Package 'CalSim'

September 10, 2020

Type Package

Title The Calibration Simplex

Version 0.5.1

Author Johannes Resin

Maintainer Johannes Resin <johannes.resin@h-its.org>

Depends R (>= 3.5)

Imports spatstat, stats, ExactMultinom

Description Generates the calibration simplex (a generalization of the reliability diagram) for three-category probability forecasts, as proposed by Wilks (2013) <doi:10.1175/WAF-D-13-00027.1>.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-09-10 16:00:02 UTC

R topics documented:

`calibration_simplex` *Calibration Simplex*

Description

Generates an object of class `calibration_simplex` which can be used to assess the calibration of ternary probability forecasts. The Calibration Simplex can be seen as generalization of the reliability diagram for binary probability forecasts. For details on the interpretation of the calibration simplex, see Wilks (2013). Be aware that some minor changes have been made compared to the calibration simplex as suggested by Wilks (2013) (see note below).

As a somewhat experimental feature, multinomial p-values can be used for uncertainty quantification, that is, as a tool to judge whether the observed discrepancies may be merely coincidental or whether the predictions may in fact be miscalibrated, see Resin (2020, Section 4.2).

Usage

```
calibration_simplex(n, p1, p2, p3, obs, test_stat, percentagewise)

## Default S3 method:
calibration_simplex(
  n = 10,
  p1 = NULL,
  p2 = NULL,
  p3 = NULL,
  obs = NULL,
  test_stat = "LLR",
  percentagewise = FALSE
)
```

Arguments

`n`	A natural number.
`p1`	A vector containing the forecasted probabilities for the first (1) category, e.g. below-normal.
`p2`	A vector containing the forecasted probabilities for the second (2) category, e.g. near-normal.
`p3`	A vector containing the forecasted probabilities for the third (3) category, e.g. above-normal.
`obs`	A vector containing the observed outcomes (Categories are encoded as 1 (e.g. below-normal), 2 (e.g. near-normal) and 3 (e.g. above-normal)).
`test_stat`	A string indicating which test statistic is to be used for the multinomial test in each bin. Options are "LLR" (log-likelihood ratio; default), "Chisq" (Pearson's chi-square) and "Prob" (probability mass statistic). See details
`percentagewise`	Logical, specifying whether probabilities are percentagewise (summing to 100) or not (summing to 1).

Details

Only two of the three forecast probability vectors (`p1`, `p2` and `p3`) need to be specified.

The p-values are based on multinomial tests comparing the observed frequencies within a bin with the average forecast probabilities within the bin as outlined in Resin (2020, Section 4.2). The p-values are exact and do not rely on asymptotics, however, it is assumed that the true distribution (under the hypothesis of forecast calibration) within each bin is approximated well by the multinomial distribution. If `n` is small the approximation may be poor, resulting in unreliable p-values. p-Values less than 0.0001 are not exact but merely indicate a value less than 0.0001.

Value

A list with class "calibration_simplex" containing

`n`	As input by user or default.
`n_bins`	Computed from `n`. Number of hexagons.
`n_obs`	Total number of observations.
`freq`	Vector of length `n_bins` containing the number of observations within each bin.
`cond_rel_freq`	Matrix containing the observed outcome frequencies within each bin.
`cond_ave_prob`	Matrix containing the average forecast probabilities within each bin.
`pvals`	Exact multinomial p-values within each bin. See details.

Object of class `calibration_simplex`.

Note

In contrast to the calibration simplex proposed by Daniel S. Wilks, 2013, the simplex has been mirrored at the diagonal through the left bottom hexagon. The miscalibration error is by default calculated precisely (in each bin as the difference of the relative frequencies of each class and the average forecast probabilities) instead of approximately (using Wilks original formula). Approximate errors can be used by setting `true_error = FALSE` when using `plot.calibration_simplex`.

References

Daniel S. Wilks, 2013, The Calibration Simplex: A Generalization of the Reliability Diagram for Three-Category Probability Forecasts, *Weather and Forecasting*, **28**, 1210-1218

Resin, J. (2020), A Simple Algorithm for Exact Multinomial Tests, *Preprint* https://arxiv.org/abs/2008.12682

See Also

`plot.calibration_simplex`

`ternary_forecast_example`

Examples

```
attach(ternary_forecast_example)   #see also documentation of sample data
#?ternary_forecast_example

# Calibrated forecast sample
calsim0 = calibration_simplex(p1 = p1, p3 = p3, obs = obs0)
plot(calsim0,use_pvals = TRUE) # with multinomial p-values

# Overconfident forecast sample
calsim1 = calibration_simplex(p1 = p1, p3 = p3, obs = obs1)
plot(calsim1)

# Underconfident forecast sample
calsim2 = calibration_simplex(p1 = p1, p3 = p3, obs = obs2)
plot(calsim2,use_pvals = TRUE) # with multinomial p-values

# Unconditionally biased forecast sample
calsim3 = calibration_simplex(p1 = p1, p3 = p3, obs = obs3)
plot(calsim3)

# Using a different number of bins
calsim = calibration_simplex(n=4, p1 = p1, p3 = p3, obs = obs3)
plot(calsim)

calsim = calibration_simplex(n=13, p1 = p1, p3 = p3, obs = obs3)
plot(calsim,               # using some additional plotting parameters:
     error_scale = 0.5,    # errors are less pronounced (smaller shifts)
     min_bin_freq = 100,   # dots are plotted only for bins,
                           # which contain at least 100 forecast-outcome pairs
     category_labels = c("below-normal","near-normal","above-normal"),
     main = "Sample calibration simplex")

detach(ternary_forecast_example)
```

`plot.calibration_simplex` *Plot Calibration Simplex*

Description

Plot Calibration Simplex

Usage

```
## S3 method for class 'calibration_simplex'
plot(
  x,
  true_error = TRUE,
  error_scale = 0.3,
```

```
  min_bin_freq = 10,
  plot_error_scale = TRUE,
  scale_area = NULL,
  indicate_bins = TRUE,
  category_labels = c("1", "2", "3"),
  use_pvals = FALSE,
  alphas = c(0.1, 0.01),
  ...
)
```

Arguments

x	Object of class `calibration_simplex`
true_error	Logical, specifying whether to use true miscalibration errors or approximate miscalibration errors.
error_scale	A number specifying the magnitude of the miscalibration errors (greater 0, usually should be less than 1, cf. note below).
min_bin_freq	A number. Lower bound for (absolute) frequencies, i.e. how many observations have to lie in a bin for it to be plotted.
plot_error_scale	Logical, specifying whether to plot a scale showing the magnitude of miscalibration errors.
scale_area	Optional. A number by which the areas of the points are scaled. Use if points are to small or to big.
indicate_bins	Logical, specifying whether to connect points to their respective bin (center of hexagon).
category_labels	A vector of length 3 containing the category names, e.g. `c("1","2","3")` (default)
use_pvals	Logical, determines whether multinomial p-values are used for uncertainty quantification, see details.
alphas	Vector of length 2 with values 1 > `alphas[1]` > `alphas[2]` >= 0.0001. Only relevant if `use_pvals = TRUE`.
...	Arguments concerning the title (e.g. `main`, `cex.main`, `col.main` and `font.main`) and subtitle (e.g. `sub`, `cex.sub`, `col.sub` and `font.sub`) may be passed here.

Details

If multinomial p-values are used (`use_pvals = TRUE`), the dots are colored in the following way:

- Blue: p-value greater `alphas[1]` (0.1 by default).
- Orange: p-value between `alphas[1]` and `alphas[2]` (0.1 and 0.01 by default)
- Red: p-value less than `alphas[2]` (0.01 by default)
- Black: p-value is exactly 0. This only happens if a category which is assigned 0 probability realizes.

Many small p-values (orange and red dots) indicate miscalibrated predictions, whereas many blue dots indicate that the predictions may in fact be calibrated. WARNING: The use of the multinomial p-values is more of an experimental feature and may not yield reliable p-values, especially if `n` is small. For details regarding the calculation of the p-values see also `calibration_simplex`.

Note

For details on the meaning of the error scale, cf. Wilks, 2013, especially Fig. 2. Note that the miscalibration error in each category is in "probability units" (as it is the average difference in relative frequency and forecast probability in each bin).

`ternary_forecast_example`

Ternary probability forecast and observations.

Description

10,000 realizations of a ternary probability forecast, which exhibits different characteristics, depending on the realizing outcome variable. Idealized forecast example, generated as described in Wilks (2013).

Usage

```
data(ternary_forecast_example)
```

Format

A data frame with 10,000 rows and 6 variables.

p1 forecast probability for outcome 1

p3 forecast probability for outcome 3

obs0 outcomes, such that the forecast is well-calibrated

obs1 outcomes, such that the forecast is overconfident

obs2 outcomes, such that the forecast is underconfident

obs3 outcomes, such that the forecast is unconditionally biased

Source

Data generated by package author.

References

Daniel S. Wilks, 2013, The Calibration Simplex: A Generalization of the Reliability Diagram for Three-Category Probability Forecasts, *Weather and Forecasting*, **28**, 1210-1218

Index