Package: Sshaped (via r-universe)

October 27, 2026

Type Package

Title Nonparametric, Tuning-Free Estimation of S-Shaped Functions

Version 1.2

Date 2025-09-22

Description Estimation of an S-shaped function and its corresponding inflection point via a least squares approach. A sequential mixed primal-dual based algorithm is implemented for the fast computation. Details can be found in Feng et al. (2022) <doi:10.1111/rssb.12481>.

License GPL-2

Imports Rcpp (>= 1.0.5)

Depends R (>= 3.4.0)

LinkingTo Rcpp, RcppArmadillo

NeedsCompilation yes

Author Oliver Y. Feng [aut], Yining Chen [aut, cre], Qiyang Han [aut], Raymond J. Carroll [aut], Richard J. Samworth [aut]

Maintainer Yining Chen <y.chen101@lse.ac.uk>

Repository https://cran.r-universe.dev

Date/Publication 2025-09-22 10:40:08 UTC

RemoteUrl https://github.com/cran/Sshaped

RemoteRef HEAD

RemoteSha 69fd7a943b38a56cf88941fb23bb7b25d2f00aec

Contents

Sshaped-package 2
cvxreg 3
plot.sshaped 4
predict.sshaped 5
sshapedreg 6

Index **8**

Sshaped-package *Nonparametric, Tuning-Free Estimation of S-Shaped Functions*

Description

Estimation of an S-shaped function and its corresponding inflection point via a least squares approach. A sequential mixed primal-dual bases algorithm is implemented for the fast computation of the estimator. The same algorithm can also be used to solve other shape-restricted regression problems, such as convex regression. See Fraser and Massam (1989) and Feng et al. (2022).

Details

Consider the nonparametric estimation of an S-shaped regression function. The least squares estimator provides a very natural, tuning-free approach, but results in a non-convex optimisation problem, since the inflection point is unknown. Nevertheless, the estimator may be regarded as a projection onto a finite union of convex cones, which allows us to propose a mixed primal-dual bases algorithm for its efficient, sequential computation.

In the current version of the package, we use this algorithm to implement the least squares regression estimator under the following shape-restrictions: S-shaped functions, i.e. increasing convex to the left of the inflection point and increasing concave to the right of the inflection point; and increasing and convex functions (as a by-product of the former). The corresponding plot and predict methods are also included. In the future, we plan to also include the estimation of additive S-shaped functions, where the covariates are multivariate for the regression.

Author(s)

- Oliver Y. Feng
- Yining Chen
- Qiyang Han
- Raymond J. Carroll
- Richard J. Samworth

References

Fraser, D. A. S. and Massam, H. (1989). A mixed primal-dual bases algorithm for regression under inequality constraints. Application to concave regression. Scandinavian Journal of Statistics, Volume 16, Pages 65-74.

Feng, O. Y., Chen, Y., Han, Q., Carroll, R. J. and Samworth, R. J. (2022). Nonparametric, tuning-free estimation of S-shaped functions. Journal of the Royal Statistical Society: Series B (Statistical Methodology), Volume 84, Issue 4, Pages 1324-1352. <doi:10.1111/rssb.12481>

Examples

```
# Generate data
set.seed(1)
x <- seq(-1,1,0.005)
y <- sin(x*pi/2) + rnorm(length(x))

# Fit S-shape
output <- sshapedreg(x,y)

# Plot
plot(output)

# prediction at new design points
xnew=rnorm(5)
predict(output,xnew)
```

---

cvxreg *Estimation of an increasing and convex function*

Description

This function computes the least squares increasing and convex regression estimator by a sequential mixed primal-dual bases algorithm.

Usage

```
cvxreg(x,y)
```

Arguments

x	a numeric vector that contains all the design points. NB. for the current version, we require all values to be distinct (but not necessarily in ascending order).
y	a numeric vector that contains the values of the response with respect to the design points.

Details

For more details, see Sshaped.

Value

An object of class sshaped, which contains the following fields:

x	covariates copied from input
y	response copied from input
fitted	the fitted values of the regression function with respect to the design points.
rss	the value of the minimised residual sum of squares of the fit
inflection	the location of the inflection point, which equals max(x) here
shape	the shape enforced in the fit, here equals "convex"

Author(s)

- Oliver Y. Feng
- Yining Chen
- Qiyang Han
- Raymond J. Carroll
- Richard J. Samworth

Examples

```
x<-seq(0,5,0.01)
y<-x^2 + rnorm(length(x))
output<-cvxreg(x,y)
plot(output)
xnew=rexp(5); predict(output,xnew)
```

`plot.sshaped` *Plot of a* `sshaped` *object*

Description

This function takes a fitted `sshaped` object produced by `sshapedreg` or `cvxreg`, and plots the observations, the fitted function and the location of the estimated inflection point (where applicable).

Usage

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

Arguments

`x`	A fitted `sshaped` object produced by `sshapedreg` or `cvxreg`.
`...`	Other arguments passed to `plot`.

Details

A plot that shows the observations, the fitted function and the location of the estimated inflection point (where applicable, i.e. when `object$shape` equals "`sshaped`".).

Value

No return value. A plot is produced.

Author(s)

- Oliver Y. Feng
- Yining Chen
- Qiyang Han
- Raymond J. Carroll
- Richard J. Samworth

See Also

sshapedreg, cvxreg

Examples

```
## See examples for the functions sshapedreg and cvxreg.
```

---

predict.sshaped *Predict method for a* sshaped *fit*

Description

This function obtains predictions at new design points from a fitted sshaped object.

Usage

```
## S3 method for class 'sshaped'
predict(object, xnew, ...)
```

Arguments

object	A fitted sshaped object produced by sshapedreg or cvxreg.
xnew	An optional numeric vector specifying the location at which prediction is required. Its values can lie outside the range of the original design points. This argument can be missing, in which case predictions are made at the same values of the covariates used to compute the object.
...	Other arguments passed to predict.

Value

A numeric vector containing predictions at new data points.

Author(s)

- Oliver Y. Feng
- Yining Chen
- Qiyang Han
- Raymond J. Carroll
- Richard J. Samworth

See Also

sshapedreg, cvxreg

Examples

```
## See examples for the functions sshapedreg and cvxreg.
```

---

sshapedreg *Estimation of an S-shaped function*

Description

This function computes the least squares S-shaped regression estimator and its corresponding inflection point by a sequential mixed primal-dual bases algorithm.

Usage

```
sshapedreg(x,y)
```

Arguments

x	a numeric vector that contains all the design points. NB. for the current version, we require all values to be distinct (but not necessarily in ascending order).
y	a numeric vector that contains the values of the response with respect to the design points.

Details

For more details, see Sshaped.

Value

An object of class sshaped, which contains the following fields:

x	covariates copied from input
y	response copied from input
fitted	the fitted values of the regression function with respect to the design points.
rss	the value of the minimised residual sum of squares of the fit
inflection	the estimated location of the inflection point
shape	the shape enforced in the fit, here equals "sshaped"

Author(s)

- Oliver Y. Feng
- Yining Chen
- Qiyang Han
- Raymond J. Carroll
- Richard J. Samworth

Examples

```
x<-seq(-1,1,0.005)
y<-sin(x*pi/2) + rnorm(length(x))
output<-sshapedreg(x,y)
plot(output)
xnew=rnorm(5); predict(output,xnew)
```

Index

cvxreg, 3, *4–6*

plot.sshaped, 4
predict.sshaped, 5

Sshaped, *3*, *6*
Sshaped *(*Sshaped-package*)*, 2
Sshaped-package, 2
sshapedreg, *4, 5*, 6, *6*