Package 'ceRtainty'

October 12, 2022

Type Package

Title Certainty Equivalent

Version 1.0.0

Description Compute the certainty equivalents and premium risks as tools for risk-efficiency analysis. For more technical information, please refer to: Hardaker, Richardson, Lien, & Schumann (2004) <doi:10.1111/j.1467-8489.2004.00239.x>, and Richardson, & Outlaw (2008) <doi:10.2495/RISK080231>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr, tidyr, RColorBrewer, stats, base

RoxygenNote 6.1.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Ariel Soto-Caro [aut, cre] (<https://orcid.org/0000-0001-7008-4009>)

Maintainer Ariel Soto-Caro <arielsotocaro@gmail.com>

Repository CRAN

Date/Publication 2019-06-14 14:40:03 UTC

R topics documented:

`add_legend` *Customized Legend for CE and RP plots.*

Description

Customized Legend for CE and RP plots.

Usage

```
add_legend(...)
```

Arguments

`...`	free

`certainty` *Certainty equivalent computation*

Description

Certainty equivalent computation

Usage

```
certainty(data, ival, fval, utility, wealth = 0)
```

Arguments

`data`	`data.set` with profit for each treatment/project. Each column is a treatment and each row a different profit observation.
`ival`	The initial value for the RAC vector to employ (scalar).
`fval`	The final value for the RAC vector to employ (scalar).
`utility`	Indicator of utility function: "ExpNeg" for the Exponential Negative utility, and "Power" for the Power utility function.
`wealth`	The initial agent wealth. By default is zero.

Details

This function computes the certainty equivalent values using profit as inputs. Works with `data.frames` with 3 or more observations. Consider each column as a different treatment or project.

Value

This function produces three objects: `CE_values` is a table with treatment by columns and certainty values by row; `RAC` is a vector with the absolute risk aversion coefficients (ARAC) if the Power utility function was implemented, or the relative risk aversion coefficient (RRAC) if the Exponential Negative utility function was implemented. The length of this vector is the same as the number of profit observations in the original dataset; and, `CE_plot` is a graph using plot function, to compare the different CEs computed.

References

Hardaker, J.B., Richardson, J.W., Lien, G., & Schumann, K.D. (2004). Stochastic efficiency analysis with risk aversion bounds: a simplified approach. Australian Journal of Agricultural and Resource Economics, 48(2), 253-270.

Examples

```
## Example 1. Using profit data from ceRtainty package
data(profitSWG)

# Storing CE values using Power utility function
c1 <- certainty(data    = profitSWG,
                ival    = .5,
                fval    = 4,
                utility = "Power")
c1$CE_values # Table with CE values
c1$RAC       # RAC vector used in CE computation
c1$CE_plot() # Invoking the CE plot

# To use the ExpNeg function, it is required the RRAC (ARAC/wealth)
# so we can compute the mean value among all profit in the dataset.

# Mean value among all profit value
mean(sapply(profitSWG,mean)) # 5081.844

# Storing CE values using Power utility function
c1 <- certainty(data    = profitSWG,
                ival    = .5/5082,
                fval    = 4/5082,
                utility = "ExpNeg")

c1$CE_values # Table with CE values
c1$RAC       # RAC vector used in CE computation
c1$CE_plot() # Invoking the CE plot

## Example 2. Using the example values of Hardaker et al. (2004)
dt <- data.frame(treatment=c(100,125,135,142,147,150,153,158,163,175,195))
```

```
# Storing CE values using Power utility function. Hardaker use an
# unique RAC value (0.005)
c2 <- certainty(data    = dt,
                ival    = .005,
                fval    = .005,
                utility = "Power")
# or
c2 <- certainty(data    = dt,
                ival    = .005,
                fval    = .005,
                utility = "ExpNeg")

c2$CE_values
c2$RAC
c2$CE_plot()
```

---

ce_en *Certainty Equivalent Function for Negative Exponential Function*

Description

Certainty Equivalent Function for Negative Exponential Function

Usage

```
ce_en(profit, rac, weight = 0)
```

Arguments

profit	data.frame with profit values
rac	scalar of RAC value
weight	original wealth

Value

Scalar with the CE value

---

ce_epnegative *Certainty Equivalent Computation using Power Utility Function*

Description

Certainty Equivalent Computation using Power Utility Function

Usage

```
ce_epnegative(data, rac_ini, rac_fin, weight = 0)
```

Arguments

data	data.frame with profit values
rac_ini	Initial value for the RAC sequence
rac_fin	Final value for the RAC sequence
weight	Original wealth

Value

Generate three objects: A table with the CEs, a vector of risk aversion coefficients RAC, and a plot to compare the CEs.

---

ce_p *Certainty Equivalent Function with Power Utility Function*

Description

Certainty Equivalent Function with Power Utility Function

Usage

```
ce_p(profit, rac, weight = 0)
```

Arguments

profit	data.frame with profit values
rac	scalar of RAC value
weight	original wealth

Value

Scalar with the CE value

---

ce_power *Certainty Equivalent Computation using Power Utility Function*

Description

Certainty Equivalent Computation using Power Utility Function

Usage

```
ce_power(data, rac_ini, rac_fin, weight = 0)
```

Arguments

data	data.frame with profit values
rac_ini	Initial value for the RAC sequence
rac_fin	Final value for the RAC sequence
weight	Original wealth

Value

Generate three objects: A table with the CEs, a vector of risk aversion coefficients RAC, and a plot to compare the CEs.

---

plot_ce_en *Plot for CE using Exponential Negative Utility Function*

Description

Plot for CE using Exponential Negative Utility Function

Usage

```
plot_ce_en(data, rac = 0, rac_ini = 0, rac_fin = 1, rac_len = 10)
```

Arguments

data	Data set with CE already computed
rac	Scalar with the RAC to use in the CE computation. When the analysis consider only one value of RAC
rac_ini	Vector of the RAC to use in the CE computation. When the analysis consider a sequence of RAC values
rac_fin	Final value for the RAC vector
rac_len	RAC vector length

Value

Plot of CE to compare treatments/projects

plot_ce_power *Plot for CE using Power Utility Function*

Description

Plot for CE using Power Utility Function

Usage

```
plot_ce_power(data, rac = 0, rac_ini = 0, rac_fin = 1,
  rac_len = 10)
```

Arguments

data	Data set with CE already computed
rac	Scalar with the RAC to use in the CE computation. When the analysis consider only one value of RAC
rac_ini	Vector of the RAC to use in the CE computation. When the analysis consider a sequence of RAC values
rac_fin	Final value for the RAC vector
rac_len	RAC vector length

Value

Plot of CE to compare treatments/projects

plot_risk_premium_en *Plot of the Risk Premium values using Exponential Negative Utility Function*

Description

Plot of the Risk Premium values using Exponential Negative Utility Function

Usage

```
plot_risk_premium_en(data, rac_ini, rac_fin, rac_len)
```

Arguments

data	data.frame of CE computed by Exponential Negative function
rac_ini	Initial RAC values used in the CE computation
rac_fin	Final RAC values used in the CE computation
rac_len	Length of the RAC vector used in the CE computation

Value

Plot object

`plot_risk_premium_p` *Plot of the Risk Premium values using Exponential Negative Utility Function*

Description

Plot of the Risk Premium values using Exponential Negative Utility Function

Usage

```
plot_risk_premium_p(data, rac_ini, rac_fin, rac_len)
```

Arguments

`data`	`data.frame` of CE computed by Power Utility function
`rac_ini`	Initial RAC values used in the CE computation
`rac_fin`	Final RAC values used in the CE computation
`rac_len`	Length of the RAC vector used in the CE computation

Value

plot object

`premium` *Risk Premium computation*

Description

Risk Premium computation

Usage

```
premium(tbase, ce_data, rac, utility)
```

Arguments

`tbase`	Name of the base treatment/project
`ce_data`	`data.frame` with CE values previously computed
`rac`	Vector with RAC sequence used in the CE computation
`utility`	The utility function: "ExpNeg" if CE it was computed using Exponential Negative utility function. "Power" if the utility function was Power

Details

This function computes the risk premium values, regarding a project or treatment arbitrarily chosen by the user, using a CEs dataset (a 'certainty' object) already computed.

Value

Generates three objects: A data.frame with the total values of the premium risks; a `data.frame` with the percentage of difference with respect the base treatment; and a plot with the treatments' premium risk.

Examples

```
## Example using profit dataset
data(profitSWG)

# First, compute the CE values
c1 <- certainty(data = profitSWG,ival = .5,fval = 4,utility = "Power")

ce_values <- c1$CE_values  # CE table
ce_rac <- c1$RAC           # RAC vector

# The Risk premium values respect to Serenade treatment
rp <- premium(tbase = "serenade", ce_data = ce_values,rac = ce_rac, utility = "Power")

rp$PremiumRisk       # absolute values
rp$PremiumRiskPer100 # values in percentage
rp$RP_plot()         # plot
```

`profitSWG` *Profit SWG*

Description

The data come from strawberry trials experiments in Florida, USA, performed by the Gulf Coast Research and Education Center, University of Florida.

Usage

```
data(profitSWG)
```

Format

A data frame with 8 rows of profit and 4 pesticide treatments:

control Control (non-treated case), in US dollars

fracture Fracture treatment, in US dollars

milstop Milstop treatment, in US dollars

serenade Serenade Optimum treatment, in US dollars

Details

Correspond to the profit for three pesticide treatments plus the case without treatment. Four observations for each season, 2014-15 and 2015-16.

Source

Soto-Caro, Wu, Guan (2019). "Evaluating Pest Management Strategies: A Robust Method and Its Application to Strawberry Disease Management". AAEA 2019 Conference.

References

Soto-Caro, Wu, Guan (2019). "Evaluating Pest Management Strategies: A Robust Method and Its Application to Strawberry Disease Management". AAEA 2019 Conference.

Examples

```
data(profitSWG)
summary(profitSWG)
```

`RACa` *Adjusted Risk Aversion Coefficient (RACa)*

Description

Adjusted Risk Aversion Coefficient (RACa)

Usage

```
RACa(rac, data)
```

Arguments

`rac`	An scalar with the value of the relative RAC
`data`	Dataset to weight the RAC

Value

This function create an adjustment to the relative risk aversion coefficient, following Hardaker et al (2004).

rac_generator *RAC Generator*

Description

RAC Generator

Usage

```
rac_generator(data, ini, fin)
```

Arguments

data	data.frame object to weight the RAC
ini	The initial value of the risk aversion coefficient (RAC) sequence
fin	The final value of the risk aversion coefficient (RAC) sequence

Details

Create a vector with the adjusted relative risk aversion coefficients to be used in the CE computation, under Power utility function.

Value

Produce a single vector of adjusted RACs.

Examples

```
# Example
data("profitSWG")
rac_generator(data = profitSWG$control, ini = 0.5, fin = 4.0)
```

rac_len *Define the length of the Risk Aversion Coefficient, RAC.*

Description

Define the length of the Risk Aversion Coefficient, RAC.

Usage

```
rac_len(ini, fin, data)
```

Arguments

ini	The initial value of the RAC sequence
fin	The final value of the RAC sequence
data	Original data, could be a vector or a matrix

Value

Two elements are generated: "r" is the RAC vector, and "length" is a scalar with the number of elements on RAC vector.

---

rac_seq *RAC Sequence generator*

Description

RAC Sequence generator

Usage

```
rac_seq(ini, fin, len)
```

Arguments

ini	The initial value for the RAC
fin	The final value for the RAC
len	The Length of the vector to creates

Value

Vector of RACs

Index