Package ‘deamer’

February 19, 2015

Type Package

Title Deconvolution density estimation with adaptive methods for a variable prone to measurement error

Version 1.0

Date 2012-07-03

Author Julien Stirnemann, Adeline Samson, Fabienne Comte. Contribution from Claire Lacour.

Maintainer j.stirnemann <j.stirnemann@gmail.com>

Description deamer provides deconvolution algorithms for the non-parametric estimation of the density f of an error-prone variable x with additive noise e. The model is y = x + e where the noisy variable y is observed, while x is unobserved. Estimation may be performed for i) a known density of the error ii) with an auxiliary sample of pure noise and iii) with an auxiliary sample of replicate (repeated) measurements. Estimation is performed using adaptive model selection and penalized contrasts.

License GPL

Repository CRAN

Date/Publication 2012-08-05 06:07:55

NeedsCompilation no

R topics documented:

`deamer-package` *Non-parametric deconvolution density estimation of variables prone to measurement-error.*

Description

The `deamer` package provides routines for non-parametric estimation of the density f of a variable with additive noise e. The general error model is $y = x + e$, where y is the noisy observation, x is the unobserved variable and e is a measurement error.

Technically, the estimation is performed using penalized deconvolution contrasts and data-driven adaptive model-selection. The estimator is projected on an orthonormal sinus cardinal basis using Fast Fourrier Transform for efficiency.
This technical framework is implemented for three situations depending on the available information or data regarding the noise e. It is assumed in all cases that the noise is homoscedastic and that its charateristic function never vanishes.
Each situation is encapsulated in a specific function:

1. `deamerKE` for estimation with known error density. The density of the error is assumed Gaussian or Laplace with a known mean and standard deviation (Gaussian) or scale parameter (Laplace).
2. `deamerSE` for estimation with an auxiliary sample of errors. This situation arises when the density of e is unknown, but an auxiliary sample of independent and identically distributed pure errors is available. Examples of such situations are found in engineering for example, when errors can be freely generated by a controlled system (like a measuring device).
3. `deamerRO` for estimation with an auxiliary sample of replicate observations (see `deamerRO` for a formal definition of replicate observations). Here, the density of the noise is specifically assumed symmetric around zero. This situation is likely to arise in biological and medical research, when pure errors cannot be observed but replicate (or repeated) noisy observations can be achieved in a sample of individuals.

Each of these functions will produce an object of class `deamer` for which generic methods exist. Alternatively, estimation can also be conducted using the default function `deamer` for users familiar with all three situations by specifying an argument `'method'` and the appropriate arguments (see `deamer-class` and the example within).
It is worth mentionning that unlike any other deconvolution procedure, `deamer` does not require an estimation of a "bandwidth" parameter prior to density estimation, thus making the usage much easier. Furthermore, `deamerKE` and `deamerSE` directly handle non-centered noise.
However, none of the `deamer` functions are implemented for heteroscedastic errors.

Details

Package:	deamer
Type:	Package
Version:	1.0
Date:	2012-06-29
License:	GPL

Author(s)

Julien Stirnemann, Adeline Samson, Fabienne Comte with contribution from Claire Lacour.

Maintainer: Julien Stirnemann <j.stirnemann@gmail.com>

References

Stirnemann JJ, Comte F, Samson A. Density estimation of a biomedical variable subject to measurement error using an auxiliary set of replicate observations. Statistics in medicine. 2012 May 17 [Epub ahead of print]

Comte F, Lacour C. Data-driven density estimation in the presence of additive noise with unknown distribution. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2011 Sep 1;73(4):601-27.

Comte F, Rozenholc Y, Taupin M-L. Penalized Contrast Estimator for Adaptive Density Deconvolution. The Canadian Journal of Statistics / La Revue Canadienne de Statistique. 2006; 34(3):431-52.

Comte F, Samson A, Stirnemann J. Deconvolution estimation of onset of pregnancy with replicate observations [Internet]. 2011 [cited 2011 Oct 25]. Available from: http://hal.archives-ouvertes.fr/hal-00588235_v2/

See Also

`deamer-class`, `deamerKE`, `deamerSE`, `deamerRO`

`deamer-class` *Objects of class 'deamer'*

Description

The `deamer` class defines the objects produced by `deamer.default` or any of `deamerKE`, `deamerSE` or `deamerRO`. Objects of class `deamer` can be used in generic functions such as `plot`, `print` and `predict`.
The default function `deamer` assumes the user is familiar with all 3 methods "se", "ke" and "ro" (see `deamer` and details), whereas method-specific wrappers `deamerKE`, `deamerSE`, `deamerRO` are intended for those who are not.

Usage

```
## Default S3 method:
deamer(y, errors, replicates, mu,  sigma, noise.type, method,
       grid.length, from, to, na.rm)
## S3 method for class 'deamer'
predict(object, newdata, na.rm,...)
```

Arguments

`y` Numeric. The vector of noisy observations.

`errors` Numeric. The vector of the auxiliary sample of errors. Does not need to be the same length as `'y'`.

`replicates` Numeric. A 2-column matrix or 2-column numeric data-frame. Contains one replicate observation in each row. The number of rows does not need to match `length(y)`.

`mu` Numeric. The (known) mean of the noise. Defaults to zero.

`sigma` Numeric. The (known) standard deviation of the noise if `noise.type="Gaussian"` or scale if `noise.type="Laplace"`

`noise.type` Character. Defines the type of density for the noise. Only `"Gaussian"` or `"Laplace"` are supported. Defaults to `"Gaussian"`

`method` Character. Only one of `"ke"`, `"se"`, `"ro"`. Defines the estimation method. See details.

`grid.length` Numeric. Optional. The number of points of the grid the estimation is performed on. Defaults to 100.

`from` Numeric. Optional. The lower bound of the grid the estimation is performed on. Defaults to `min(y)`.

`to` Numeric. Optional. The upper bound of the grid the estimation is performed on. Defaults to `max(y)`.

`na.rm` Logical. Optional. If `na.rm=TRUE`, NAs will be removed before estimation. Defaults to `FALSE`.

`object` An object of class `deamer`

`newdata` Numeric vector (possibly single valued).

`...` Further arguments for generic functions

Details

The estimation method is chosen according to the `method` argument. For known density noise, `method="ke"` and arguments `'mu'` and `'sigma'` should be supplied. For estimation with an auxiliary sample of errors `method="se"` and argument `'errors'` should be supplied. For estimation with an auxiliary sample of replicates, `method="ro"` and argument `'replicates'` should be supplied. For further details on each of these models, see `deamer` and functions `deamerKE`, `deamerSE` and `deamerRO` respectively. These functions are wrappers for deamer.default and have a more straightforward usage.

Value

`y` The input vector.

`f` The deconvolution estimate of the density of x, estimated over `supp`.

`n` Length of input vector.

`M` Sample size of pure errors (argument `'errors'` with `method="se"` or `deamerSE`) or replicate observations (argument `'replicates'` with `method="ro"` or `deamerRO`). For `method="ke"` or `deamerKE`, `M=NULL`

`method` The method of estimation. Possible values: `"kegauss"` for known Gaussian noise, `"kelap"` for known Laplace noise, `"se"` for sample of pure errors, `"ro"` for replicate noisy observations.

`mu` The mean of the error density for `method="ke"` or `deamerKE`. For other methods, `mu=NULL`.

`sigma` The standard deviation (resp. scale parameter) of the error density for `method="ke"` or `deamerKE` with Gaussian noise (resp. Laplace noise). For other methods, `sigma=NULL`.

`supp` The grid of values used for estimation.

`m` The estimated parameter for adaptive model selection.

`ahat` Values of the estimated projection coefficients using Fast Fourier Transform.

Generic function `predict` yields a vector of predictions.

Warning

Heteroscedastic errors are not supported in any of `deamerKE`, `deamerSE`, `deamerRO`.

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

References

Stirnemann JJ, Comte F, Samson A. Density estimation of a biomedical variable subject to measurement error using an auxiliary set of replicate observations. Statistics in medicine. 2012 May 17 [Epub ahead of print]

Comte F, Lacour C. Data-driven density estimation in the presence of additive noise with unknown distribution. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2011 Sep 1;73(4):601-27.

Comte F, Rozenholc Y, Taupin M-L. Penalized Contrast Estimator for Adaptive Density Deconvolution. The Canadian Journal of Statistics / La Revue Canadienne de Statistique. 2006; 34(3):431-52.

Comte F, Samson A, Stirnemann J. Deconvolution estimation of onset of pregnancy with replicate observations [Internet]. 2011 [cited 2011 Oct 25]. Available from: http://hal.archives-ouvertes.fr/hal-00588235_v2/

See Also

deamer, deamerKE, deamerRO, deamerSE

Examples

```
#this example based on simulated data presents each method implemented in deamer.
#the deamer function is presented but the wrappers deamerKE, deamerRO
#and deamerSE would yield the same results.

set.seed(12345)
n=1000; M=500
rff=function(x){
  u=rbinom(x, 1, 0.5)
  X=u*rnorm(x, -2, 1)+(1-u)*rnorm(x,2,1)
  return(X)
}
x <- rff(n) #a mixed gaussian distribution

# true density function:
f.true=function(x) (0.5/(sqrt(2*pi)))*(exp(-0.5*(x+2)^2) + exp(-0.5*(x-2)^2))

e <- rlaplace(n, 0, 0.5) # laplace noise
y <- x + e  # observations with additive noise

eps <- rlaplace(M, 0, 0.5) # a sample of pure errors for method="se"
# a 2-column matrix of replicate noisy observations for method="ro"
rep <- matrix(rep(rff(M),each=2)+rlaplace(2*M,0,0.5), byrow=TRUE, ncol=2)

#estimation with known error
# the same as deamerKE(y, noise.type="laplace", sigma=0.5)
est.ke <- deamer(y, noise.type="laplace", sigma=0.5, method="ke")
#will generate a warning since we are assuming mu=0
est.ke

#estimation with an auxiliary sample of errors
# the same as deamerSE(y, errors=eps)
est.se <- deamer(y, errors=eps, method="se")
est.se

#estimation with replicate noisy observations
# the same as deamerRO(y, replicates=rep)
est.ro <- deamer(y, replicates=rep, method="ro")
est.ro

curve(f.true(x), from=-6, to=6,lwd=2, lty=2)
lines(est.ke, lwd=1, col="green3")
lines(est.se, lwd=1, col="blue2")
lines(est.ro, lwd=1, col="orange")
legend("topright", lty=c(2,1,1,1), col=c("black", "green3", "blue2","orange"),
        legend=c("true density", "method='ke'", "method='se'", "method='ro'"),
        bty='n')
```

```
#compare predictions for each method for newx
newx=c(-2,0,2)
rbind(
      predict(est.ke, newdata=newx),
      predict(est.se, newdata=newx),
      predict(est.ro, newdata=newx)
      ) -> preds
dimnames(preds)<-list(c("ke","se","ro"),newx)
        #predictions are made at newdata
preds
```

deamerKE *Density estimation with known error density*

Description

deamerKE performs a deconvolution estimation of the density of a noisy variable ('y') under the hypothesis of a known density of the noise ("KE" for "known error"). deamerKE allows to choose between a Gaussian or a Laplace density for the noise. The standard deviation of the noise (resp. the scale parameter) is required. By default, deamerKE will consider the noise centered around zero.

Usage

```
deamerKE(y, mu, sigma, noise.type, grid.length = 100, from, to, na.rm = FALSE)
```

Arguments

y — Numeric. The vector of noisy observations

mu — Numeric. The (known) mean of the noise. Defaults to zero.

sigma — Numeric. The (known) standard deviation of the noise if noise.type="Gaussian" or scale if noise.type="Laplace"

noise.type — Character. Defines the type of density for the noise. Only "Gaussian" or "Laplace" are supported. Defaults to "Gaussian"

grid.length — Numeric. Optional. The number of points of the grid the estimation is performed on. Defaults to 100.

from — Numeric. Optional. The lower bound of the grid the estimation is performed on. Defaults to min(y).

to — Numeric. Optional. The upper bound of the grid the estimation is performed on. Defaults to max(y).

na.rm — Logical. Optional. If na.rm=TRUE, NAs will be removed before estimation. Defaults to FALSE.

Details

The model is $y = x + e$ where x has an unknown density f and e is a symmetric variable around mu (either Laplace or Gaussian). Therefore, deamerKE can directly handle non-centered noise by specifying mu.
The Gaussian mean and standard deviation have the general meaning. The Laplace density function is parameterized as:

$$\frac{1}{2\sigma} exp\Big(- \frac{|x-\mu|}{\sigma}\Big)$$

Value

An object of class 'deamer'

Warning

deamerKE is not implemented for heteroscedastic errors.

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

References

Comte F, Rozenholc Y, Taupin M-L. Penalized Contrast Estimator for Adaptive Density Deconvolution. The Canadian Journal of Statistics / La Revue Canadienne de Statistique. 2006; 34(3):431-52.

See Also

deamer, deamerRO, deamerSE, deamer-class

Examples

```
##########################################################
#EXAMPLE 1: known error, Laplacian

set.seed(12345)
n=1000
rff=function(x){
  u=rbinom(x, 1, 0.5)
  X=u*rnorm(x, -2, 1)+(1-u)*rnorm(x,2,1)
  return(X)
}
x <- rff(n) #a mixed gaussian distribution

# true density function:
f.true=function(x) (0.5/(sqrt(2*pi)))*(exp(-0.5*(x+2)^2) + exp(-0.5*(x-2)^2))

e <- rlaplace(n, 0, 0.5)
y <- x + e
```

```
est <- deamerKE(y, noise.type="laplace", sigma=0.5)
est

curve(f.true(x), -6, 6, lwd=2, lty=3)
lines(est, lwd=2)
lines(density(y), lwd=2, lty=4)
legend("topleft", bty="n", lty=c(1,3,4), lwd=2, legend=c("deamerKE", "true density",
       "kernel density\nof noisy obs."))

#########################################################
#EXAMPLE 2: known error, Laplacian and non-centered

set.seed(12345)
n=1000
rff=function(x){
  u=rbinom(x, 1, 0.5)
  X=u*rnorm(x, -2, 1)+(1-u)*rnorm(x,2,1)
  return(X)
}
x <- rff(n) #a mixed gaussian distribution

# true density function:
f.true=function(x) (0.5/(sqrt(2*pi)))*(exp(-0.5*(x+2)^2) + exp(-0.5*(x-2)^2))

e <- rlaplace(n, 2, 0.5) #mean=2 and not zero!
y <- x + e

est <- deamerKE(y, noise.type="laplace", mu=2, from=-4, to=4, sigma=0.5)
est

curve(f.true(x), -6, 6, lwd=2, lty=3)
lines(est, lwd=2)
lines(density(y), lwd=2, lty=4)
legend("topleft", bty="n", lty=c(1,3,4), lwd=2, legend=c("deamerKE", "true density",
       "kernel density\nof noisy obs."))
```

deamerRO — *Density estimation using an auxiliary sample of replicate noisy observations.*

Description

deamerRO performs a deconvolution estimation of the density of a noisy variable ('y') under the hypothesis of an unknown density of the noise using an auxiliary sample of replicate observations ("RO" for "replicate observations"). Therefore deamerRO requires two samples: one with single noisy observations and another with replicate noisy observations (see details).

Usage

```
deamerRO(y, replicates, grid.length = 100, from, to, na.rm = FALSE)
```

Arguments

`y` Numeric. The vector of noisy observations.

`replicates` Numeric. A 2-column matrix or 2-column numeric data-frame. Contains one replicate observation in each row. The number of rows does not need to match `length(y)`.

`grid.length` Numeric. Optional. The number of points of the grid the estimation is performed on. Defaults to 100.

`from` Numeric. Optional. The lower bound of the grid the estimation is performed on. Defaults to `min(y)`.

`to` Numeric. Optional. The upper bound of the grid the estimation is performed on. Defaults to `max(y)`.

`na.rm` Logical. Optional. If `na.rm=TRUE`, NAs will be removed before estimation. Defaults to `FALSE`.

Details

The model is defined as $y = x + e$, where x and e both have unknown densities.
Replicate observations are defined as

$$z_1 = x + e_1$$

$$z_2 = x + e_2$$

The main underlying hypotheses are:

1. Homoscedasticity of the errors.
2. The errors e_1 and e_2 are independent.
3. The samples are independent.
4. Errors are symmetric, 0-mean variables.
5. Errors e, e_1 and e_2 have the same distribution.

Value

an object of class `'deamer'`

Warning

- `deamerRO` is not implemented for heteroscedastic errors.
- Unlike `deamerKE` and `deamerSE`, `deamerRO` assumes the errors are centered around 0.

Note

`deamerRO` only allows for 2 replicates per observation for the moment (argument ’replicates’ is a 2-column matrix or data-frame). Future versions should allow using more than 2.

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

References

Stirnemann JJ, Comte F, Samson A. Density estimation of a biomedical variable subject to measurement error using an auxiliary set of replicate observations. Statistics in medicine. 2012 May 17 [Epub ahead of print]

Comte F, Samson A, Stirnemann J. Deconvolution estimation of onset of pregnancy with replicate observations [Internet]. 2011 [cited 2011 Oct 25]. Available from: http://hal.archives-ouvertes.fr/hal-00588235_v2/

See Also

deamer, deamerKE, deamerSE, deamer-class

Examples

```
set.seed(123)
n=1000 #sample size of single noisy observtions
M=500 #sample size of replicate observations

rff=function(x){
  u=rbinom(x, 1, 0.5)
  X=u*rnorm(x, -2, 1)+(1-u)*rnorm(x,2,1)
  return(X)
}
x <- rff(n) #a mixed gaussian distribution

# true density function:
f.true=function(x) (0.5/(sqrt(2*pi)))*(exp(-0.5*(x+2)^2) + exp(-0.5*(x-2)^2))

e <- rnorm(n,0,0.5)
y <- x + e

x. <- rff(M)

e1 <- rnorm(M,0,0.5)
e2 <- rnorm(M,0,0.5)
rep<-data.frame(y1=x.+e1, y2=x.+e2)

est<-deamerRO(y, replicates=rep)
est

plot(est, lwd=2)
curve(f.true(x), add=TRUE, lwd=2, lty=3)
lines(density(y), lwd=2, lty=4)
legend("topleft", bty="n", lty=c(1,3,4), lwd=2, legend=c("deamerRO", "true density",
       "kernel density\nof noisy obs."))
```

`deamerSE` *Density estimation using an auxiliary sample of pure errors*

Description

`deamerSE` performs a deconvolution estimation of the density of a noisy variable (`'y'`) under the hypothesis of an unknown density of the noise using an auxiliary sample of pure errors ("SE" for "sample error"). Therefore, `deamerSE` requires two samples: one with single noisy observations and another with pure errors.

Usage

```
deamerSE(y, errors, grid.length = 100, from, to, na.rm = FALSE)
```

Arguments

`y` Numeric. The vector of noisy observations.

`errors` Numeric. The vector of the auxiliary sample of errors. Does not need to be the same length as `'y'`.

`grid.length` Numeric. Optional. The number of points of the grid the estimation is performed on. Defaults to 100.

`from` Numeric. Optional. The lower bound of the grid the estimation is performed on. Defaults to `min(y)`.

`to` Numeric. Optional. The upper bound of the grid the estimation is performed on. Defaults to `max(y)`.

`na.rm` Logical. Optional. If `na.rm=TRUE`, NAs will be removed before estimation. Defaults to `FALSE`.

Details

The model is $y = x + e$ where x and e both have unknown densities. The density of x is estimated by using an independant auxiliary sample of pure errors eps (argument ’`errors`’) that will be used for computing the characteristic function of the noise. It is therefore essential to ensure that e and eps arise from the same distribution (generally experimentally).
`deamerSE` will handle non-centered errors. Therefore, the input vector for argument `errors` does not necessarily need to be centered before estimation.

Value

An object of class `'deamer'`

Warning

`deamerSE` is not implemented for heteroscedastic errors.

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

References

Comte F, Lacour C. Data-driven density estimation in the presence of additive noise with unknown distribution. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2011 Sep 1;73(4):601-27.

See Also

`deamer`, `deamerKE`, `deamerRO`, `deamer-class`

Examples

```
#################################################################################
# Example 1: centered errors
set.seed(23456)
n = 1000; M = 500
x <- rchisq(n,3)
b=0.5
e <- rlaplace(n, 0, b)
y <- x + e

eps <- rlaplace(M, 0, b)

est <- deamerSE(y, eps)
est

curve(dchisq(x, 3), 0, 12, lwd=2, lty=3)
lines(est, lwd=2)
lines(density(y), lwd=2, lty=4)
legend("topright", bty="n", lty=c(1,3,4), lwd=2, legend=c("deamerSE", "true density",
       "kernel density\nof noisy obs."))

#################################################################################
# Example 2: non-centered errors
set.seed(23456)
n = 1000; M = 500
x <- rchisq(n,3)
mu=2; b=0.5
e <- rlaplace(n, mu, b)
y <- x + e

eps <- rlaplace(M, mu, b)

est <- deamerSE(y, eps, from=0, to=12)
est

curve(dchisq(x, 3), 0, 12, lwd=2, lty=3)
```

```
lines(est, lwd=2)
lines(density(y), lwd=2, lty=4)
legend("topright", bty="n", lty=c(1,3,4), lwd=2, legend=c("deamerSE", "true density",
       "kernel density\nof noisy obs."))
```

laplace *Laplace distribution*

Description

density and random generation for the Laplace distribution of mean 'mu' and scale parameter 'b'

Usage

```
dlaplace(x, mu=0, b=1)
rlaplace(n, mu=0, b=1)
```

Arguments

x	vector of quantiles.
n	number of observations.
mu	mean. Should be a single value. Defaults to 0.
b	scale. Should be a single value. Defaults to 1.

Details

The Laplace density function is parameterized as:

$$\frac{1}{2b} exp\Big(-\frac{|x-\mu|}{b}\Big)$$

Value

Returns a vector of n draws from a Laplace distribution

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

Examples

```
set.seed(1234)

n=1000
x <- rchisq(n,3)
b=0.4
e <- rlaplace(n, 0, b)
y <- x + e  #noisy observations with laplace noise
```

mise *Mean integrated squared error*

Description

Computes the mean integrated squared error between a theoretical density and an estimate given by deamer

Usage

```
mise(density, obj)
```

Arguments

density	a theoretical density. Should be a single argument function
obj	an object of class 'deamer'. See deamer-class.

Value

Returns the value (scalar) of the mean integrated squared error.

Note

This function is mainly for simulation and comparison of methods.

Author(s)

Julien Stirnemann <j.stirnemann@gmail.com>

See Also

deamerKE,deamerSE,deamerRO, deamer, deamer-class

Examples

```
n=1000
x <- rchisq(n, df=3)
e <- rnorm(n,0,0.1)
y <- x + e
estimate <- deamerKE(y, noise.type="gaussian", sigma=0.1)
f_th <- function(x) dchisq(x, df=3)

mise(f_th, estimate)
```

Index