

# Package ‘RBF’

January 20, 2025

**Type** Package

**Title** Robust Backfitting

**Version** 2.1.1

**Date** 2023-08-31

**Description** A robust backfitting algorithm for additive models based on (robust) local polynomial kernel smoothers. It includes both bounded and re-descending (kernel) M-estimators, and it computes predictions for points outside the training set if desired. See Boente, Martinez and Salibian-Barrera (2017) <[doi:10.1080/10485252.2017.1369077](https://doi.org/10.1080/10485252.2017.1369077)> and Martinez and Salibian-Barrera (2021) <[doi:10.21105/joss.02992](https://doi.org/10.21105/joss.02992)> for details.

**License** GPL (>= 3.0)

**RoxygenNote** 7.2.3

**Encoding** UTF-8

**Imports** stats, graphics

**Suggests** knitr, rmarkdown, gam, RobStatTM, MASS

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Author** Matias Salibian-Barrera [aut, cre],  
Alejandra Martinez [aut]

**Maintainer** Matias Salibian-Barrera <[matias@stat.ubc.ca](mailto:matias@stat.ubc.ca)>

**Repository** CRAN

**Date/Publication** 2023-08-31 17:30:07 UTC

## Contents

RBF-package	2
backf.cl	2
backf.rob	4
deviance.backf	5
fitted.values.backf	6
formula.backf	7
k.epan	7

plot.backf . . . . .	8
predict.backf . . . . .	9
print.backf . . . . .	9
psi.huber . . . . .	10
psi.tukey . . . . .	11
residuals.backf . . . . .	11
summary.backf . . . . .	12

## Index 13

---

RBF-package	<i>A robust backfitting algorithm for additive models.</i>
-------------	--

---

### Description

A robust backfitting algorithm for additive models.

### Author(s)

Matias Salibian-Barrera, Alejandra Martinez

Maintainer: Matias Salibian-Barrera <matias@stat.ubc.ca>

### References

Boente G, Martinez A, Salibian-Barrera M. Robust estimators for additive models using backfitting. *Journal of Nonparametric Statistics*, 2017; 29:744-767. <https://doi.org/10.1080/10485252.2017.1369077>

---

backf.cl	<i>Classic Backfitting</i>
----------	----------------------------

---

### Description

This function computes the standard backfitting algorithm for additive models.

### Usage

```
backf.cl(
  formula,
  data,
  subset,
  point = NULL,
  windows,
  epsilon = 1e-06,
  degree = 0,
  prob = NULL,
  max.it = 100
)
```

## Arguments

formula	an object of class <code>formula</code> (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by <code>as.data.frame</code> to a data frame) containing the variables in the model. If not found in <code>data</code> , the variables are taken from <code>environment(formula)</code> , typically the environment from which the function was called.
subset	an optional vector specifying a subset of observations to be used in the fitting process.
point	matrix of points where predictions will be computed and returned.
windows	vector of bandwidths for the local polynomial smoother, one per explanatory variable.
epsilon	convergence criterion. Maximum allowed relative difference between consecutive estimates
degree	degree of the local polynomial smoother. Defaults to 0 (local constant).
prob	vector of probabilities of observing each response (length <code>n</code> ). Defaults to <code>NULL</code> and in that case it is ignored.
max.it	Maximum number of iterations for the algorithm.

## Details

This function computes the standard backfitting algorithm for additive models, using a squared loss function and local polynomial smoothers.

## Value

A list with the following components:

alpha	Estimate for the intercept.
g.matrix	Matrix of estimated additive components ( <code>n</code> by <code>p</code> ).
prediction	Matrix of estimated additive components for the points listed in the argument <code>point</code> .

## Author(s)

Matias Salibian-Barrera, <matias@stat.ubc.ca>, Alejandra Martinez

## References

Hasie, TJ and Tibshirani, RJ. Generalized Additive Models, 1990. Chapman and Hall, London.

## Examples

```
data(airquality)
tmp <- backf.cl(Ozone ~ Solar.R + Wind + Temp, data=airquality,
subset=complete.cases(airquality), windows=c(130, 9, 10), degree=1)
```

---

`backf.rob`*Robust Backfitting*

---

### Description

This function computes a robust backfitting algorithm for additive models

### Usage

```
backf.rob(  
  formula,  
  data,  
  subset,  
  windows,  
  point = NULL,  
  epsilon = 1e-06,  
  degree = 0,  
  sigma.hat = NULL,  
  prob = NULL,  
  max.it = 50,  
  k.h = 1.345,  
  k.t = 4.685,  
  type = "Huber"  
)
```

### Arguments

<code>formula</code>	an object of class <code>formula</code> (or one that can be coerced to that class): a symbolic description of the model to be fitted.
<code>data</code>	an optional data frame, list or environment (or object coercible by <a href="#">as.data.frame</a> to a data frame) containing the variables in the model. If not found in <code>data</code> , the variables are taken from <code>environment(formula)</code> , typically the environment from which the function was called.
<code>subset</code>	an optional vector specifying a subset of observations to be used in the fitting process.
<code>windows</code>	vector of bandwidths for the local polynomial smoother, one per explanatory variable.
<code>point</code>	matrix of points where predictions will be computed and returned.
<code>epsilon</code>	convergence criterion. Maximum allowed relative difference between consecutive estimates
<code>degree</code>	degree of the local polynomial smoother. Defaults to 0 (local constant).
<code>sigma.hat</code>	estimate of the residual standard error. If NULL (default) we use the <a href="#">mad</a> of the residuals obtained with local medians.
<code>prob</code>	vector of probabilities of observing each response (length <code>n</code> ). Defaults to NULL and in that case it is ignored.

max.it	Maximum number of iterations for the algorithm.
k.h	tuning constant for a Huber-type loss function.
k.t	tuning constant for a Tukey-type loss function.
type	one of either 'Tukey' or 'Huber'.

### Details

This function computes a robust backfitting algorithm for additive models using robust local polynomial smoothers.

### Value

A list with the following components:

alpha	Estimate for the intercept.
g.matrix	Matrix of estimated additive components (n by p).
prediction	Matrix of estimated additive components for the points listed in the argument point.
sigma.hat	Estimate of the residual standard error.

### Author(s)

Matias Salibian-Barrera, <matias@stat.ubc.ca>, Alejandra Martinez

### References

Boente G, Martinez A, Salibian-Barrera M. Robust estimators for additive models using backfitting. *Journal of Nonparametric Statistics*, 2017; 29:744-767. <https://doi.org/10.1080/10485252.2017.1369077>

### Examples

```
data(airquality)
tmp <- backf.rob(Ozone ~ Solar.R + Wind + Temp, data=airquality,
subset=complete.cases(airquality), windows=c(136.7, 8.9, 4.8), degree=1)
```

---

deviance.backf                      *Deviance for objects of class backf*

---

### Description

This function returns the deviance of the fitted additive model using one of the three classical or robust marginal integration estimators, as computed with [backf.cl](#) or [backf.rob](#).

### Usage

```
## S3 method for class 'backf'
deviance(object, ...)
```

**Arguments**

object            an object of class backf, a result of a call to `backf.cl` or `backf.rob`.  
...                additional other arguments. Currently ignored.

**Value**

A real number.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

---

fitted.values.backf    *Fitted values for objects of class backf*

---

**Description**

This function returns the fitted values given the covariates of the original sample under an additive model using a classical or robust marginal integration procedure estimator computed with `backf.cl` or `backf.rob`.

**Usage**

```
## S3 method for class 'backf'  
fitted.values(object, ...)
```

**Arguments**

object            an object of class backf, a result of a call to `backf.cl` or `backf.rob`.  
...                additional other arguments. Currently ignored.

**Value**

A vector of fitted values.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

---

formula.backf	<i>Additive model formula</i>
---------------	-------------------------------

---

**Description**

Description of the additive model formula extracted from an object of class backf.

**Usage**

```
## S3 method for class 'backf'  
formula(x, ...)
```

**Arguments**

x                    an object of class backf, a result of a call to [backf.cl](#) or [backf.rob](#).  
...                   additional other arguments. Currently ignored.

**Value**

A model formula.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

---

k.epan	<i>Epanechnikov kernel</i>
--------	----------------------------

---

**Description**

This function evaluates an Epanechnikov kernel

**Usage**

```
k.epan(x)
```

**Arguments**

x                    a vector of real numbers

**Details**

This function evaluates an Epanechnikov kernel

**Value**

A vector of the same length as x where each entry is  $0.75 * (1 - x^2)$  if  $x < 1$  and 0 otherwise.

**Author(s)**

Matias Salibian-Barrera, <matias@stat.ubc.ca>, Alejandra Martinez

**Examples**

```
x <- seq(-2, 2, length=10)
k.epan(x)
```

---

plot.backf

*Diagnostic plots for objects of class backf*

---

**Description**

Plot method for objects of class backf.

**Usage**

```
## S3 method for class 'backf'
plot(x, ask = FALSE, which = 1:np, ...)
```

**Arguments**

x	an object of class backf, a result of a call to <a href="#">backf.cl</a> or <a href="#">backf.rob</a> .
ask	logical value. If TRUE, the graphical device will prompt for confirmation before going to the next page/screen of output.
which	vector of indices of explanatory variables for which partial residuals plots will be generated. Defaults to all available explanatory variables.
...	additional other arguments. Currently ignored.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

**Examples**

```
tmp <- backf.rob(Ozone ~ Solar.R + Wind + Temp, data=airquality,
subset=complete.cases(airquality), windows=c(136.7, 8.9, 4.8), degree=1)
plot(tmp, which=1:2)
```



---

predict.backf	<i>Fitted values for objects of class backf.</i>
---------------	--

---

**Description**

This function returns the fitted values given the covariates of the original sample under an additive model using the classical or robust backfitting approach computed with [backf.cl](#) or [backf.rob](#).

**Usage**

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

**Arguments**

object	an object of class backf, a result of a call to <a href="#">backf.cl</a> or <a href="#">backf.rob</a> .
...	additional other arguments. Currently ignored.

**Value**

A vector of fitted values.

**Author(s)**

Alejandra Mercedes Martinez <[ale\\_m\\_martinez@hotmail.com](mailto:ale_m_martinez@hotmail.com)>

---

print.backf	<i>Print a Marginal Integration procedure</i>
-------------	---

---

**Description**

The default print method for a backf object.

**Usage**

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

**Arguments**

x	an object of class backf, a result of a call to <a href="#">backf.cl</a> or <a href="#">backf.rob</a> .
...	additional other arguments. Currently ignored.

**Value**

A real number.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

---

psi.huber

*Derivative of Huber's loss function.*

---

**Description**

This function evaluates the first derivative of Huber's loss function.

**Usage**

```
psi.huber(r, k = 1.345)
```

**Arguments**

r                    a vector of real numbers  
k                    a positive tuning constant.

**Details**

This function evaluates the first derivative of Huber's loss function.

**Value**

A vector of the same length as x.

**Author(s)**

Matias Salibian-Barrera, <matias@stat.ubc.ca>, Alejandra Martinez

**Examples**

```
x <- seq(-2, 2, length=10)  
psi.huber(r=x, k = 1.5)
```

---

psi.tukey	<i>Derivative of Tukey's bi-square loss function.</i>
-----------	---

---

**Description**

This function evaluates the first derivative of Tukey's bi-square loss function.

**Usage**

```
psi.tukey(r, k = 4.685)
```

**Arguments**

r	a vector of real numbers
k	a positive tuning constant.

**Details**

This function evaluates the first derivative of Tukey's bi-square loss function.

**Value**

A vector of the same length as x.

**Author(s)**

Matias Salibian-Barrera, <matias@stat.ubc.ca>, Alejandra Martinez

**Examples**

```
x <- seq(-2, 2, length=10)
psi.tukey(r=x, k = 1.5)
```

---

residuals.backf	<i>Residuals for objects of class backf</i>
-----------------	---

---

**Description**

This function returns the residuals of the fitted additive model using the classical or robust backfitting estimators, as computed with [backf.cl](#) or [backf.rob](#).

**Usage**

```
## S3 method for class 'backf'
residuals(object, ...)
```

**Arguments**

object            an object of class backf, a result of a call to [backf.cl](#) or [backf.rob](#).  
...                additional other arguments. Currently ignored.

**Value**

A vector of residuals.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

---

summary.backf

*Summary for additive models fits using backfitting*

---

**Description**

Summary method for class backf.

**Usage**

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

**Arguments**

object            an object of class backf, a result of a call to [backf.cl](#) or [backf.rob](#).  
...                additional other arguments. Currently ignored.

**Details**

This function returns the estimation of the intercept and also the five-number summary and the mean of the residuals for both classical and robust estimators. For the classical estimator, it also returns the R-squared. For the robust estimator it returns a robust version of the R-squared and the estimate of the residual standard error.

**Author(s)**

Alejandra Mercedes Martinez <ale\_m\_martinez@hotmail.com>

# Index

## \* Robust Backfitting

RBF-package, 2

as.data.frame, 3, 4

backf.ci, 2, 5–9, 11, 12

backf.rob, 4, 5–9, 11, 12

deviance.backf, 5

fitted.values.backf, 6

formula.backf, 7

k.epan, 7

mad, 4

plot.backf, 8

predict.backf, 9

print.backf, 9

psi.huber, 10

psi.tukey, 11

RBF (RBF-package), 2

RBF-package, 2

residuals.backf, 11

summary.backf, 12