

# Package ‘plugdensity’

July 23, 2025

**Version** 0.8-5

**Date** 2023-12-08

**Title** Plug-in Kernel Density Estimation

**Author** Eva Herrmann <eherrmann@mathematik.tu-darmstadt.de> (C original);  
R interface etc by Martin Maechler

**Maintainer** Martin Maechler <maechler@stat.math.ethz.ch>

**Description** Kernel density estimation with global bandwidth selection  
via ``plug-in''.

**Imports** utils

**License** GPL (>= 2)

**URL** <http://curves-etc.r-forge.r-project.org/>,  
[https://r-forge.r-project.org/R/?group\\_id=846](https://r-forge.r-project.org/R/?group_id=846),  
<https://r-forge.r-project.org/scm/viewvc.php/pkg/plugdensity/?root=curves-etc>  
,  
<svn://svn.r-forge.r-project.org/svnroot/curves-etc/pkg/plugdensity>

**BugReports** [https://r-forge.r-project.org/tracker/?group\\_id=846&atid=3357](https://r-forge.r-project.org/tracker/?group_id=846&atid=3357)

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2023-12-08 10:30:02 UTC

## Contents

plugin.density . . . . .	2
<b>Index</b>	<b>4</b>

---

 plugin.density

 Kernel Density Estimation by Plug-In Bandwidth Selection
 

---

### Description

The function `plugin.density()` provides kernel density estimation with iterative plug-in bandwidth selection.

`bw.EH()` computes the bandwidth (smoothing parameter) which `plugin.density()` also uses.

### Usage

```
plugin.density(x, nout = 201, xout = NULL, na.rm = FALSE)
bw.EH(x)
```

### Arguments

<code>x</code>	vector of number whose density is to be estimated.
<code>nout</code>	integer specifying the number of equispaced <code>xout</code> values to use <i>only when</i> <code>xout = NULL</code> (as by default).
<code>xout</code>	numeric vector of abscissa values at which the density is to be evaluated. By default, an equispaced sequence of values covering (slightly more than) the range of <code>x</code> .
<code>na.rm</code>	logical; if TRUE, missing values are removed from <code>x</code> . If FALSE any missing values cause an error.

### Value

`bw.EH()` returns the bandwidth  $h$  computed by Eva Herrmann's plugin bandwidth selector.

`plugin.density()` returns an object of class "densityEHpi" inheriting also from class "density". It is a `list` with components

<code>x</code>	the <code>n</code> coordinates of the points where the density is estimated.
<code>y</code>	the estimated density values.
<code>bw</code>	the bandwidth used.
<code>n</code>	the sample size after elimination of missing values.
<code>call</code>	the call which produced the result.
<code>data.name</code>	the deparsed name of the <code>x</code> argument.

### Author(s)

Algorithm and C code: Eva Herrmann <eherrmann@mathematik.tu-darmstadt.de>; R interface: Martin Maechler <maechler@R-project.org>.

### Source

Original code, no longer available, from '<http://www.unizh.ch/biostat/Software/>'.

## References

J. Engel, Eva Herrmann and Theo Gasser (1994). An iterative bandwidth selector for kernel estimation of densities and their derivatives. *Journal of Nonparametric Statistics* **4**, 21–34.

## See Also

[density](#).

## Examples

```
data(co2)
plot(dco2 <- density(co2), ylim = c(0, 0.03))
(pdco2 <- plugin.density(co2, xout = dco2$x))
lines(pdco2, col = "red")

plot(pdco2)# calls 'plot.density' method

str(pdco2 <- plugin.density(co2))
xo <- pdco2 $x
str(d.co2 <- density(co2, n = length(xo), from=xo[1],to=max(xo),
                    width= 4 * pdco2$bw))
nms <- c("x","y", "bw", "n")
all.equal(d.co2[nms], pdco2[nms])
## are close: "Component 2 (= 'y'): Mean relative difference: 0.0009..."
## be.EH() directly computes the bandwidth only:
stopifnot(all.equal(bw.EH(co2), pdco2$bw, tol = 3e-16))
```

# Index

\* **distribution**

plugin.density, 2

\* **smooth**

plugin.density, 2

bw.EH(plugin.density), 2

density, 2, 3

list, 2

plugin.density, 2

print.densityEHpi(plugin.density), 2