Package 'pmledecon'

May 30, 2022

Title Deconvolution Density Estimation using Penalized MLE

Version 0.2.1

Description Given a sample with additive measurement error, the package estimates the deconvolution density - that is, the density of the underlying distribution of the sample without measurement error. The method maximises the log-likelihood of the estimated density, plus a quadratic smoothness penalty. The distribution of the measurement error can be either a known family, or can be estimated from a ``pure error'' sample. For known error distributions, the package supports Normal, Laplace or Beta distributed error. For unknown error distribution, a pure error sample independent from the data is used.

Depends R (>= 3.6.0)

Imports stats,splitstackshape,rmutil

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.1.1

Maintainer Yun Cai <Yun.Cai@dal.ca>

NeedsCompilation no

Author Yun Cai [aut, cre],
Hong Gu [aut],
Tobias Kenney [aut]

Repository CRAN

Date/Publication 2022-05-30 06:40:02 UTC

R topics documented:

pmledecon *Deconvolution density estimation using penalized MLE*

Description

Given a sample with additive measurement error, pmledecon estimates the deconvolution density - that is, the density of the underlying distribution of the sample without measurement error. The method maximises the log-likelihood of the estimated density, plus a quadratic smoothness penalty. The distribution of the measurement error can be either a known family, or can be estimated from a "pure error" sample. For known error distributions, pmledecon supports Normal, Laplace or Beta distributed error. For unknown error distribution, a pure error sample independent from the data is used.

Usage

```
pmledecon(ob,error,supp,n,lmd,R,tsz,stsz,bsz,subid,conv)
```

Arguments

ob	Vector. The contaminated observed data.
error	Either a vector containing a sample from the "pure" error distribution; Or a list. The first element of the list is a character string, specifying a known error family. Options are *"Normal"*, *"Laplace"* and *"beta"*. The remaining two elements of the list are the two parameters for the distribution. For the normal distribution, the first parameter is the mean μ and the second is the standard deviation σ. For the Laplace distribution the first parameter is the mean and the second is the scale parameter b. For the beta distribution, the two parameters are the two shape parameters. Scaled beta distributions are not currently supported.
supp	Vector. Optional. User defined grid values for deconvolution density estimation. The default is a sequence of evenly spaced points on the estimated boundaries. The function returns a vector of density values at the provided points.
n	If $supp$ is not specified, an evenly-spaced vector of support points between estimated endpoints is used. This parameter determines the number of support points. The default is 1000.
lmd	Optional. The penalty parameter for the smoothness term.
R	Optional. This is used for setting the penalty parameter lmd depending on the likelihood and smoothness. The default is $R = 10^5$.
tsz	Number of points for numerical integration (default 1000). Whenever numerical integrations are calculated, the interval is divided into this number of steps
stsz	Optional. Initial step-size for numerical integrations. Whenever numerical integrations are calculated, the interval is divided into a fixed number of steps (default 1000). This sets the number of steps so that for the initial range, the step size is stsz.
bsz	Optional. The subsample size of the basis. Defalt is rounded result of sample size devided by 10 if the data's sample size is greater than 30 or 20 otherwise.

subid	Logical. Indicates whether to print the number of subbases completed. This can be helpful for diagnosing cases where the function fails to return a value, which can happen if too many subbases cause problems. The default is $subid = TRUE$.
conv	Logical. If $conv = TRUE$, the log-likelihood of the convolution density will be returned. Defaults to *FALSE*.

Details

The estimated density maximises a combination of the log-likelihood of the observed data, plus 'lmd' times a smoothness penalty. Larger values of 'lmd' result in smoother estimated densities, while smaller values follow the data more closely. The parameter 'R' is used to set 'lmd' based on the likelihood of the data. For the initial density estimate, 'lmd' is computed as the ratio of the derivatives of log-likelihood to the derivatives of the smoothness penalty, divided by 'R'. Thus, smaller values of 'R' result in smoother estimated densities. The default is $R=10^4$ for sample size less than 100, and $R=10^5$ for larger sample size. This has produced good results in a number of simulation studies. Appropriate values of 'R' will changed depending on the signal-noise ratio and on the smoothness of the true density.

stsz' relates to the accuracy of our numerical evaluation of the convolution. Smaller 'stsz' means smaller sampling period and the evaluation will be closer to the theoretical convolution. The default of 1000 steps along the range is usually accurate enough, but for some heavy-tailed distributions, smaller step-size may be needed.

Optimisation is performed using the $optim$ function. Because of the non-negativity constraint, this sometimes returns an error. 'pmledecon' selects a new basis in these cases, and repeats. In a few cases, it is possible that a large proportion of the sub-bases will have this problem. When $subid = TRUE$, the function will print the number of successfully optimised sub-bases after each iteration. If this number is not increasing, it indicates problems with the starting values. This is particularly common in cases where there are outliers in the data, so it can often be resolved by removing the outliers.

To ensure stability of the log-likelihood of the final solution, the 'convll' value uses the censored log-likelihood. Observations with likelihood less than 10^-10 are replaced by 10^-10.

Value

A list containing the following elements

sup	The grid of values where deconvolution density is estimated.
f	The estimated deconvolution density at the points in 'sup'.
conll	The estimated convolved loglikelihood if $conv = TRUE$.
lmd.sub	The vector of 'lmd' values used for different subsamples.

Examples

```
## Not run:
###############example for unknown error
sz=esz=30
set.seed(45217)
truth=rnorm(sz,0,1)
```

```
error=rnorm(esz,0,2)
ob=truth+error
error1=rnorm(esz,0,2)

## In order for this example to run quickly, we set tsz=200.
## This is not accurate enough for practical use.
est=pmledecon(ob,error1,tsz=200)

### compare the estimate with the truth
plot(density(ob,n=1000),col="red",lwd=2,lty=3,type="l",ylim=c(0,0.4),xlab="",main="unknown error")
lines(seq(-10,10,length.out=1000),dnorm(seq(-10,10,length.out=1000),0,1),lwd=2,lty=4,col="green")
lines(est$sup,est$f,lwd=2)

###########example of known normal error
sz=esz=30
set.seed(45217)
truth=rnorm(sz,0,1)
error=rnorm(esz,0,2)
ob=truth+error

## In order for this example to run quickly, we set tsz=200.
## This is not accurate enough for practical use.

est=pmledecon(ob,error=list("Normal",0,2),tsz=200)

### compare the estimate with the truth
plot(density(ob,n=1000),col="red",lwd=2,lty=3,type="l",ylim=c(0,0.4),xlab="",main="Normal error")
lines(seq(-10,10,length.out=1000),dnorm(seq(-10,10,length.out=1000),0,1),lwd=2,lty=4,col="green")
lines(est$sup,est$f,lwd=2)

## End(Not run)
```

Index