

Package: EconEvalR (via r-universe)

September 22, 2026

Type Package

Title Economic Evaluation Methods for Cost-Benefit, Partial Budgeting,
and Cost-Effectiveness Analyses

Version 0.1.0

Description Provides functions for economic evaluation, including Cost-Benefit Analysis (CBA), Benefit-Cost Ratio (BCR), Net Present Value (NPV), Internal Rate of Return (IRR), Partial Budgeting, Budget Impact Analysis (BIA), Cost-Effectiveness Analysis (CEA), Decision Tree Analysis, Sensitivity Analysis, and Probabilistic Sensitivity Analysis. The package produces summaries, graphical displays, and reproducible workflows for applications in veterinary science, agriculture, public health, epidemiology, health economics, and related fields.

License MIT + file LICENSE

URL <https://github.com/vinodhpm/EconEvalR>

BugReports <https://github.com/vinodhpm/EconEvalR/issues>

Depends R ($\geq 4.3.0$)

Imports cli ($\geq 3.6.2$), mgcv

Suggests covr, testthat ($\geq 3.0.0$)

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 8.0.0

Repository <https://vinodhpm.r-universe.dev>

Date/Publication 2026-08-02 17:49:09 UTC

RemoteUrl <https://github.com/vinodhpm/econevalr>

RemoteRef HEAD

RemoteSha e45e295cf1651c69f16dc1cd3ebf781d66646897

Contents

as.data.frame.EconBCR	3
as.data.frame.EconBIA	4
as.data.frame.EconCEAC	4
as.data.frame.EconCEAF	5
as.data.frame.EconEVPI	6
as.data.frame.EconEVPPI	7
as.data.frame.EconPSA	7
as.data.frame.EconTornado	8
as.data.frame.EconTree	8
benefit_cost_ratio	9
budget_impact_analysis	10
cost_effectiveness	11
cost_effectiveness_acceptability_curve	12
cost_effectiveness_acceptability_frontier	13
decision_tree_analysis	14
discount_cashflow	16
discounted_payback_period	17
dominance_analysis	17
expected_value_partial_perfect_information	18
expected_value_perfect_information	19
icer	21
internal_rate_return	22
modified_internal_rate_return	22
multi_way_sensitivity_analysis	23
net_health_benefit	24
net_monetary_benefit	25
net_present_value	26
one_way_sensitivity_analysis	26
partial_budget	27
payback_period	27
plot.EconBCR	28
plot.EconBIA	29
plot.EconCEA	30
plot.EconCEAC	30
plot.EconCEAF	31
plot.EconEVPI	31
plot.EconEVPPI	32
plot.EconMWSA	33
plot.EconPSA	33
plot.EconThreshold	34
plot.EconTornado	34
plot.EconTree	35
present_value	35
print.EconBCR	36
print.EconBIA	37
print.EconCEAC	37
print.EconCEAF	38

print.EconEVPI	38
print.EconEVPPI	39
print.EconPSA	39
print.EconTornado	40
print.EconTree	40
probabilistic_sensitivity_analysis	41
profitability_index	42
summary.EconBCR	42
summary.EconBIA	43
summary.EconCEA	43
summary.EconCEAC	44
summary.EconCEAF	44
summary.EconEVPI	45
summary.EconEVPPI	46
summary.EconPSA	47
summary.EconTornado	47
summary.EconTree	48
threshold_analysis	49
tornado_diagram	49
two_way_sensitivity_analysis	50
Index	51

as.data.frame.EconBCR

Convert an EconBCR object to a data frame

Description

Convert an EconBCR object to a data frame

Usage

```
## S3 method for class 'EconBCR'
as.data.frame(x, ...)
```

Arguments

x An object of class "EconBCR".

... Additional arguments (unused).

Value

A data frame with one row containing the benefit, cost, benefit-cost ratio, and decision.

```
as.data.frame.EconBIA
```

Convert a Budget Impact Analysis Object to a Data Frame

Description

Converts an object of class "EconBIA" into a data frame suitable for exporting or further analysis.

Usage

```
## S3 method for class 'EconBIA'
as.data.frame(x, ...)
```

Arguments

`x` An object of class "EconBIA".
`...` Additional arguments (unused).

Value

A data frame containing the annual budget impact results.

Examples

```
## Not run:
df <- as.data.frame(bia)
head(df)

## End(Not run)
```

```
as.data.frame.EconCEAC
```

Convert a Cost-Effectiveness Acceptability Curve Object to a Data Frame

Description

Convert a Cost-Effectiveness Acceptability Curve Object to a Data Frame

Usage

```
## S3 method for class 'EconCEAC'
as.data.frame(x, ...)
```

Arguments

x An object of class "EconCEAF".
... Additional arguments (unused).

Value

A data frame containing the willingness-to-pay thresholds, probabilities of cost-effectiveness, and mean incremental net monetary benefit.

```
as.data.frame.EconCEAF
```

Convert a Cost-Effectiveness Acceptability Frontier Object to a Data Frame

Description

Converts an object of class "EconCEAF" into a data frame suitable for exporting or further analysis.

Usage

```
## S3 method for class 'EconCEAF'
as.data.frame(x, ...)
```

Arguments

x An object of class "EconCEAF".
... Additional arguments (unused).

Value

A data frame containing:

WTP Willingness-to-pay threshold.

OptimalStrategy Optimal strategy at each WTP.

Probability Probability that the optimal strategy is cost-effective.

ExpectedNMB Expected Net Monetary Benefit of the optimal strategy.

Examples

```
## Not run:
df <- as.data.frame(ceaf)
head(df)

## End(Not run)
```

```
as.data.frame.EconEVPI
```

Convert an Expected Value of Perfect Information Object to a Data Frame

Description

Converts an object of class "EconEVPI" into a data frame suitable for exporting or further analysis.

Usage

```
## S3 method for class 'EconEVPI'
as.data.frame(x, ...)
```

Arguments

x An object of class "EconEVPI".

... Additional arguments (unused).

Value

A data frame containing:

WTP Willingness-to-pay threshold.

EVPI Expected Value of Perfect Information.

ExpectedPerfectInformation Expected value under perfect information, $E[\max(NMB)]$.

ExpectedCurrentInformation Expected value under current information, $\max(E[NMB])$.

Examples

```
## Not run:
df <- as.data.frame(evpi)
head(df)

## End(Not run)
```

```
as.data.frame.EconEVPPI
```

Convert an Expected Value of Partial Perfect Information Object to a Data Frame

Description

Convert an Expected Value of Partial Perfect Information Object to a Data Frame

Usage

```
## S3 method for class 'EconEVPPI'  
as.data.frame(x, ...)
```

Arguments

x An object of class "EconEVPPI".
... Additional arguments (unused).

Value

A data frame.

```
as.data.frame.EconPSA
```

Convert a Probabilistic Sensitivity Analysis Object to a Data Frame

Description

Convert a Probabilistic Sensitivity Analysis Object to a Data Frame

Usage

```
## S3 method for class 'EconPSA'  
as.data.frame(x, ...)
```

Arguments

x An object of class "EconPSA".
... Additional arguments (unused).

Value

A data frame containing every Monte Carlo simulation.

`as.data.frame.EconTornado`*Convert a Tornado Diagram Object to a Data Frame*

Description

Convert a Tornado Diagram Object to a Data Frame

Usage

```
## S3 method for class 'EconTornado'  
as.data.frame(x, ...)
```

Arguments

<code>x</code>	An object of class "EconTornado".
<code>...</code>	Additional arguments (unused).

Value

A data frame.

`as.data.frame.EconTree`*Convert a Decision Tree Analysis Object to a Data Frame*

Description

Converts an object of class "EconTree" into a data frame suitable for exporting or further analysis.

Usage

```
## S3 method for class 'EconTree'  
as.data.frame(x, ...)
```

Arguments

<code>x</code>	An object of class "EconTree".
<code>...</code>	Additional arguments (unused).

Value

A data frame containing the terminal node results.

Examples

```
## Not run:  
df <- as.data.frame(tree)  
head(df)  
  
## End(Not run)
```

benefit_cost_ratio	<i>Benefit-Cost Ratio</i>
--------------------	---------------------------

Description

Computes the Benefit-Cost Ratio (BCR) for an investment, intervention, or project.

Usage

```
benefit_cost_ratio(benefit, cost)
```

Arguments

benefit	Numeric. Total discounted benefits.
cost	Numeric. Total discounted costs.

Details

The Benefit-Cost Ratio is calculated as

$$BCR = \frac{Benefit}{Cost}$$

Value

An object of class "EconBCR".

Examples

```
benefit_cost_ratio(  
  benefit = 150000,  
  cost = 100000  
)
```

`budget_impact_analysis`*Budget Impact Analysis*

Description

Performs a Budget Impact Analysis (BIA) by comparing the projected healthcare expenditure under current practice with a new intervention over multiple years.

Usage

```
budget_impact_analysis(  
  population,  
  uptake,  
  current_cost,  
  new_cost,  
  years = length(uptake),  
  discount = 0  
)
```

Arguments

<code>population</code>	Numeric scalar representing the eligible population.
<code>uptake</code>	Numeric vector representing the annual uptake proportions (0-1) of the new intervention.
<code>current_cost</code>	Numeric scalar giving the annual cost per patient under current practice.
<code>new_cost</code>	Numeric scalar giving the annual cost per patient under the new intervention.
<code>years</code>	Integer giving the number of years.
<code>discount</code>	Annual discount rate (default = 0).

Details

The function calculates, for each year:

- Eligible population
- Number treated
- Current budget
- New budget
- Incremental budget impact
- Discounted budget impact
- Cumulative discounted impact

Value

An object of class "EconBIA".

Examples

```
bia <- budget_impact_analysis(  
  population = 10000,  
  uptake = c(0.10,0.20,0.35,0.45,0.60),  
  current_cost = 1200,  
  new_cost = 1500,  
  years = 5,  
  discount = 0.03  
)  
  
print(bia)  
summary(bia)  
plot(bia)  
as.data.frame(bia)
```

cost_effectiveness	<i>Cost Effectiveness Analysis</i>
--------------------	------------------------------------

Description

Performs a cost-effectiveness analysis.

Usage

```
cost_effectiveness(strategy, cost, effectiveness)
```

Arguments

strategy	Character vector of strategy names.
cost	Numeric vector of costs.
effectiveness	Numeric vector of effectiveness.

Value

Object of class "EconCEA".

```
cost_effectiveness_acceptability_curve
```

Cost-Effectiveness Acceptability Curve

Description

Computes the probability that an intervention is cost-effective across a range of willingness-to-pay (WTP) thresholds.

Usage

```
cost_effectiveness_acceptability_curve(
  incremental_cost,
  incremental_effect,
  wtp = seq(0, 1e+05, by = 1000)
)
```

Arguments

<code>incremental_cost</code>	Numeric vector of incremental costs.
<code>incremental_effect</code>	Numeric vector of incremental effects.
<code>wtp</code>	Numeric vector of willingness-to-pay thresholds.

Details

The Incremental Net Monetary Benefit (INMB) is computed as

$$INMB = \lambda \times \Delta E - \Delta C$$

where

- λ = willingness-to-pay threshold
- ΔE = incremental effect
- ΔC = incremental cost

The CEAC is the proportion of simulations with $INMB > 0$.

Value

An object of class "EconCEAC".

Examples

```
set.seed(1)

cost <- rnorm(1000,500,100)

effect <- rnorm(1000,0.35,0.08)

ceac <- cost_effectiveness_acceptability_curve(
  incremental_cost=cost,
  incremental_effect=effect,
  wtp=seq(0,5000,100)
)
```

```
cost_effectiveness_acceptability_frontier
```

Cost-Effectiveness Acceptability Frontier

Description

Computes the Cost-Effectiveness Acceptability Frontier (CEAF) for multiple competing strategies.

Usage

```
cost_effectiveness_acceptability_frontier(
  cost,
  effect,
  wtp = seq(0, 1e+05, by = 1000)
)
```

Arguments

cost	A numeric matrix of costs. Rows represent Monte Carlo simulations and columns represent competing strategies.
effect	A numeric matrix of effects with the same dimensions as cost .
wtp	A numeric vector of willingness-to-pay (WTP) thresholds.

Details

Net Monetary Benefit (NMB) is calculated as

$$NMB = \lambda \times Effect - Cost$$

where

- λ is the willingness-to-pay threshold.

For each WTP threshold:

- Expected NMB is calculated for every strategy.
- The optimal strategy is identified.
- The probability that this strategy is optimal is estimated.

Value

An object of class "EconCEAF".

Examples

```
set.seed(123)

cost <- cbind(
  rnorm(1000,1000,100),
  rnorm(1000,1200,120),
  rnorm(1000,1400,150)
)

effect <- cbind(
  rnorm(1000,0.70,0.05),
  rnorm(1000,0.80,0.05),
  rnorm(1000,0.90,0.05)
)

ceaf <- cost_effectiveness_acceptability_frontier(
  cost = cost,
  effect = effect
)
```

decision_tree_analysis

Decision Tree Analysis

Description

Performs a simple decision tree analysis by calculating expected costs and expected effectiveness for terminal nodes.

Usage

```
decision_tree_analysis(strategy, probability, cost, effectiveness)
```

Arguments

strategy	Character vector of terminal strategy names.
probability	Numeric vector of probabilities.
cost	Numeric vector of costs.
effectiveness	Numeric vector of effectiveness.

Details

Expected values are calculated as:

$$Expected\ Cost = Probability \times Cost$$

$$Expected\ Effectiveness = Probability \times Effectiveness$$

Value

An object of class "EconTree".

Examples

```
strategy <- c(
  "Treatment A",
  "Treatment B",
  "Treatment C"
)

probability <- c(
  0.30,
  0.45,
  0.25
)

cost <- c(
  1200,
  1800,
  2400
)

effectiveness <- c(
  0.70,
  0.82,
  0.91
)

tree <- decision_tree_analysis(
  strategy,
  probability,
  cost,
  effectiveness
)

print(tree)
summary(tree)
plot(tree)
as.data.frame(tree)
```

discount_cashflow	<i>Discount Cash Flows</i>
-------------------	----------------------------

Description

Discounts one or more future cash flows to their present values.

Usage

```
discount_cashflow(cashflow, rate, period)
```

Arguments

cashflow	Numeric vector of cash flows.
rate	Discount rate expressed as a decimal (e.g., 0.10 for 10%).
period	Integer vector of time periods corresponding to each cash flow.

Details

The discounted cash flow is calculated as

$$PV = \frac{CF_t}{(1+r)^t}$$

where:

- CF = Cash flow
- r = Discount rate
- t = Time period

Value

An object of class "EconDCF".

Examples

```
discount_cashflow(  
  cashflow = c(1000, 1200, 1500),  
  rate = 0.10,  
  period = 1:3  
)
```

discounted_payback_period
Discounted Payback Period

Description

Calculates the discounted payback period of an investment.

Usage

```
discounted_payback_period(cashflow, rate)
```

Arguments

cashflow	Numeric vector of cash flows. The first value should be the initial investment (negative).
rate	Discount rate expressed as a decimal.

Value

An object of class "EconDiscountedPayback".

Examples

```
dpb <- discounted_payback_period(
  cashflow = c(-50000,12000,15000,18000,20000),
  rate = 0.10
)

print(dpb)
```

dominance_analysis *Dominance Analysis*

Description

Identifies strongly dominated strategies.

Usage

```
dominance_analysis(strategy, cost, effectiveness)
```

Arguments

strategy	Character vector of strategy names.
cost	Numeric vector of costs.
effectiveness	Numeric vector of effectiveness.

Details

A strategy is strongly dominated if another strategy has lower cost and greater or equal effectiveness.

Value

An object of class "EconDominance".

Examples

```
dominance_analysis(
  strategy = c("A","B","C"),
  cost = c(1000,1200,900),
  effectiveness = c(0.70,0.75,0.80)
)
```

expected_value_partial_perfect_information

Expected Value of Partial Perfect Information

Description

Computes the Expected Value of Partial Perfect Information (EVPPI) using regression-based approximation.

Usage

```
expected_value_partial_perfect_information(
  cost,
  effect,
  parameter,
  wtp = seq(0, 1e+05, by = 1000),
  method = "gam"
)
```

Arguments

cost	Numeric matrix of costs. Rows are PSA simulations and columns are competing strategies.
effect	Numeric matrix of effects having the same dimensions as cost .
parameter	Numeric vector representing the uncertain parameter of interest.
wtp	Numeric vector of willingness-to-pay thresholds.
method	Regression method. Currently only "gam" is supported.

Details

The EVPPI is estimated by regression of the Net Monetary Benefit (NMB) on the parameter of interest.

This implementation uses Generalized Additive Models (`mgcv::gam`) to approximate the conditional expectation of NMB.

Value

An object of class "EconEVPPI".

Examples

```
## Not run:

library(mgcv)

set.seed(123)

cost <- cbind(
  rnorm(1000,1000,100),
  rnorm(1000,1200,100)
)

effect <- cbind(
  rnorm(1000,0.70,0.05),
  rnorm(1000,0.82,0.05)
)

theta <- runif(1000)

evppi <- expected_value_partial_perfect_information(
  cost = cost,
  effect = effect,
  parameter = theta
)

## End(Not run)
```

`expected_value_perfect_information`

Expected Value of Perfect Information

Description

Computes the Expected Value of Perfect Information (EVPI) across a range of willingness-to-pay (WTP) thresholds using probabilistic sensitivity analysis results.

Usage

```
expected_value_perfect_information(
  cost,
  effect,
  wtp = seq(0, 1e+05, by = 1000)
)
```

Arguments

cost	A numeric matrix of costs. Rows represent Monte Carlo simulations and columns represent competing strategies.
effect	A numeric matrix of effectiveness values with identical dimensions to cost .
wtp	Numeric vector of willingness-to-pay thresholds.

Details

Net Monetary Benefit (NMB) is computed as

$$NMB = \lambda \times Effect - Cost$$

where

- λ is the willingness-to-pay threshold.

EVPI is calculated as

$$EVPI = E[\max(NMB)] - \max(E[NMB])$$

where

- $E[\max(NMB)]$ is the expected value under perfect information.
- $\max(E[NMB])$ is the best expected value under current information.

Value

An object of class "EconEVPI".

Examples

```
set.seed(123)

cost <- cbind(
  rnorm(1000,1000,100),
  rnorm(1000,1200,120),
  rnorm(1000,1500,140)
)

effect <- cbind(
  rnorm(1000,0.70,0.05),
```

```

      rnorm(1000,0.82,0.05),
      rnorm(1000,0.92,0.05)
    )

    evpi <- expected_value_perfect_information(
      cost = cost,
      effect = effect
    )

```

icer

Incremental Cost-Effectiveness Ratio

Description

Calculates the Incremental Cost-Effectiveness Ratio (ICER) between two competing interventions.

Usage

```
icer(cost, effectiveness)
```

Arguments

cost Numeric vector of costs.
effectiveness Numeric vector of effectiveness.

Details

The ICER is calculated as

$$ICER = \frac{\Delta Cost}{\Delta Effectiveness}$$

where

- $\Delta Cost$ = Difference in cost
- $\Delta Effectiveness$ = Difference in effectiveness

Value

An object of class "EconICER".

Examples

```

icer(
  cost = c(1000,1300,1700),
  effectiveness = c(0.60,0.75,0.90)
)

```

`internal_rate_return` *Internal Rate of Return*

Description

Calculates the Internal Rate of Return (IRR).

Usage

```
internal_rate_return(cashflow, interval = c(-0.99, 10), tol = 1e-08)
```

Arguments

<code>cashflow</code>	Numeric vector of cash flows.
<code>interval</code>	Search interval.
<code>tol</code>	Numerical tolerance.

Value

Object of class "EconIRR".

`modified_internal_rate_return`
Modified Internal Rate of Return

Description

Calculates the Modified Internal Rate of Return (MIRR).

Usage

```
modified_internal_rate_return(cashflow, finance_rate, reinvest_rate)
```

Arguments

<code>cashflow</code>	Numeric vector of cash flows.
<code>finance_rate</code>	Finance rate (decimal).
<code>reinvest_rate</code>	Reinvestment rate (decimal).

Value

Object of class "EconMIRR".

Examples

```
mirr <- modified_internal_rate_return(  
  cashflow = c(-50000,12000,15000,18000,20000),  
  finance_rate = 0.08,  
  reinvest_rate = 0.10  
)  
  
print(mirr)
```

```
multi_way_sensitivity_analysis  
      Multi-Way Sensitivity Analysis
```

Description

Performs deterministic multi-way sensitivity analysis.

Usage

```
multi_way_sensitivity_analysis(parameters, model)
```

Arguments

<code>parameters</code>	A data.frame containing parameter values.
<code>model</code>	A function accepting one row (as a named list) and returning a single numeric outcome.

Value

An object of class "EconMWSA".

Examples

```
pars <- data.frame(  
  Cost = c(1000,1200,1400),  
  Effectiveness = c(0.70,0.75,0.82),  
  Probability = c(0.90,0.85,0.80)  
)  
  
model <- function(x){  
  x$Effectiveness * 5000 -  
  x$Cost * x$Probability  
}  
  
mw <- multi_way_sensitivity_analysis(  
  parameters = pars,  
  model = model  
)
```

net_health_benefit	<i>Net Health Benefit</i>
--------------------	---------------------------

Description

Calculates the Net Health Benefit (NHB).

Usage

```
net_health_benefit(strategy, cost, effectiveness, willingness_to_pay)
```

Arguments

strategy	Character vector of strategy names.
cost	Numeric vector of costs.
effectiveness	Numeric vector of effectiveness.
willingness_to_pay	Numeric scalar representing the willingness-to-pay threshold.

Details

Net Health Benefit is calculated as

$$NHB = E - \frac{C}{\lambda}$$

where

- E = Effectiveness
- C = Cost
- λ = Willingness-to-pay threshold

Value

An object of class "EconNHB".

Examples

```
nhb <- net_health_benefit(
  strategy = c("Current", "Treatment A", "Treatment B"),
  cost = c(1000, 1300, 1700),
  effectiveness = c(0.60, 0.75, 0.90),
  willingness_to_pay = 5000
)

print(nhb)
```

`net_monetary_benefit` *Net Monetary Benefit*

Description

Calculates the Net Monetary Benefit (NMB).

Usage

```
net_monetary_benefit(strategy, cost, effectiveness, willingness_to_pay)
```

Arguments

`strategy` Character vector of strategy names.

`cost` Numeric vector of costs.

`effectiveness` Numeric vector of effectiveness.

`willingness_to_pay` Numeric scalar specifying the willingness-to-pay threshold per unit of effectiveness.

Details

Net Monetary Benefit is calculated as

$$NMB = \lambda \times E - C$$

where

- λ = willingness-to-pay threshold
- E = effectiveness
- C = cost

Value

An object of class "EconNMB".

Examples

```
nmb <- net_monetary_benefit(
  strategy = c("Current", "Treatment A", "Treatment B"),
  cost = c(1000, 1300, 1700),
  effectiveness = c(0.60, 0.75, 0.90),
  willingness_to_pay = 5000
)

print(nmb)
```

<code>net_present_value</code>	<i>Net Present Value</i>
--------------------------------	--------------------------

Description

Calculates the Net Present Value (NPV) of a project.

Usage

```
net_present_value(cashflow, rate)
```

Arguments

<code>cashflow</code>	Numeric vector of cash flows.
<code>rate</code>	Discount rate expressed as a decimal.

Value

An object of class "EconNPV".

<code>one_way_sensitivity_analysis</code>	<i>One-Way Sensitivity Analysis</i>
---	-------------------------------------

Description

Performs deterministic one-way sensitivity analysis by varying a single parameter.

Usage

```
one_way_sensitivity_analysis(base_case, low, high, model)
```

Arguments

<code>base_case</code>	Numeric scalar.
<code>low</code>	Lower value.
<code>high</code>	Upper value.
<code>model</code>	Function returning the outcome.

Value

Object of class "EconOWSA".

partial_budget	<i>Partial Budget Analysis</i>
----------------	--------------------------------

Description

Performs a complete partial budget analysis.

Usage

```
partial_budget(added_returns, reduced_costs, added_costs, reduced_returns)
```

Arguments

`added_returns` Numeric vector.
`reduced_costs` Numeric vector.
`added_costs` Numeric vector.
`reduced_returns` Numeric vector.

Value

Object of class "EconPB".

Examples

```
partial_budget(  
  added_returns = c(5000,2500),  
  reduced_costs = c(1500,500),  
  added_costs = c(2000,1000),  
  reduced_returns = c(500)  
)
```

payback_period	<i>Payback Period</i>
----------------	-----------------------

Description

Calculates the payback period of an investment.

Usage

```
payback_period(cashflow)
```

Arguments

cashflow Numeric vector of cash flows. The first value should be the initial investment (negative).

Details

The payback period is the time required for cumulative cash inflows to recover the initial investment. Linear interpolation is used when recovery occurs during a period.

Value

An object of class "EconPayback".

Examples

```
pb <- payback_period(
  cashflow = c(-50000,12000,15000,18000,20000)
)

print(pb)
```

plot.EconBCR

Plot Benefit-Cost Ratio Results

Description

Produces a bar chart comparing total benefits and total costs.

Usage

```
## S3 method for class 'EconBCR'
plot(x, ...)
```

Arguments

x An object of class "EconBCR".

... Additional graphical arguments passed to [graphics::barplot\(\)](#).

Value

Invisibly returns the midpoints of the bars.

Examples

```
x <- benefit_cost_ratio(
  benefit = 150000,
  cost = 100000
)

plot(x)
```

plot.EconBIA	<i>Plot Budget Impact Analysis</i>
--------------	------------------------------------

Description

Produces a visualization of the annual incremental budget impact and cumulative budget impact over time.

Usage

```
## S3 method for class 'EconBIA'
plot(x, type = c("annual", "cumulative"), ...)
```

Arguments

x	An object of class "EconBIA".
type	Type of plot. One of: "annual" Annual incremental budget impact. "cumulative" Cumulative discounted budget impact.
...	Additional graphical arguments.

Value

Produces a budget impact plot.

Examples

```
## Not run:
plot(bia)
plot(bia, type = "cumulative")

## End(Not run)
```

plot.EconCEA	<i>Plot Cost-Effectiveness Plane</i>
--------------	--------------------------------------

Description

Plot Cost-Effectiveness Plane

Usage

```
## S3 method for class 'EconCEA'
plot(x, ...)
```

Arguments

x An object of class "EconCEA".
... Additional graphical arguments.

Value

A Cost-Effectiveness Plane.

plot.EconCEAC	<i>Plot Cost-Effectiveness Acceptability Curve</i>
---------------	--

Description

Produces a Cost-Effectiveness Acceptability Curve (CEAC) showing the probability that an intervention is cost-effective over a range of willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconCEAC'
plot(x, ...)
```

Arguments

x An object of class "EconCEAC".
... Additional graphical arguments.

Value

Produces a CEAC plot.

plot.EconCEAF	<i>Plot Cost-Effectiveness Acceptability Frontier</i>
---------------	---

Description

Produces a Cost-Effectiveness Acceptability Frontier (CEAF) showing the probability that the optimal strategy is cost-effective over a range of willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconCEAF'
plot(x, ...)
```

Arguments

x	An object of class "EconCEAF".
...	Additional graphical arguments.

Value

Produces a CEAF plot.

Examples

```
## Not run:
plot(ceaf)

## End(Not run)
```

plot.EconEVPI	<i>Plot Expected Value of Perfect Information</i>
---------------	---

Description

Produces an Expected Value of Perfect Information (EVPI) curve across willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconEVPI'
plot(x, ...)
```

Arguments

x	An object of class "EconEVPI".
...	Additional graphical arguments.

Value

Produces an EVPI plot.

Examples

```
## Not run:  
plot(evpi)  
  
## End(Not run)
```

plot.EconEVPPI

Plot Expected Value of Partial Perfect Information

Description

Produces an Expected Value of Partial Perfect Information (EVPPI) curve across willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconEVPPI'  
plot(x, ...)
```

Arguments

x An object of class "EconEVPPI".
... Additional graphical arguments.

Value

Produces an EVPPI plot.

Examples

```
## Not run:  
plot(evppi)  
  
## End(Not run)
```

plot.EconMWSA	<i>Plot Multi-Way Sensitivity Analysis</i>
---------------	--

Description

Plot Multi-Way Sensitivity Analysis

Usage

```
## S3 method for class 'EconMWSA'  
plot(x, ...)
```

Arguments

x	An object of class "EconMWSA".
...	Additional graphical arguments.

Value

Produces a line plot of model outcomes.

plot.EconPSA	<i>Plot Probabilistic Sensitivity Analysis</i>
--------------	--

Description

Displays the distribution of simulated outcomes.

Usage

```
## S3 method for class 'EconPSA'  
plot(x, ...)
```

Arguments

x	An object of class "EconPSA".
...	Additional graphical arguments.

Value

Produces a histogram with density curve.

plot.EconThreshold	<i>Plot Threshold Analysis</i>
--------------------	--------------------------------

Description

Plot Threshold Analysis

Usage

```
## S3 method for class 'EconThreshold'  
plot(x, ...)
```

Arguments

x	An object of class "EconThreshold".
...	Additional graphical arguments.

Value

Produces a threshold analysis plot.

plot.EconTornado	<i>Plot Tornado Diagram</i>
------------------	-----------------------------

Description

Plot Tornado Diagram

Usage

```
## S3 method for class 'EconTornado'  
plot(x, ...)
```

Arguments

x	An object of class "EconTornado".
...	Additional graphical arguments.

plot.EconTree	<i>Plot Decision Tree Analysis Results</i>
---------------	--

Description

Produces a bar plot of expected costs or expected effectiveness for each terminal strategy.

Usage

```
## S3 method for class 'EconTree'
plot(x, type = c("cost", "effectiveness"), ...)
```

Arguments

x	An object of class "EconTree".
type	Character string specifying the quantity to plot. One of "cost" or "effectiveness".
...	Additional graphical arguments.

Value

Produces a bar plot.

Examples

```
## Not run:
plot(tree)
plot(tree, type = "effectiveness")

## End(Not run)
```

present_value	<i>Present Value</i>
---------------	----------------------

Description

Calculates the present value of one or more future cash flows.

Usage

```
present_value(future_value, rate, period)
```

Arguments

future_value	Numeric vector of future values.
rate	Discount rate expressed as a decimal.
period	Integer vector of periods.

Details

Present value is calculated as

$$PV = \frac{FV}{(1+r)^t}$$

where

- FV = Future Value
- r = Discount rate
- t = Time period

Value

An object of class "EconPV".

Examples

```
pv <- present_value(
  future_value = c(1000,1500,2000),
  rate = 0.10,
  period = 1:3
)

print(pv)
```

<code>print.EconBCR</code>	<i>Print an EconBCR object</i>
----------------------------	--------------------------------

Description

Print an EconBCR object

Usage

```
## S3 method for class 'EconBCR'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconBCR".
<code>...</code>	Additional arguments (unused).

Value

Invisibly returns the object.

<code>print.EconBIA</code>	<i>Print a Budget Impact Analysis Object</i>
----------------------------	--

Description

Print a Budget Impact Analysis Object

Usage

```
## S3 method for class 'EconBIA'  
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconBIA".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconCEAC</code>	<i>Print a Cost-Effectiveness Acceptability Curve Object</i>
-----------------------------	--

Description

Print a Cost-Effectiveness Acceptability Curve Object

Usage

```
## S3 method for class 'EconCEAC'  
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconCEAC".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconCEAF</code>	<i>Print a Cost-Effectiveness Acceptability Frontier Object</i>
-----------------------------	---

Description

Print a Cost-Effectiveness Acceptability Frontier Object

Usage

```
## S3 method for class 'EconCEAF'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconCEAF".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconEVPI</code>	<i>Print an Expected Value of Perfect Information Object</i>
-----------------------------	--

Description

Print an Expected Value of Perfect Information Object

Usage

```
## S3 method for class 'EconEVPI'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconEVPI".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconEVPPI</code>	<i>Print an Expected Value of Partial Perfect Information Object</i>
------------------------------	--

Description

Print an Expected Value of Partial Perfect Information Object

Usage

```
## S3 method for class 'EconEVPPI'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconEVPPI".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconPSA</code>	<i>Print a Probabilistic Sensitivity Analysis Object</i>
----------------------------	--

Description

Print a Probabilistic Sensitivity Analysis Object

Usage

```
## S3 method for class 'EconPSA'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconPSA".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconTornado</code>	<i>Print a Tornado Diagram Object</i>
--------------------------------	---------------------------------------

Description

Print a Tornado Diagram Object

Usage

```
## S3 method for class 'EconTornado'  
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconTornado".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

<code>print.EconTree</code>	<i>Print a Decision Tree Analysis Object</i>
-----------------------------	--

Description

Print a Decision Tree Analysis Object

Usage

```
## S3 method for class 'EconTree'  
print(x, ...)
```

Arguments

<code>x</code>	An object of class "EconTree".
<code>...</code>	Additional arguments (unused).

Value

The input object, invisibly.

probabilistic_sensitivity_analysis*Probabilistic Sensitivity Analysis*

Description

Performs Monte Carlo simulation for economic evaluation.

Usage

```
probabilistic_sensitivity_analysis(n = 1000, sampler, model, seed = NULL)
```

Arguments

n	Integer. Number of Monte Carlo simulations.
sampler	A function generating one random parameter set. The function should return a list.
model	A function accepting the sampled parameter list and returning a single numeric outcome.
seed	Optional integer random seed.

Details

This function performs probabilistic sensitivity analysis (PSA) using Monte Carlo simulation. During each iteration, random parameters are generated by **sampler()** and passed to **model()**. The model must return one finite numeric outcome.

Value

An object of class "EconPSA".

Examples

```
sampler <- function() {  
  list(  
    cost = rnorm(1, 1000, 100),  
    effectiveness = rbeta(1, 20, 5)  
  )  
}  
  
model <- function(par) {  
  par$effectiveness * 5000 -  
  par$cost  
}
```

```
psa <- probabilistic_sensitivity_analysis(
  n = 1000,
  sampler = sampler,
  model = model,
  seed = 123
)
```

profitability_index	<i>Profitability Index</i>
---------------------	----------------------------

Description

Calculates the Profitability Index (PI).

Usage

```
profitability_index(cashflow, rate)
```

Arguments

cashflow	Numeric vector of cash flows.
rate	Discount rate.

Value

Object of class "EconPI".

summary.EconBCR	<i>Summarize an EconBCR object</i>
-----------------	------------------------------------

Description

Summarize an EconBCR object

Usage

```
## S3 method for class 'EconBCR'
summary(object, ...)
```

Arguments

object	An object of class "EconBCR".
...	Additional arguments (unused).

Value

A data frame summarizing the benefit-cost analysis.

summary.EconBIA	<i>Summarize a Budget Impact Analysis Object</i>
-----------------	--

Description

Returns a summary of the annual and cumulative budget impact over the specified time horizon.

Usage

```
## S3 method for class 'EconBIA'  
summary(object, ...)
```

Arguments

object	An object of class "EconBIA".
...	Additional arguments (unused).

Value

A data frame containing annual budget impact results.

Examples

```
## Not run:  
summary(bia)  
  
## End(Not run)
```

summary.EconCEA	<i>Summarize a Cost-Effectiveness Analysis</i>
-----------------	--

Description

Summarize a Cost-Effectiveness Analysis

Usage

```
## S3 method for class 'EconCEA'  
summary(object, ...)
```

Arguments

object	An object of class "EconCEA".
...	Additional arguments (unused).

Value

A data frame summarizing the cost-effectiveness analysis.

summary.EconCEAC	<i>Summarize a Cost-Effectiveness Acceptability Curve Object</i>
------------------	--

Description

Summarize a Cost-Effectiveness Acceptability Curve Object

Usage

```
## S3 method for class 'EconCEAC'
summary(object, ...)
```

Arguments

object	An object of class "EconCEAC".
...	Additional arguments (unused).

Value

A data frame summarizing the Cost-Effectiveness Acceptability Curve.

summary.EconCEAF	<i>Summarize a Cost-Effectiveness Acceptability Frontier Object</i>
------------------	---

Description

Returns a summary of the Cost-Effectiveness Acceptability Frontier (CEAF) results across all willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconCEAF'
summary(object, ...)
```

Arguments

object	An object of class "EconCEAF".
...	Additional arguments (unused).

Value

A data frame containing:

WTP Willingness-to-pay threshold.

OptimalStrategy Optimal strategy at each WTP.

Probability Probability that the optimal strategy is truly cost-effective.

ExpectedNMB Expected Net Monetary Benefit.

Examples

```
## Not run:
summary(ceaf)

## End(Not run)
```

summary.EconEVPI

Summarize an Expected Value of Perfect Information Object

Description

Returns a summary of the Expected Value of Perfect Information (EVPI) across willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconEVPI'
summary(object, ...)
```

Arguments

object An object of class "EconEVPI".

... Additional arguments (unused).

Value

A data frame containing:

WTP Willingness-to-pay threshold.

EVPI Expected Value of Perfect Information.

ExpectedPerfectInformation Expected value under perfect information, $E[\max(NMB)]$.

ExpectedCurrentInformation Expected value under current information, $\max(E[NMB])$.

Examples

```
## Not run:
summary(evpi)

## End(Not run)
```

summary.EconEVPPI	<i>Summarize an Expected Value of Partial Perfect Information Object</i>
-------------------	--

Description

Returns a summary of the Expected Value of Partial Perfect Information (EVPPI) across willingness-to-pay (WTP) thresholds.

Usage

```
## S3 method for class 'EconEVPPI'
summary(object, ...)
```

Arguments

object	An object of class "EconEVPPI".
...	Additional arguments (unused).

Value

A data frame containing:

WTP Willingness-to-pay threshold.

EVPPI Expected Value of Partial Perfect Information.

Examples

```
## Not run:
summary(evppi)

## End(Not run)
```

summary.EconPSA	<i>Summarize a Probabilistic Sensitivity Analysis Object</i>
-----------------	--

Description

Summarize a Probabilistic Sensitivity Analysis Object

Usage

```
## S3 method for class 'EconPSA'  
summary(object, ...)
```

Arguments

object	An object of class "EconPSA".
...	Additional arguments (unused).

Value

A data frame containing summary statistics.

summary.EconTornado	<i>Summarize a Tornado Diagram Object</i>
---------------------	---

Description

Summarize a Tornado Diagram Object

Usage

```
## S3 method for class 'EconTornado'  
summary(object, ...)
```

Arguments

object	An object of class "EconTornado".
...	Additional arguments (unused).

Value

A data frame summarizing the tornado analysis.

summary.EconTree	<i>Summarize a Decision Tree Analysis Object</i>
------------------	--

Description

Returns a summary of the expected costs and expected effectiveness for each terminal node.

Usage

```
## S3 method for class 'EconTree'
summary(object, ...)
```

Arguments

object	An object of class "EconTree".
...	Additional arguments (unused).

Value

A data frame containing:

Strategy Terminal strategy.

Probability Terminal node probability.

Cost Observed cost.

Effectiveness Observed effectiveness.

ExpectedCost Expected cost.

ExpectedEffectiveness Expected effectiveness.

Examples

```
## Not run:
summary(tree)

## End(Not run)
```

threshold_analysis	<i>Threshold Analysis</i>
--------------------	---------------------------

Description

Identifies the threshold value of a parameter at which a model outcome reaches a specified target.

Usage

```
threshold_analysis(lower, upper, target = 0, model, tol = 1e-08)
```

Arguments

lower	Lower search bound.
upper	Upper search bound.
target	Target outcome.
model	Function returning the model outcome.
tol	Numerical tolerance.

Value

Object of class "EconThreshold".

tornado_diagram	<i>Tornado Diagram</i>
-----------------	------------------------

Description

Creates data for a tornado diagram.

Usage

```
tornado_diagram(parameter, low, high, base_case)
```

Arguments

parameter	Character vector of parameter names.
low	Numeric vector of outcomes at the low value.
high	Numeric vector of outcomes at the high value.
base_case	Numeric scalar.

Value

Object of class "EconTornado".

`two_way_sensitivity_analysis`*Two-Way Sensitivity Analysis*

Description

Performs deterministic two-way sensitivity analysis by varying two model parameters simultaneously.

Usage

```
two_way_sensitivity_analysis(parameter1, parameter2, model)
```

Arguments

<code>parameter1</code>	Numeric vector of values for parameter 1.
<code>parameter2</code>	Numeric vector of values for parameter 2.
<code>model</code>	Function accepting two arguments and returning a numeric outcome.

Value

An object of class "EconTWSA".

Index

`as.data.frame.EconBCR`, 3
`as.data.frame.EconBIA`, 4
`as.data.frame.EconCEAC`, 4
`as.data.frame.EconCEAF`, 5
`as.data.frame.EconEVPI`, 6
`as.data.frame.EconEVPPI`, 7
`as.data.frame.EconPSA`, 7
`as.data.frame.EconTornado`, 8
`as.data.frame.EconTree`, 8

`benefit_cost_ratio`, 9
`budget_impact_analysis`, 10

`cost_effectiveness`, 11
`cost_effectiveness_acceptability_curve`,
12
`cost_effectiveness_acceptability_frontier`,
13

`decision_tree_analysis`, 14
`discount_cashflow`, 16
`discounted_payback_period`, 17
`dominance_analysis`, 17

`expected_value_partial_perfect_information`,
18
`expected_value_perfect_information`,
19

`graphics::barplot()`, 28

`icer`, 21
`internal_rate_return`, 22

`modified_internal_rate_return`, 22
`multi_way_sensitivity_analysis`, 23

`net_health_benefit`, 24
`net_monetary_benefit`, 25
`net_present_value`, 26

`one_way_sensitivity_analysis`, 26

`partial_budget`, 27
`payback_period`, 27
`plot.EconBCR`, 28
`plot.EconBIA`, 29
`plot.EconCEA`, 30
`plot.EconCEAC`, 30
`plot.EconCEAF`, 31
`plot.EconEVPI`, 31
`plot.EconEVPPI`, 32
`plot.EconMWSA`, 33
`plot.EconPSA`, 33
`plot.EconThreshold`, 34
`plot.EconTornado`, 34
`plot.EconTree`, 35
`present_value`, 35
`print.EconBCR`, 36
`print.EconBIA`, 37
`print.EconCEAC`, 37
`print.EconCEAF`, 38
`print.EconEVPI`, 38
`print.EconEVPPI`, 39
`print.EconPSA`, 39
`print.EconTornado`, 40
`print.EconTree`, 40
`probabilistic_sensitivity_analysis`,
41
`profitability_index`, 42

`summary.EconBCR`, 42
`summary.EconBIA`, 43
`summary.EconCEA`, 43
`summary.EconCEAC`, 44
`summary.EconCEAF`, 44
`summary.EconEVPI`, 45
`summary.EconEVPPI`, 46
`summary.EconPSA`, 47
`summary.EconTornado`, 47
`summary.EconTree`, 48

`threshold_analysis`, 49

tornado_diagram, [49](#)
two_way_sensitivity_analysis, [50](#)