

# encoDnaseI

October 5, 2010

---

ALICOR

*aligned and interpolated correlation for local maximum traces for two scatterplots*

---

## Description

aligned and interpolated correlation for local maximum traces from two scatterplots

## Usage

```
alicor(x1, y1, x2, y2, bin = 50000)
ALICOR(ssr, dns = rawCD4, bin = 50000)
```

## Arguments

x1	domain points for first scatterplot
y1	range points for first scatterplot
x2	domain points for second scatterplot
y2	range points for second scatterplot
ssr	an instance of class <code>snpScreenResult</code>
dns	an instance of class <code>hg18track</code>
bin	bin size, units are base pairs

## Details

We define a scatterplot to be a pair of vectors  $(x, y)$ . We are interested in measuring the distance between two scatterplots, focusing on the locations of local peaks and valleys. Two scatterplots are close if their peaks and valleys in  $y$  are nearby in  $x$ .

We have no restrictions on commonalities between the scatterplots, but this only makes sense if there is reasonable overlap between their  $x$  ranges.

The algorithm, implemented in low-level function `alicor`, is as follows. Use parameter `bin` to define a grid in  $x$  for each scatterplot, and compute the maximum  $y$  value in each  $x$ -grid interval. Compute a common domain for the two scatterplots based solely on  $x1$ . Linearly interpolate the maximal series for each scatterplot on the common domain. Compute the correlation coefficient for the resulting interpolated series.

The higher-level function `ALICOR` adapts this to `snpScreenResult` `plot_mlp` display data and to `hg18` annotation track data.

**Value**

scalar correlation coefficient

**Note**

Many variations on this algorithm are possible, but the code is not very flexible at this time.

**Author(s)**

VJ Carey <stvjc@channing.harvard.edu>

**Examples**

```
data(sOSR2)
data(c19g)
juxtaPlot(c19g, sOSR2)
ALICOR(sOSR2, c19g)
```

---

rawCd4DnaseI

*A data frame with information on the UCSC browser track related to DNaseI hypersensitivity*

---

**Description**

A data frame with information on the UCSC browser track related to DNaseI hypersensitivity; the rawCD4 object is an eSet extension representing the same information; rawHelaDnaseI is like rawCD4 but results on Hela cells.

**Usage**

```
data(rawCd4DnaseI)
data(rawHelaDnaseI)
data(rawCD4)
```

**Details**

Obtained from a MySQL representation of the data distributed at the Genome Browser FTP site

**Value**

a data.frame

**Author(s)**

Vince Carey <stvjc@channing.harvard.edu>

**References**

[hgdownload.cse.ucsc.edu](http://hgdownload.cse.ucsc.edu) ... it appears that they do not offer the MYD/MYI representations, just the txt.gz and sql files now. So if you obtain the encodeNhgriDnaseHsChipRawCd4.txt and .sql files at goldenPath/currentGenomes/Homo\_Sapiens/encode/database, you can reconstruct the underlying data for this data.frame (hg18, Nov 2007).

**Examples**

```

data(rawCd4DnaseI)
dim(rawCd4DnaseI)
rawCd4DnaseI[1:5,]
library(lattice)
xyplot(dataValue~chromStart|chrom, data=rawCd4DnaseI, subset=chrom %in%
       c("chr1", "chr10", "chr19", "chr20"), scales=list(x=list(relation="free")))

```

---

hg18track-class	<i>Class "hg18track" container for hg18 annotation found in genome browser track files; class "chrnum" extends numeric for an indexing application on hg18track objects.</i>
-----------------	--

---

**Description**

container for hg18 annotation found in genome browser track files

**Objects from the Class**

Objects can be created by calls of the form `new("hg18track", assayData, featureData, experimentData, annotation, dataVals, ...)`. These are single-sample `eSet` instances.

Note that `demoTrk19` is a restriction of the `rawCD4` structure to the interval of chromosome 19 that was assayed in the ENCODE project for DnaseI hypersensitivity.

**Slots**

```

assayData: Object of class "AssayData" ~~
phenoData: Object of class "AnnotatedDataFrame" ~~
featureData: Object of class "AnnotatedDataFrame" ~~
experimentData: Object of class "MIAME" ~~
annotation: Object of class "character" ~~
.__classVersion__: Object of class "Versions" ~~

```

**Extends**

Class `"eSet"`, directly. Class `"VersionedBiobase"`, by class `"eSet"`, distance 2. Class `"Versioned"`, by class `"eSet"`, distance 3.

**Methods**

```

[ signature(x = "hg18track"): select using numeric, logical, or chrnum indices.
chrnum signature(object = "hg18track"): extract numeric tokens for chromosome
  number at which data values are obtained; note that chrnum is also used as name of a class.
dataVals signature(object = "hg18track"): actual data values
getTrkXY signature(object = "hg18track", type = "character"): obtain a
  list with components x, y indicating location and data value respectively; location is within
  chromosome; default type is 'midpoint' of locations given as intervals

```

**getTrkXY** signature(object = "hg18track", type = "missing"): take default midpoint x values corresponding to data values

**rangeLocs** signature(object = "hg18track"): if measures from only one chromosome are present, this returns low and high values of chromStart and chromEnd respectively, otherwise error.

**clipTrk** signature(obj = "hg18track", low="numeric", hi="numeric", attr="ANY"): create a restriction of the track using an interval specification. by default the chromStart featureData component is used for coordinates to clip; if attr is non-missing, the featureData component named by attr will be used.

**initialize** signature(.Object = "hg18track"): create a new instance

### Author(s)

VJ Carey <stvjc@channing.harvard.edu>

### Examples

```
showClass("hg18track")
data(rawCD4)
rawCD4
rawCD4.chr1 = rawCD4[ chrnum(1), ]
rangeLocs(rawCD4.chr1)
plot(getTrkXY(rawCD4.chr1), ylab="data value", xlab="interval midpt on chr 1" )
c52 = clipTrk(rawCD4[ chrnum(5), ], 1.30e8, 1.33e8 )
plot(getTrkXY(c52))
```

---

juxtaPlot

*two-panel plot with track info and snp screen t-values*

---

### Description

two-panel plot with track info and snp screen t-values

### Usage

```
juxtaPlot(trk, ssr)
```

### Arguments

trk	instance of <a href="#">hg18track</a>
ssr	instance of <a href="#">GGtools snpScreenResult</a>

### Details

xyplot of lattice package is used.

### Value

xyplot output; use print in Sweave.

**Author(s)**

VJ Carey <stvjc@channing.harvard.edu>

**Examples**

```
data(sOSR2)
data(c19g)
juxtaPlot(c19g, sOSR2)
```

# Index

## \*Topic classes

hg18track-class, 3

## \*Topic models

ALICOR, 1

juxtaPlot, 4

rawCd4DnaseI, 2

[, hg18track-method  
(hg18track-class), 3

ALICOR, 1

alicor (ALICOR), 1

chrnum (hg18track-class), 3

chrnum, hg18track-method  
(hg18track-class), 3

chrnum, numeric-method  
(hg18track-class), 3

chrnum-class (hg18track-class), 3

chromStarts (hg18track-class), 3

chromStarts, hg18track-method  
(hg18track-class), 3

clipTrk (hg18track-class), 3

clipTrk, hg18track, numeric, numeric, ANY-method  
(hg18track-class), 3

dataVals (hg18track-class), 3

dataVals, hg18track-method  
(hg18track-class), 3

demoPGM1ssr (ALICOR), 1

demoTrk19 (hg18track-class), 3

eSet, 3

getTrkXY (hg18track-class), 3

getTrkXY, hg18track, character-method  
(hg18track-class), 3

getTrkXY, hg18track, missing-method  
(hg18track-class), 3

hg18track, 1, 4

hg18track-class, 3

initialize, hg18track-method  
(hg18track-class), 3

juxtaPlot, 4

locmax (ALICOR), 1

makeHg18track (hg18track-class), 3

rangeLocs (hg18track-class), 3

rangeLocs, hg18track-method  
(hg18track-class), 3

rawCD4 (rawCd4DnaseI), 2

rawCd4DnaseI, 2

rawHelaDnaseI (rawCd4DnaseI), 2

Versioned, 3

VersionedBiobase, 3

WRC (ALICOR), 1

wrc (ALICOR), 1