Package ‘unbalhaar’

February 20, 2015

Type Package
Title Function estimation via Unbalanced Haar wavelets
Version 2.0
Date 2010-08-09
Author Piotr Fryzlewicz
Maintainer Piotr Fryzlewicz <p.fryzlewicz@lse.ac.uk>
Description The package implements top-down and bottom-up algorithms
    for nonparametric function estimation in Gaussian noise using
    Unbalanced Haar wavelets.
License GPL-2
LazyLoad yes
Repository CRAN
Date/Publication 2012-10-29 08:59:57
NeedsCompilation no

R topics documented:

unbalhaar-package ............................................. 2
best.unbal.haar .............................................. 2
best.unbal.haar.bu .......................................... 3
hard.thresh .................................................... 4
hard.thresh.bu ............................................... 5
inner.prod.iter ............................................... 6
inner.prod.max ............................................... 7
inner.prod.max.p .......................................... 7
med ............................................................. 8
reconstr ...................................................... 9
reconstr.bu .................................................. 9
uh ............................................................. 10
uh.bu ........................................................ 11
unbal.haar.vector .......................................... 12

Index 13
unbalhaar-package  Function estimation via Unbalanced Haar wavelets

Description

The package implements top-down and bottom-up algorithms for nonparametric function estimation in Gaussian noise using Unbalanced Haar wavelets.

Details

<table>
<thead>
<tr>
<th>Package:</th>
<th>unbalhaar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>2.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2010-08-09</td>
</tr>
<tr>
<td>License:</td>
<td>GPL-2</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

The main functions of the package are uh and uh.bu.

Author(s)

Piotr Fryzlewicz
Maintainer: Piotr Fryzlewicz <p.fryzlewicz@lse.ac.uk>

References


Examples

```r
x <- c(rep(0, 100), rep(1, 200)) + rnorm(300)
est.topdown <- uh(x)
est.bottomup <- uh.bu(x)
```

best.unbal.haar  Best top-down Unbalanced Haar decomposition

Description

The function finds the “best” top-down Unbalanced Haar (UH) decomposition of the input vector `x`, according to a selection rule (criterion) which specifies which UH vector gets chosen at each scale and location.
Usage

```r
best.unbal.haar(x, criterion = inner.prod.max)
```

Arguments

- `x`: a vector
- `criterion`: a function which takes a vector of length `n` and returns an integer between 1 and `n-1`

Value

- `tree`: A list of J matrices, where J represents the number of "scales". Each matrix is of size 5 x (the number of UH coefficients at a given scale). Each column (= vector of length 5) contains an Unbalanced Haar coefficient in the following format: 1st component - an index of the coefficient; 2nd component - the value of the coefficient; 3rd component - time point where the corresponding UH vector starts; 4th component - last time point before the breakpoint of the UH vector; 5th component - end point of the UH vector.
- `smooth`: the "smooth" component of `x`, equal to `sum(x) / sqrt(n)`, where `n` is the length of `x`

Author(s)

Piotr Fryzlewicz

See Also

`inner.prod.max`, `inner.prod.max.p`, `best.unbal.haar.bu`

Examples

```r
best.unbal.haar(rnorm(100), inner.prod.max.p)
```

---

**Best bottom-up Unbalanced Haar decomposition**

Description

The function finds the “best” bottom-up Unbalanced Haar (UH) decomposition of the input vector `x`.

Usage

```r
best.unbal.haar.bu(x, stretch = length(x))
```
Arguments

x a vector

stretch at each iteration, only the first 1:stretch elements of the current input vector (whose length decreases by one with each iteration) get scanned in the search for the worst-fitting fine-scale Unbalanced Haar wavelet

Value

detail A matrix of size 3 x n-1, where n is the length of x, containing the detail coefficients of x in the order they were chosen. Each column corresponds to a single coefficient and contains, from top to bottom: location of the coefficient, the associated weight, and the value of the coefficient.

smooth the “smooth” component of x, equal to \( \frac{\text{sum}(x)}{\sqrt{n}} \), where n is the length of x

Author(s)

Piotr Fryzlewicz

See Also

best.unbal.haar

Examples

```r
best.unbal.haar.bu(rnorm(100))
```

hardtresh

Hard thresholding of a top-down Unbalanced Haar decomposition

Description

Presented with an object returned by best.unbal.haar, the function sets to zero those Unbalanced Haar coefficients which fall below a certain threshold sigma.

Usage

```r
hardtresh(buh, sigma = 1)
```

Arguments

buh an object returned by best.unbal.haar containing the decomposition to be thresholded

sigma the threshold (a positive scalar)
**Value**

a thresholded object, of the same class as `buh`

**Author(s)**

Piotr Fryzlewicz

**See Also**

`best.unbal.haar, hard.thresh.bu`

**Examples**

```r
x <- rnorm(1000)
x.uh <- best.unbal.haar(x)
x.uh.th <- hard.thresh(x.uh)
x.uh.th.r <- reconstr(x.uh.th)
hs.plot(x.uh.th.r)
```

---

**hard.thresh.bu**  
*Hard thresholding of a bottom-up Unbalanced Haar decomposition*

**Description**

Presented with an object returned by `best.unbal.haar.bu`, the function sets to zero those Unbalanced Haar coefficients which fall below a certain threshold `sigma`.

**Usage**

```r
hard.thresh.bu(buh.bu, sigma = 1)
```

**Arguments**

- `buh.bu`  
an object returned by `best.unbal.haar.bu` containing the decomposition to be thresholded
- `sigma`  
the threshold (a positive scalar)

**Value**

a thresholded object, of the same class as `buh.bu`

**Author(s)**

Piotr Fryzlewicz

**See Also**

`best.unbal.haar.bu, hard.thresh`
inner.prod.iter

Inner products with Unbalanced Haar wavelets

Description

For an input vector of length n, the function computes inner products between the input vector and all possible n-1 Unbalanced Haar vectors of length n.

Usage

inner.prod.iter(x)

Arguments

x a vector of length n

Details

The computation is iterative and is performed in computational time \(O(n)\).

Value

a vector of length n-1, containing inner products between x and consecutive Unbalanced Haar wavelets of length n

Author(s)

Piotr Fryzlewicz

Examples

inner.prod.iter(rnorm(100))

Examples

x <- rnorm(1000)
x.uh <- best.unbal.haar.bu(x)x.uh.th <- hard.thresh.bu(x.uh)x.uh.th.r <- reconstr.bu(x.uh.th)ts.plot(x.uh.th.r)
inner.prod.max

Unbalanced Haar wavelet which maximises the inner product

Description

The function finds the Unbalanced Haar vector which yields the largest (in absolute value) inner product with the input vector.

Usage

inner.prod.max(x)

Arguments

x

a vector

Value

The index where abs(inner.prod.iter(x)) is maximised. If two or more maxima are found, the med of their locations is returned.

Author(s)

Piotr Fryzlewicz

See Also

inner.prod.iter.med, inner.prod.max.p

Examples

inner.prod.max(c(rep(0, 100), rep(1, 200)))

inner.prod.max.p

Unbalanced Haar wavelet which maximises the inner product

Description

The function finds the Unbalanced Haar vector which yields the largest (in absolute value) inner product with the input vector, amongst those Unbalanced Haar vectors whose breakpoint is located between 100(1-p)% and 100p% of their support.

Usage

inner.prod.max.p(x, p = 0.8)
Arguments

\[ x \quad \text{a vector} \]

\[ p \quad \text{a scalar in } (0.5, 1] \]

Value

The index where \( \text{abs(inner.prod.iter}(x)) \) is maximised on the subinterval \((1+\text{floor}(1-p)*n):\text{ceiling}(p*n)\), where \( n \) is the length of \( x \). If two or more maxima are found, the \( \text{med} \) of their locations is returned.

Author(s)

Piotr Fryzlewicz

See Also

\emph{inner.prod.iter, med, inner.prod.max}

Examples

\[
\text{inner.prod.max.p(c(rep(0, 100), rep(1, 200)), .55)}
\]

\[
\text{med(1:4)}
\]

\[
\text{median(1:4)}
\]
reconstr

**Description**

Reconstructs a vector from its top-down Unbalanced Haar decomposition stored in an object returned by `best.unbal.haar` or `hard.thresh`.

**Usage**

```r
reconstr(buh)
```

**Arguments**

- **buh**: an object of the type returned by `best.unbal.haar` and `hard.thresh`.

**Value**

The inverse Unbalanced Haar transform of `buh`.

**Author(s)**

Piotr Fryzlewicz

**See Also**

- `best.unbal.haar`, `hard.thresh`, `reconstr.bu`

**Examples**

```r
x <- rnorm(1000)
x.uh <- best.unbal.haar(x)
x.uh.th <- hard.thresh(x.uh)
x.uh.th.r <- reconstr(x.uh.th)
ts.plot(x.uh.th.r)
```

---

**reconstr.bu**

**Reconstruct a bottom-up Unbalanced Haar decomposition**

**Description**

Reconstructs a vector from its bottom-up Unbalanced Haar decomposition stored in an object returned by `best.unbal.haar.bu` or `hard.thresh.bu`.

**Usage**

```r
reconstr.bu(buh.bu)
```
Arguments

buh.bu an object of the type returned by best.unbal.haar.bu and hard.thresh.bu

Value

the inverse Unbalanced Haar transform of buh.bu

Author(s)

Piotr Fryzlewicz

See Also

best.unbal.haar.bu, hard.thresh.bu, reconstr

Examples

x <- rnorm(1000)
x.uh <- best.unbal.haar.bu(x)
x.uh.th <- hard.thresh.bu(x.uh)
x.uh.th.r <- reconstr.bu(x.uh.th)
ts.plot(x.uh.th.r)

---

**uh**

*Denoising via top-down Unbalanced Haar*

Description

Given an input vector of the form “signal + iid Gaussian noise”, the function estimates the noise level via Median Absolute Deviation, finds the best top-down Unbalanced Haar decomposition (according to the selection rule criterion), thresholds it with the universal threshold, and performs the inverse Unbalanced Haar transform to yield an estimate of the signal.

Usage

uh(x, criterion = inner.prod.max)

Arguments

x a vector of the form “signal + iid Gaussian noise”
criterion a function which takes a vector of length n and returns an integer between 1 and n-1

Value

an estimate of the signal
uh.bu

**Author(s)**

Piotr Fryzlewicz

**References**


**See Also**

`uh.bu, best.unbal.haar, inner.prod.max, inner.prod.max.p, hard.thresh, reconstr`

**Examples**

```r
x <- c(rep(0, 100), rep(1, 200)) + rnorm(300)
est <- uh(x)
```

---

**uh.bu**

_Denoising via bottom-up Unbalanced Haar_

**Description**

Given an input vector of the form “signal + iid Gaussian noise”, the function estimates the noise level via Median Absolute Deviation, finds the best bottom-up Unbalanced Haar decomposition, thresholds it with the universal threshold, and performs the inverse Unbalanced Haar transform to yield an estimate of the signal.

**Usage**

```r
uh.bu(x, stretch = length(x))
```

**Arguments**

- `x`: a vector of the form “signal + iid Gaussian noise”
- `stretch`: at each iteration, only the first `1:stretch` elements of the current input vector (whose length decreases by one with each iteration) get scanned in the search for the worst-fitting fine-scale Unbalanced Haar wavelet

**Value**

an estimate of the signal

**Author(s)**

Piotr Fryzlewicz
References

See Also
uh, best.unbal.haar.bu, hard.thresh.bu, reconstr.bu

Examples
\[
x <- c(rep(0, 100), rep(1, 200)) + rnorm(300)
est <- uh.bu(x)
\]

---

unbal.haar.vector

*Unbalanced Haar vector*

Description
Computes the non-zero part of an Unbalanced Haar vector with a given start-, break- and end-point.

Usage
unbal.haar.vector(a)

Arguments
a
  a three-component vector of integers such that \(a[1] \leq a[2] < a[3]\). The three components specify, respectively, the start point, the time point just before the breakpoint, and the endpoint of the desired Unbalanced Haar vector.

Value
the non-zero part of the corresponding Unbalanced Haar vector

Author(s)
Piotr Fryzlewicz

Examples
unbal.haar.vector(c(1, 1, 2))
unbal.haar.vector(c(2, 5, 12))
Index

*Topic math
  best.unbal.haar, 2
  best.unbal.haar.bu, 3
  hard.thresh, 4
  hard.thresh.bu, 5
  inner.prod.iter, 6
  inner.prod.max, 7
  inner.prod.max.p, 7
  med, 8
  reconstr, 9
  reconstr.bu, 9
  uh, 10
  uh.bu, 11
  unbal.haar.vector, 12

*Topic package
  unbalhaar-package, 2

best.unbal.haar, 2, 4, 5, 9, 11
best.unbal.haar.bu, 3, 3, 5, 10, 12

hard.thresh, 4, 5, 9, 11
hard.thresh.bu, 5, 5, 10, 12

inner.prod.iter, 6, 7, 8
inner.prod.max, 3, 7, 8, 11
inner.prod.max.p, 3, 7, 7, 11

med, 7, 8, 8

reconstr, 9, 10, 11
reconstr.bu, 9, 9, 12

uh, 2, 10, 12
uh.bu, 2, 11, 11
unbal.haar.vector, 12
unbalhaar (unbalhaar-package), 2
unbalhaar-package, 2