Package 'fairadapt'

November 28, 2019

Type Package

Title Fair Data Adaptation with Quantile Preservation

Version 0.1.0

Author Drago Plecko

Maintainer Drago Plecko <drago.plecko@stat.math.ethz.ch>

Description An implementation of the fair data adaptation with quantile preservation described in Plecko & Meinshausen (2019) <arXiv:1911.06685>.
The adaptation procedure uses the specified causal graph to pre-process the given training and testing data in such a way to remove the bias caused by the protected attribute. The procedure uses tree ensembles for quantile regression.

License GPL-3

Encoding UTF-8

LazyData true

Imports ranger

RoxygenNote 7.0.0

NeedsCompilation no

Repository CRAN

Date/Publication 2019-11-28 15:10:05 UTC

R topics documented:

fairadapt 2

Index **4**

`fairadapt` *fairadapt*

Description

Implementation of fair data adaptation with quantile preservation (Plecko & Meinshausen 2019). Uses only plain `R`.

Usage

```
fairadapt(formula, train.data, test.data, adj.mat, protect.A, res.vars = NULL)
```

Arguments

`formula`	Object of class `formula` describing the response and the covariates.
`train.data, test.data`	Training data & testing data, both of class `data.frame`.
`adj.mat`	Matrix of class `matrix` encoding the relationships in the causal graph. `M[i,j] == 1` implies the existence of an edge from node i to node j. Must include all the
`protect.A`	A value of class `character` describing the binary protected attribute. Must be one of the entries of `colnames(adj.mat)`.
`res.vars`	A vector of class `character` listing all the resolving variables, which should not be changed by the adaption procedure. Default value is `NULL`, corresponding to no resolving variables. Resolving variables should be a subset of the descendants of the protected attribute.

Details

The procedure takes the training and testing data as an input, together with the causal graph given by an adjacency matrix and the list of resolving variables, which should be kept fixed during the adaptation procedure. The procedure then calculates a fair representation of the data, after which any classification method can be used. There are, however, several valid training options yielding fair predictions, and the best of them can be chosen with cross-validation. For more details we refer the user to the original paper. Most of the running time is due to the quantile regression step using the ranger package.

Value

A `list` of length two. The two elements of the list are of class `data.frame` and contain the adapted training and testing data respectively.

Author(s)

Drago Plecko

References

Plecko, D. & Meinshausen, N. (2019). Fair Data Adaptation with Quantile Preservation

Examples

```
library(fairadapt)
n1 <- n2 <- 100
n <- n1 + n2
A <- rbinom(n, size = 1, prob = 0.5)
X1 <- rnorm(n) + (A-1)
X2 <- rnorm(n) + (2*A-1)
Y <- rbinom(n, size = 1, prob = exp(X1+X2)/(1+exp(X1+X2)))
data <- data.frame(cbind(A, X1, X2, Y))
adjacency.matrix <- array(0, dim = c(4,4))
colnames(adjacency.matrix) <- rownames(adjacency.matrix) <- c("A", "X1", "X2", "Y")
adjacency.matrix["A", c("X1", "X2")] <- 1
adjacency.matrix[c("X1", "X2"), "Y"] <- 1
L <- fairadapt(Y ~ ., train.data = data[1:n1, ], test.data = data[-(1:n1), ],
               protect.A = "A", adj.mat = adjacency.matrix, res.vars = "X1")

library(fairadapt)

# UCI Adult example
L <- fairadapt(income ~ ., train.data = adult.train,
               test.data = adult.test, protect.A = "sex",
               adj.mat = adjacency.matrix)
adjusted.train.data <- L[[1]]
adjusted.test.data <- L[[2]]
```

Index

fairadapt, 2