

Package ‘apor’

November 4, 2025

Type Package

Title Assessment of Predictions for an Ordinal Response

Version 0.1.1

Description Produces several metrics to assess the prediction of ordinal categories based on the estimated probability distribution for each unit of analysis produced by any model returning a matrix with these probabilities.

License GPL-2

Encoding UTF-8

RoxigenNote 7.3.3

NeedsCompilation no

Author Javier Espinosa-Brito [aut, cre]

Maintainer Javier Espinosa-Brito <javier.espinosa.b@usach.cl>

Repository CRAN

Date/Publication 2025-11-04 00:00:10 UTC

Contents

nopa	1
opdRef	3
ordPredArgmax	4
ordPredRandom	6
Index	8

nopa

Normalized Ordinal Prediction Agreement (NOPA)

Description

Compute the Normalized Ordinal Prediction Agreement (NOPA) metric, a performance measure for models with ordinal-scaled response variables that output estimated probability distributions (EPDs) instead of predicted labels.

This function assesses the predictive quality of a model for an ordinal response by aggregating the predicted probability mass as a function of the level of disagreement with respect to the observed category. It provides a normalized and interpretable score between 0 and 1, where 1 indicates perfect agreement and 0 represents the worst possible prediction.

NOPA compares the estimated probability distribution produced by a model for each unit of analysis against the observed ordinal response of the same unit. The maximum disagreement is $k - 1$, where k is the number of ordinal categories of the response variable, and the minimum disagreement is 0. Then, aggregates the disagreements of all units of analysis into one single measure.

The function internally computes:

- OPD — Ordinal Prediction Disagreement, the average level of disagreement between the predicted and observed categories.
- w — The worst possible OPD given the dataset, representing the maximum disagreement achievable.
- NOPA — The normalized agreement metric defined as $1 - OPD/w$.
- OPDempDist, OPDur, NOPAempDist, NOPAur: Reference values for empirical and uniform-random baselines to contextualize model performance assessment provided by OPD and NOPA.

Usage

```
nopa(predMat, obsVect)
```

Arguments

<code>predMat</code>	A numeric matrix with k columns and n rows, where k is the number of ordinal categories and n is the number of units of analysis. Each row must be the estimated probability distribution for the unit of analysis to respond each one of the k categories.
<code>obsVect</code>	A numeric or integer vector of observed categories, with values from 1 to k , where k is the number of categories of the ordinal response variable (matching the number of columns in <code>predMat</code>).

Value

A list containing:

`predMat` Input matrix of predicted probabilities.

`obsVect` Input vector of observed categories.

`disagreementsObs` A matrix with k columns (number of ordinal categories of the response variable), and n rows. Each row shows the level of disagreement of each ordinal category with respect to the observed one for the same unit of analysis.

`rearrangedProbObs` Matrix of probabilities aggregated by level of disagreement.

`meanDistObs` Mean aggregated disagreement profile.

`OPD` Observed Ordinal Prediction Disagreement.

`w` OPD for the worst prediction possible (maximum disagreement).

`NOPA` Normalized Ordinal Prediction Agreement (main metric).

`OPDempDist` A version of a reference point for OPD. It considers an ordinal prediction disagreement measure for the case where the estimated probability distribution for the k categories of the ordinal response follows the same distribution as the empirical one.

`OPDur` A version of a reference point for OPD. It considers an ordinal prediction disagreement measure for the case where the observed response variable has its own empirical distribution and the estimated probability distribution for the k categories of the ordinal response follows a uniform distribution.

`NOPAempDist` A version of a reference point for NOPA. It considers a normalized ordinal prediction agreement measure for the case where the estimated probability distribution for the k categories of the ordinal response follows the same distribution as the empirical one.

`NOPAur` A version of a reference point for NOPA. It considers a normalized ordinal prediction agreement measure for the case where the estimated probability distribution for the k categories of the ordinal response follows a uniform distribution.

References

Javier

See Also

[ordPredArgmax](#), [ordPredRandom](#) `opdRef`

Examples

```
EPD <- t(apply(matrix(runif(100), ncol=5), 1, function(y) y/sum(y)))
sum(rowSums(EPD)) == nrow(EPD)
ordResponse <- sample(1:5, 20, replace=TRUE)
nopा( predMat=EPD, obsVect=ordResponse)
```

`opdRef`

OPD Reference Points: Empirical vs Uniform Baselines

Description

Computes two reference values for the Ordinal Prediction Disagreement (OPD): (i) the expected OPD when the predicted label \hat{Y} follows the *same* empirical distribution as Y ; and (ii) the expected OPD when \hat{Y} is *uniform* over the k ordered categories while Y retains its empirical distribution. These values are useful as dataset-specific anchors for interpreting raw OPD and for constructing normalized benchmarks.

Usage

`opdRef(p)`

Arguments

- p** A probability vector of length k giving the empirical distribution of the observed ordinal outcome $Y \in \{1, \dots, k\}$. Each entry must be nonnegative and the entries must sum to 1.

Details

Let $p = (p_1, \dots, p_k)$ denote the empirical distribution of Y . The function returns two scalars:

- OPDempDist: $\mathbb{E}|\hat{Y} - Y|$ when $\hat{Y} \sim p$ independently of $Y \sim p$.
- OPDur: $\mathbb{E}|\hat{Y} - Y|$ when $\hat{Y} \sim \text{Unif}\{1, \dots, k\}$ independently of $Y \sim p$.

Both are computed via the disagreement-level decomposition

$$\mathbb{E}|\hat{Y} - Y| = \sum_{d=0}^{k-1} d \mathbb{P}(|\hat{Y} - Y| = d),$$

where, for the uniform case,

$$\text{OPD}_{UR} = \frac{1}{k} \sum_{d=0}^{k-1} d \left[\mathbb{P}\{Y \leq k-d\} - \mathbb{P}\{Y \leq d\} + \mathbb{P}\{Y \geq d+1\} \right],$$

which is the discrete- $\{1, \dots, k\}$ version of the expression shown in the manuscript.

Value

A named numeric vector of length two: `c(OPDempDist = . . . , OPDur = . . .)`.

See Also

[nopa](#), [ordPredArgmax](#), [ordPredRandom](#)

Examples

```
# Example with k = 5 categories and an empirical distribution p:
p <- c(0.10, 0.20, 0.40, 0.20, 0.10)
opdRef(p)
```

[ordPredArgmax](#)

Argmax Mapping from an Estimated Probability Distribution (EPD) to a Predicted Class

Description

Deterministically maps each row of an estimated probability distribution (EPD) matrix to a single predicted class by taking the index of the maximum probability. Rows are normalized to sum to one (within tolerance). Ties can be broken by first, last, or at random among maximizers.

Usage

```
ordPredArgmax(P, tie_break = c("first", "random", "last"), tol = 1e-12)
```

Arguments

P	A numeric matrix of size $n \times k$, where each row contains the estimated probabilities $\hat{\pi}_{ij}$ for subject i and classes $j = 1, \dots, k$. Values must be nonnegative; rows are normalized to sum to one if needed.
tie_break	Character string indicating how to break ties among equal maxima. One of "first" (default), "last", or "random".
tol	Numeric tolerance used for (i) row-sum checks and (ii) equality when identifying ties among maximum probabilities. Defaults to 1e-12.

Details

The function normalizes each row of P to sum to one (within tol). Rows with (near) zero total probability trigger an error.

If multiple classes achieve the same (within tol) maximum probability, the returned class depends on tie_break:

- "first" — smallest index among maximizers (default).
- "last" — largest index among maximizers.
- "random" — one index sampled uniformly from the set of maximizers.

Value

An integer vector of length n with the predicted class indices in $\{1, \dots, k\}$ for each row of P.

See Also

[nopa](#), [ordPredRandom](#), [opdRef](#)

Examples

```
P <- rbind(
  c(0.05, 0.10, 0.25, 0.60),
  c(0.40, 0.40, 0.10, 0.10), # tie between classes 1 and 2
  c(NA, 0.20, 0.80, 0.00) # NA treated as 0
)
```

ordPredRandom

Randomized Mapping from an Estimated Probability Distribution (EPD) to a Predicted Class

Description

Stochastically maps each row of an estimated probability distribution (EPD) matrix to a single predicted class by drawing one sample from the row's categorical distribution. Rows are normalized to sum to one (within tolerance), and the cut-points method is used with intervals $(c_{i,j-1}, c_{i,j}]$, ensuring $z_i = 1$ maps to class k .

Usage

```
ordPredRandom(P, z = NULL, tol = 1e-12)
```

Arguments

- P A numeric matrix of size $n \times k$, where each row contains the estimated probabilities $\hat{\pi}_{ij}$ for subject i and classes $j = 1, \dots, k$. Values must be nonnegative; rows are normalized to sum to one if needed.
- z Optional numeric vector of length n with values in $(0, 1]$ providing external uniforms for reproducibility or control. If NULL (default), draws are generated internally via `runif(n)`.
- tol Numeric tolerance used for row-sum checks and for guarding against underflow when normalizing. Defaults to `1e-12`.

Details

The mapping follows the cumulative cut-points $c_{i,0} = 0$, $c_{i,j} = \sum_{\ell=1}^j \hat{\pi}_{i\ell}$ for $j = 1, \dots, k$, and assigns class j whenever $c_{i,j-1} < z_i \leq c_{i,j}$. When z is supplied, values are clipped to $(0, 1]$ to respect interval boundaries. Rows with (near) zero total probability trigger an error.

Value

An integer vector of length n with the predicted class indices in $\{1, \dots, k\}$ for each row of P.

See Also

[nopa](#), [ordPredArgmax](#) [opdRef](#)

Examples

```
set.seed(1)
P <- rbind(
  c(0.05, 0.10, 0.25, 0.60),
  c(0.40, 0.40, 0.10, 0.10),
  c(0.00, 0.20, 0.80, 0.00)
)
```

```
# Stochastic draws from each row's EPD  
ordPredRandom(P)  
  
# Reproducible draws using provided uniforms  
z <- c(0.2, 0.85, 1.0)  
ordPredRandom(P, z = z)
```

Index

nopa, 1, [4–6](#)

opdRef, [3](#), [3](#), [5](#), [6](#)

ordPredArgmax, [3](#), [4](#), [4](#), [6](#)

ordPredRandom, [3–5](#), [6](#)