Package ‘rRAP’

October 14, 2022

Type Package
Title Real-Time Adaptive Penalization for Streaming Lasso Models
Version 1.1
Date 2016-09-29
Author Ricardo Pio Monti
Maintainer Ricardo Pio Monti <ricardo.monti08@gmail.com>
Depends lars, lassoshooting, MASS
Description An implementation of the Real-time Adaptive Penalization (RAP) algorithm through which to iteratively update a regularization parameter in a streaming context.
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2016-10-31 16:53:55

R topics documented:

  rRAP-package .................................................. 1
  predict.RAP ..................................................... 2
  RAP ............................................................. 3
  update.RAP ..................................................... 5

Index

rRAP-package Real-Time Adaptive Penalization for Streaming Lasso Models

Description

This package provides an implementation of the Real-time adaptive penalization (RAP) algorithm through which to iteratively update a regularization parameter in a streaming context.

Details
Author(s)

Ricardo Pio Monti
Maintainer: Ricardo Pio Monti <ricardo.monti08@gmail.com>

References


See Also

RAP, update.RAP, predict.RAP

Examples

# Recreate Figure 1 from Monti et al 2016
library(lars)
data(diabetes)
Data = cbind(diabetes$y, diabetes$x)
# initialize RAP object
R = RAP(X = matrix(diabetes$x[1,], nrow=1), y = diabetes$y[1], r = .995, eps = 0.0005, l0 = .1)
# iteratively update:
## Not run:
for (i in 2:nrow(Data)){
  R = update.RAP(RAPobj=R, Ynew = diabetes$y[i], Xnew=matrix(diabetes$x[i,], nrow=1))
}
## End(Not run)

predict.RAP

Predict method for RAP objects

Description

Obtain prediction based on current estimate of sparse linear regression model

Usage

## S3 method for class 'RAP'
predict(object, Xnew, ...)

**Arguments**

- **object**: Current RAP object
- **Xnew**: New observation from which to predict
- ... Additional arguments

**Value**

Produces a matrix of predicted values

**Author(s)**

Ricardo Pio Monti

**References**


**See Also**

RAP, update.RAP

**Examples**

```r
# library(lars)
data(diabetes)
Data = cbind(diabetes$y, diabetes$x)
# initialize RAP object with a burn in of 50 observations
R = RAP(X = matrix(diabetes$x[1:50,], nrow=50),
       y = diabetes$y[1:50], r = .995, eps = 0.0005, l0 = .1)
# make predictions:
#predict.RAP(object = R, Xnew = diabetes$x[50:70,])
```

**Description**

This function initializes and RAP object. This contains a Lasso regression model together with methods to iteratively update the regularization parameter.

**Usage**

```r
RAP(X, y, r = 0.95, eps = 0.01, l0 = 0.1, Approx = FALSE)
```
Arguments

- X: Burn in training data. Can either be a single observation (in this case a matrix with 1 row) or several. This must be a matrix.
- y: Burn in response data
- r: Fixed forgetting factor used to update
- eps: Fixed stepsize used to update regularization parameter
- l0: Initial guess for regularization parameter
- Approx: Boolean indicating whether exact or approximate gradient should be calculated when updating regularization parameter.

Details


Value

A RAP object is returned with the following elements:

- r: Fixed forgetting factor
- eps: Stepsize used to update regularization parameter
- w: Current measure of effective sample size
- xbar
- St
- regParam: Current estimate of regularization parameter
- l1Track: Vector storing all past estimates of regularization parameter
- beta: Current estimate of regression coefficients
- Approx: Boolean indicating if exact or approximate gradients where employed

The object has the following methods:

- update: Update regularization parameters and regression coefficients based on new data
- predict: Predict based on current model

Note

Warning that this implementation uses the shooting algorithm (co-ordinate gradient descent) to update regression coefficients. A more efficient implementation would employ stochastic gradient descent.

Author(s)

Ricardo Pio Monti

References

update.RAP

See Also
update.RAP, update.RAP

Examples

# Recreate Figure 1 from
library(lars)
data(diabetes)
Data = cbind(diabetes$y, diabetes$x)
# initialize RAP object
R = RAP(X = matrix(diabetes$x[1,], nrow=1), y = diabetes$y[1], r = .995, eps = 0.0005, l0 = .1)
# iteratively update:
## Not run:
for (i in 2:nrow(Data)){
  R = update.RAP(RAPobj=R, Ynew = diabetes$y[i], Xnew=matrix(diabetes$x[i,], nrow=1))
}
## End(Not run)

update.RAP  
Update sparsity parameter and regression coefficients

Description

Update regularization parameter and the associated Lasso regression coefficients, Updates can either
be mini-batch or single observations.

Usage

## S3 method for class 'RAP'
update(object, Ynew, Xnew, ...)

Arguments

object  Current RAP object
Ynew  New response. In the case of mini-batch updates a vector should be provided.
Xnew  New covariates. This should be a matrix.
...  Additional arguments

Details

See Monti et al 2016

Value

A RAP objecti is returned where the regularization parameter and the estimated regression coeffi-
cients have been updated.
Note

Warning that this implementation uses the shooting algorithm (co-ordinate gradient descent) to update regression coefficients. A more efficient implementation would employ stochastic gradient descent.

Author(s)

Ricardo Pio Monti

References


See Also

RAP, predict.RAP

Examples

# Recreate Figure 1 from
library(lars)
data(diabetes)
Data = cbind(diabetes$y, diabetes$x)
# initialize RAP object
R = RAP(X = matrix(diabetes$x[1,], nrow=1), y = diabetes$y[1], r = .995, eps = 0.0005, l0 = .1)
# iteratively update:
## Not run:
for (i in 2:nrow(Data)){
  R = update.RAP(object=R, Ynew = diabetes$y[i], Xnew=matrix(diabetes$x[i,], nrow=1))
}
## End(Not run)
Index

* package
  rRAP-package, 1

predict.RAP, 2, 2, 6

RAP, 2, 3, 3, 6
rRAP (rRAP-package), 1
rRAP-package, 1

update.RAP, 2, 3, 5, 5