

# R documentation

of 'cosso.Rd' etc.

January 3, 2003

## R topics documented:

cosso . . . . .	1
predict.cosso . . . . .	3
print.cosso . . . . .	4
summary.cosso . . . . .	5
SIM1x . . . . .	6
SIM2x . . . . .	6
SIM1y . . . . .	7
SIM2y . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

cosso	<i>COmponent Selection and Smoothing Operator</i>
-------	---

---

## Description

COSSEO is used for model selection and model fitting in the smoothing spline analysis of variance models by employing a norm penalty, instead of the squared norm penalty used in the common smoothing spline methods. The tuning parameter is chosen by cross validation.

## Usage

```
cosso(folds, x, y, order)
```

## Arguments

<code>folds</code>	numeric, the number of folds for cross validation
<code>x</code>	the input matrix of covariates
<code>y</code>	numeric vector, the response
<code>order</code>	numeric, the order of the ANOVA model; = 1 for additive model, = 2 for the model including all main effects and two-way interactions

## Details

It calls Fortran subroutines using the .Fortran interface.

**Value**

'cosso' returns an object of class "cosso".

The function 'predict' makes prediction for any input  $x$ . The functions 'summary' and 'print' are used to summarize the fitted results.

An object of class "cosso" is a list containing the following components:

<code>size</code>	the number of observations
<code>dim</code>	the number of variables
<code>x</code>	the matrix of input variables
<code>y</code>	the response vector
<code>lambda</code>	the parameter $\lambda_0$
<code>M</code>	the tuning parameter
<code>theta</code>	the vector $\theta$ for the estimated function
<code>c</code>	the coefficient vector $c$ for the estimated function
<code>d</code>	the coefficient $d$ for the estimated function
<code>fitted.values</code>	the fitted mean responses
<code>fitted.components</code>	the fitted components
<code>call</code>	the matched call to <code>cosso</code>

**Note**

The covariates should be scaled into the range [0,1]

**Author(s)**

Yi Lin (yilin@stat.wisc.edu) & Hao Helen Zhang (hzhang2@stat.ncsu.edu)

**References**

Lin & Zhang (2002) *Component Selection and Smoothing Spline Analysis of Variance Models*

**See Also**

[predict.cosso](#), [print.cosso](#), [summary.cosso](#) for prediction and summary of the COSSO fitting

**Examples**

```
## --- example 1 in the reference paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
print(addsol)
summary(addsol)
predict(addsol, simx1)

## --- circuit example in the reference paper: estimating the impedance Z.
data(simx2)
data(simy2)
```

```

folds <- 5
fullsol <- cosso(folds, simx2, simy2, 2)
print(fullsol)
summary(fullsol)
predict(fullsol,simx2)

```

---

predict.cosso

*Prediction method for COSSO fits*

---

## Description

Predicted values based on a “cosso” object.

## Usage

```
predict.cosso(object,newdata)
```

## Arguments

object	Object of class inheriting from “cosso”
newdata	Data frame in which to predict

## Details

‘predict.cosso’ produces predicted values, obtained by evaluating the regression function in the frame ‘newdata’. It calls a Fortran subroutine with the .Fortran interface.

## Value

‘predict.cosso’ produces a vector of predictions.

## Note

The covariates should be scaled into the range [0,1]

## Author(s)

Yi lin & Hao Helen Zhang

## References

Lin & Zhang (2002) *Component Selection and Smoothing Spline Analysis of Variance Models*

## See Also

‘cosso’, ‘print.cosso’, ‘summary.cosso’

## Examples

```
## --- example 1 in the reference paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
predict(addsol, simx1)
fullsol <- cosso(folds, simx1, simy1, 2)
predict(fullsol, simx1)
```

---

print.cosso

*Printing the main components of a COSSO Object*

---

## Description

‘print’ method for an object of class “cosso”.

## Usage

```
print.cosso(x)
print(x)
```

## Arguments

x: a “cosso” object, usually, a result of a call to ‘cosso’

## Author(s)

Yi Lin & Hao Helen Zhang

## References

Lin & Zhang (2002) *Component Selection and Smoothing Spline Analysis of Variance Models*

## See Also

‘cosso’, ‘predict.cosso’, ‘summary.cosso’

## Examples

```
## --- example1 in the reference paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
print(addsol)
fullsol <- cosso(folds, simx1, simy1, 2)
print(fullsol)
```

---

`summary.cosso`*Summarizing the fitted COSSO Object*

---

## Description

Give the values of the fitted “cosso” object.

## Usage

```
summary.cosso(object)
summary(object)
```

## Arguments

`object`: a “cosso” object, usually, a result of a call to ‘cosso’

## Value

the fitted results of a “cosso” object:

<code>c</code>	the coefficient vector <code>c</code> of the estimated function
<code>d</code>	the coefficients <code>d</code> of the estimated function
<code>fhat</code>	the fitted mean response values
<code>f_i</code>	the fitted components of the estimated function

## Author(s)

Yi Lin & Hao Helen Zhang

## References

Lin & Zhang (2002) *Component Selection and Smoothing Spline Analysis of Variance Models*

## See Also

‘cosso’, ‘predict.cosso’, ‘print.cosso’

## Examples

```
## --- example 1 in the reference paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
summary(adsol)
fullsol <- cosso(folds, simx1, simy1, 2)
summary(fullsol)
```

---

**SIM1x***Simulation 1: the input data frame X*

---

**Description**

X is a random vector generated uniformly from  $[0, 1]^{10}$ . The sample size is 100.

**Format**

This data frame contains ten columns with names 'x1', ..., 'x10'

**Examples**

```
## --- example 1 in the paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
print(addsol)
```

---

**SIM2x***Simulation 2 (circuit example): the input X*

---

**Description**

X is a random vector generated uniformly from  $[0, 1]^4$ . The sample size is 100.

**Format**

This data frame contains four columns with names 'x1', ..., 'x4'

**Examples**

```
## --- circuit example in the reference paper: estimating the impedance Z
data(simx2)
data(simy2)
folds <- 5
fullsol <- cosso(folds, simx2, simy2, 2)
print(fullsol)
```

---

SIM1y

*Simulation 1: the response Y1*

---

### Description

Y1 is generated according to the true regression function and noise level as specified by example 1 in the reference paper

### Format

Y1 is a vector

### Examples

```
## --- example 1 in the paper
data(simx1)
data(simy1)
folds <- 5
addsol <- cosso(folds, simx1, simy1, 1)
print(addsol)
predict(addsol, simx1)
```

---

SIM2y

*Simulation 2 (circuit example): the response Y2*

---

### Description

Y2 is generated according to the true regression function and noise level as specified by the circuit example in the reference

### Format

Y2 is a vector

### Examples

```
## --- circuit example in the reference paper: estimating the impedance Z
data(simx2)
data(simy2)
folds <- 5
fullsol <- cosso(folds, simx2, simy2, 2)
print(fullsol)
predict(fullsol, simx2)
```

# Index

- \*Topic **keyword**
  - cosso, 1
- \*Topic **cosso**
  - predict.cosso, 3
  - print.cosso, 4
  - summary.cosso, 4
- \*Topic **datasets**
  - SIM1x, 5
  - SIM1y, 6
  - SIM2x, 6
  - SIM2y, 7
- \*Topic **methods**
  - predict.cosso, 3
  - print.cosso, 4
  - summary.cosso, 4

cosso, 1

predict.cosso, 2, 3

print.cosso, 2, 4

SIM1x, 5

SIM1y, 6

SIM2x, 6

SIM2y, 7

summary.cosso, 2, 4