Package 'scoringfunctions'

November 10, 2024

Version 0.0.7

Date 2024-11-10

Title A Collection of Scoring Functions for Assessing Point Forecasts

Description Implements multiple consistent scoring functions (Gneiting T (2011) <doi:10.1198/jasa.2011.r10138>) for assessing point forecasts and point predictions. Detailed documentation of scoring functions' properties is included for facilitating interpretation of results.

Depends R (>= 4.0.0)

License GPL-3

Author Hristos Tyralis [aut, cre] (<https://orcid.org/0000-0002-8932-4997>), Georgia Papacharalampous [aut] (<https://orcid.org/0000-0001-5446-954X>)

Maintainer Hristos Tyralis <montchrister@gmail.com>

Repository CRAN

NeedsCompilation no

Date/Publication 2024-11-10 17:00:04 UTC

Contents

`scoringfunctions-package`

Overview of the functions in the scoringfunctions package

Description

The scoringfunctions package implements consistent scoring (loss) functions and identification functions

Details

The package functions are categorized into the following classes:

- Consistent scoring functions for one-dimensional functionals
- Consistent scoring functions for two-dimensional functionals
- Identification functions for one-dimensional functionals
- Supporting functions

.

1. Consistent scoring functions for one-dimensional functionals

`aerr_sf`: Absolute error scoring function

`aperr_sf`: Absolute percentage error scoring function

`bmedian_sf`: β-median scoring function

`bregman1_sf`: Bregman scoring function (type 1)

`bregman2_sf`: Bregman scoring function (type 2, Patton scoring function)

`bregman3_sf`: Bregman scoring function (type 3, QLIKE scoring function)

`bregman4_sf`: Bregman scoring function (type 4, Patton scoring function)

`expectile_sf`: Asymmetric piecewise quadratic scoring function (expectile scoring function, expectile loss function)

`ghuber_sf`: Generalized Huber scoring function

`gpl1_sf`: Generalized piecewise linear power scoring function (type 1)

`gpl2_sf`: Generalized piecewise linear power scoring function (type 2)

`huber_sf`: Huber scoring function

`maelog_sf`: MAE-LOG scoring function

`maesd_sf`: MAE-SD scoring function

`nmoment_sf`: n-th moment scoring function

`obsweighted_sf`: Observation-weighted scoring function

`quantile_sf`: Asymmetric piecewise linear scoring function (quantile scoring function, quantile loss function)

`relerr_sf`: Relative error scoring function (MAE-PROP scoring function)

`serr_sf`: Squared error scoring function

`sperr_sf`: Squared percentage error scoring function

`srelerr_sf`: Squared relative error scoring function

2. Consistent scoring functions for two-dimensional functionals

`interval_sf`: Interval scoring function (Winkler scoring function)

`mv_sf`: Mean - variance scoring function

3. Identification functions for one-dimensional functionals

`expectile_if`: Expectile identification function

`mean_if`: Mean identification function

`nmoment_if`: n-th moment identification function

`quantile_if`: Quantile identification function

4. Identification functions for two-dimensional functionals

`mv_if`: Mean - variance identification function

5. Supporting functions

capping_function: Capping function

---

aerr_sf *Absolute error scoring function*

Description

The function aerr_sf computes the absolute error scoring function when y materializes and x is the predictive median functional.

The absolute error scoring function is defined in Table 1 in Gneiting (2011).

Usage

```
aerr_sf(x, y)
```

Arguments

x	Predictive median functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The absolute error scoring function is defined by:

$$S(x, y) := |x - y|$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

Range of function:

$$S(x, y) \geq 0, \forall x, y \in \mathbb{R}$$

Value

Vector of absolute errors.

Note

For details on the absolute error scoring function, see Gneiting (2011).

The median functional is the median of the probability distribution F of y (Gneiting 2011).

The absolute error scoring function is negatively oriented (i.e. the smaller, the better).

The absolute error scoring function is strictly $\mathbb{F}$-consistent for the median functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ exists and is finite (Raiffa and Schlaifer 1961, p.196; Ferguson 1967, p.51; Thomson 1979; Saerens 2000; Gneiting 2011).

References

Ferguson TS (1967) Mathematical Statistics: A Decision-Theoretic Approach. Academic Press, New York.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Raiffa H,Schlaifer R (1961) Applied Statistical Decision Theory. Colonial Press, Clinton.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the absolute error scoring function.

df <- data.frame(
    y = rep(x = 0, times = 5),
    x = -2:2
)

df$absolute_error <- aerr_sf(x = df$x, y = df$y)

print(df)
```

`aperr_sf` *Absolute percentage error scoring function*

Description

The function aperr_sf computes the absolute percentage error scoring function when y materializes and x is the predictive $\mathrm{med}^{(-1)}(F)$ functional.

The absolute percentage error scoring function is defined in Table 1 in Gneiting (2011).

Usage

```
aperr_sf(x, y)
```

Arguments

x	Predictive $\text{med}^{(-1)}(F)$ functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The absolute percentage error scoring function is defined by:

$$S(x, y) := |(x - y)/y|$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of absolute percentage errors.

Note

For details on the absolute percentage error scoring function, see Gneiting (2011).

The β-median functional, $\text{med}^{(\beta)}(F)$ is the median of a probability distribution whose density is proportional to $y^{\beta} f(y)$, where f is the density of the probability distribution F of y (Gneiting 2011).

The absolute percentage error scoring function is negatively oriented (i.e. the smaller, the better).

The absolute percentage error scoring function is strictly $\mathbb{F}^{(w)}$-consistent for the $\text{med}^{(-1)}(F)$ functional. $\mathbb{F}$ is the family of probability distributions for which $\text{E}_F[Y]$ exists and is finite. $\mathbb{F}^{(w)}$ is the subclass of probability distributions in $\mathbb{F}$, which are such that $w(y)f(y)$, $w(y) = 1/y$ has finite integral over $(0, \infty)$, and the probability distribution $F^{(w)}$ with density proportional to $w(y)f(y)$ belongs to $\mathbb{F}$ (see Theorems 5 and 9 in Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the absolute percentage error scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$absolute_percentage_error <- aperr_sf(x = df$x, y = df$y)

print(df)
```

`bmedian_sf` *β-median scoring function*

Description

The function bmedian_sf computes the β-median scoring function when y materializes and x is the predictive $\text{med}^{(\beta)}(F)$ functional.

The β-median scoring function is defined in eq. (4) in Gneiting (2011).

Usage

```
bmedian_sf(x, y, b)
```

Arguments

x	Predictive $\text{med}^{(\beta)}(F)$ functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
b	It can be a vector of length n (must have the same length as y).

Details

The β-median scoring function is defined by:

$$S(x, y, b) := |1 - (y/x)^b|$$

Domain of function:

$$x > 0$$

$$y > 0$$

$$b \neq 0$$

Range of function:

$$S(x, y, b) \geq 0, \forall x, y > 0, b \neq 0$$

Value

Vector of β-median losses.

Note

For details on the β-median scoring function, see Gneiting (2011).

The β-median functional, $\text{med}^{(\beta)}(F)$ is the median of a probability distribution whose density is proportional to $y^{\beta} f(y)$, where f is the density of the probability distribution F of y (Gneiting 2011).

The β-median scoring function is negatively oriented (i.e. the smaller, the better).

The β-median scoring function is strictly $\mathbb{F}^{(w)}$-consistent for the $\text{med}^{(\beta)}(F)$ functional. $\mathbb{F}$ is the family of probability distributions for which $\text{E}_F[Y]$ exists and is finite. $\mathbb{F}^{(w)}$ is the subclass of probability distributions in $\mathbb{F}$, which are such that $w(y)f(y)$, $w(y) = y^{\beta}$ has finite integral over $(0, \infty)$, and the probability distribution $F^{(w)}$ with density proportional to $w(y)f(y)$ belongs to $\mathbb{F}$ (see Theorems 5 and 9 in Gneiting 2011)

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the bmedian scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3,
    b = c(-1, 1, 2)
)

df$bmedian_error <- bmedian_sf(x = df$x, y = df$y, b = df$b)

print(df)
```

`bregman1_sf` *Bregman scoring function (type 1)*

Description

The function bregman1_sf computes the Bregman scoring function when y materializes and x is the predictive mean functional.

The Bregman scoring function is defined by eq. (18) in Gneiting (2011) and the form implemented here for $\phi(x) = |x|^a$ is defined by eq. (19) in Gneiting (2011).

Usage

```
bregman1_sf(x, y, a)
```

Arguments

`x`	Predictive mean functional (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
`a`	It can be a vector of length n (must have the same length as y).

Details

The Bregman scoring function (type 1) is defined by:

$$S(x, y, a) := |y|^a - |x|^a - a\text{sign}(x)|x|^{a-1}(y - x)$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$a > 1$$

Range of function:

$$S(x, y, a) \geq 0, \forall x, y \in \mathbb{R}, a > 1$$

Value

Vector of Bregman losses.

Note

The implemented function is denoted as type 1 since it corresponds to a specific type of $\phi(x)$ of the general form of the Bregman scoring function defined by eq. (18) in Gneiting (2011).

For details on the Bregman scoring function, see Savage (1971), Banerjee et al. (2005) and Gneiting (2011).

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The Bregman scoring function is negatively oriented (i.e. the smaller, the better).

The herein implemented Bregman scoring function is strictly $\mathbb{F}$-consistent for the mean functional. $\mathbb{F}$ is the family of probability distributions for which $\mathrm{E}_F[Y]$ and $\mathrm{E}_F[|Y|^a]$ exist and are finite (Savage 1971; Gneiting 2011).

References

Banerjee A, Guo X, Wang H (2005) On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory* **51(7)**:2664–2669. doi:10.1109/TIT.2005.850145.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Savage LJ (1971) Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* **66(337)**:783–810. doi:10.1080/01621459.1971.10482346.

Examples

```
# Compute the Bregman scoring function (type 1).

df <- data.frame(
    y = rep(x = 0, times = 7),
    x = c(-3, -2, -1, 0, 1, 2, 3),
    a = rep(x = 3, times = 7)
)

df$bregman1_penalty <- bregman1_sf(x = df$x, y = df$y, a = df$a)

print(df)

# Equivalence of Bregman scoring function (type 1) and squared error scoring
# function, when a = 2.

set.seed(12345)

n <- 100

x <- runif(n, -20, 20)
y <- runif(n, -20, 20)
a <- rep(x = 2, times = n)

u <- bregman1_sf(x = x, y = y, a = a)

v <- serr_sf(x = x, y = y)
```

```
max(abs(u - v)) # values are slightly higher than 0 due to rounding error
min(abs(u - v))
```

---

bregman2_sf *Bregman scoring function (type 2, Patton scoring function)*

Description

The function bregman2_sf computes the Bregman scoring function when y materializes and x is the predictive mean functional.

The Bregman scoring function is defined by eq. (18) in Gneiting (2011) and the form implemented here for $\phi(x) = \frac{1}{b(b-1)}x^b$, $b \in \mathsf{R} \setminus \{0, 1\}$ is defined by eq. (20) in Gneiting (2011).

Usage

```
bregman2_sf(x, y, b)
```

Arguments

x	Predictive mean functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
b	It can be a vector of length n (must have the same length as y).

Details

The Bregman scoring function (type 2) is defined by:

$$S(x, y, b) := \frac{1}{b(b-1)}(y^b - x^b) - \frac{1}{b-1}x^{b-1}(y - x)$$

Domain of function:

$$x > 0$$

$$y > 0$$

$$b \in \mathbb{R} \setminus \{0, 1\}$$

Range of function:

$$S(x, y, b) \geq 0, \forall x, y > 0, b \in \mathbb{R} \setminus \{0, 1\}$$

Value

Vector of Bregman losses.

Note

The implemented function is denoted as type 2 since it corresponds to a specific type of $\phi(x)$ of the general form of the Bregman scoring function defined by eq. (18) in Gneiting (2011).

For details on the Bregman scoring function, see Savage (1971), Banerjee et al. (2005) and Gneiting (2011). For details on the specific form implemented here, see Patton (2011).

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The Bregman scoring function is negatively oriented (i.e. the smaller, the better).

The herein implemented Bregman scoring function is strictly $\mathbb{F}$-consistent for the mean functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ and $\mathrm{E}_F[\frac{1}{b(b-1)}Y^b]$ exist and are finite (Savage 1971; Gneiting 2011).

References

Banerjee A, Guo X, Wang H (2005) On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory* **51(7)**:2664–2669. doi:10.1109/TIT.2005.850145.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Savage LJ (1971) Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* **66(337)**:783–810. doi:10.1080/01621459.1971.10482346.

Examples

```
# Compute the Bregman scoring function (type 2).

df <- data.frame(
    y = rep(x = 2, times = 6),
    x = rep(x = 1:3, times = 2),
    b = rep(x = c(-3, 3), each = 3)
)

df$bregman2_penalty <- bregman2_sf(x = df$x, y = df$y, b = df$b)

print(df)

# The Bregman scoring function (type 2) is half the squared error scoring
# function, when b = 2.

df <- data.frame(
    y = rep(x = 5.5, times = 10),
    x = 1:10,
    b = rep(x = 2, times = 10)
```

```
)

df$bregman2_penalty <- bregman2_sf(x = df$x, y = df$y, b = df$b)

df$squared_error <- serr_sf(x = df$x, y = df$y)

df$ratio <- df$bregman2_penalty/df$squared_error

print(df)


# When a = b > 1 the Bregman scoring function (type 2) coincides with the
# Bregman scoring function (type 1) up to a multiplicative constant.

df <- data.frame(
    y = rep(x = 5.5, times = 10),
    x = 1:10,
    b = rep(x = c(3, 4), each = 5)
)

df$bregman2_penalty <- bregman2_sf(x = df$x, y = df$y, b = df$b)

df$bregman1_penalty <- bregman1_sf(x = df$x, y = df$y, a = df$b)

df$ratio <- df$bregman2_penalty/df$bregman1_penalty

print(df)
```

`bregman3_sf` *Bregman scoring function (type 3, QLIKE scoring function)*

Description

The function bregman3_sf computes the Bregman scoring function when y materializes and x is the predictive mean functional.

The Bregman scoring function is defined by eq. (18) in Gneiting (2011) and the form implemented here for $\phi(x) = -\log(x)$ is defined by eq. (20) in Gneiting (2011).

Usage

```
bregman3_sf(x, y)
```

Arguments

`x`	Predictive mean functional (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The Bregman scoring function (type 3) is defined by:

$$S(x, y) := (y/x) - \log(y/x) - 1$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of Bregman losses.

Note

The implemented function is denoted as type 3 since it corresponds to a specific type of $\phi(x)$ of the general form of the Bregman scoring function defined by eq. (18) in Gneiting (2011).

For details on the Bregman scoring function, see Savage (1971), Banerjee et al. (2005) and Gneiting (2011). For details on the specific form implemented here, see the QLIKE scoring function in Patton (2011).

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The Bregman scoring function is negatively oriented (i.e. the smaller, the better).

The herein implemented Bregman scoring function is strictly $\mathbb{F}$-consistent for the mean functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ and $\mathrm{E}_F[\log(Y)]$ exist and are finite (Savage 1971; Gneiting 2011).

References

Banerjee A, Guo X, Wang H (2005) On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory* **51(7)**:2664–2669. doi:10.1109/TIT.2005.850145.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Savage LJ (1971) Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* **66(337)**:783–810. doi:10.1080/01621459.1971.10482346.

Examples

```
# Compute the Bregman scoring function (type 3, QLIKE scoring function).

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$bregman3_penalty <- bregman3_sf(x = df$x, y = df$y)

print(df)
```

`bregman4_sf` *Bregman scoring function (type 4, Patton scoring function)*

Description

The function bregman4_sf computes the Bregman scoring function when y materializes and x is the predictive mean functional.

The Bregman scoring function is defined by eq. (18) in Gneiting (2011) and the form implemented here for $\phi(x) = x\log(x)$ is defined by eq. (20) in Gneiting (2011).

Usage

```
bregman4_sf(x, y)
```

Arguments

`x`	Predictive mean functional (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The Bregman scoring function (type 4) is defined by:

$$S(x, y) := y\log(y/x) - y + x$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of Bregman losses.

Note

The implemented function is denoted as type 4 since it corresponds to a specific type of $\phi(x)$ of the general form of the Bregman scoring function defined by eq. (18) in Gneiting (2011).

For details on the Bregman scoring function, see Savage (1971), Banerjee et al. (2005) and Gneiting (2011). For details on the specific form implemented here, see Patton (2011).

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The Bregman scoring function is negatively oriented (i.e. the smaller, the better).

The herein implemented Bregman scoring function is strictly $\mathbb{F}$-consistent for the mean functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ and $\mathrm{E}_F[Y \log(Y)]$ exist and are finite (Savage 1971; Gneiting 2011).

References

Banerjee A, Guo X, Wang H (2005) On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory* **51(7)**:2664–2669. doi:10.1109/TIT.2005.850145.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Savage LJ (1971) Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* **66(337)**:783–810. doi:10.1080/01621459.1971.10482346.

Examples

```
# Compute the Bregman scoring function (type 4).

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$bregman4_penalty <- bregman4_sf(x = df$x, y = df$y)

print(df)
```

`capping_function` *Capping function*

Description

The function capping_function computes the value of the capping function, defined in Taggart (2022), p.205.

It is used by the generalized Huber loss function among others (see Taggart 2022).

Usage

```
capping_function(t, a, b)
```

Arguments

`t`	It can be a vector of length n.
`a`	It can be a vector of length n (must have the same length as t).
`b`	It can be a vector of length n (must have the same length as t).

Details

The capping function $\kappa_{a,b}(t)$ is defined by:

$$\kappa_{a,b}(t) := \max\{\min\{t, b\}, -a\}$$

or equivalently,

$$\kappa_{a,b}(t) := \begin{cases} -a, & t \leq -a \\ t, & -a < t \leq b \\ b, & t > b \end{cases}$$

Domain of function:

$$t \in \mathbb{R}$$

$$a \geq 0$$

$$b \geq 0$$

Value

Vector of values of the capping function.

Note

For the definition of the capping function, see Taggart (2022), p.205.

References

Taggart RJ (2022) Point forecasting and forecast evaluation with generalized Huber loss. *Electronic Journal of Statistics* **16**:201–231. doi:10.1214/21EJS1957.

Examples

```
# Compute the capping function.

df <- data.frame(
    t = c(1, -1, 1, -1, 1, -1, 1, -1, 1, 1, 2.5, 2.5, 3.5, 3.5),
    a = c(0, 0, 0, 0, Inf, Inf, Inf, Inf, 2, 3, 2, 3, 2, 3),
    b = c(0, 0, Inf, Inf, 0, 0, Inf, Inf, 3, 2, 3, 2, 3, 2)
)

df$cf <- capping_function(t = df$t, a = df$a, b = df$b)

print(df)
```

---

expectile_if *Expectile identification function*

Description

The function expectile_if computes the expectile identification function at a specific level p, when y materializes and x is the predictive expectile at level p.

The expectile identification function is defined in Table 9 in Gneiting (2011).

Usage

```
expectile_if(x, y, p)
```

Arguments

x	Predictive expectile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).

Details

The expectile identification function is defined by:

$$V(x, y, p) := 2|\mathbf{1}\{x \geq y\} - p|(x - y)$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

Range of function:

$$V(x, y, p) \in \mathbb{R}$$

Value

Vector of values of the expectile identification function.

Note

For the definition of expectiles, see Newey and Powell (1987).

The expectile identification function is a strict $\mathbb{F}$-identification function for the p-expectile functional (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

$\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ exists and is finite (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

References

Dimitriadis T, Fissler T, Ziegel JF (2024) Osband's principle for identification functions. *Statistical Papers* **65**:1125–1132. doi:10.1007/s0036202301428x.

Fissler T, Ziegel JF (2016) Higher order elicitability and Osband's principle. *The Annals of Statistics* **44(4)**:1680–1707. doi:10.1214/16AOS1439.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Newey WK, Powell JL (1987) Asymmetric least squares estimation and testing. *Econometrica* **55(4)**:819–847. doi:10.2307/1911031.

Examples

```
# Compute the expectile identification function.

df <- data.frame(
    y = rep(x = 0, times = 6),
    x = c(2, 2, -2, -2, 0, 0),
    p = rep(x = c(0.05, 0.95), times = 3)
)

df$expectile_if <- expectile_if(x = df$x, y = df$y, p = df$p)
```

`expectile_sf` *Asymmetric piecewise quadratic scoring function (expectile scoring function, expectile loss function)*

Description

The function expectile_sf computes the asymmetric piecewise quadratic scoring function (expectile scoring function) at a specific level p, when y materializes and x is the predictive expectile at level p.

The asymmetric piecewise quadratic scoring function is defined by eq. (27) in Gneiting (2011).

Usage

```
expectile_sf(x, y, p)
```

Arguments

`x`	Predictive expectile (prediction) at level p. It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
`p`	It can be a vector of length n (must have the same length as y).

Details

The asymmetric piecewise quadratic scoring function is defined by:

$$S(x, y, p) := |\mathbf{1}\{x \geq y\} - p|(x - y)^2$$

or equivalently,

$$S(x, y, p) := p(\max\{-(x - y), 0\})^2 + (1 - p)(\max\{x - y, 0\})^2$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

Range of function:

$$S(x, y, p) \geq 0, \forall x, y \in \mathbb{R}, p \in (0, 1)$$

Value

Vector of expectile losses.

Note

For the definition of expectiles, see Newey and Powell (1987).

The asymmetric piecewise quadratic scoring function is negatively oriented (i.e. the smaller, the better).

The asymmetric piecewise quadratic scoring function is strictly $\mathbb{F}$-consistent for the p-expectile functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^2]$ exists and is finite (Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Newey WK, Powell JL (1987) Asymmetric least squares estimation and testing. *Econometrica* **55(4)**:819–847. doi:10.2307/1911031.

Examples

```
# Compute the asymmetric piecewise quadratic scoring function (expectile scoring
# function).

df <- data.frame(
    y = rep(x = 0, times = 6),
    x = c(2, 2, -2, -2, 0, 0),
    p = rep(x = c(0.05, 0.95), times = 3)
)

df$expectile_penalty <- expectile_sf(x = df$x, y = df$y, p = df$p)

print(df)

# The asymmetric piecewise quadratic scoring function (expectile scoring
# function) at level p = 0.5 is half the squared error scoring function.
```

```
df <- data.frame(
    y = rep(x = 0, times = 3),
    x = c(-2, 0, 2),
    p = rep(x = c(0.5), times = 3)
)

df$expectile_penalty <- expectile_sf(x = df$x, y = df$y, p = df$p)

df$squared_error <- serr_sf(x = df$x, y = df$y)

print(df)
```

---

ghuber_sf *Generalized Huber scoring function*

Description

The function ghuber_sf computes the generalized Huber scoring function at a specific level p and parameters a and b, when y materializes and x is the predictive Huber functional at level p.

The generalized Huber scoring function is defined by eq. (4.7) in Taggart (2022) for $\phi(t) = t^2$.

Usage

```
ghuber_sf(x, y, p, a, b)
```

Arguments

x	Predictive Huber functional (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).
a	It can be a vector of length n (must have the same length as y).
b	It can be a vector of length n (must have the same length as y).

Details

The generalized Huber scoring function is defined by:

$$S(x, y, p, a, b) := |\mathbf{1}\{x \geq y\} - p|(y^2 - (\kappa_{a,b}(x - y) + y)^2 + 2x\kappa_{a,b}(x - y))$$

or equivalently

$$S(x, y, p, a, b) := |\mathbf{1}\{x \geq y\} - p| f_{a,b}(x - y)$$

or

$$S(x, y, p, a, b) := pf_{a,b}(-\max\{-(x-y), 0\}) + (1-p)f_{a,b}(\max\{x-y, 0\})$$

where

$$f_{a,b}(t) := \kappa_{a,b}(t)(2t - \kappa_{a,b}(t))$$

and $\kappa_{a,b}(t)$ is the capping function defined by:

$$\kappa_{a,b}(t) := \max\{\min\{t, b\}, -a\}$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

$$a > 0$$

$$b > 0$$

Range of function:

$$S(x, y, p, a, b) \geq 0, \forall x, y \in \mathbb{R}, p \in (0, 1), a, b > 0$$

Value

Vector of generalized Huber losses.

Note

For the definition of Huber functionals, see definition 3.3 in Taggart (2022). The value of eq. (4.7) is twice the value of the equation in definition 4.2 in Taggart (2002).

The generalized Huber scoring function is negatively oriented (i.e. the smaller, the better).

The generalized Huber scoring function is strictly $\mathbb{F}$-consistent for the p-Huber functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^2 - (Y-a)^2]$ and $\mathrm{E}_F[Y^2 - (Y+b)^2]$ (or equivalently $\mathrm{E}_F[Y]$) exist and are finite (Taggart 2022).

References

Taggart RJ (2022) Point forecasting and forecast evaluation with generalized Huber loss. *Electronic Journal of Statistics* **16**:201–231. doi:10.1214/21EJS1957.

Examples

```
# Compute the generalized Huber scoring function.

set.seed(12345)

n <- 10

df <- data.frame(
    x = runif(n, -2, 2),
    y = runif(n, -2, 2),
    p = runif(n, 0, 1),
    a = runif(n, 0, 1),
    b = runif(n, 0, 1)
)

df$ghuber_penalty <- ghuber_sf(x = df$x, y = df$y, p = df$p, a = df$a, b = df$b)

print(df)

# Equivalence of the generalized Huber scoring function and the asymmetric
# piecewise quadratic scoring function (expectile scoring function), when
# a = Inf and b = Inf.

set.seed(12345)

n <- 100

x <- runif(n, -20, 20)
y <- runif(n, -20, 20)
p <- runif(n, 0, 1)
a <- rep(x = Inf, times = n)
b <- rep(x = Inf, times = n)

u <- ghuber_sf(x = x, y = y, p = p, a = a, b = b)
v <- expectile_sf(x = x, y = y, p = p)

max(abs(u - v)) # values are slightly higher than 0 due to rounding error
min(abs(u - v))

# Equivalence of the generalized Huber scoring function and the Huber scoring
# function when p = 1/2 and a = b.

set.seed(12345)

n <- 100

x <- runif(n, -20, 20)
y <- runif(n, -20, 20)
p <- rep(x = 1/2, times = n)
a <- runif(n, 0, 20)

u <- ghuber_sf(x = x, y = y, p = p, a = a, b = a)
```

```
v <- huber_sf(x = x, y = y, a = a)

max(abs(u - v)) # values are slightly higher than 0 due to rounding error
min(abs(u - v))
```

---

gpl1_sf *Generalized piecewise linear power scoring function (type 1)*

Description

The function gpl1_sf computes the generalized piecewise linear power scoring function at a specific level p for $g(x) = x^b/|b|$, $b > 0$, when y materializes and x is the predictive quantile at level p.

The generalized piecewise linear power scoring function is defined by eq. (25) in Gneiting (2011) and the form implemented here for the specific $g(x)$ is defined by eq. (26) in Gneiting (2011).

Usage

```
gpl1_sf(x, y, p, b)
```

Arguments

x	Predictive quantile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).
b	It can be a vector of length n (must have the same length as y).

Details

The generalized piecewise linear power scoring function (type 1) is defined by:

$$S(x, y, p, b) := (1/|b|)(\mathbf{1}\{x \geq y\} - p)(x^b - y^b)$$

or equivalently

$$S(x, y, p, b) := (1/|b|)(p|\max\{-(x^b - y^b), 0\}| + (1 - p)|\max\{x^b - y^b, 0\}|)$$

Domain of function:

$$x > 0$$

$$y > 0$$

$$0 < p < 1$$

$$b > 0$$

Range of function:

$$S(x, y, p, b) \geq 0, \forall x, y > 0, p \in (0, 1), b > 0$$

Value

Vector of generalized piecewise linear power losses.

Note

The implemented function is denoted as type 1 since it corresponds to a specific type of $g(x)$ of the general form of the generalized piecewise linear power scoring function defined by eq. (25) in Gneiting (2011).

For the definition of quantiles, see Koenker and Bassett Jr (1978).

The generalized piecewise linear scoring function is negatively oriented (i.e. the smaller, the better).

The herein implemented generalized piecewise linear power scoring function is strictly $\mathbb{F}$-consistent for the p-quantile functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^b]$ exists and is finite (Thomson 1979; Saerens 2000; Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Koenker R, Bassett Jr G (1978) Regression quantiles. *Econometrica* **46(1)**:33–50. doi:10.2307/1913643.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the generalized piecewise linear scoring function (type 1).

df <- data.frame(
    y = rep(x = 2, times = 6),
    x = c(1, 2, 3, 1, 2, 3),
    p = c(rep(x = 0.05, times = 3), rep(x = 0.95, times = 3)),
    b = rep(x = 2, times = 6)
)

df$gpl1_penalty <- gpl1_sf(x = df$x, y = df$y, p = df$p, b = df$b)

print(df)
```

```
# Equivalence of generalized piecewise linear scoring function (type 1) and
# asymmetric piecewise linear scoring function (quantile scoring function), when
# b = 1.

set.seed(12345)

n <- 100

x <- runif(n, 0, 20)
y <- runif(n, 0, 20)
p <- runif(n, 0, 1)
b <- rep(x = 1, times = n)

u <- gpl1_sf(x = x, y = y, p = p, b = b)
v <- quantile_sf(x = x, y = y, p = p)

max(abs(u - v))

# Equivalence of generalized piecewise linear scoring function (type 1) and
# MAE-SD scoring function, when p = 1/2 and b = 1/2.

set.seed(12345)

n <- 100

x <- runif(n, 0, 20)
y <- runif(n, 0, 20)
p <- rep(x = 0.5, times = n)
b <- rep(x = 1/2, times = n)

u <- gpl1_sf(x = x, y = y, p = p, b = b)
v <- maesd_sf(x = x, y = y)

max(abs(u - v))
```

---

gpl2_sf *Generalized piecewise linear power scoring function (type 2)*

Description

The function gpl2_sf computes the generalized piecewise linea power scoring function at a specific level p for $g(x) = \log(x)$, when y materializes and x is the predictive quantile at level p.

The generalized piecewise linear power scoring function is negatively oriented (i.e. the smaller, the better).

The generalized piecewise linear scoring function is defined by eq. (25) in Gneiting (2011) and the form implemented here for the specific $g(x)$ is defined by eq. (26) in Gneiting (2011) for $b = 0$.

Usage

```
gpl2_sf(x, y, p)
```

Arguments

x	Predictive quantile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).

Details

The generalized piecewise linear power scoring function (type 2) is defined by:

$$S(x, y, p) := (\mathbf{1}\{x \geq y\} - p) \log(x/y)$$

or equivalently

$$S(x, y, p) := p|\max\{-(\log(x) - \log(y)), 0\}| + (1 - p)|\max\{\log(x) - \log(y), 0\}|$$

Domain of function:

$$x > 0$$

$$y > 0$$

$$0 < p < 1$$

Range of function:

$$S(x, y, p) \geq 0, \forall x, y > 0, p \in (0, 1)$$

Value

Vector of generalized piecewise linear losses.

Note

The implemented function is denoted as type 2 since it corresponds to a specific type of $g(x)$ of the general form of the generalized piecewise linear power scoring function defined by eq. (25) in Gneiting (2011).

For the definition of quantiles, see Koenker and Bassett Jr (1978).

The herein implemented generalized piecewise linear power scoring function is strictly $\mathbb{F}$-consistent for the p-quantile functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[\log(Y)]$ exists and is finite (Thomson 1979; Saerens 2000; Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Koenker R, Bassett Jr G (1978) Regression quantiles. *Econometrica* **46(1)**:33–50. doi:10.2307/1913643.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the generalized piecewise linear scoring function (type 2).

df <- data.frame(
    y = rep(x = 2, times = 6),
    x = c(1, 2, 3, 1, 2, 3),
    p = c(rep(x = 0.05, times = 3), rep(x = 0.95, times = 3))
)

df$gpl2_penalty <- gpl2_sf(x = df$x, y = df$y, p = df$p)

print(df)

# The generalized piecewise linear scoring function (type 2) is half the MAE-LOG
# scoring function.

df <- data.frame(
    y = rep(x = 5.5, times = 10),
    x = 1:10,
    p = rep(x = 0.5, times = 10)
)

df$gpl2_penalty <- gpl2_sf(x = df$x, y = df$y, p = df$p)

df$mae_log_penalty <- maelog_sf(x = df$x, y = df$y)

df$ratio <- df$gpl2_penalty/df$mae_log_penalty

print(df)
```

---

huber_sf *Huber scoring function*

Description

The function huber_sf computes the Huber scoring function with parameter a, when y materializes and x is the predictive Huber mean.

The Huber scoring function is defined in Huber (1964).

Usage

```
huber_sf(x, y, a)
```

Arguments

x	Predictive Huber mean (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
a	It can be a vector of length n (must have the same length as y).

Details

The Huber scoring function is defined by:

$$S(x, y, a) := \begin{cases} \frac{1}{2}(x-y)^2, & |x-y| \leq a \\ a|x-y| - \frac{1}{2}a^2, & |x-y| > a \end{cases}$$

or equivalently

$$S(x, y, a) := (1/2)\kappa_{a,a}(x-y)(2(x-y) - \kappa_{a,a}(x-y))$$

where $\kappa_{a,b}(t)$ is the capping function defined by:

$$\kappa_{a,b}(t) := \max\{\min\{t, b\}, -a\}$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$a > 0$$

Range of function:

$$S(x, y, a) \geq 0, \forall x, y \in \mathbb{R}, a > 0$$

Value

Vector of Huber losses.

Note

For the definition of Huber mean, see Taggart (2022).

The Huber scoring function is negatively oriented (i.e. the smaller, the better).

The Huber scoring function is strictly $\mathbb{F}$-consistent for the Huber mean. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^2 - (Y-a)^2]$ and $\mathrm{E}_F[Y^2 - (Y+a)^2]$ (or equivalently $\mathrm{E}_F[Y]$) exist and are finite (Taggart 2022).

References

Huber PJ (1964) Robust estimation of a location parameter. *Annals of Mathematical Statistics* **35(1)**:73–101. doi:10.1214/aoms/1177703732.

Taggart RJ (2022) Point forecasting and forecast evaluation with generalized Huber loss. *Electronic Journal of Statistics* **16**:201–231. doi:10.1214/21EJS1957.

Examples

```
# Compute the Huber scoring function.

df <- data.frame(
    x = c(-3, -2, -1, 0, 1, 2, 3),
    y = c(0, 0, 0, 0, 0, 0, 0),
    a = c(2.7, 2.5, 0.6, 0.7, 0.9, 1.2, 5)
)

df$huber_penalty <- huber_sf(x = df$x, y = df$y, a = df$a)

print(df)
```

---

interval_sf *Interval scoring function (Winkler scoring function)*

Description

The function interval_sf computes the interval scoring function (Winkler scoring function) when y materializes and $[x_1, x_2]$ is the central $1-p$ prediction interval.

The interval scoring function is defined by eq. (43) in Gneiting and Raftery (2007).

Usage

```
interval_sf(x1, x2, y, p)
```

Arguments

x1	Predictive quantile (prediction) at level $p/2$. It can be a vector of length n (must have the same length as y).
x2	Predictive quantile (prediction) at level $1 - p/2$. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x_1).
p	It can be a vector of length n (must have the same length as y).

Details

The interval scoring function is defined by:

$$S(x_1, x_2, y, p) := (x_2 - x_1) + (2/p)(x_1 - y)\mathbf{1}\{y < x_1\} + (2/p)(y - x_2)\mathbf{1}\{y > x_2\}$$

Domain of function:

$$x_1 \in \mathbb{R}$$

$$x_2 \in \mathbb{R}$$

$$x_1 < x_2$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

Range of function:

$$S(x_1, x_2, y, p) \geq 0, \forall x_1, x_2, y \in \mathbb{R}, x_1 < x_2, p \in (0, 1)$$

Value

Vector of interval losses.

Note

For the definition of quantiles, see Koenker and Bassett Jr (1978).

The interval scoring function is negatively oriented (i.e. the smaller, the better).

The interval scoring function is strictly $\mathbb{F}$-consistent for the central $1-p$ prediction interval $[x_1, x_2]$. x_1 and x_2 are quantile functionals at levels $p/2$ and $1 - p/2$ respectively.

$\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ exists and is finite (Dunsmore 1968; Winkler 1972; Gneiting and Raftery 2007; Winkler and Murphy 1979; Fissler and Ziegel 2016; Brehmer and Gneiting 2021).

References

Brehmer JR, Gneiting T (2021) Scoring interval forecasts: Equal-tailed, shortest, and modal interval. *Bernoulli* **27(3)**:1993–2010. doi:10.3150/20BEJ1298.

Dunsmore IR (1968) A Bayesian approach to calibration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **30(2)**:396–405. doi:10.1111/j.25176161.1968.tb00740.x.

Fissler T, Ziegel JF (2016) Higher order elicitability and Osband's principle. *The Annals of Statistics* **44(4)**:1680–1707. doi:10.1214/16AOS1439.

Gneiting T, Raftery AE (2007) Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association* **102(477)**:359–378. doi:10.1198/016214506000001437.

Koenker R, Bassett Jr G (1978) Regression quantiles. *Econometrica* **46(1)**:33–50. doi:10.2307/1913643.

Winkler RL (1972) A decision-theoretic approach to interval estimation. *Journal of the American Statistical Association* **67(337)**:187–191. doi:10.1080/01621459.1972.10481224.

Winkler RL, Murphy AH (1979) The use of probabilities in forecasts of maximum and minimum temperatures.*Meteorological Magazine* **108(1288)**:317–329.

Examples

```
# Compute the interval scoring function (Winkler scoring function).

df <- data.frame(
    y = rep(x = 0, times = 6),
    x1 = c(-3, -2, -1, 0, 1, 2),
    x2 = c(1, 2, 3, 4, 5, 6),
    p = rep(x = c(0.05, 0.95), times = 3)
)

df$interval_penalty <- interval_sf(x1 = df$x1, x2 = df$x2, y = df$y, p = df$p)

print(df)
```

`maelog_sf` *MAE-LOG scoring function*

Description

The function maelog_sf computes the MAE-LOG scoring function when y materializes and x is the predictive median functional.

The MAE-LOG scoring function is defined by eq. (11) in Patton (2011).

Usage

```
maelog_sf(x, y)
```

Arguments

x	Predictive median functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The MAE-LOG scoring function is defined by:

$$S(x, y) := |\log(x/y)|$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of MAE-LOG losses.

Note

For details on the MAE-LOG scoring function, see Gneiting (2011) and Patton (2011).

The median functional is the median of the probability distribution F of y (Gneiting 2011).

The MAE-LOG scoring function is negatively oriented (i.e. the smaller, the better).

The MAE-LOG scoring function is strictly $\mathbb{F}$-consistent for the median functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[\log(Y)]$ exists and is finite (Thomson 1979; Saerens 2000; Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the MAE-LOG scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$mae_log_penalty <- maelog_sf(x = df$x, y = df$y)

print(df)
```

`maesd_sf` *MAE-SD scoring function*

Description

The function maesd_sf computes the MAE-SD scoring function when y materializes and x is the predictive median functional.

The MAE-SD scoring function is defined by eq. (12) in Patton (2011).

Usage

```
maesd_sf(x, y)
```

Arguments

x	Predictive median functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The MAE-SD scoring function is defined by:

$$S(x, y) := |x^{1/2} - y^{1/2}|$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of MAE-SD losses.

Note

For details on the MAE-SD scoring function, see Gneiting (2011) and Patton (2011).

The median functional is the median of the probability distribution F of y (Gneiting 2011).

The MAE-SD scoring function is negatively oriented (i.e. the smaller, the better).

The MAE-SD scoring function is strictly $\mathbb{F}$-consistent for the median functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^{1/2}]$ exists and is finite (Thomson 1979; Saerens 2000; Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the MAE-SD scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$mae_sd_penalty <- maesd_sf(x = df$x, y = df$y)

print(df)
```

`mean_if` *Mean identification function*

Description

The function mean_if computes the mean identification function , when y materializes and x is the predictive mean.

The mean identification function is defined in Table 9 in Gneiting (2011).

Usage

```
mean_if(x, y)
```

Arguments

x	Predictive expectile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The mean identification function is defined by:

$$V(x, y) := x - y$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

Range of function:

$$V(x, y) \in \mathbb{R}$$

Value

Vector of values of the mean identification function.

Note

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The mean identification function is a strict $\mathbb{F}$-identification function for the mean functional. (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

$\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ exists and is finite (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

References

Dimitriadis T, Fissler T, Ziegel JF (2024) Osband's principle for identification functions. *Statistical Papers* **65**:1125–1132. doi:10.1007/s0036202301428x.

Fissler T, Ziegel JF (2016) Higher order elicitability and Osband's principle. *The Annals of Statistics* **44(4)**:1680–1707. doi:10.1214/16AOS1439.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Newey WK, Powell JL (1987) Asymmetric least squares estimation and testing. *Econometrica* **55(4)**:819–847. doi:10.2307/1911031.

Examples

```
# Compute the mean identification function.

df <- data.frame(
    y = rep(x = 0, times = 3),
    x = c(-2, 0, 2)
)

df$mean_if <- mean_if(x = df$x, y = df$y)
```

`mv_if` *Mean - variance identification function*

Description

The function mv_if computes the mean - variance identification function, when y materializes, x_1 is the predictive mean and x_2 is the predictive variance.

The mean - variance identification function is defined in proposition (3.11) in Fissler and Ziegel (2019).

Usage

```
mv_if(x1, x2, y)
```

Arguments

`x1`	Predictive mean (prediction). It can be a vector of length n (must have the same length as y).
`x2`	Predictive variance (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The mean - variance identification function is defined by:

$$V(x_1, x_2, y) := (x_1 - y, x_2 + x_1^2 - y^2)$$

Domain of function:

$$x_1 \in \mathbb{R}$$

$$x_2 > 0$$

$$y \in \mathbb{R}$$

Value

Matrix of mean - variance values of the identification function.

Note

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The variance functional is the variance $\mathrm{Var}_F[Y] := \mathrm{E}_F[Y^2] - (\mathrm{E}_F[Y])^2$ of the probability distribution F of y (Gneiting 2011)

The mean - variance identification function is a strict $\mathbb{F}$-identification function for the pair (mean, variance) functional (Gneiting 2011; Fissler and Ziegel 2019; Dimitriadis et al. 2024).

$\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ and $\mathrm{E}_F[Y^2]$ exist and are finite (Gneiting 2011; Fissler and Ziegel 2019; Dimitriadis et al. 2024).

References

Dimitriadis T, Fissler T, Ziegel JF (2024) Osband's principle for identification functions. *Statistical Papers* **65**:1125–1132. doi:10.1007/s0036202301428x.

Fissler T, Ziegel JF (2019) Order-sensitivity and equivariance of scoring functions. *Electronic Journal of Statistics* **13(1)**:1166–1211. doi:10.1214/19EJS1552.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the mean - variance identification function.

df <- data.frame(
    y = rep(x = 0, times = 6),
    x1 = c(2, 2, -2, -2, 0, 0),
    x2 = c(1, 2, 1, 2, 1, 2)
)

v <- as.data.frame(mv_if(x1 = df$x1, x2 = df$x2, y = df$y))

print(cbind(df, v))
```

`mv_sf` *Mean - variance scoring function*

Description

The function mv_sf computes the mean - variance scoring function, when y materializes, x_1 is the predictive mean and x_2 is the predictive variance.

The mean - variance scoring function is defined by eq. (3.11) in Fissler and Ziegel (2019).

Usage

```
mv_sf(x1, x2, y)
```

Arguments

x1	Predictive mean (prediction). It can be a vector of length n (must have the same length as y).
x2	Predictive variance (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The mean - variance scoring function is defined by:

$$S(x_1, x_2, y) := x_2^{-2}(x_1^2 - 2x_2 - 2x_1 y + y^2)$$

Domain of function:

$$x_1 \in \mathbb{R}$$

$$x_2 > 0$$

$$y \in \mathbb{R}$$

Value

Vector of mean - variance losses.

Note

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The variance functional is the variance $\mathrm{Var}_F[Y] := \mathrm{E}_F[Y^2] - (\mathrm{E}_F[Y])^2$ of the probability distribution F of y (Gneiting 2011)

The mean - variance scoring function is negatively oriented (i.e. the smaller, the better).

The mean - variance scoring function is strictly consistent for the pair (mean, variance) functional (Osband 1985, p.9; Gneiting 2011; Fissler and Ziegel 2019).

References

Fissler T, Ziegel JF (2019) Order-sensitivity and equivariance of scoring functions. *Electronic Journal of Statistics* **13(1)**:1166–1211. doi:10.1214/19EJS1552.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Osband KH (1985) Providing Incentives for Better Cost Forecasting. PhD thesis, University of California, Berkeley. doi:10.5281/zenodo.4355667.

Examples

```
# Compute the mean - variance scoring function.

df <- data.frame(
    y = rep(x = 0, times = 6),
    x1 = c(2, 2, -2, -2, 0, 0),
    x2 = c(1, 2, 1, 2, 1, 2)
)

df$mv_penalty <- mv_sf(x1 = df$x1, x2 = df$x2, y = df$y)

print(df)
```

---

nmoment_if *n-th moment identification function*

Description

The function nmoment_if computes the n-th moment identification function, when y materializes and x is the predictive n-th moment.

The expectile identification function is defined in Table 9 in Gneiting (2011) by setting $r(t) = t^n$ and $s(t) = 1$.

Usage

```
nmoment_if(x, y, n)
```

Arguments

x	Predictive n-th moment. It can be a vector of length m (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length m (must have the same length as x).
n	n) is the moment order. It can be a vector of length m (must have the same length as x).

Details

The n-th moment identification function is defined by:

$$V(x, y, n) := x - y^n$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$n \in \mathbb{N}$$

Value

Vector of values of the n-th moment identification function.

Note

The n-th moment functional is the expectation $\mathrm{E}_F[Y^n]$ of the probability distribution F of y.

The n-th moment identification function is a strict $\mathbb{F}$-identification function for the n-th moment functional (Gneiting 2011; Fissler and Ziegel 2016).

$\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^n]$ exists and is finite (Gneiting 2011; Fissler and Ziegel 2016).

References

Fissler T, Ziegel JF (2016) Higher order elicitability and Osband's principle. *The Annals of Statistics* **44(4)**:1680–1707. doi:10.1214/16AOS1439.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the n-th moment scoring function.

df <- data.frame(
    y = rep(x = 2, times = 6),
    x = c(1, 2, 3, 1, 2, 3),
    n = c(2, 2, 2, 3, 3, 3)
)

df$nmoment_if <- nmoment_if(x = df$x, y = df$y, n = df$n)

print(df)
```

`nmoment_sf` *n-th moment scoring function*

Description

The function nmoment_sf computes the n-th moment scoring function, when y materializes, and $\mathrm{E}_F[Y^n]$ is the predictive n-th moment.

The n-th moment scoring function is defined by eq. (22) in Gneiting (2011) by setting $r(t) = t^n$, $s(t) = 1$, $\phi(t) = t^2$ and removing all terms that are not functions of x.

Usage

```
nmoment_sf(x, y, n)
```

Arguments

x	Predictive n-th moment. It can be a vector of length m (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length m (must have the same length as x).
n	n) is the moment order. It can be a vector of length m (must have the same length as x).

Details

The n-th moment scoring function is defined by:

$$S(x, y, n) := -x^2 - 2x(y^n - x)$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$n \in \mathbb{N}$$

Value

Vector of n-th moment losses.

Note

The n-th moment functional is the expectation $\mathrm{E}_F[Y^n]$ of the probability distribution F of y.

The n-th moment scoring function is negatively oriented (i.e. the smaller, the better).

The n-th moment scoring function is strictly $\mathbb{F}$-consistent for the n-th moment functional $\mathrm{E}_F[Y^n]$ (Theorem 8 in Gneiting 2011). $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y^]$, $\mathrm{E}_F[Y^2]$, $\mathrm{E}_F[Y^n]$ and $\mathrm{E}_F[Y^{n+1}]$ exist and are finite (Theorem 8 in Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the n-th moment scoring function.

df <- data.frame(
    y = rep(x = 2, times = 6),
    x = c(1, 2, 3, 1, 2, 3),
    n = c(2, 2, 2, 3, 3, 3)
)

df$nmoment_penalty <- nmoment_sf(x = df$x, y = df$y, n = df$n)

print(df)
```

`obsweighted_sf` *Observation-weighted scoring function*

Description

The function obsweighted_sf computes the observation-weighted scoring function when y materializes and x is the predictive $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional.

The observation-weighted scoring function is defined in p. 752 in Gneiting (2011).

Usage

```
obsweighted_sf(x, y)
```

Arguments

`x`	Predictive $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The observation-weighted scoring function is defined by:

$$S(x, y) := y(x - y)^2$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of observation-weighted errors.

Note

For details on the observation-weighted scoring function, see Gneiting (2011).

The observation-weighted scoring function is negatively oriented (i.e. the smaller, the better).

The observation-weighted scoring function is strictly consistent for the $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional.

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the observation-weighted scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$squared_relative_error <- obsweighted_sf(x = df$x, y = df$y)

print(df)
```

---

quantile_if *Quantile identification function*

Description

The function quantile_if computes the quantile identification function at a specific level p, when y materializes and x is the predictive quantile at level p.

The quantile identification function is defined in Table 9 in Gneiting (2011).

Usage

```
quantile_if(x, y, p)
```

Arguments

x	Predictive quantile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).

Details

The quantile identification function is defined by:

$$V(x, y, p) := \mathbf{1}\{x \geq y\} - p$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

Range of function:

$$V(x, y, p) \in (-1, 1)$$

Value

Vector of values of the quantile identification function.

Note

For the definition of quantiles, see Koenker and Bassett Jr (1978).

The quantile identification function is a strict $\mathbb{F}_p$-identification function for the p-quantile functional (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

$\mathbb{F}_p$ is the family of probability distributions F for which there exists an y with $F(y) = p$ (Gneiting 2011; Fissler and Ziegel 2016; Dimitriadis et al. 2024).

References

Dimitriadis T, Fissler T, Ziegel JF (2024) Osband's principle for identification functions. *Statistical Papers* **65**:1125–1132. doi:10.1007/s0036202301428x.

Fissler T, Ziegel JF (2016) Higher order elicitability and Osband's principle. *The Annals of Statistics* **44(4)**:1680–1707. doi:10.1214/16AOS1439.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Koenker R, Bassett Jr G (1978) Regression quantiles. *Econometrica* **46(1)**:33–50. doi:10.2307/1913643.

Examples

```
# Compute the quantile identification function.

df <- data.frame(
    y = rep(x = 0, times = 6),
    x = c(2, 2, -2, -2, 0, 0),
    p = rep(x = c(0.05, 0.95), times = 3)
)

df$quantile_if <- quantile_if(x = df$x, y = df$y, p = df$p)
```

`quantile_sf` *Asymmetric piecewise linear scoring function (quantile scoring function, quantile loss function)*

Description

The function quantile_sf computes the asymmetric piecewise linear scoring function (quantile scoring function) at a specific level p, when y materializes and x is the predictive quantile at level p.

The asymmetric piecewise linear scoring function is defined by eq. (24) in Gneiting (2011).

Usage

```
quantile_sf(x, y, p)
```

Arguments

x	Predictive quantile (prediction) at level p. It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).
p	It can be a vector of length n (must have the same length as y).

Details

The assymetric piecewise linear scoring function is defined by:

$$S(x, y, p) := (\mathbf{1}\{x \geq y\} - p)(x - y)$$

or equivalently,

$$S(x, y, p) := p|\max\{-(x - y), 0\}| + (1 - p)|\max\{x - y, 0\}|$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

$$0 < p < 1$$

Range of function:

$$S(x, y, p) \geq 0, \forall x, y \in \mathbb{R}, p \in (0, 1)$$

Value

Vector of quantile losses.

Note

For the definition of quantiles, see Koenker and Bassett Jr (1978).

The asymmetric piecewise linear scoring function is negatively oriented (i.e. the smaller, the better).

The asymmetric piecewise linear scoring function is strictly $\mathbb{F}$-consistent for the p-quantile functional. $\mathbb{F}$ is the family of probability distributions F for which $\mathrm{E}_F[Y]$ exists and is finite (Schlaifer 1961, p.196; Ferguson 1967, p.51; Thomson 1979; Saerens 2000; Gneiting 2011).

References

Ferguson TS (1967) Mathematical Statistics: A Decision-Theoretic Approach. Academic Press, New York.

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Koenker R, Bassett Jr G (1978) Regression quantiles. *Econometrica* **46(1)**:33–50. doi:10.2307/1913643.

Raiffa H,Schlaifer R (1961) Applied Statistical Decision Theory. Colonial Press, Clinton.

Saerens M (2000) Building cost functions minimizing to some summary statistics. *IEEE Transactions on Neural Networks* **11(6)**:1263–1271. doi:10.1109/72.883416.

Thomson W (1979) Eliciting production possibilities from a well-informed manager. *Journal of Economic Theory* **20(3)**:360–380. doi:10.1016/00220531(79)900425.

Examples

```
# Compute the asymmetric piecewise linear scoring function (quantile scoring
# function).

df <- data.frame(
    y = rep(x = 0, times = 6),
    x = c(2, 2, -2, -2, 0, 0),
    p = rep(x = c(0.05, 0.95), times = 3)
)

df$quantile_penalty <- quantile_sf(x = df$x, y = df$y, p = df$p)
```

```
print(df)

# The absolute error scoring function is twice the asymmetric piecewise linear
# scoring function (quantile scoring function) at level p = 0.5.

df <- data.frame(
    y = rep(x = 0, times = 3),
    x = c(-2, 0, 2),
    p = rep(x = c(0.5), times = 3)
)

df$quantile_penalty <- quantile_sf(x = df$x, y = df$y, p = df$p)

df$absolute_error <- aerr_sf(x = df$x, y = df$y)

print(df)
```

`relerr_sf` *Relative error scoring function (MAE-PROP scoring function)*

Description

The function relerr_sf computes the relative error scoring function when y materializes and x is the predictive $\text{med}^{(1)}(F)$ functional.

The relative error scoring function is defined in Table 1 in Gneiting (2011).

The relative error scoring function is referred to as MAE-PROP scoring function in eq. (13) in Patton (2011).

Usage

```
relerr_sf(x, y)
```

Arguments

`x`	Predictive $\text{med}^{(1)}(F)$ functional (prediction). It can be a vector of length n (must have the same length as y).
`y`	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The relative error scoring function is defined by:

$$S(x, y) := |(x - y)/x|$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of relative errors.

Note

For details on the relative error scoring function, see Gneiting (2011).

The β-median functional, $\text{med}^{(\beta)}(F)$ is the median of a probability distribution whose density is proportional to $y^{\beta} f(y)$, where f is the density of the probability distribution F of y (Gneiting 2011).

The relative error scoring function is negatively oriented (i.e. the smaller, the better).

The relative error scoring function is strictly $\mathbb{F}^{(w)}$-consistent for the $\text{med}^{(1)}(F)$ functional. $\mathbb{F}$ is the family of probability distributions for which $\text{E}_F[Y]$ exists and is finite. $\mathbb{F}^{(w)}$ is the subclass of probability distributions in $\mathbb{F}$, which are such that $w(y)f(y)$, $w(y) = y$ has finite integral over $(0, \infty)$, and the probability distribution $F^{(w)}$ with density proportional to $w(y)f(y)$ belongs to $\mathbb{F}$ (see Theorems 5 and 9 in Gneiting 2011)

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Patton AJ (2011) Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics* **160(1)**:246–256. doi:10.1016/j.jeconom.2010.03.034.

Examples

```
# Compute the relative error scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$relative_error <- relerr_sf(x = df$x, y = df$y)

print(df)
```

serr_sf *Squared error scoring function*

Description

The function serr_sf computes the squared error scoring function when y materializes and x is the predictive mean functional.

The squared error scoring function is defined in Table 1 in Gneiting (2011).

Usage

```
serr_sf(x, y)
```

Arguments

x	Predictive mean functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The squared error scoring function is defined by:

$$S(x, y) := (x - y)^2$$

Domain of function:

$$x \in \mathbb{R}$$

$$y \in \mathbb{R}$$

Range of function:

$$S(x, y) \geq 0, \forall x, y \in \mathbb{R}$$

Value

Vector of squared errors.

Note

For details on the squared error scoring function, see Savage (1971), Gneiting (2011).

The mean functional is the mean $\mathrm{E}_F[Y]$ of the probability distribution F of y (Gneiting 2011).

The squared error scoring function is negatively oriented (i.e. the smaller, the better).

The squared error scoring function is strictly $\mathbb{F}$-consistent for the mean functional. $\mathbb{F}$ is the family of probability distributions F for which the second moment exists and is finite (Savage 1971; Gneiting 2011).

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Savage LJ (1971) Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* **66(337)**:783–810. doi:10.1080/01621459.1971.10482346.

Examples

```
# Compute the squarer error scoring function.

df <- data.frame(
    y = rep(x = 0, times = 5),
    x = -2:2
)

df$squared_error <- serr_sf(x = df$x, y = df$y)

print(df)
```

---

sperr_sf *Squared percentage error scoring function*

Description

The function sperr_sf computes the squared percentage error scoring function when y materializes and x is the predictive $\frac{\mathrm{E}_F[Y^{-1}]}{\mathrm{E}_F[Y^{-2}]}$ functional.

The squared percentage error scoring function is defined in p. 752 in Gneiting (2011).

Usage

```
sperr_sf(x, y)
```

Arguments

x	Predictive $\frac{\mathrm{E}_F[Y^{-1}]}{\mathrm{E}_F[Y^{-2}]}$ functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The squared percentage error scoring function is defined by:

$$S(x, y) := ((x - y)/y)^2$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of squared percentage errors.

Note

For details on the squared percentage error scoring function, see Park and Stefanski (1998) and Gneiting (2011).

The squared percentage error scoring function is negatively oriented (i.e. the smaller, the better).

The squared percentage error scoring function is strictly consistent for the $\frac{\mathrm{E}_F[Y^{-1}]}{\mathrm{E}_F[Y^{-2}]}$ functional.

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Park H, Stefanski LA (1998) Relative-error prediction. *Statistics and Probability Letters* **40(3)**:227–236. doi:10.1016/S01677152(98)000881.

Examples

```
# Compute the squared percentage error scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$squared_percentage_error <- sperr_sf(x = df$x, y = df$y)

print(df)
```

`srelerr_sf` *Squared relative error scoring function*

Description

The function srelerr_sf computes the squared relative error scoring function when y materializes and x is the predictive $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional.

The squared relative error scoring function is defined in p. 752 in Gneiting (2011).

Usage

```
srelerr_sf(x, y)
```

Arguments

x	Predictive $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional (prediction). It can be a vector of length n (must have the same length as y).
y	Realization (true value) of process. It can be a vector of length n (must have the same length as x).

Details

The squared relative error scoring function is defined by:

$$S(x, y) := ((x - y)/x)^2$$

Domain of function:

$$x > 0$$

$$y > 0$$

Range of function:

$$S(x, y) \geq 0, \forall x, y > 0$$

Value

Vector of squared relative errors.

Note

For details on the squared relative error scoring function, see Gneiting (2011).

The squared relative error scoring function is negatively oriented (i.e. the smaller, the better).

The squared relative error scoring function is strictly consistent for the $\frac{\mathrm{E}_F[Y^2]}{\mathrm{E}_F[Y]}$ functional.

References

Gneiting T (2011) Making and evaluating point forecasts. *Journal of the American Statistical Association* **106(494)**:746–762. doi:10.1198/jasa.2011.r10138.

Examples

```
# Compute the squared percentage error scoring function.

df <- data.frame(
    y = rep(x = 2, times = 3),
    x = 1:3
)

df$squared_relative_error <- srelerr_sf(x = df$x, y = df$y)

print(df)
```

Index