Package 'wevid'

June 15, 2018

Type Package

Title Quantifying Performance of a Binary Classifier Through Weight of Evidence

Version 0.4.2

Date 2018-06-15

Description
The distributions of the weight of evidence (log Bayes factor) favouring case over noncase status in a test dataset (or test folds generated by cross-validation) can be used to quantify the performance of a diagnostic test (McKeigue P., Quantifying performance of a diagnostic test as the expected information for discrimination: relation to the C-statistic. Statistical Methods for Medical Research 2018, in press). The package can be used with any test dataset on which you have observed case-control status and have computed prior and posterior probabilities of case status using a model learned on a training dataset. To quantify how the predictor will behave as a risk stratifier, the quantiles of the distributions of weight of evidence in cases and controls can be calculated and plotted.

Depends R (>= 2.10)

License GPL-3

URL http://www.homepages.ed.ac.uk/pmckeigu/preprints/classify/wevidtutorial.html

LazyLoad yes

Imports ggplot2, pROC, reshape2, zoo

ByteCompile TRUE

RoxygenNote 6.0.1

NeedsCompilation no

Author Paul McKeigue [aut],
Marco Colombo [ctb, cre]

Maintainer Marco Colombo <m.colombo@ed.ac.uk>

Repository CRAN

Date/Publication 2018-06-15 11:17:55 UTC

R topics documented:

wevid-package . . . 2
plotcumfreqs . . . 3
plotroc . . . 4
plotWdists . . . 4
prop.belowthreshold . . . 5
summary-densities . . . 6
Wdensities . . . 7
weightsofevidence . . . 8
wevid.datasets . . . 9

Index **10**

`wevid-package` *Quantifying performance of a diagnostic test using the sampling distribution of the weight of evidence favouring case over noncase status*

Description

The **wevid** package provides functions for quantifying the performance of a diagnostic test (or any other binary classifier) by calculating and plotting the distributions in cases and noncases of the weight of evidence favouring case over noncase status.

Details

The distributions of the weight of evidence (log Bayes factor) favouring case over noncase status in a test dataset (or test folds generated by cross-validation) can be used to quantify the performance of a diagnostic test.

In comparison with the C-statistic (area under ROC curve), the expected weight of evidence (expected information for discrimination) has several advantages as a summary measure of predictive performance. To quantify how the predictor will behave as a risk stratifier, the quantiles of the distributions of weight of evidence in cases and controls can be calculated and plotted.

This package can be used with any test dataset on which you have observed case-control status and have computed prior and posterior probabilities of case status using a model learned on a training dataset. Therefore, you should have computed on a test dataset (or on test folds used for cross-validation):

1. The prior probability of case status (this may be just the frequency of cases in the training data).
2. The posterior probability of case status (using the model learned on the training data to predict on the test data).
3. The observed case status (coded as 0 for noncases, 1 for cases).

The main function of the package is `Wdensities` which computes the crude and model-based densities of weight of evidence in cases and controls. Once these are computed, they can be plotted with `plotWdists` and `plotcumfreqs`. Summary statistics can be reported with `summary`.

Author(s)

Paul McKeigue <paul.mckeigue@ed.ac.uk>

References

McKeigue P., Quantifying performance of a diagnostic test as the expected information for discrimination: relation to the C-statistic. *Statistical Methods for Medical Research*, 2018, in press.

See Also

Useful links:

- http://www.homepages.ed.ac.uk/pmckeigu/preprints/classify/wevidtutorial.html

---

`plotcumfreqs` *Plot the cumulative frequency distributions in cases and in controls*

Description

Plot the cumulative frequency distributions in cases and in controls

Usage

```
plotcumfreqs(densities)
```

Arguments

`densities`	Densities object produced by `Wdensities`.

Value

A ggplot object representing the cumulative frequency distributions of the smoothed densities of the weights of evidence in cases and in controls.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))
plotcumfreqs(densities)
```

plotroc *Plot crude and model-based ROC curves*

Description

While the crude ROC curve can be non-concave and is generally not smooth, the model-based ROC curve is always concave, as the corresponding densities have been adjusted to be mathematically consistent.

Usage

```
plotroc(densities)
```

Arguments

densities	Densities object produced by Wdensities.

Value

A ggplot object representing crude and model-based ROC curves.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))
plotroc(densities)
```

plotWdists *Plot the distribution of the weight of evidence in cases and in controls*

Description

Plot the distribution of the weight of evidence in cases and in controls

Usage

```
plotWdists(densities, distlabels = c("Crude", "Model-based"))
```

Arguments

densities	Densities object produced by Wdensities.
distlabels	Character vector of length 2 to be used to label the crude and the model-based curves (in that order).

Value

A ggplot object representing the distributions of crude and model-based weights of evidence in cases and in controls.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))
plotWdists(densities)

# Example which requires fitting a mixture distribution
data("fitonly")
densities <- with(fitonly, Wdensities(y, posterior.p, prior.p,
                                      in.spike=posterior.p < 0.1))

# truncate spike
plotWdists(densities) + ggplot2::scale_y_continuous(limits=c(0, 0.5))
```

---

prop.belowthreshold *Proportions of cases and controls below a threshold of weight of evidence*

Description

Proportions of cases and controls below a threshold of weight of evidence

Usage

```
prop.belowthreshold(densities, w.threshold)
```

Arguments

densities	Densities object produced by Wdensities.
w.threshold	Threshold value of weight of evidence (natural logs).

Value

Numeric vector of length 2 listing the proportions of controls and cases with weight of evidence below the given threshold.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))
w.threshold <- log(4) # threshold Bayes factor of 4
prop.belowthreshold(densities, w.threshold)
```

summary-densities *Summary evaluation of predictive performance*

Description

Summary evaluation of predictive performance

Usage

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

## S3 method for class 'Wdensities'
mean(x, ...)

auroc.model(densities)

lambda.model(densities)
```

Arguments

object, x, densities	Densities object produced by Wdensities.
...	Further arguments passed to or from other methods. These are currently ignored.

Value

summary returns a data frame that reports the number of cases and controls, the test log-likelihood, the crude and model-based C-statistic and expected weight of evidence.

mean returns a numeric vector listing the mean densities of the weight of evidence in controls and in cases.

auroc.model returns the area under the ROC curve according to the model-based densities of weight of evidence.

lambda.model returns the expected weight of evidence (expected information for discrimination) in bits from the model-based densities.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))

summary(densities)
mean(densities)
auroc.model(densities)
lambda.model(densities)
```

`Wdensities` *Compute densities of weights of evidence in cases and controls*

Description

The function computes smoothed densities of the weight of evidence in cases and in controls from the crude probabilities, then adjusts them to make them mathematically consistent.

Usage

```
Wdensities(y, posterior.p, prior.p, range.xseq = c(-25, 25),
  x.stepsize = 0.01, adjust.bw = 1, in.spike = NULL)
```

Arguments

`y`	Binary outcome label (0 for controls, 1 for cases).
`posterior.p`	Vector of posterior probabilities generated by using model to predict on test data.
`prior.p`	Vector of prior probabilities.
`range.xseq`	Range of points where the curves should be sampled.
`x.stepsize`	Distance between each point.
`adjust.bw`	Bandwidth adjustment for the Gaussian kernel density estimator. By default it's set to 1 (no adjustment), setting it to a value smaller/larger than 1 reduces/increases the smoothing of the kernel. This argument is ignored if `in.spike` is not `NULL`.
`in.spike`	If `NULL`, the distributions of the weights of evidence are assumed to be approximately gaussian. If instead a spike-slab mixture distribution should be assumed, the this must be a logical vector of the same length as `y`, with elements set to `TRUE` if in the spike component, `FALSE` otherwise. Typically used where high proportion of values of the predictor are zero.

Details

If the model probabilities reflect a spike-slab mixture distribution, where a high proportion of values of the predictor are zero, these can be indicated through the `in.spike` argument. In the general case, it's fine to leave it set to `NULL`.

Value

A densities object that contains the information necessary to compute summary measures and generate plots.

Examples

```
data("cleveland")
densities <- with(cleveland, Wdensities(y, posterior.p, prior.p))

# Example which requires fitting a mixture distribution
data("fitonly")
densities <- with(fitonly, Wdensities(y, posterior.p, prior.p,
                                      in.spike=posterior.p < 0.1))
```

`weightsofevidence` *Calculate weights of evidence in natural log units*

Description

Calculate weights of evidence in natural log units

Usage

```
weightsofevidence(posterior.p, prior.p)
```

Arguments

`posterior.p`	Vector of posterior probabilities generated by using model to predict on test data.
`prior.p`	Vector of prior probabilities.

Value

The weight of evidence in nats for each observation.

Examples

```
data("cleveland") # load example dataset
W <- with(cleveland, weightsofevidence(posterior.p, prior.p))
```

`wevid.datasets` *Example datasets*

Description

The **wevid** package comes with the following dataset:

- `cleveland` is based on cross-validated prediction of coronary disease in the Cleveland Heart Study (297 observations).
- `pima` is based on cross-validated prediction of diabetes in Pima Native Americans (768 observations).
- `fitonly` is based on cross-validated prediction of colorectal cancer from fecal immunochemical test (FIT) only in Michigan (242 observations). As most controls and some cases have have zero values in the FIT test, to fit densities to the sampled values of weight of evidence in controls and cases it is necessary to specify spike-slab mixtures.

Format

Each dataset consists of a data frame with the following variables:

- prior.p: Prior probabilities of case status.
- posterior.p: Posterior probabilities of case status.
- y: Case-control status.

Source

http://www.homepages.ed.ac.uk/pmckeigu/preprints/classify/wevidtutorial.html

Index

∗Topic **datasets**
wevid.datasets, 9

auroc.model *(*summary-densities*)*, 6

cleveland *(*wevid.datasets*)*, 9

fitonly *(*wevid.datasets*)*, 9

lambda.model *(*summary-densities*)*, 6

mean.Wdensities *(*summary-densities*)*, 6

pima *(*wevid.datasets*)*, 9
plotcumfreqs, *2*, 3
plotroc, 4
plotWdists, *2*, 4
prop.belowthreshold, 5

summary, *2*
summary-densities, 6
summary.Wdensities *(*summary-densities*)*, 6

Wdensities, *2–6*, 7
weightsofevidence, 8
wevid *(*wevid-package*)*, 2
wevid-package, 2
wevid.datasets, 9