Package ‘SML’
February 19, 2015

Type Package
Title Statistical Machine Learning
Version 0.1
Date 2013-08-25
Author Tuo Zhao
Maintainer Tuo Zhao <tourzhao@gmail.com>
Depends glmnet, Matrix, lattice,
Description The SML package is a collection of statistical machine learning methods.
License GPL-2
Repository CRAN
Date/Publication 2013-08-26 22:16:57
NeedsCompilation no

R topics documented:

SML-package ................................................................. 1
lasso.stars ................................................................. 2
plot.stars ................................................................. 4
print.stars ............................................................... 5

Index

SML-package Statistical Machine Learning

Description

a collection of statistical machine learning methods.

Details
Author(s)

Tuo Zhao Maintainers: tourzhao <tourzhao@gmail.com>

References


See Also

*lasso.stars*

---

**lasso.stars**

*Stability Approach to Regularization Selection for Lasso*

**Description**

Implements the Stability Approach to Regularization Selection (StARS) for Lasso

**Usage**

```r
lasso.stars(x, y, rep.num = 20, lambda = NULL, nlambda = 100,
            lambda.min.ratio = 0.001, stars.thresh = 0.1, sample.ratio = NULL,
            alpha = 1, verbose = TRUE)
```

**Arguments**

- **x**
  - The n by d data matrix representing n observations in d dimensions
- **y**
  - The n-dimensional response vector
- **rep.num**
  - The number of subsampling for StARS. The default value is 20.
**lambda**  
A sequence of decreasing positive numbers to control regularization. Typical usage is to leave the input lambda = NULL and have the program compute its own lambda sequence based on nlambda and lambda.min.ratio. Users can also specify a sequence to override this. Use with care - it is better to supply a decreasing sequence values than a single (small) value.

**nlambda**  
The number of regularization parameters. The default value is 100.

**lambda.min.ratio**  
The smallest value for lambda, as a fraction of the upperbound (MAX) of the regularization parameter which makes all estimates equal to 0. The program can automatically generate lambda as a sequence of length = nlambda starting from MAX to lambda.min.ratio*MAX in log scale. The default value is 0.001.

**stars.thresh**  
The threshold of the variability in StARS. The default value is 0.1. The alternative value is 0.05. Only applicable when criterion = "stars".

**sample.ratio**  
The subsampling ratio. The default value is 10*sqrt(n)/n when n>11T and 0.8 when n<=11T, where n is the sample size.

**alpha**  
The tuning parameter for the elastic-net regression. The default value is 1 (lasso).

**verbose**  
If verbose = FALSE, tracing information printing is disabled. The default value is TRUE.

**Details**
StARS selects the optimal regularization parameter based on the variability of the solution path. It chooses the least sparse graph among all solutions with the same variability. An alternative threshold 0.05 is chosen under the assumption that the model is correctly specified. In applications, the model is usually an approximation of the true model, 0.1 is a safer choice. The implementation is based on the popular package "glmnet".

**Value**
An object with S3 class "stars" is returned:

**path**  
The solution path of regression coefficients (in an d by nlambda matrix)

**lambda**  
The regularization parameters used in Lasso

**opt.index**  
The index of the optimal regularization parameter.

**opt.beta**  
The optimal regression coefficients.

**opt.lambda**  
The optimal regularization parameter.

**Variability**  
The variability along the solution path.

**Note**
This function can only work under the setting when d>1

**Author(s)**
Tuo Zhao  
Maintainers: Tuo Zhao <tourzhao@gmail.edu>
References


See Also

*SML-package*

Examples

```r
#generate data
x = matrix(rnorm(U0*X0),U0,LX0)
beta = c(rep(0,67))
y = rnorm(U0) + x*%*%beta

#stars for lasso
z1 = lasso.stars(x,y)
summary(z1)
plot(z1)

#stars for lasso
zR = lasso.stars(x,y, stars.thresh = 0.05)
summary(zR)
plot(zR)

#stars for lasso
zS = lasso.stars(x,y, rep.num = 50)
summary(zS)
plot(zS)
```

---

**plot.stars**: Plot function for S3 class "stars"

**Description**

Visualize the solution path and plot the optimal solution by model selection

**Usage**

```r
## S3 method for class 'stars'
plot(x, ...)
```

**Arguments**

- `x` An object with S3 class "stars"
- `...` System reserved (No specific usage)
print.stars

Author(s)

Tuo Zhao
Maintainers: Tuo Zhao <tourzhao@gmail.com>

See Also

lasso.stars

print.stars

Print function for S3 class "stars"

Description

Print the information about the solution path length and the degree of freedom’s along the solution path.

Usage

## S3 method for class 'stars'
print(x, ...)

Arguments

x An object with S3 class "stars"

... System reserved (No specific usage)

Author(s)

Tuo Zhao
Maintainers: Tuo Zhao <tourzhao@gmail.com>

See Also

lasso.stars
Index

lasso.stars, 2, 2, 5
plot.stars, 4
print.stars, 5
SML (SML-package), 1
SML-package, 1