Package ‘quantsmooth’

November 21, 2016

Type Package
Title Quantile smoothing and genomic visualization of array data
Version 1.40.0
Date 2014-10-07
Author Jan Oosting, Paul Eilers, Renee Menezes
Maintainer Jan Oosting <j.oosting@lumc.nl>
Depends R(>= 2.10.0), quantreg, grid
Description Implements quantile smoothing as introduced in: Quantile
smoothing of array CGH data; Eilers PH, de Menezes RX;
License GPL-2
biocViews Visualization, CopyNumberVariation
NeedsCompilation no

R topics documented:

  chrom.bands .......................... 2
  Chromosome14 .......................... 3
  drawSimpleChrom ....................... 3
  getChangedRegions ..................... 4
  getLambdaMin ......................... 5
  grid.chromosome ...................... 6
  lengthChromosome ..................... 7
  numericCHR ............................ 8
  paintCytobands ....................... 9
  plotChromosome ...................... 10
  plotSmoothed ......................... 11
  position2Cytoband .................... 12
  prepareGenomePlot .................... 13
  quantsmooth ......................... 14
  quantsmooth.cv ....................... 14
  quantsmooth.seg ...................... 15
  scaleto .............................. 16

Index 17
Dataset of human chromosomes and their banding patterns

**Description**

Dataset used to produce human chromosomal ideograms for plotting purposes.

**Usage**

```r
data(chrom.bands)
```

**Format**

A data frame with 4068 observations on the following 12 variables.

- `chr` a character vector
- `arm` a character vector
- `band` a character vector
- `ISCN.top` a numeric vector
- `ISCN.bot` a numeric vector
- `bases.top` a numeric vector
- `bases.bot` a numeric vector
- `stain` a character vector
- `cM.top` a numeric vector
- `cM.bot` a numeric vector
- `n.markers` a numeric vector
- `p.markers` a numeric vector

**Details**

The original file gives only the physical map positions. The genetic map positions are interpolated from the Rutgers linkage map (Kong et al 2004).

**Source**


**References**

Chromosome 14

Example data from several quantitative genomic methods

Description

A collection of arrays that contains data of chromosome 14 of 3 colorectal tumors. The first tumor shows 1 region of loss, the second tumor shows no aberration, while the third tumor shows loss of 1 copy of the chromosome.

affy.cn Copy number values of 358 probes from Affymetrix 10K genechip. Data was obtained from DChip

affy.pos corresponding probe positions

bac.cn Copy number values of 112 probes from a 1 mb spaced BAC array-CGH

bac.pos corresponding probe positions

ill.cn Copy number values of 207 probes from Illumina GoldenGate Linkage IV data

ill.pos corresponding probe positions

Usage

data(chr14)

Format

Matrices of copy number values and vectors of chromosomal probe positions

Author(s)

Jan Oosting

drawSimpleChrom

Draw chromosome-like icons

Description

This function paints chromosomal icons on an existing plot

Usage

drawSimpleChrom(x, y, len = 3, width = 1, fill, col, orientation = c("h", "v"), centromere.size = 0)
getChangedRegions

Arguments

- **x**: start x-position
- **y**: start y-position
- **len**: total length of the chromosome
- **width**: width of the chromosome
- **fill**: character, {"a", "p", "q", "q[1-3]", "p[1-3]"}. Events to a chromosome can be depicted by coloring "a"ll of the chromosome, the complete p or q-arm, or a subsegment of the arms
- **col**: color(s) of fill
- **orientation**: either "h"orizontal or "v"ertical
- **centromere.size**: The size of the centromere as fraction of the width

Value

This function is executed for its side effects

Author(s)

Jan Oosting

Examples

```r
plot(c(0,4),c(0,3),type="n",xaxt="n",yaxt="n",xlab="",ylab="")
drawSimpleChrom(2,3,fill=c("p","q3"),col=c("red","blue"),orientation="v")
```

Description

retrieve regions of interest in a vector of intensities using quantile smoothing

Usage

```r
getChangedRegions(intensities, positions, normalized.to=1, interval, threshold, minlength=2, ...)
```

Arguments

- **intensities**: numeric vector
- **positions**: numeric vector of the same length as intensities. If this argument is not given the results contain the indexes of the intensities vector, else the values in positions are used. Both vectors are sorted in the order of positions.
- **normalized.to**: numeric, reference value. Changes are compared to this value
- **interval**: numeric [0,1], bandwidth around reference. If the smoothed line at the higher quantile drops below the normalized.to value, a deleted region is recognized, and vice versa.
- **threshold**: numeric, if the median smoothed value drops below normalized.to - threshold, or above normalized.to + threshold a changed region is called
- **minlength**: integer, not used currently
- **...**: extra arguments for quantsmooth function
getLambdaMin

Details

This function uses quantsmooth to detect regions in the genome that are abnormal. If interval is set then a smoothed line is calculated for \( \tau = 0.5 - \text{interval}/2 \), and a region is determined as upregulated if this line is above the reference. Down regulation is determined when the smoothed line for \( \tau = 0.5 + \text{interval}/2 \) is below the reference value. If threshold is set then a smoothed line is calculated for \( \tau = 0.5 \) and up- or down regulation are determined when this line is outside the range \([\text{normalized} \cdot \tau - \text{threshold}; \text{normalized} \cdot \tau + \text{threshold}]\)

Value

A data.frame with 3 columns is returned. Each row contains a region with columns up, start and end. start and end indicate positions in the vector of the first and last position that were up- or downregulated

Author(s)

Jan Oosting

Examples

data(chr14)
getChangedRegions(ill.cn[,1],ill.pos,normalized.to=2,interval=0.5)

getLambdaMin

Description

Test a set of smoothing parameters to find best fit to data

Usage

getLambdaMin(intensities,lambdas,...)

Arguments

intensities numeric vector
lambdas numeric vector; see quantsmooth
... extra parameters for quantsmooth.cv; currently only ridge.kappa

Details

Cross validation is performed using a set of lambda values in order to find the lambda value that shows the best fit to the data.

Value

This function returns the lambda value that has the lowest cross validation value on this dataset

Author(s)

Jan Oosting
grid.chromosome

See Also

quantsmooth.cv

Examples

data(chr14)
lambdas<-2^seq(from=-2,to=5,by=0.25)
getLambdaMin(bac.cn[,1],lambdas)

grid.chromosome

Draw a chromosome using the grid package

Description

A chromosome is drawn including the cytobands

Usage

grid.chromosome(chrom, side = 1, units = "hg19", chrom.width = 0.5, length.out,
bands = "major", legend = c("chrom", "band", "none"), cex.leg = 0.7, bleach = 0, ...)

Arguments

chrom numeric or character, id of chromosome to plot
side numeric [1:4], side of rectangle to draw, 4 sides, side 2 and 4 are vertical
units character or data.frame, type of units for genomic data, or a dataframe with
UCSC cytoband data, see lengthChromosome
chrom.width numeric [0,1], The width relative to the width (sides 2 and 4) or height(sides 1
and 3) of the viewport
length.out numeric, size of native units of viewport
bands character, draw either major or minor bands
legend character, type of legend
cex.leg numeric, relative size of legend text
bleach numeric [0,1], proportion by which to bleach the chromosome
... arguments for viewport(), especially x,y, width, and height

Details

The chromosome is drawn within a rectangle defined by x, y, width, and height, which is pushed as
a viewport. The legend is drawn within the same rectangle in the space left over by chrom.width.

Value

This function is executed for its side effects

Author(s)

David L. Duffy, Jan Oosting
lengthChromosome

References

lodplot package

See Also

paintCytobands

Examples

grid.newpage()
grid.chromosome(1, units = "bases", height = 0.15)

lengthChromosome

Retrieve chromosomal length

Description

Retrieve human chromosomal length from NCBI data

Usage

lengthChromosome(chrom, units = "hg19")

Arguments

chrom vector of chromosomal id, 1:22,X,Y
units character, or data.frame, see details

Details

The cytoband data was originally obtained from the lodplot package by David Duffy, which contained basepair data from genome version hg17, but also the linkage related positions in cM. These datasets have units "bases" and "cM" respectively. Cytoband data for genome versions "hg18", "hg19", "hg38" and "mm10" has been included, and can be referenced by these strings. It is also possible to use cytoband data as obtained from the UCSC site, by downloading the cytoBand.txt.gz or cytoBandIdeo.txt.gz annotation file for a species (see example below). Note however that this information is not available for most species.

Value

A numeric vector in the requested units

Author(s)

Jan Oosting
Examples

# Show length of chromosome 1 in several types of units
lengthChromosome(1,"cM")
lengthChromosome(1,"bases")
lengthChromosome(1,"hg38")

# mm9 cytoband data
temp <- tempfile(fileext = ".txt.gz")
download.file("http://hgdownload.soe.ucsc.edu/goldenPath/mm9/database/cytoBand.txt.gz",temp)
mm9cytobands <- read.table(temp,sep="\t")
lengthChromosome(1,mm9cytobands)

# remove temp file
unlink(temp)

numericCHR
Conversion of chromosome IDs between numeric and character

Description
The function converts chromosomal ids to their numeric form, and the sex chromosomes to values between 98 and 100. This simplifies sorting on chromosome ID

Usage
numericCHR(CHR, prefix="chr")
characterCHR(CHR, prefix="")

Arguments

CHR
character/numeric vector for both functions the mode of the input is not forced.
For numericCHR strings "X","Y" and "XY" are converted to 98,99 and 100 respectively.

prefix
character, string is excluded from (numericCHR) or prepended to (characterCHR) all items of the output

Value
numericCHR returns a numeric vector of same length as CHR characterCHR returns a character vector of same length as CHR

Author(s)
Jan Oosting

Examples

chaps<-c("3","2","8","X","7","Y","5","1","9","10","11","12","4","6")
sort(chaps)
sort(numericCHR(chaps))
characterCHR(sort(sort(numericCHR(chaps))),prefix="chr")
**paintCytobands**

**Description**

Paints a human chromosomal idiogram in an existing plot. Adapted from the paint.chromosome function in the lodplot package by David L. Duffy.

**Usage**

```r
paintCytobands(chrom, pos = c(0, 0), units = "hg19", width = 0.4,
               length.out, bands = "major", orientation = c("h","v"), legend = TRUE,
               cex.leg = 0.7, bleach = 0, ...)```

**Arguments**

- **chrom**: chromosomal id, chromosome to plot 1:22,X,Y.
- **pos**: numeric vector of length 2, position in the plot to start the plot.
- **units**: character or data.frame, type of units for genomic data, or a dataframe with UCSC cytoband data, see `lengthChromosome`.
- **width**: numeric, width of the chromosome, the chromosome is plotted between `pos[2]` and `pos[2]-width`.
- **length.out**: numeric, if given, the chromosome will have this length in the plot.
- **bands**: if not equal to "major", then also the minor bands will be plotted.
- **orientation**: chromosome is plotted either Horizontally to the right of the starting point or Vertically down from the starting point.
- **legend**: logical, if TRUE then the bandnames are plotted next to the chromosome.
- **cex.leg**: numeric, relative size of legend text.
- **bleach**: numeric [0,1], proportion by which to bleach the chromosome.
- ... extra parameters for `plot`.

**Value**

This function is executed for its side effects.

**Author(s)**

David L. Duffy, Jan Oosting.

**References**

lodplot package

**Examples**

```r
plot(c(0,lengthChromosome(14,"bases")),c(-2,2),type="n",xaxt="n",yaxt="n",xlab="",ylab="")
paintCytobands(14,units="bases")```
plotChromosome  

Wrapper for plotSmoothed

Description

This function is a wrapper for plotSmoothed, to make data subsetting easier

Usage

plotChromosome(gendata, chrompos, chromosome, dataselection = NULL, ylim = NULL, normalized.to = NULL, grid = NULL, smooth.lambda = 2, interval = 0.5, ...)

Arguments

gendata  numeric matrix or data.frame
chrompos chrompos object with same number of rows as gendata
chromosome numeric, chromosome to show
dataselection optional, subset of samples/columns in gendata
ylim limits for plot
normalized.to y-value(s) for line
grid x-value(s) for line
smooth.lambda smoothing parameter, see quantsmooth
interval position of extra lines besides median, see plotSmoothed
... extra arguments for plotSmoothed

Value

The function is used for its side effects

Author(s)

Jan Oosting

See Also

plotSmoothed, quantsmooth
plotSmoothed

Description

Plot a smoothed line together with the original data values

Usage

plotSmoothed(intensities, position, ylim=NULL, ylab="intensity", xlab="position", normalized.to=NULL, grid=NULL, smooth.lambda=2, interval=0.5, plotnew=TRUE, cols, cex.pts = 0.6, ...)

Arguments

- `intensities`: numeric vector or matrix, data are plotted by column
- `position`: numeric vector; the length should be the number of rows in `intensities`
- `ylim`: numeric vector of length 2, limits for plot. If `NULL` then the minimal and maximal value in `intensities` is used
- `ylab`: character, label for y-position
- `xlab`: character, label for x-position
- `normalized.to`: numeric, a line(s) is drawn at this horizontal position
- `grid`: numeric, a line(s) is drawn at this vertical position
- `smooth.lambda`: numeric, smoothing parameter see `quantsmooth`
- `interval`: numeric (0..1), plotting of extra smoothed lines around median. With `interval = 0.5` the 0.25 and 0.75 quartiles are plotted, with `interval = 0.9` the 0.05 and 0.95 quantiles are plotted,
- `plotnew`: logical, if TRUE a new plot is created, else the data are plotted into an existing plot
- `cols`: color vector, colors for columns in `intensities`
- `cex.pts`: size of the dots in the plot. Set to 0 to skip plotting the dots
- `...`: extra parameters for `plot`

Details

This function plots the raw data values as dots and the median smoothed values as a continuous line. If `interval` is supplied these are plotted as lines in different line types. More than 1 `interval` can be given.

Value

This function is used for its side effects

Author(s)

Jan Oosting

See Also

`quantsmooth`
position2CytoBand

**Description**

Determine cytoband position based on location of probe

**Usage**

`position2CytoBand(chrom, position, units = "hg19", bands = c("major", "minor"))`

**Arguments**

- `chrom`: chromosomal id, chromosome to plot 1:22, X, Y
- `position`: numeric vector
- `units`: character, type of positional unit
- `bands`: character, type of cytoband

**Value**

Character vector with cytobands, if an illegal position was used, the value "-" is returned. All positions within a single function call should be for a single chromosome

**Author(s)**

Jan Oosting

**See Also**

`lengthChromosome`

**Examples**

```r
position2CytoBand(1, c(50e6, 125e6, 200e6), units = "bases")
position2CytoBand(1, c(50, 125, 200), units = "cM", bands = "minor")
```
**prepareGenomePlot**  

*Set up a full genome plot*

**Description**

This function starts up a plot consisting of all chromosomes of a genome, including axes with chromosome names.

**Usage**

```r
prepareGenomePlot(chrompos, cols = "grey50", paintCytobands = FALSE, bleach = 0, topspace = 1, organism, sexChromosomes = FALSE, units = "hg19",...)
```

**Arguments**

- `chrompos`: chrompos object, data.frame with CHR column identifying the chromosome of probes, and a MapInfo column identifying the position on the chromosome
- `cols`: color(s) for the chromosome lines
- `paintCytobands`: logical, use paintCytoband to plot ideograms for all chromosomes
- `bleach`: numeric [0,1], proportion by which to bleach the ideograms
- `topspace`: numerical, extra space on top of plot, i.e. for legends
- `organism`: character, if given a 2 column plot is created with the chromosomes for the given species. Currently "hsa", "mmu", and "rno" are supported
- `sexChromosomes`: logical, if TRUE then also the sex chromosomes X and Y are plotted
- `units`: character or data.frame, type of units for genomic data, or a dataframe with UCSC cytoband data, see `lengthChromosome`
- `...`: extra arguments for `plot` function

**Details**

If `organism` is not supplied then a single column is plotted of the available chromosomes in `chrompos$CHR`. The arguments `paintCytobands`, `bleach`, and `sexChromosomes` are not used in that case. If `organism` is supplied and `chrompos` is NULL then a result is generated with the starting Y and X position of each chromosome

**Value**

A matrix with 2 columns that contain the Y and X positions for the probes on the plot

**Author(s)**

Jan Oosting
Description
Quantile smoothing of array data

Usage
quantsmooth(intensities, smooth.lambda=2, tau=0.5, ridge.kappa=0, smooth.na=TRUE, segment)

Arguments
- intensities: numeric vector
- smooth.lambda: numeric
- tau: numeric [0..1], the quantile desired; see rq.fit
- ridge.kappa: fudge parameter; see details
- smooth.na: logical; handling of NA
- segment: integer, length of overlapping segments

Value
This function returns a vector of the same length as intensities, or a matrix if the length of tau is greater than 1.

Author(s)
Jan Oosting

Examples
data(chr14)
plot(quantsmooth(bac.cn[,1], smooth.lambda=2.8), type="l")

Description
Cross validation of smoothing parameters

Usage
quantsmooth.cv(intensities, smooth.lambda=2, ridge.kappa=0)

Arguments
- intensities: numeric vector
- smooth.lambda: numeric; see quantsmooth
- ridge.kappa: fudge parameter; see quantsmooth
Details

Cross validation is performed by calculating the fit from the even indices on the odd indices and vice versa.

Value

This function returns the sum of squared differences or NA if the fitting function gave an error.

Author(s)

Jan Oosting

See Also

getLambdaMin

Examples

```r
data(chr14)
# A low value is indicative of a better fit to the data
quantsmooth.cv(bac.cn[,1],1)
quantsmooth.cv(bac.cn[,1],2.8)
```

Description

segmented Quantile smoothing of array data

Usage

```r
quantsmooth.seg(y, x = 1:length(y), lambda = 2, tau = 0.5, kappa = 0, nb = length(x))
```

Arguments

- `y`: numeric vector
- `x`: numeric vector of same length as `y`. Position of values
- `lambda`: numeric
- `tau`: numeric [0..1], the quantile desired; see `rq.fit`
- `kappa`: fudge parameter; see details
- `nb`: integer, basis

Value

This function returns a vector of the same length as `y`

Author(s)

Jan Oosting
Examples

```r
data(chr14)
plot(quantsmooth.seg(bac.cn[,1],lambda=2.8,nb=50),type="l")
```

**scaleto**  
Scales data within a range to a new range

**Description**

This function scales data to a new range while enforcing the boundaries. This can be helpful in preventing overlap between chromosomal plots that display multiple chromosomes in the same plot.

**Usage**

```r
scaleto(x, fromlimits = c(0, 50), tolimits = c(0.5, -0.5), adjust = TRUE)
```

**Arguments**

- `x`: numeric
- `fromlimits`: numeric vector with length 2, original range of data
- `tolimits`: numeric vector with length 2, target range of data
- `adjust`: logical, if TRUE then the target values are clipped to the target range

**Value**

numeric of same size as `x`

**Author(s)**

Jan Oosting
Index

* Topic **aplott**
  - drawSimpleChrom, 3
  - grid.chromosome, 6
  - paintCytobands, 9

* Topic **attribute**
  - getChangedRegions, 4

* Topic **datasets**
  - chrom.bands, 2
  - Chromosome14, 3

* Topic **data**
  - lengthChromosome, 7

* Topic **hplot**
  - plotChromosome, 10
  - plotSmoothed, 11
  - prepareGenomePlot, 13

* Topic **htest**
  - quantsmooth.cv, 14

* Topic **manip**
  - numericCHR, 8
  - position2Cytoband, 12
  - scaleto, 16

* Topic **smooth**
  - getLambdaMin, 5
  - quantsmooth, 14
  - quantsmooth.cv, 14
  - quantsmooth.seg, 15

affy.cn (Chromosome14), 3
affy.pos (Chromosome14), 3
bac.cn (Chromosome14), 3
bac.pos (Chromosome14), 3
characterCHR (numericCHR), 8
chr14 (Chromosome14), 3
chrom.bands, 2
Chromosome14, 3
drawSimpleChrom, 3
getChangedRegions, 4
getLambdaMin, 5, 15
grid.chromosome, 6
ill.cn (Chromosome14), 3
ill.pos (Chromosome14), 3
lengthChromosome, 6, 7, 9, 12, 13
numericCHR, 8
paintCytobands, 7, 9
plotChromosome, 10
plotSmoothed, 10, 11
position2Cytoband, 12
prepareGenomePlot, 13
quantsmooth, 5, 10, 11, 14, 14
quantsmooth.cv, 6, 14
quantsmooth.seg, 15
rq.fit, 14, 15
scaleto, 16