Package ‘log.location’

November 24, 2020

Title What the Package Does (One Line, Title Case)

Version 0.0.0.9000

Description What the package does (one paragraph).

License MIT + file LICENSE

Imports logcondens,
logcondens.mode,
pracma,
rmutil,
parallel

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

R topics documented:

`beran.est` *The ocation estimator of Beran (1974)*

Description

Suppose the observations are sampled from a density $f(x - m)$ where m is the location parameter and f is an unknown density. This function, based on Beran (1974), gives a nonparametric estimator of m. This estimator uses Fourier basis expansion to estimate the score function corresponding to the location model, which has the form

$$\phi_F(t) = f'(F^{-1}(t))/f(F^{-1}(t)).$$

The Fourier coefficients of the score function are computed by `score.coeff`.

Usage

```
beran.est(x, theta, M, init, alpha)
```

Arguments

`x`	An array of length n; represents the data.
`theta`	A small number, should be of order $O_p(n^{-1/2})$. The default is $4n^{-1/2}$. This is the tuning parameter for Fourier coefficient estimation.
`M`	The number of Fourier basis to be used to approximate ϕ_F.
`init`	Optional. A vector of initial estimators of m. The default value is the sample median.
`alpha`	The confidence level for the confidence bands. An (1-alpha) percent confidence interval is constructed. Alpha should lie in the interval (0, 0.50). The default value is 0.05.

Details

`theta`: theta do not need to depend on the range of the data because the estimators depend only on the rank of the dataponts. If $theta = z_n$ which equals $zn^{-1/2}$, then the estimated coefficients are root-n consistent by Theorem 2.1 of Beran (1974).

`M`: A higher value of M decreases bias, but increases the variance. Theorem 4.1 of Beran (1974) states that if $M \to \infty$ as the sample size n grows and $\lim_{n\to\infty} M^6/n = 0$, then this estimator is asymptotically efficient. A larger M always gives a more conservative confidence interval.

`init`: Beran (1974) recommends using sample median and warns that the method will be sensitive to the choice of the preliminary estimator. This should be a root-n consistent estimator of m. Some other choices are the mean, or the trimmed mean.

Value

A list of length two.

- `estimate`: An array of same length as init , giving the estimators of m based on the corresponding initial estimators in init. If init is missing, only one estimate is produced, which uses the sample median as the initial estimator.
- `CI`: A matrix giving the (1-alpha) percent confidence intervals (CI). Each row corresponds to an initial estimator in init (if missing, the sample median is used). The first column corresponds to the lower CI, and the second column corresponds to the upper CI.

Author(s)

Nilanjana Laha (maintainer), `<nlaha@hsph.harvard.edu>`.

References

Beran, R. (1974). *Asymptotically efficient adaptive rank estimates in location models*. Ann. Statist., 2, 63-74.

Examples

```
x <- rlogis(100); beran.est(x, M=10)
beran.est(x, M=20)
```

`beran.select` *Tuning parameter selector for Beran (1974)'s location estimator*

Description

`beran.est` computes the location estimator in a location shift model using Beran (1974)'s estimator. This function depends on tuning parameters theta and M. We estimate the MSE of the estimators and to choose the best (theta, M) from a set of options.

Usage

```
beran.select(x, t, Mvec, inth, B)
```

Arguments

`x`	An array of length n; the dataset.
`t`	Optional, an array of real numbers, contains values of theta, parameter needed for estimating the Fourier coefficient.
`Mvec`	Optional, an array of integers, the number of basis functions to use. See details.
`inth`	A number; the initial estimator. The default is the median.
`B`	Optional, the number of bootstrap samples. The default is 100.

Details

To this end, we generate B Bootstrap samples and for each pair of (theta, M), we estimate the MSE by computing

$$\frac{\sum_{i=1}^{B}(\hat{\theta}_i(theta, M) - \theta)^2}{B}$$

where $\theta_i(theta, M)$ is the estimator based on the i-th bootstrap sample and θ is the true theta.

We take theta to be in c(0.01, seq(0.10, 0.80, by=0.05)) and take M to be in (2, 4, 6, ..., 10).

Value

A vector of foure numbers.

- `param`: A 2 columns, the number of rows is same as the length of inth. Each row gives the optimal (dn, tn) for the corresponding inth value.
- `estimte`: Gives the estimators of θ using the optimal (dn, tn)'s.

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Laha, N. *Location estimation fr symmetric and log-concave densities*. Submitted.

Stone, C. (1975). *Adaptive maximum likelihood estimators of a location parameter*, Ann. Statist., 3, 267-284.

See Also

`beran.est`

Examples

```
{}
  @export
```

`giveth` *Stone (1975)'s estimator*

Description

Gives a truncated one step estimator for the location parameter in a symmetric location family. This estimator is constructed using Stone (1975)'s methods. This estimator uses the kernel density estimator (KDE) to estimate the scores. Similar to all other one-step estimators, this estimator requires a preliminary estimator.

Usage

```
giveth(x, inthv)
```

Arguments

`x`	A vector of length n; the dataset
`dn`	A parameter for the truncation. The default value is 20.
`tn`	A parameter associated with the bandwidth of the kernel density estimator. The default value is 0.60.
`inth`	A preliminary estimator for θ. The default is the median.
`alpha`	The confidence level for the confidence bands. An (1-alpha) percent confidence interval is constructed. Alpha should lie in the interval (0, 0.50). The default value is 0.05.

Details

Stone (1975) uses $r_n = \hat{\sigma} t_n$ as the bandwidth of the Gaussian kernel, where $\hat{\sigma}$ is the median of the $X_i - inth$'s. For asymptotic efficiency of the estimators, a) dn$\to \infty$ and tn$\to 0$ b)

$$\frac{(dn)^2}{n^{1-\epsilon}(tn)^6} = O(1)$$

for some $\epsilon > 0$. A rule of thumb plug-in estimate for the optimal kernel width is $1.059\hat{\sigma}n^{-1/5}$ where $\hat{\sigma}$ is an estimator of the standard deviation, which does not satisfy the above relation. Stone(1975) takes inth to be the median in his simulations. The default choices of dn and tn are taken from Stone (1975), who considered a sample of size 40. Stone(1975)'s estimator uses Gaussian Kernels to estimate the unknown symmetric density.

Value

A list of length two.

- `estimate`: An array of same length as init , giving the estimators of m based on the corresponding initial estimators in init. If init is missing, only one estimate is produced, which uses the sample median as the initial estimator.
- `CI`: A matrix giving the (1-alpha) percent confidence intervals (CI). Each row corresponds to an initial estimator in init (if missing, the sample median is used). The first column corresponds to the lower CI, and the second column corresponds to the upper CI.

Author(s)

Nilanjana Laha (maintainer), `<nlaha@hsph.harvard.edu>`.

References

Laha, N. *Location estimation fr symmetric and log-concave densities*. Submitted.

Stone, C. (1975). *Adaptive maximum likelihood estimators of a location parameter*, Ann. Statist., 3, 267-284.

Examples

```
x <- rnorm(100); inth <- mean(x);
 giveth(x, inth=inth)
 giveth(x)
```

`MLE_location` *The MLE estimator of Laha (2020).*

Description

Suppose n univariate observations are sampled from a density $f(x - m)$ where m is the location parameter and f is an unknown symmetric density. This function computes a one step estimator to estimte m. This estimator uses the log-concave MLE estimator from the package `logcondens.mode` to estimate f.

Usage

```
MLE_location(x, alpha = 0.05)
```

Arguments

x	An array of length n; represents the data.
alpha	The confidence level for the confidence bands. An (1-alpha) percent confidence interval is constructed. Alpha should lie in the interval (0, 0.50). The default value is 0.05.

Value

A list of length two.

- `estimate`: A scalar.
- `CI`: A vector of length two.

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Laha N. (2020). *Location estimation for symmetric and log-concave densities*. submitted.

See Also

`p.mle`, `s.sym`

Examples

```
x <- rnorm(100); MLE_location(x)
```

`mle_score` *Scores of log-concave MLE*

Description

Calculates

$$\frac{\hat{\phi}_n(X_{i+1}) - \hat{\phi}_n(X_i)}{X_{i+1} - X_i}$$

for i=1,...,n., where $\hat{\phi}_n = \log \hat{f}_n$, the latter being the unsmoothed log-concave MLE given by `logcondens`.

Usage

```
mle_score(x)
```

Arguments

x	a vector of length n; the dataset

Value

scores evaluated in between the data points, a vector of size n-1.

`p.mle` *The partial MLE estimator of Laha (2020).*

Description

Suppose n univariate observations are sampled from a density $f(x - m)$ where m is the location parameter and f is an unknown symmetric density. This function computes a one step estimator to estimte m. This estimator uses the log-concave MLE estimator from the package `logcondens.mode` to estimate f, and is root-n consistent for m provided f is log-concave.

Usage

```
p.mle(x, init, q = 0, alpha = 0.05)
```

Arguments

`x`	An array of length n; represents the data.
`init`	Optional. An initial estimator of m. The default value is the sample median.
`q`	A fraction between 0 and 1/2. Corresponds to the truncation parameter. The default is 0, which indicates no truncation.
`alpha`	The confidence level for the confidence bands. An (1-alpha) percent confidence interval is constructed. Alpha should lie in the interval (0, 0.50). The default value is 0.05.

Details

`q`: If q is positive, the function (1-2*q) percent observations from both tails while computing the one step estimator. The scores are estimated using the full data. See Laha et al. for more details.

`init`: The default is mean. If init is the median, some jitter (0.0001) is added.

Value

A list of length two.

- `estimate`: A matrix of two columns, and the rownumber equals the length of q. The firs column is q, and the second column is estimates corresponding to q.
- `CI`: A matrix of three columns, the first column is q, the second column is the left point and the third column is the right point of the confidence intervals corresponding to q.

Author(s)

Nilanjana Laha (maintainer), `<nlaha@hsph.harvard.edu>`.

References

Laha N. (2020). *Location estimation for symmetric and log-concave densities*. submitted.

Examples

```
x <- rlogis(100); p.mle(x, q=c(0, 0.001, 0.01))
```

`s.sym` *Smoothed symmetrized estimator of location from Laha (2020).*

Description

Suppose n univariate observations are sampled from a density $f(x-m)$ where m is the location parameter and f is an unknown symmetric density. This function computes a one step estimator to estimte m. This estimator uses the smoothed log-concave MLE estimator from the package `logcondens` to estimate f, and is root-n consistent for m provided f is log-concave.

Usage

```
s.sym(x, q = 0, inth, alpha = 0.05)
```

Arguments

`x`	An array of length n; represents the data.
`q`	A fraction between 0 and 1/2. Corresponds to the truncation parameter. The default is 0, which indicates no truncation.
`alpha`	The confidence level for the confidence bands. An (1-alpha) percent confidence interval is constructed. Alpha should lie in the interval (0, 0.50). The default value is 0.05.
`init`	Optional. An initial estimator of m. The default value is the sample median.

Details

`q`: If q is positive, the function (1-2*q) percent observations from both tails while computing the one step estimator. The scores are estimated using the full data. See Laha et al. for more details.

`init`: The default is mean. If init is the median, some jitter (0.0001) is added.

Value

A list of length two.

- `estimate`: A matrix of two columns, and the rownumber equals the length of q. The firs column is q, and the second column is estimates corresponding to q.
- `CI`: A matrix of three columns, the first column is q, the second column is the left point and the third column is the right point of the confidence intervals corresponding to q.

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Laha N. (2020). *Location estimation for symmetric and log-concave densities*. submitted.

Examples

```
x <- rlogis(100); p.mle(x, q=c(0, 0.001, 0.01))
```

`score.coeff` *Fourier coefficients of the score in a location model*

Description

Estimates the Fourier coefficients of the scores in a location shift model, with is the class of densities obtained by a location shift of a fixed unknown density f. The score, given by

$$\phi'(x) = -f' \circ F^{-1}(t)/f \circ F^{-1}(x),$$

does not depend on the unknown shift. The Fourier coefficients of the score is estimated from the data, whose density is assumed to be a location shift of the unknown density f. We use Beran (1974)'s nonparametric estimator here.

Usage

```
score.coeff(x, theta, indices, which)
```

Arguments

`x`	An array of length n; represents the data from a location model whose score the Fourier coefficients seek to estimate.
`theta`	A small number, should be of order $O_p(n^{-1/2})$. The default is $4n^{-1/2}$
`indices`	An array of positive integers, for each integer j in "indices", Fourier coefficient corresponding to the basis function $t \mapsto \exp(i2\pi jt)$ is computed.
`which`	Optional. Takes value 1 or 2. If "which" is 1, only the real part of the Fourier coefficient is computed. If "which" is 2, only the imaginary part of the k-th coefficient is calculated. The default is to calculate both real and imaginary parts.

Details

`theta`: theta do not need to depend on the range of the data because the estimators depend only on the rank of the dataponts. If $theta = z_n$ which equals $zn^{-1/2}$, then the estimated coefficients are root-n consistent by Theorem 2.1 of Beran (1974).

`which`: If it is known that the density f is symmetric, then the Fourier coefficients are real. Therefore, there is no need to calculate the imaginary part. Similarly, if the density is odd, then the Fourier coefficinets of the scores are purely imaginary. Otherwise, one generally requires both the real and the imaginary parts, and "which" should be left unspecified in those cases.

Value

- If "which=1", an array (with the same length as "indices"), give the real parts of the Fourier coefficients.
- If "which=2, an array (with the same length as "indices"), give the imaginary parts of the Fourier coefficients.
- The default is to return a matrix whose first and second column give the real and imaginary parts of the Fourier coefficients, respectively. The number of rows equal the length of the array "indices".

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Beran, R. (1974). *Asymptotically efficient adaptive rank estimates in location models*. Ann. Statist., 2, 63-74.

Examples

```
x <- rnorm(100); score.coeff(x, length(x)^(-1/2), 1)
```

`stone.select` *Tuning parameter selector for Stone (1975)'s location estimator*

Description

`giveth` computes the location estimator in a location shift model using Stone (1975)'s estimator. This function depends on tuning parameters dn and tn. We estimate the MSE of the estimators and to choose the best (dn,tn) from a set of options.

Usage

```
stone.select(x, inth, D, t, B)
```

Arguments

`x`	An array of length n; the dataset.
`inth`	A number; the initial estimator. The default is the median.
`D`	Optional, an array of real numbers, contains values of D, parameter needed for tuning dn. See details.
`t`	Optional, an array of real numbers, contains values of t, parameter needed for tuning tn. See details.
`B`	Optional, the number of bootstrap samples. The default is 100.

Details

To this end, we generate B Bootstrap samples and for each pair of (dn, tn), we estimate the MSE by computing

$$\frac{\sum_{i=1}^{B}(\hat{\theta}_i(dn, tn) - inth)^2}{B}$$

where $\theta_i(dn, tn)$ is the estimator based on the i-th bootstrap sample.

For asymptotic efficiency of the estimators, a) dn$\to\infty$ and tn$\to 0$ b)

$$\frac{(dn)^2}{n^{1-\epsilon}(tn)^6} = O(1)$$

for some $\epsilon > 0$. We take dn to be the minimum of $\min(Dn^{1/2}(tn)^3$ and max(|x|)+2 (sd(x)*tn). Since dn is the truncation parameter and Stone (1975) uses kernel smoothening, we use the bound max(|x|)+2 (sd(x)tn). and we take tn to be the minimum between one and $tn^{-1/7}/100$. We generate a set of (dn, tn) by varying D and t acrros a grid. The default choice of the grid for t is c(0.01, seq(0.10, 0.80, by=0.05)) and that for D is (0.5, 1, 1.5, 2, 3, 4,....., 20).

Value

A list is returned:

- `param`: A 2 columns, the number of rows is same as the length of inth. Each row gives the optimal (dn, tn) for the corresponding inth value.
- `estimte`: Gives the estimators of θ using the optimal (dn, tn)'s.
- `default_estimate`:A

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Laha, N. *Location estimation fr symmetric and log-concave densities*. Submitted.

Stone, C. (1975). *Adaptive maximum likelihood estimators of a location parameter*, Ann. Statist., 3, 267-284.

See Also

`giveth`

`giveth`

Examples

```
{}
  @export
```

`stone.select.ww` *Tuning parameter selector for Stone (1975)'s location estimator*

Description

`giveth` computes the location estimator in a location shift model using Stone (1975)'s estimator. This function depends on tuning parameters dn and tn. We estimate the MSE of the estimators and to choose the best (dn,tn) from a set of options.

Usage

```
stone.select.ww(x, inth)
```

Arguments

`x`	An array of length n; the dataset.
`inth`	A vector of initial estimators
`D`	An array of real numbers, contains values of D, parameter needed for tuning dn. See details.
`t`	An array of real numbers, contains values of t, parameter needed for tuning tn. See details.

Details

To this end, we split the dataset in tan parts. Then we compute estimators $\hat{\theta}_i(dn, tn)$ for each (dn,tn) under consideration from the i-th part of the data, where i=1,...,10. We then estimate the MSE corresponding to each (dn, tn) by computing

$$\frac{\sum_{i=1}^{1} 0(\hat{\theta}_i(dn, tn) - \hat{\theta}(dn, tn))^2}{10}.$$

We choose the (dn, tn) pairs which minimize the estimated MSE.

For asymptotic efficiency of the estimators, a) dn$\rightarrow \infty$ and tn$\rightarrow 0$ b)

$$\frac{(dn)^2}{n^{1-\epsilon}(tn)^6} = O(1)$$

for some $\epsilon > 0$. We take $dn = Dn^{1/2}(tn)^3$ and $tn = tn^{-1/7}$. We generate a set of (dn, tn) by varying D and t in the set (0.5, 1, 1.5, 2, 3, 4,....., 20).

Value

A list is returned:

- `param`: A matrix with 2 columns, the number of rows is same as the length of inth. Each row gives the optimal (dn, tn) for the corresponding inth value.
- `estimte`: An array of same length as inth. Gives the estimators of θ using the optimal (dn, tn)'s.

Author(s)

Nilanjana Laha (maintainer), <nlaha@hsph.harvard.edu>.

References

Laha, N. *Location estimation fr symmetric and log-concave densities*. Submitted.

Stone, C. (1975). *Adaptive maximum likelihood estimators of a location parameter*, Ann. Statist., 3, 267-284.

See Also

giveth

Examples

```
x <- rnorm(100); inth <- mean(x);
 giveth(x, inth=inth)
 giveth(x)
```

Index