Package 'ccboost'

January 25, 2021

Type Package

Title Robust Boosting

Version 0.1-1.3

Date 2021-01-19

Author Zhu Wang [aut, cre] (<https://orcid.org/0000-0002-0773-0052>)

Maintainer Zhu Wang <zhuwang@gmail.com>

Description Robust functional descent algorithm for concave-convex family through composite optimization by conjugation operator. Wang (2021) <arXiv:2101.07718>.

Depends R (>= 3.5.0)

Imports mpath, xgboost

Suggests R.rsp, DiagrammeR

VignetteBuilder R.rsp

License GPL (>= 2)

Encoding UTF-8

LazyLoad yes

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-01-25 13:30:06 UTC

R topics documented:

ccboost *fit a predictive model with robust boosting algorithm*

Description

Fit a predictive model with robust boosting algorithm. For loss functions in the CC-family (concave-convex), apply composite optimization by conjugation operator (COCO), where optimization is conducted by functional descent boosting algorithm. Models include the generalized linear models.

Usage

```
ccboost(
  x,
  y,
  weights,
  cfun = "ccave",
  s = 1,
  delta = 0.1,
  dfun = "gaussian",
  iter = 10,
  nrounds = 100,
  del = 1e-10,
  trace = FALSE,
  ...
)
```

Arguments

x	input matrix, of dimension nobs x nvars; each row is an observation vector. Can accept dgCMatrix
y	response variable. Quantitative for family="gaussian", or family="poisson" (non-negative counts). For family="binomial" should be a factor with two levels
weights	vector of nobs with non-negative weights
cfun	concave component of CC-family, can be "hacve","acave","bcave","ccave", "dcave","ecave","gcave","hcave"
s	tuning parameter of cfun. s > 0 and can be equal to 0 for cfun="tcave". If s is too close to 0 for cfun="acave","bcave","ccave", the calculated weights can become 0 for all observations, thus crash the program
delta	a small positive number provided by user only if cfun="gcave" and 0 < s <1
dfun	type of convex component in the CC-family, can be "gaussian","binomial", "poisson"
iter	number of iteration in the COCO algorithm
nrounds	boosting iterations

del	convergency criteria in the COCO algorithm
trace	if TRUE, fitting progress is reported
...	other arguments passing to xgboost

Value

An object with S3 class xgboost.

weight_update	weight in the last iteration of the COCO algorithm

Author(s)

Zhu Wang
Maintainer: Zhu Wang <zhuwang@gmail.com>

References

Wang, Zhu (2021), *Unified Robust Boosting*, arXiv eprint, https://arxiv.org/abs/2101.07718

Examples

```
x <- matrix(rnorm(100*2),100,2)
g2 <- sample(c(0,1),100,replace=TRUE)
fit1 <- ccboost(x, g2, cfun="acave",s=0.5, dfun="gaussian", trace=TRUE,
                verbose=0, max.depth=1, nrounds=50)
fit2 <- ccboost(x, g2, cfun="acave",s=0.5, dfun="binomial", trace=TRUE,
                verbose=0, max.depth=1, nrounds=50)
fit3 <- ccboost(x, g2, cfun="acave",s=0.5, dfun="poisson", trace=TRUE,
                verbose=0, max.depth=1, nrounds=50)
```

---

dataLS *generate random data for classification as in Long and Servedio (2010)*

Description

generate random data for classification as in Long and Servedio (2010)

Usage

```
dataLS(ntr, ntu = ntr, nte, percon)
```

Arguments

ntr	number of training data
ntu	number of tuning data, default is the same as ntr
nte	number of test data
percon	proportion of contamination, must between 0 and 1. If percon > 0, the labels of the corresponding percenrage of response variable in the training and tuning data are flipped.

Value

a list with elements xtr, xtu, xte, ytr, ytu, yte for predictors of disjoint training, tuning and test data, and response variable -1/1 of training, tuning and test data.

Author(s)

Zhu Wang
Maintainer: Zhu Wang <zhuwang@gmail.com>

References

P. Long and R. Servedio (2010), *Random classification noise defeats all convex potential boosters*, *Machine Learning Journal*, 78(3), 287–304.

Examples

```
dat <- dataLS(ntr=100, nte=100, percon=0)
```

Index