Package ‘pros’

December 10, 2018

Title Penalized Regression on Steroids

Version 0.1

Author Austin David Brown <brow5079@umn.edu>

Maintainer Austin David Brown <brow5079@umn.edu>

Description This is a project for STAT8053 at the University of Minnesota.

Depends R (>= 3.5.1)

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

`cv.pros` *Cross-validation*

Description

The K-fold cross-validation function.

Usage

```
cv.pros(X, y, K_fold = 10, alpha = c(1, 0, 0, 0, 0, 0),
  lambdas = seq(0.001, 1.001, 0.01), step_size = 1/1000,
  algorithm = "proximal_gradient_cd", max_iter = 10000,
  tolerance = 10^(-8), random_seed = 0)
```

Arguments

`X`	the matrix of the data
`y`	the vector of response values
`alpha`	the convex combination of length 7 corresponding to the penalties: • l1 penalty • l2 penalty • l4 penalty • l6 penalty • l8 penalty • l10 penalty
`lambdas`	A vector of dual penalization values to be evaluated
`step_size`	step size
`algorithm`	the optimization algorithm • `proximal_gradient_cd` (proximal gradient coordinate descent) • `subgradient_cd` (subgradient coordinate algorithm)
`max_iter`	maximum iterations. This also tunes the step size.
`tolerance`	tolerance
`random_seed`	random seed

Value

A class `cv_pros`

Examples

```
cv = cv.pros(X_train, y_train)
pred = predict(cv, X_test)
```

`predict.cv_pros` *Cross-validation Prediction*

Description

The cross-validation prediction function.

Usage

```
## S3 method for class 'cv_pros'
predict(cv_prosObj, X_new)
```

Arguments

`cv_prosObj`	an object of class `cv_pros`
`X_new`	the matrix of the data to predict

Value

A `vector` of prediction values.

Examples

```
cv = cv.pros(X_train, y_train)
pred = predict(cv, X_test)
```

`predict.pros` *Pros Prediction*

Description

The prediction function.

Usage

```
## S3 method for class 'pros'
predict(prosObj, X)
```

Arguments

`prosObj`	an object of class `pros`
`X`	the matrix of the data to predict

Value

A `vector` of prediction values.

Examples

```
fit = pros(X_train, y_train, lambda = .1)
pred = predict(fit, X_test)
```

`pros` *Pros*

Description

The fit function for a specific lambda value.

Usage

```
pros(X, y, alpha = c(1, 0, 0, 0, 0, 0), lambda, step_size = 1/1000,
  algorithm = "proximal_gradient_cd", max_iter = 10000,
  tolerance = 10^(-8), random_seed = 0)
```

Arguments

- `X` — the matrix of the data
- `y` — the vector of response values
- `alpha` — the convex combination of length 7 corresponding to the penalties:
  - l1 penalty
  - l2 penalty
  - l4 penalty
  - l6 penalty
  - l8 penalty
  - l10 penalty
- `lambda` — the dual penalization value
- `step_size` — step size
- `algorithm` — the optimization algorithm
  - `proximal_gradient_cd` (proximal gradient coordinate descent)
  - `subgradient_cd` (subgradient coordinate algorithm)
- `max_iter` — maximum iterations. This also tunes the step size.
- `tolerance` — tolerance
- `random_seed` — random seed

Value

A class `pros`

Examples

```
fit = pros(X_train, y_train, lambda = .1)
pred = predict(fit, X_test)
```

Index