Package 'predfairness'

July 28, 2021

Type Package

Title Discrimination Mitigation for Machine Learning Models

Version 0.1.0

Date 2021-07-14

Maintainer Thaís de Bessa Gontijo de Oliveira <thais.bgo@gmail.com>

Description Based on different statistical definitions of discrimination, several methods have been proposed to detect and mitigate social inequality in machine learning models. This package aims to provide an alternative to fairness treatment in predictive models. The ROC method implemented in this package is described by Kamiran, Karim and Zhang (2012) <https://ieeexplore.ieee.org/document/6413831/>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests caret, stats

NeedsCompilation no

Author Thaís de Bessa Gontijo de Oliveira [aut, cre],
Leonardo Paes Vieira [aut],
Gustavo Rodrigues Lacerda Silva [ctb],
Barbara Bianca Alves Cardoso [ctb],
Douglas Alexandre Gomes Vieira [ctb]

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2021-07-28 11:50:02 UTC

R topics documented:

`adult.data` *Adult Dataset*

Description

Sample extracted from the 1994 USA census. Each observation is related to an individual.

Format

Data frame with 32,561 rows and 15 columns.

- income: factor, >50K / <=50K. Income of each person.
- age: integer, age of each person.
- workclass factor, category of work. Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
- fnlwgt: numeric
- education: factor. Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- education_num: numeric. Variable education converted to numeric.
- marital-status: factor of marital status. Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
- occupation: factor. Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
- relationship:: factor. Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
- race: factor. White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- capital_gain: numeric.
- capital_loss. numeric.
- hours-per-week: numeric. Worked hours per week.
- native-country:: factor. Country of origin.

Value

Returns a data frame with 32,561 rows and 15 columns.

Source

Dataset extracted from: UCL Machine Learning Repository

References

Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Examples

```
data('adult.data')
```

`roc_method` *Reject Option based Classification method*

Description

Reject Option based Classification (ROC) method for discrimination reduction in predictive models. Given a probabilistic model for binary classifications, ROC is a post processing method which changes instances' classification labels in a defined probability interval. In a probabilistic model, the decision criteria is defined by simply choosing the category with the higher estimated probability. Considering a binary classification, a model returns two complementary probabilities.

Assuming that discriminatory classifications occurs near rejection boundary (when probabilities are near 0.5), the ROC method defines an interval in which probabilities can be considered next to the boundary. Then, once the interval size ([0.5, theta]) is defined, the method looks for the higher probability between the two classes. If a privileged person receives a positive classification with probability between 0.5 and theta, the method turn this classification to negative. Conversely, if the method finds a deprived person with negative classification probability between 0.5 and theta, then it changes her to positive.

Usage

```
roc_method(
  pred_mod,
  positive_col,
  positive_class,
  negative_col,
  sensible_col,
  privileged_group,
  classification_col,
  theta
)
```

Arguments

`pred_mod`	data frame - predictions and its probabilities with respect to each category.
`positive_col`	string - positive classification probabilities column name
`positive_class`	string - positive classification label
`negative_col`	string - negative classification probabilities column name
`sensible_col`	string - sensible attribute column name
`privileged_group`	string - privileged group label
`classification_col`	string - classifications column name
`theta`	numeric - classification probabilities threshold

Details

In a binary classification, the highest probability is always greater than 0.5. Considering already classified instances, and selecting people from the privileged-positive classified group and deprived-negative classified group, the method searches for those with the maximum probability less than theta. In this case, the function will change the instance's classification label and replace the two probabilities with their complementary. The user must run the data frame with predictions, the column name with the sensible attribute, as well as the privileged group name through the ROC method. Also, the user must add the classification column name, the categories probabilities columns names and the name of the category considered the positive one. This function returns a data frame with updated probabilities and classifications.

Value

Returns a new data frame with updated classifications and probabilities, maintaining the structure (columns and its names) of the original data frame, ran in the method.

Author(s)

Leonardo Paes Vieira

References

F. Kamiran, A. Karim and X. Zhang, "Decision Theory for Discrimination-Aware Classification," 2012 IEEE 12th International Conference on Data Mining, 2012, pp. 924-929, doi: 10.1109/ICDM.2012.45.

Examples

```
data('adult.data')

adult.data$income = ifelse(test =  adult.data$income == '>50K',
                           yes = 1, no = 0)

adult.data = adult.data[, colnames(adult.data) %in%
                         c('age', 'education', 'sex',
                           'income', 'capital_gain')]

adult.data = adult.data[sample(1:nrow(adult.data), size = 100, replace = FALSE), ]

##### Logistic Regression

if (!requireNamespace("stats", quietly = TRUE)) {
stop("Package \"stats\" needed for this example to work.",
call. = FALSE)}

mod = glm(formula =  income ~., data = adult.data, family = binomial(link = 'logit'))

### The 'predict' function returns the classes probabilities
### automatically for caret (package) models

pred = data.frame(greater = mod$fitted.values, less =  1 - mod$fitted.values, sex = adult.data$sex,
```

```
                    classification = ifelse(mod$fitted.values >= 0.5, 'greater', 'less'))

theta = 0.6

pred_changed = roc_method(pred_mod = pred, positive_col = 'greater',
                          positive_class = 'greater', negative_col = 'less',
                          sensible_col = 'sex', privileged_group = 'Male',
                          classification_col = 'classification',
                          theta = theta)


pred_changed
```

Index