

# Package ‘smoothic’

May 9, 2026

**Type** Package

**Title** Variable Selection Using a Smooth Information Criterion

**Version** 1.2.1

**Depends** R (>= 3.5.0)

**Maintainer** Meadhbh O'Neill <meadhbhon@gmail.com>

**Description** Implementation of the SIC epsilon-telescope method, either using single or distributional (multiparameter) regression. Includes classical regression with normally distributed errors and robust regression, where the errors are from the Laplace distribution. The “smooth generalized normal distribution” is used, where the estimation of an additional shape parameter allows the user to move smoothly between both types of regression. See O'Neill and Burke (2022) “Robust Distributional Regression with Automatic Variable Selection” for more details. <doi:10.48550/arXiv.2212.07317>. This package also contains the data analyses from O'Neill and Burke (2023). “Variable selection using a smooth information criterion for distributional regression models”. <doi:10.1007/s11222-023-10204-8>.

**License** GPL-3

**URL** <https://meadhbh-oneill.github.io/smoothic/>,  
<https://github.com/meadhbh-oneill/smoothic>

**BugReports** <https://github.com/meadhbh-oneill/smoothic/issues>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.3

**Imports** data.table, dplyr, ggplot2, MASS, numDeriv, purrr, rlang,  
stringr, tibble, tidyr, toOrdinal

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Meadhbh O'Neill [aut, cre],  
Kevin Burke [aut]

**Repository** CRAN

**Date/Publication** 2025-09-16 20:20:02 UTC

## Contents

bostonhouseprice . . . . .	2
bostonhouseprice2 . . . . .	3
citycrime . . . . .	4
diabetes . . . . .	5
pcancer . . . . .	6
plot_effects . . . . .	7
plot_paths . . . . .	8
predict.smoothic . . . . .	9
smoothic . . . . .	10
sniffer . . . . .	12
summary.smoothic . . . . .	13

<b>Index</b>	<b>15</b>
--------------	-----------

---

bostonhouseprice	<i>Boston House Price Data (Original)</i>
------------------	---

---

## Description

Original data, which come from a study by Harrison Jr and Rubinfeld (1978), examining the association between median house prices in a particular community with various community characteristics. See [bostonhouseprice2](#) for the corrected version, with additional variables.

## Usage

bostonhouseprice

## Format

A data frame with 506 rows and 9 variables:

**crime** crimes committed per capita

**rooms** average number of rooms per house

**radial** index of accessibility to radial highways

**stratio** average student-teacher ratio of schools in the community

**lowstat** percentage of the population that are "lower status"

**lnox** log(annual average nitrogen oxide concentration (pphm))

**lproptax** log(property tax per \$1000)

**ldist** log(weighted distances to five employment centres in the Boston region)

**lprice** log(median house price (\$))

## Source

<https://CRAN.R-project.org/package=wooldridge>

## References

- Harrison Jr, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81-102.
- Wooldridge, J. M. (2015). *Introductory econometrics: A modern approach*. Cengage learning.

---

bostonhouseprice2      *Boston House Price Data (Corrected Version)*

---

## Description

Corrected data, which come from a study by Harrison Jr and Rubinfeld (1978), examining the association between median house prices in a particular community with various community characteristics. See [bostonhouseprice](#) for the original version.

## Usage

bostonhouseprice2

## Format

A data frame with 506 rows and 13 variables:

- crim** per capita crime rate by town  
**zn** proportion of residential land zoned for lots over 25,000 sq.ft  
**indus** proportion of non-retail business acres per town  
**rm** average number of rooms per dwelling  
**age** proportion of owner-occupied units built prior to 1940  
**rad** index of accessibility to radial highways  
**ptratio** pupil-teacher ratio by town  
**lnox** log(nitric oxides concentration (parts per 10 million))  
**ldis** log(weighted distances to five Boston employment centres)  
**ltax** log(full-value property-tax rate per USD 10,000)  
**lstat** log(percentage of lower status of the population)  
**chast** Charles River dummy variable (=1 if tract bounds river; 0 otherwise)  
**lmedv** log(corrected median value of owner-occupied homes in USD 1000's)

## Source

<https://CRAN.R-project.org/package=mlbench>

## References

- Harrison Jr, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81-102.
- Leisch F, Dimitriadou E (2021). *mlbench: Machine Learning Benchmark Problems*. R package version 2.1-3.

---

citycrime

*City Crime Data*

---

### Description

Data relating to crime rates per one million residents in 50 U.S cities, taken from Thomas (1990).

### Usage

citycrime

### Format

A data frame with 50 rows and 7 variables:

**violent** reported violent crime rate per 100,000 residents

**funding** annual police funding per resident (\$)

**hs** percentage of people 25 years+ with 4 years of high school

**not\_hs** percentage of 16 to 19 year-olds not in high school and not high school graduates

**college** percentage of 18 to 24 year-olds in college

**college4** percentage of people 25 years+ with at least 4 years of college

**crime\_rate** total overall reported crime rate per 1 million residents

### Source

[https://hastie.su.domains/StatLearnSparsity\\_files/DATA/crime.txt](https://hastie.su.domains/StatLearnSparsity_files/DATA/crime.txt)

### References

Thomas, G.S., 1990. The Rating Guide to Life in America's Small Cities. Prometheus Books, 59 John Glenn Drive, Amherst, NY 14228-2197.

Hastie, T., Tibshirani, R. and Wainwright, M., 2015. Statistical learning with sparsity: the lasso and generalizations. CRC press.

---

diabetes

*Diabetes Data*

---

### Description

Data relating to a study of disease progression one year after baseline.

### Usage

diabetes

### Format

A data frame with 442 rows and 11 variables:

**AGE** age of the patient

**SEX** sex of the patient

**BMI** body mass index of the patient

**BP** blood pressure of the patient

**S1** blood serum measurement 1

**S2** blood serum measurement 2

**S3** blood serum measurement 3

**S4** blood serum measurement 4

**S5** blood serum measurement 5

**S6** blood serum measurement 6

**Y** quantitative measure of disease progression one year after baseline

### Source

<https://CRAN.R-project.org/package=lars>

### References

Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., et al. (2004). Least angle regression. *The Annals of Statistics*.

---

pcancer

*Prostate Cancer Data*

---

### Description

Data, which come from a study by Stamey et al. (1989), examining the correlation between the level of prostate-specific antigen (PSA) and various clinical measures in men who were about to receive a radical prostatectomy.

### Usage

pcancer

### Format

A data frame with 97 rows and 9 variables:

**lcavol** log(cancer volume (cm<sup>3</sup>))

**lweight** log(prostate weight (g))

**age** age of the patient

**lbph** log(amount of benign prostatic hyperplasia (cm<sup>2</sup>))

**svi** presence of seminal vesicle invasion (1=yes, 0=no)

**lcp** log(capsular penetration (cm))

**gleason** Gleason score

**pgg45** percentage of Gleason scores four of five

**lpsa** log(PSA (ng/mL))

### Source

<https://web.stanford.edu/~hastie/ElemStatLearn/datasets/prostate.data>

### References

Stamey, T. A., Kabalin, J. N., McNeal, J. E., Johnstone, I. M., Freiha, F., Redwine, E. A., and Yang, N. (1989). Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. ii. radical prostatectomy treated patients. *The Journal of urology*, 141(5):1076-1083.

---

plot\_effects                      *Plot conditional density curves*

---

### Description

This function plots the model-based conditional density curves for different effect combinations. For example, take a particular covariate that is selected in the final model. The other selected covariates are fixed at their median values by default (see `covariate_fix` to fix at other values) and then the plotted red and blue densities correspond to the modification of the chosen covariate as “low” (25th quantile by default) and “high” (75th quantile by default).

### Usage

```
plot_effects(  
  obj,  
  what = "all",  
  show_average_indiv = TRUE,  
  p = c(0.25, 0.75),  
  covariate_fix,  
  density_range  
)
```

### Arguments

<code>obj</code>	An object of class “smoothic” which is the result of a call to <a href="#">smoothic</a> .
<code>what</code>	The covariate effects to be plotted, default is <code>what = "all"</code> . The user may supply a vector of covariate names to be plotted (only covariates selected in the final model can be plotted).
<code>show_average_indiv</code>	Should a “baseline” or “average” individual be shown, default is <code>show_average_indiv = TRUE</code> . If <code>show_average_indiv = FALSE</code> then this is not shown.
<code>p</code>	The probabilities given to the <a href="#">quantile</a> function. This corresponds to the plotted red and blue density curves where the chosen covariate is modified as “low” and “high”. The default is <code>p = c(0.25, 0.75)</code> to show the 25th and 75th quantiles.
<code>covariate_fix</code>	Optional values to fix the covariates at that are chosen in the final model. When not supplied, the covariates are fixed at their median values. See the example for more detail.
<code>density_range</code>	Optional range for which the density curves should be plotted.

### Value

A plot of the conditional density curves.

### Author(s)

Meadhbh O’Neill

**Examples**

```

# Sniffer Data -----
# MPR Model ----
results <- smoothic(
  formula = y ~ .,
  data = sniffer,
  family = "normal",
  model = "mpr"
)
plot_effects(results)

# Only plot gastemp and gaspres
# Do not show the average individual plot
# Plot the lower and upper density curves using 10th quantile (lower) and 90th quantile (upper)
# Fix violent to its violent to 820 and funding to 40

plot_effects(results,
  what = c("gastemp", "gaspres"),
  show_average_indiv = FALSE,
  p = c(0.1, 0.9),
  covariate_fix = c("gastemp" = 70,
                    "gaspres" = 4))

# The curves for the gastemp variable are computed by fixing gaspres = 4 (as is specified
# in the input). The remaining variables that are not specified in covariate_fix are fixed
# to their median values (i.e., tanktemp is fixed at its median). gastemp is then modified
# to be low (10th quantile) and high (90th quantile), as specified by p in the function.

```

---

plot\_paths

*Plot the  $\epsilon$ -telescope coefficient paths*


---

**Description**

This function plots the standardized coefficient values with respect to the  $\epsilon$ -telescope for the location (and dispersion) components.

**Usage**

```

plot_paths(
  obj,
  log_scale_x = TRUE,
  log_scale_x_pretty = TRUE,
  facet_scales = "fixed"
)

```

**Arguments**

obj	An object of class “smoothic” which is the result of a call to <a href="#">smoothic</a> .
log_scale_x	Default is log_scale_x = TRUE, which uses a log scale on the x-axis. If log_scale_x = FALSE, then the raw values of the $\epsilon$ -telescope are plotted.
log_scale_x_pretty	Default is log_scale_x_pretty = TRUE, where the x-axis labels are “pretty”. epsilon_1 and epsilon_T must be a number to the power of 10 for this to apply.
facet_scales	Default is facet_scales = “fixed”. This is supplied to <a href="#">facet_wrap</a> .

**Value**

A plot of the standardized coefficient values through the  $\epsilon$ -telescope.

**Author(s)**

Meadhbh O’Neill

**Examples**

```
# Sniffer Data -----
# MPR Model ----
results <- smoothic(
  formula = y ~ .,
  data = sniffer,
  family = "normal",
  model = "mpr"
)
plot_paths(results)
```

---

predict.smoothic	<i>Predict smoothic</i>
------------------	-------------------------

---

**Description**

predict method class “smoothic”

**Usage**

```
## S3 method for class 'smoothic'
predict(object, newdata, ...)
```

**Arguments**

object	an object of class “smoothic” which is the result of a call to <a href="#">smoothic</a> .
newdata	new data object
...	further arguments passed to or from other methods.

**Value**

a matrix containing the predicted values for the location  $\mu$  and scale  $s$

**Author(s)**

Meadhbh O'Neill

**Examples**

```
# Sniffer Data -----
# MPR Model ----
results <- smoothic(
  formula = y ~ .,
  data = sniffer,
  family = "normal",
  model = "mpr"
)
predict(results)
```

---

smoothic

*Variable Selection Using a Smooth Information Criterion (SIC)*

---

**Description**

Implements the SIC  $\epsilon$ -telescope method, either using single or multiparameter regression. Returns estimated coefficients, estimated standard errors and the value of the penalized likelihood function. Note that the function will scale the predictors to have unit variance, however, the final estimates are converted back to their original scale.

**Usage**

```
smoothic(
  formula,
  data,
  family = "sgnd",
  model = "mpr",
  lambda = "log(n)",
  epsilon_1 = 10,
  epsilon_T = 1e-04,
  steps_T = 100,
  zero_tol = 1e-05,
  max_it = 10000,
  kappa,
  tau,
  max_it_vec,
  stepmax_nlm
)
```

**Arguments**

formula	An object of class " <code>formula</code> ": a two-sided object with response on the left hand side and the model variables on the right hand side.
data	A data frame containing the variables in the model; the data frame should be unstandardized.
family	The family of the model, default is <code>family = "sgnd"</code> for the "Smooth Generalized Normal Distribution" where the shape parameter $\kappa$ is also estimated. Classical regression with normally distributed errors is performed when <code>family = "normal"</code> . If <code>family = "laplace"</code> , this corresponds to a robust regression with errors from a Laplace-like distribution. If <code>family = "laplace"</code> , then the default value of <code>tau = 0.15</code> , which is used to approximate the absolute value in the Laplace density function.
model	The type of regression to be implemented, either <code>model = "mpr"</code> for multiparameter regression (i.e., location and scale), or <code>model = "spr"</code> for single parameter regression (i.e., location only). Defaults to <code>model="mpr"</code> .
lambda	Value of penalty tuning parameter. Suggested values are " <code>log(n)</code> " and " <code>2</code> " for the BIC and AIC respectively. Defaults to <code>lambda = "log(n)"</code> for the BIC case. This is evaluated as an R expression, so it may be a number of some function of $n$ .
epsilon_1	Starting value for $\epsilon$ -telescope. Defaults to 10.
epsilon_T	Final value for $\epsilon$ -telescope. Defaults to $1e-04$ .
steps_T	Number of steps in $\epsilon$ -telescope. Defaults to 100, must be greater than or equal to 10.
zero_tol	Coefficients below this value are treated as being zero. Defaults to $1e-05$ .
max_it	Maximum number of iterations to be performed before the optimization is terminated. Defaults to $1e+04$ .
kappa	Optional user-supplied positive $\kappa$ value ( $> 0.2$ to avoid computational issues) if <code>family = "sgnd"</code> . If supplied, the shape parameter $\kappa$ will be fixed to this value in the optimization. If not supplied, $\kappa$ is estimated from the data.
tau	Optional user-supplied positive smoothing parameter value in the "Smooth Generalized Normal Distribution" if <code>family = "sgnd"</code> or <code>family = "laplace"</code> . If not supplied then <code>tau = 0.15</code> . If <code>family = "normal"</code> then <code>tau = 0</code> is used. Smaller values of $\tau$ bring the approximation closer to the absolute value function, but this can cause the optimization to become unstable. Some issues with standard error calculation with smaller values of $\tau$ when using the Laplace distribution in the robust regression setting.
max_it_vec	Optional vector of length <code>steps_T</code> that contains the maximum number of iterations to be performed in each $\epsilon$ -telescope step. If not supplied, <code>max_it</code> is the maximum number of iterations performed for 10 steps and then the maximum number of iterations to be performed reduces to 10 for the remainder of the telescope.
stepmax_nlm	Optional maximum allowable scaled step length (positive scalar) to be passed to <code>nlm</code> . If not supplied, default values in <code>nlm</code> are used.

**Value**

A list with estimates and estimated standard errors.

- `coefficients` - vector of coefficients.
- `see` - vector of estimated standard errors.
- `model` - the matched type of model which is called.
- `plike` - value of the penalized likelihood function.
- `kappa` - value of the estimated/fixed shape parameter `kappa` if `family = "sgnd"`.

**Author(s)**

Meadhbh O'Neill

**References**

O'Neill, M. and Burke, K. (2023) Variable selection using a smooth information criterion for distributional regression models. <doi:10.1007/s11222-023-10204-8>

O'Neill, M. and Burke, K. (2022) Robust Distributional Regression with Automatic Variable Selection. <arXiv:2212.07317>

**Examples**

```
# Sniffer Data -----
# MPR Model ----
results <- smoothic(
  formula = y ~ .,
  data = sniffer,
  family = "normal",
  model = "mpr"
)
summary(results)
```

---

sniffer

*Sniffer Data*

---

**Description**

Data examining the factors that impact the amount of hydrocarbon vapour released when gasoline is pumped into a tank.

**Usage**

```
sniffer
```

**Format**

A data frame with 125 rows and 5 variables:

**tanktemp** initial tank temperature (degrees F)  
**gastemp** temperature of the dispensed gasoline (degrees F)  
**tankpres** initial vapour pressure in the tank (psi)  
**gaspres** vapour pressure of the dispensed gasoline (psi)  
**y** hydrocarbons emitted (g)

**Source**

<https://CRAN.R-project.org/package=a1r4>

**References**

Bedrick, E.J. (2000). Checking for lack of fit in linear models with parametric variance functions. *Technometrics* 42 (3), 226–236.  
 Weisberg, S. (2014). *Applied Linear Regression*, 4th edition. Hoboken NJ: Wiley.

---

summary.smoothic

*Summarising Smooth Information Criterion (SIC) Fits*

---

**Description**

summary method class “smoothic”

**Usage**

```
## S3 method for class 'smoothic'
summary(object, ...)
```

**Arguments**

**object** an object of class “smoothic” which is the result of a call to `smoothic`.  
**...** further arguments passed to or from other methods.

**Value**

A list containing the following components:

- **model** - the matched model from the smoothic object.
- **coefmat** - a typical coefficient matrix whose columns are the estimated regression coefficients, estimated standard errors (SEE) and p-values.
- **plike** - value of the penalized likelihood function.

**Author(s)**

Meadhbh O'Neill

**Examples**

```
# Sniffer Data -----  
# MPR Model ----  
results <- smoothic(  
  formula = y ~ .,  
  data = sniffer,  
  family = "normal",  
  model = "mpr"  
)  
summary(results)
```

# Index

## \* datasets

- bostonhouseprice, [2](#)
- bostonhouseprice2, [3](#)
- citycrime, [4](#)
- diabetes, [5](#)
- pcancer, [6](#)
- sniffer, [12](#)

- bostonhouseprice, [2, 3](#)
- bostonhouseprice2, [2, 3](#)

- citycrime, [4](#)

- diabetes, [5](#)

- facet\_wrap, [9](#)
- formula, [11](#)

- nlm, [11](#)

- pcancer, [6](#)
- plot\_effects, [7](#)
- plot\_paths, [8](#)
- predict.smoothic, [9](#)
- print.summary.smoothic  
(summary.smoothic), [13](#)

- quantile, [7](#)

- smoothic, [7, 9, 10, 13](#)
- sniffer, [12](#)
- summary.smoothic, [13](#)