivals.

`capital_injection_size_generator` a function indicating the random generator of capital injections' sizes.

`capital_injection_size_parameters` a named list containing parameters for the random generator of capital injections' sizes.

## References

Breuera L., Badescu A. L. *A generalised Gerber Shiu measure for Markov-additive risk processes with phase-type claims and capital injections*. Scandinavian Actuarial Journal, 2014(2): 93-115, 2014.

## See Also

`CramerLundbergCapitalInjections`

## PathCramerLundberg-class

*A formal S4 class PathCramerLundberg*

### Description

A formal S4 class to contain a simulated path of CramerLundberg model.

### Details

Objects of the class must not be created manually. Instead, objects of this class are returned by `simulate_path`.

### Slots

`model` an object of CramerLundberg class.

`path` a numeric matrix of columns `"time"` and `"X"` defining the simulated path represented by pairs time-value.

`claim_sizes` a numeric vector of claims' sizes.

`claim_arrival_times` a numeric vector of claims' interarrival times.

`time_horizon` a numeric vector of the maximum time horizon achieved.

`is_ruined` a logical vector indicating whether the process is ruined.

`elapsed_time` a numeric vector of the elapsed simulation time in seconds.

`max_time_horizon` a numeric vector of the maximum time horizon allowed.

`max_simulation_time` a numeric vector of the maximum simulation time in seconds.

`seed` a numeric vector indicating the seed used for simulation.

### See Also

CramerLundberg (class definition), `CramerLundberg` (constructor).

## PathCramerLundbergCapitalInjections-class

*A formal S4 class PathCramerLundbergCapitalInjections*

### Description

A formal S4 class to contain a simulated path of CramerLundbergCapitalInjections model.

### Details

Objects of the class must not be created manually. Instead, objects of this class are returned by `simulate_path`.

### Slots

`model` an object of CramerLundbergCapitalInjections class.

`path` a numeric matrix of columns `"time"` and `"X"` defining the simulated path represented by pairs time-value.

`claim_sizes` a numeric vector of claims' sizes.

`claim_arrival_times` a numeric vector of claims' interarrival times.

`capital_injection_sizes` a numeric vector of capital injections' sizes.

`capital_injection_arrival_times` a numeric vector of capital injections' interarrival times.

`time_horizon` a numeric vector of the maximum time horizon achieved.

`is_ruined` a logical vector indicating whether the process is ruined.

`elapsed_time` a numeric vector of the elapsed simulation time in seconds.

`max_time_horizon` a numeric vector of the maximum time horizon allowed.

`max_simulation_time` a numeric vector of the maximum simulation time in seconds.

`seed` a numeric vector indicating the seed used for simulation.

### See Also

CramerLundbergCapitalInjections (class definition), `CramerLundbergCapitalInjections` (constructor).

---

## `PathSparreAndersen-class`

*A formal S4 class PathSparreAndersen*

---

### Description

A formal S4 class to contain a simulated path of SparreAndersen model.

### Details

Objects of the class must not be created manually. Instead, objects of this class are returned by `simulate_path`.

### Slots

`model` an object of CramerLundberg class.

`path` a numeric matrix of columns `"time"` and `"X"` defining the simulated path represented by pairs time-value.

`claim_sizes` a numeric vector of claims' sizes.

`claim_arrival_times` a numeric vector of claims' interarrival times.

`time_horizon` a numeric vector of the maximum time horizon achieved.

`is_ruined` a logical vector indicating whether the process is ruined.

`elapsed_time` a numeric vector of the elapsed simulation time in seconds.

`max_time_horizon` a numeric vector of the maximum time horizon allowed.

`max_simulation_time` a numeric vector of the maximum simulation time in seconds.

`seed` a numeric vector indicating the seed used for simulation.

## See Also

SparreAndersen (class definition), `SparreAndersen` (constructor).

---

`PathSparreAndersenCapitalInjections-class`
*A formal S4 class PathSparreAndersenCapitalInjections*

---

## Description

A formal S4 class to contain a simulated path of SparreAndersenCapitalInjections model.

## Details

Objects of the class must not be created manually. Instead, objects of this class are returned by `simulate_path`.

## Slots

`model` an object of SparreAndersenCapitalInjections class.

`path` a numeric matrix of columns `"time"` and `"X"` defining the simulated path represented by pairs time-value.

`claim_sizes` a numeric vector of claims' sizes.

`claim_arrival_times` a numeric vector of claims' interarrival times.

`capital_injection_sizes` a numeric vector of capital injections' sizes.

`capital_injection_arrival_times` a numeric vector of capital injections' interarrival times.

`time_horizon` a numeric vector of the maximum time horizon achieved.

`is_ruined` a logical vector indicating whether the process is ruined.

`elapsed_time` a numeric vector of the elapsed simulation time in seconds.

`max_time_horizon` a numeric vector of the maximum time horizon allowed.

`max_simulation_time` a numeric vector of the maximum simulation time in seconds.

`seed` a numeric vector indicating the seed used for simulation.

## See Also

SparreAndersenCapitalInjections (class definition), `SparreAndersenCapitalInjections` (constructor).

## plot_path *Plot a path of the simulated ruin process*

### Description

`plot_path()` takes a simulated ruin process as the argument and plots its path.

### Usage

```
plot_path(path_object)
```

### Arguments

| | |
|---|---|
| `path_object` | an S4 object of *Path class (e.g., PathCramerLundberg). |

### Details

Under the hood, the function uses `ggplot2` package, therefore, all functionality from `ggplot2` is available.

### Value

A `ggplot2` object.

### Examples

```
model <- CramerLundberg(initial_capital = 10,
                        premium_rate = 1,
                        claim_poisson_arrival_rate = 1,
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))

path <- simulate_path(model = model, max_time_horizon = 10)

plot_path(path)
```

## ruin_probability *Estimate a ruin probability for a finite time horizon*

### Description

`ruin_probability` simulates paths for a given risk `model` and returns a crude Monte-Carlo estimate of the ruin probability for the finite time horizon.

## Usage

```
ruin_probability(model, time_horizon, simulation_number = NULL,
  ci_level = NULL, parallel = NULL, return_paths = NULL)
```

## Arguments

| | |
|---|---|
| model | an S4 object indicating a risk model (e.g., CramerLundberg). |
| time_horizon | a length one numeric finite vector specifying the time at which the ruin probability should be estimated. |
| simulation_number | a length one numeric vector giving the number of simulations that should be performed. Default: 10000. |
| ci_level | a length one numeric vector between 0 and 1 indicating the level of the confidence interval of the ruin probability. Default: 0.95. |
| parallel | a length one logical vector indicating whether the parallel computing should be used. Default: TRUE. |
| return_paths | a length one logical vector indicating whether a list of simulated paths should be returned. Default: FALSE. |

## Details

The function uses a parallel computing from the package parallel (if parallel is TRUE). The package sets up RNGkind to "L'Ecuyer-CMRG" for a safe seeding (see nextRNGStream) when it is loaded, so that user does not have to take care of seeds / RNGs. Further, the function allows computing the confidence interval, assuming the normal distribution of the ruin probability (thanks to CLT).

## Value

A list of two elements: a numeric vector of lower bound of CI, estimate, and upper bound of CI of the ruin probability; and optionally the list of simulated paths.

## Examples

```
model <- CramerLundberg(initial_capital = 0,
                        premium_rate = 1,
                        claim_poisson_arrival_rate = 1,
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))
ruin_probability(model = model,
                 time_horizon = 10,
                 simulation_number = 100,
                 return_paths = TRUE,
                 parallel = FALSE)
```

simulate_path *Generic for simulating a path of a given risk model*

## Description

`simulate_path()` simulates a path of a given risk model until one of the following conditions is met: (1) the process is ruined, (2) `max_time_horizon` is achieved, (3) the elapsed time of the simulation is greater than `max_simulation_time`.

## Usage

```
simulate_path(model, max_time_horizon = NULL, max_simulation_time = NULL,
  seed = NULL)
```

## Arguments

| | |
|---|---|
| `model` | an S4 object indicating a risk model (e.g., CramerLundberg). |
| `max_time_horizon` | a length one numeric vector specifying the maximum time horizon, until with the process will be simulated. Default: `Inf`. |
| `max_simulation_time` | a length one numeric vector indicating the maximum allowed time of simulation. The value should be specified in seconds. Default: `Inf`. |
| `seed` | an optional arbitrary length numeric vector specifying the seed. If provided, the `.Random.seed` in `.GlobalEnv` is set to its value. |

## Value

An S4 corresponding to `model` class object. For instance, for CramerLundberg, the object of class PathCramerLundberg is returned.

## Warning

Setting both `max_time_horizon` and `max_simulation_time` to `Inf` might be dangerous. In this case, the only stopping condition is a ruin of the process, which might not happen.

## Examples

```
model <- CramerLundberg(initial_capital = 10,
                        premium_rate = 1,
                        claim_poisson_arrival_rate = 1,
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))

path <- simulate_path(model = model, max_time_horizon = 10)
```

simulate_path,CramerLundberg-method
*Simulates a path of a Cramer-Lundberg model*

## Description

`simulate_path()` simulates a path of CramerLundberg model until one of the following conditions is met: (1) the process is ruined, (2) `max_time_horizon` is achieved, (3) the elapsed time of the simulation is greater than `max_simulation_time`.

## Usage

```
## S4 method for signature 'CramerLundberg'
simulate_path(model, max_time_horizon = NULL,
  max_simulation_time = NULL, seed = NULL)
```

## Arguments

| | |
|---|---|
| `model` | an S4 object of CramerLundberg class. |
| `max_time_horizon` | a length one numeric vector specifying the maximum time horizon, until with the process will be simulated. Default: `Inf`. |
| `max_simulation_time` | a length one numeric vector indicating the maximum allowed time of simulation. The value should be specified in seconds. Default: `Inf`. |
| `seed` | an optional arbitrary length numeric vector specifying the seed. If provided, the `.Random.seed` in `.GlobalEnv` is set to its value. |

## Value

PathCramerLundberg

## Warning

Setting both `max_time_horizon` and `max_simulation_time` to `Inf` might be dangerous. In this case, the only stopping condition is a ruin of the process, which might not happen.

## Examples

```
model <- CramerLundberg(initial_capital = 10,
                        premium_rate = 1,
                        claim_poisson_arrival_rate = 1,
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))

path <- simulate_path(model = model, max_time_horizon = 10)
```

`simulate_path,CramerLundbergCapitalInjections-method`
*Simulates a path of a Cramer-Lundberg model's extension with capital injections*

## Description

`simulate_path()` simulates a path of CramerLundbergCapitalInjections model until one of the following conditions is met: (1) the process is ruined, (2) `max_time_horizon` is achieved, (3) the elapsed time of the simulation is greater than `max_simulation_time`.

## Usage

```
## S4 method for signature 'CramerLundbergCapitalInjections'
simulate_path(model,
  max_time_horizon = NULL, max_simulation_time = NULL, seed = NULL)
```

## Arguments

| | |
|---|---|
| `model` | an S4 object of CramerLundbergCapitalInjections class. |
| `max_time_horizon` | a length one numeric vector specifying the maximum time horizon, until with the process will be simulated. Default: `Inf`. |
| `max_simulation_time` | a length one numeric vector indicating the maximum allowed time of simulation. The value should be specified in seconds. Default: `Inf`. |
| `seed` | an optional arbitrary length numeric vector specifying the seed. If provided, the `.Random.seed` in `.GlobalEnv` is set to its value. |

## Value

PathCramerLundbergCapitalInjections

## Warning

Setting both `max_time_horizon` and `max_simulation_time` to `Inf` might be dangerous. In this case, the only stopping condition is a ruin of the process, which might not happen.

## Examples

```
model <- CramerLundbergCapitalInjections(
  initial_capital = 10,
  premium_rate = 1,
  claim_poisson_arrival_rate = 1,
  claim_size_generator = rexp,
  claim_size_parameters = list(rate = 1),
  capital_injection_poisson_rate = 1,
```

```
  capital_injection_size_generator = rexp,
  capital_injection_size_parameters = list(rate = 2)
)

path <- simulate_path(model = model, max_time_horizon = 10)
```

---

`simulate_path,SparreAndersen-method`
*Simulates a path of a Sparre Andersen model*

---

## Description

`simulate_path()` simulates a path of SparreAndersen model until one of the following conditions is met: (1) the process is ruined, (2) `max_time_horizon` is achieved, (3) the elapsed time of the simulation is greater than `max_simulation_time`.

## Usage

```
## S4 method for signature 'SparreAndersen'
simulate_path(model, max_time_horizon = NULL,
  max_simulation_time = NULL, seed = NULL)
```

## Arguments

| | |
|---|---|
| `model` | an S4 object of SparreAndersen class. |
| `max_time_horizon` | a length one numeric vector specifying the maximum time horizon, until with the process will be simulated. Default: `Inf`. |
| `max_simulation_time` | a length one numeric vector indicating the maximum allowed time of simulation. The value should be specified in seconds. Default: `Inf`. |
| `seed` | an optional arbitrary length numeric vector specifying the seed. If provided, the `.Random.seed` in `.GlobalEnv` is set to its value. |

## Value

PathSparreAndersen

## Warning

Setting both `max_time_horizon` and `max_simulation_time` to `Inf` might be dangerous. In this case, the only stopping condition is a ruin of the process, which might not happen.

## Examples

```
model <- SparreAndersen(initial_capital = 10,
                        premium_rate = 1,
                        claim_interarrival_generator = rexp,
                        claim_interarrival_parameters = list(rate = 2),
                        claim_size_generator = rexp,
                        claim_size_parameters = list(rate = 1))

path <- simulate_path(model = model, max_time_horizon = 10)
```

---

# simulate_path,SparreAndersenCapitalInjections-method

*Simulates a path of a Sparre Andersen model's extension with capital injections*

---

## Description

`simulate_path()` simulates a path of SparreAndersenCapitalInjections model until one of the following conditions is met: (1) the process is ruined, (2) `max_time_horizon` is achieved, (3) the elapsed time of the simulation is greater than `max_simulation_time`.

## Usage

```
## S4 method for signature 'SparreAndersenCapitalInjections'
simulate_path(model,
  max_time_horizon = NULL, max_simulation_time = NULL, seed = NULL)
```

## Arguments

| | |
|---|---|
| `model` | an S4 object of SparreAndersenCapitalInjections class. |
| `max_time_horizon` | a length one numeric vector specifying the maximum time horizon, until with the process will be simulated. Default: `Inf`. |
| `max_simulation_time` | a length one numeric vector indicating the maximum allowed time of simulation. The value should be specified in seconds. Default: `Inf`. |
| `seed` | an optional arbitrary length numeric vector specifying the seed. If provided, the `.Random.seed` in `.GlobalEnv` is set to its value. |

## Value

PathSparreAndersenCapitalInjections

## Warning

Setting both `max_time_horizon` and `max_simulation_time` to `Inf` might be dangerous. In this case, the only stopping condition is a ruin of the process, which might not happen.

## Examples

```
model <- SparreAndersenCapitalInjections(
  initial_capital = 10,
  premium_rate = 1,
  claim_interarrival_generator = rexp,
  claim_interarrival_parameters = list(rate = 1),
  claim_size_generator = rexp,
  claim_size_parameters = list(rate = 1),
  capital_injection_interarrival_generator = rexp,
  capital_injection_interarrival_parameters = list(rate = 1),
  capital_injection_size_generator = rexp,
  capital_injection_size_parameters = list(rate = 2)
)

path <- simulate_path(model = model, max_time_horizon = 10)
```

---

# SparreAndersen *Constructs an object of SparreAndersen S4 class*

---

## Description

`SparreAndersen()` constructs an object of `SparreAndersen` S4 class.

## Usage

```
SparreAndersen(initial_capital = NULL, premium_rate = NULL,
  claim_interarrival_generator = NULL, claim_interarrival_parameters = NULL,
  claim_size_generator = NULL, claim_size_parameters = NULL)
```

## Arguments

`initial_capital`
: a length one numeric non-negative vector specifying an initial capital. Default: `0`.

`premium_rate`
: a length one numeric non-negative vector specifying a premium rate. Default: `1`.

`claim_interarrival_generator`
: a function indicating the random generator of claims' interarrival times. Default: `rexp`.

`claim_interarrival_parameters`
: a named list containing parameters for the random generator of claims' interarrival times. Default: `list(rate = 1)`.

`claim_size_generator`
: a function indicating the random generator of claims' sizes. Default: `rexp`.

`claim_size_parameters`
: a named list containing parameters for the random generator of claims' sizes. Default: `list(rate = 1)`.

## Details

The function constructs an object of a formal S4 class `SparreAndersen`, a representation of an extension of Cramer-Lundberg model that allows for non-exponential interarrival times and defined as follows:

$$X(t) = u + ct - \sum_{i=1}^{N(t)} Y_i,$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N(t)$ is the renewal process defined by distribution of interarrival times (`claim_interarrival_generator` and `claim_interarrival_parameters`), $Y_i$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

## Value

An object of SparreAndersen class.

## References


- Andersen, E. Sparre. *On the collective theory of risk in case of contagion between claims*. Transactions of the XVth International Congress of Actuaries, 2(6), 1957.
- Thorin O. *Some Comments on the Sparre Andersen Model in the Risk Theory*. ASTIN Bulletin: The Journal of the IAA, 8(1):104-125, 1974.


## See Also

`CramerLundberg`, `CramerLundbergCapitalInjections`, `link{SparreAndersenCapitalInjections}`.

## Examples

```
model <- SparreAndersen(
  initial_capital = 10,
  premium_rate = 1,
  claim_interarrival_generator = rexp,
  claim_interarrival_parameters = list(rate = 1),
  claim_size_generator = rexp,
  claim_size_parameters = list(rate = 1)
)
```

---

# SparreAndersen-class *A formal S4 class SparreAndersen*

---

## Description

A formal S4 class representation of classical Sparre Andersen model.

## Details

The model is defined as follows:

$$X(t) = u + ct - \sum_{i=1}^{N(t)} Y_i,$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N(t)$ is the renewal process defined by distribution of interarrival times (`claim_interarrival_generator` and `claim_interarrival_parameters`), $Y_i$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

Objects of class can be created only by using the constructor `SparreAndersen`.

## Slots

`initial_capital` a length one numeric non-negative vector specifying an initial capital.

`premium_rate` a length one numeric non-negative vector specifying a premium rate.

`claim_interarrival_generator` a function indicating the random generator of claims' interarrival times.

`claim_interarrival_parameters` a named list containing parameters for the random generator of claims' interarrival times.

`claim_size_generator` a function indicating the random generator of claims' sizes.

`claim_size_parameters` a named list containing parameters for the random generator of claims' sizes.

## References


- Andersen, E. Sparre. *On the collective theory of risk in case of contagion between claims*. Transactions of the XVth International Congress of Actuaries, 2(6), 1957.
- Thorin O. *Some Comments on the Sparre Andersen Model in the Risk Theory*. ASTIN Bulletin: The Journal of the IAA, 8(1):104-125, 1974.


## See Also

`SparreAndersen`

---

# SparreAndersenCapitalInjections

*Constructs an object of SparreAndersenCapitalInjections S4 class*

---

## Description

`SparreAndersenCapitalInjections()` constructs an object of `SparreAndersenCapitalInjections` S4 class.

## Usage

```
SparreAndersenCapitalInjections(initial_capital = NULL, premium_rate = NULL,
  claim_interarrival_generator = NULL, claim_interarrival_parameters = NULL,
  claim_size_generator = NULL, claim_size_parameters = NULL,
  capital_injection_interarrival_generator = NULL,
  capital_injection_interarrival_parameters = NULL,
  capital_injection_size_generator = NULL,
  capital_injection_size_parameters = NULL)
```

## Arguments

| | |
|---|---|
| `initial_capital` | a length one numeric non-negative vector specifying an initial capital. Default: `0`. |
| `premium_rate` | a length one numeric non-negative vector specifying a premium rate. Default: `1`. |
| `claim_interarrival_generator` | a function indicating the random generator of claims' interarrival times. Default: `rexp`. |
| `claim_interarrival_parameters` | a named list containing parameters for the random generator of claims' interarrival times. Default: `list(rate = 1)`. |
| `claim_size_generator` | a function indicating the random generator of claims' sizes. Default: `rexp`. |
| `claim_size_parameters` | a named list containing parameters for the random generator of claims' sizes. Default: `list(rate = 1)`. |
| `capital_injection_interarrival_generator` | a function indicating the random generator of capital injections' interarrival times. Default: `rexp`. |
| `capital_injection_interarrival_parameters` | a named list containing parameters for the random generator of capital injections' interarrival times. Default: `list(rate = 1)`. |
| `capital_injection_size_generator` | a function indicating the random generator of capital injections' sizes. Default: `rexp`. |
| `capital_injection_size_parameters` | a named list containing parameters for the random generator of capital injections' sizes. Default: `list(rate = 1)`. |

## Details

The function constructs an object of a formal S4 class `SparreAndersenCapitalInjections`, a representation of an extension of Sparre Andersen model that allows for positive jumps and defined as follows:

$$X_(t) = u + ct + \sum_{k=1}^{N^{(+)}(t)} Y_k^{(+)} - \sum_{i=1}^{N^{(-)}(t)} Y_i^{(-)}$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N^{(+)}(t)$ is the renewal process of positive jumps (capital injections) defined by distribution of interarrival times (`capital_injection_interarrival_generator` and `capital_injection_interarrival_parameters`), $Y_k^{(+)}$ are iid capital injections' sizes (`capital_injection_size_generator` and `capital_injection_size_parameters`), $N^{(-)}(t)$ is the renewal process of claims defined by distribution of interarrival times (`claim_interarrival_generator` and `claim_interarrival_parameters`), $Y_i^{(-)}$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

## Value

An object of SparreAndersenCapitalInjections class.

## References

Breuera L., Badescu A. L. *A generalised Gerber Shiu measure for Markov-additive risk processes with phase-type claims and capital injections*. Scandinavian Actuarial Journal, 2014(2): 93-115, 2014.

## See Also

`CramerLundberg`, `CramerLundbergCapitalInjections`, `link{SparreAndersen}`.

## Examples

```
model <- SparreAndersenCapitalInjections(
  initial_capital = 10,
  premium_rate = 1,
  claim_interarrival_generator = rexp,
  claim_interarrival_parameters = list(rate = 1),
  claim_size_generator = rexp,
  claim_size_parameters = list(rate = 1),
  capital_injection_interarrival_generator = rexp,
  capital_injection_interarrival_parameters = list(rate = 1),
  capital_injection_size_generator = rexp,
  capital_injection_size_parameters = list(rate = 1)
)
```

---

# SparreAndersenCapitalInjections-class

*A formal S4 class SparreAndersenCapitalInjections*

---

## Description

A formal S4 class representation of Sparre Andersen's extension that includes capital injections.

## Details

The model is defined as follows:

$$X_(t) = u + ct + \sum_{k=1}^{N^{(+)}(t)} Y_k^{(+)} - \sum_{i=1}^{N^{(-)}(t)} Y_i^{(-)}$$

where $u$ is the initial capital (`initial_capital`), $c$ is the premium rate (`premium_rate`), $N^{(+)}(t)$ is the renewal process of positive jumps (capital injections) defined by distribution of interarrival times (`capital_injection_interarrival_generator` and `capital_injection_interarrival_parameters`), $Y_k^{(+)}$ are iid capital injections' sizes (`capital_injection_size_generator` and `capital_injection_size_parameters`), $N^{(-)}(t)$ is the renewal process of claims defined by distribution of interarrival times (`claim_interarrival_generator` and `claim_interarrival_parameters`), $Y_i^{(-)}$ are iid claim sizes (`claim_size_generator` and `claim_size_parameters`).

Objects of class can be created only by using the constructor `SparreAndersenCapitalInjections`.

## Slots

`initial_capital` a length one numeric non-negative vector specifying an initial capital.

`premium_rate` a length one numeric non-negative vector specifying a premium rate.

`claim_interarrival_generator` a function indicating the random generator of claims' interarrival times.

`claim_interarrival_parameters` a named list containing parameters for the random generator of claims' interarrival times.

`claim_size_generator` a function indicating the random generator of claims' sizes.

`claim_size_parameters` a named list containing parameters for the random generator of claims' sizes.

`capital_injection_interarrival_generator` a function indicating the random generator of capital injections' interarrival times.

`capital_injection_interarrival_parameters` a named list containing parameters for the random generator of capital injections' interarrival times.

`capital_injection_size_generator` a function indicating the random generator of capital injections' sizes.

`capital_injection_size_parameters` a named list containing parameters for the random generator of capital injections' sizes.

## References

Breuera L., Badescu A. L. *A generalised Gerber Shiu measure for Markov-additive risk processes with phase-type claims and capital injections*. Scandinavian Actuarial Journal, 2014(2): 93-115, 2014.

## See Also

`SparreAndersenCapitalInjections`

# Index