

Package ‘haploReconstruct’

October 10, 2016

Type Package

Title Reconstruction of Haplotype-Blocks from Time Series Data

Version 0.1.2

Author Susanne U. Franssen, Nicholas H. Barton, Christian Schloetterer

Maintainer Susanne U. Franssen <susefranssen@googlemail.com>

Depends R (>= 3.2.2), data.table

Description Reconstruction of founder haplotype blocks from time series data.

Imports igraph, foreach, zoo, stringi, matrixStats, gplots, stats,
dbscan, methods

License GPL-3

LazyData TRUE

Encoding UTF-8

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-10-10 18:18:49

R topics documented:

ex_dat	2
hbr-class	4
initialize_SNP_time_series	5
inspect_window,hbr-method	7
inspect_window_avLink,hbr-method	8
inspect_window_dbScan,hbr-method	9
inspect_window_PCA,hbr-method	10
map,hbr-method	11
markers,hbr-method	12
number_hbr,hbr-method	12
plot,hbr-method	13
plot_cluster_trajectories,hbr-method	14

plot_hbr_freq,hbr-method	15
plot_marker_trajectories,hbr-method	16
reconstruct_hb,SNP_time_series-method	17
rev_map,hbr-method	19
show,hbr-method	20
show,SNP_time_series-method	20
SNP_time_series-class	21
summary,hbr-method	22
sync_to_frequencies	23

Index	25
--------------	-----------

ex_dat	<i>Example data set</i>
--------	-------------------------

Description

Example data set including the simulated data set with selection operating on two different sites 1 Mb apart from each other, each unique to a single but different out of 200 different founder haplotypes (simulated data corresponding to Fig. 1A in Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)). Data was simulated in an evolve and resequence setup using 5 replicates and time points up to generation F60 with sampled time-points every 20th generation. The data as a data.table containing already allele frequencies polarized for the minor allele in the founder populations as can be obtained by the [sync_to_frequencies](#) function from a sync formatted text file (sync format, see Kofler et al. 2011). A basic workflow for haplotype analysis is presented below.

Usage

```
data(ex_dat)
```

Format

data.table containing frequency information of all samples polarized for the minor allele in the experimental founder population

Author(s)

Susanne U. Franssen

References

Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)

Examples

```

# The following workflow provides a general example on
# haplotype analysis on the provided example data set
#
# example data was previously formatted from a sync file with an added header using:
# ex_dat=sync_to_frequencies(file="ex_dat.sync",
#                           base.pops=rep(c(TRUE,rep(FALSE,3)),times=5), header=TRUE)
# The file contains samples for F0, F20, F40 and F60, each for five replicates simulations.

# filter replicated time series data for informative SNPs
dat_filtered=initialize_SNP_time_series(chr=ex_dat$chr, pos=ex_dat$pos,
base.freq=ex_dat$basePops, lib.freqs=ex_dat[,7:ncol(ex_dat), with=FALSE],
pop.ident=rep(1:5,each=4), pop.generation=rep(c(0:3)*20,times = 5), use.libs=rep(TRUE,20))

# reconstruct haplotype-blocks
dat_reconst=reconstruct_hb(dat_filtered, chrom="2R")

# various ways of inspecting the results
#
?plot.hbr
plot(dat_reconst, indicate_shared=TRUE, addPoints=TRUE)
#
#?summary.hbr
summary(dat_reconst)
#
?plot_hbr_freq
par(mfrow=c(2,1),mar=c(4,4,1,1),oma=c(0,0,0,0))
plot_hbr_freq(dat_reconst, hbr_id=2, replicate=1, timepoint=c(0,20,40,60), window=5)
plot_hbr_freq(dat_reconst, hbr_id=2, replicate=2, timepoint=c(0,20,40,60), window=5)
# Note: For the example parameter settings reconstructed haplotype-block hbr_id=2
# corresponds to the focal selected haplotype shown in Fig. 1A
# (Franssen, Barton & Schloetterer 2016,
# Reconstruction of haplotype-blocks selected during experimental evolution, MBE).
#
?map
map(dat_reconst)
#
?rev_map
rev_map(dat_reconst)
#
?plot_cluster_trajectories
plot_cluster_trajectories(dat_reconst, window=38)
#
?plot_marker_trajectories
plot_marker_trajectories(dat_reconst, hbr_id=2)
#
?inspect_window
inspect_window(dat_reconst, window=38)
#
?inspect_window_PCA
inspect_window_PCA(dat_reconst, window=38)
#

```

```

?inspect_window_avLink
inspect_window_avLink(dat_reconst, window=38)
#
?inspect_window_dbScan
inspect_window_dbScan(dat_reconst, window=38, eps=1)
#
?markers
markers(dat_reconst, hbr_id=2)
#
?number_hbr
number_hbr(dat_reconst)

```

hbr-class

An S4 class storing results from haplotype-block reconstruction

Description

An S4 class containing input parameters and results for reconstructed haplotype-blocks

Details

An S4 class containing a `SNP_time_series` data object, the haplotype-blocks were constructed on, parameters that were used for haplotype-block reconstruction and results of haplotype block reconstruction. An `hbr` object can only be created with [reconstruct_hb](#).

Slots

`dat` A `SNP_time_series` object containing the time series data and threshold used for marker filtering (refer to `SNP_time_series` for more details)

`chromosome` The chromosome for which haplotype-blocks were calculated.

`min.cl.size` Numeric specifying the minimum number of correlated markers in a window for haplotype-block reconstruction

`min.cl.cor` Numeric specifying the correlation between marker SNPs required using average linkage clustering for markers to be assembled in one cluster

`min.inter` Numeric specifying the minimum number of markers in two clusters of overlapping windows required to be identical for cluster elongation across windows to build haplotype-blocks

`single.win` Boolean specifying that are supported by only a cluster in one window are also included in the markers of reconstructed blocks. If `FALSE` only `hbrs` are returned that span at least two windows and only markers being present in the intersection between overlapping windows are included.

`transf` Boolean indicating if time series data was sqrt transformed prior to clustering.

`scaleSNP` Boolean indicating if time series data was scaled (mean=0, var=1) for each SNP prior to clustering.

`pos.cor` Boolean indicating if negative correlations between time series between two SNPs were set to zero prior to clustering.

- `c1.chr` A list containing results from clustering of low frequency markers in the experimental starting population for each window. Each list element contains the results for a window. If clusters for the respective window were identified the respective list numeric vectors with the marker positions for each cluster. If no clusters were identified the list is empty. The clusters identified for overlapping windows are the basis for cluster elongation to haplotype-blocks as present in `c1.long.m`.
- `c1.long.m` A list containing the results of haplotype-block reconstruction. Each list element corresponds to a reconstructed # haplotype-block and contains a numeric vector with all the SNP positions. The minor allele in the experimental starting population at those positions represent marker alleles for the respective haplotype-block. Chromosome-wide haplotype-blocks can be visualized with `plot`.

Author(s)

Susanne U. Franssen

References

Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution](#), *MBE*

See Also

[ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#) [map_rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#) [number_hbr](#)

initialize_SNP_time_series

Initialization of time series data as input for haplotype reconstruction

Description

This function initializes a genome-wide time series data set that can be used as input for haplotype-block reconstruction.

Usage

```
initialize_SNP_time_series(chr, pos, base.freq, lib.freqs, pop.ident,  
  pop.generation, use.libs, minfreqchange = 0.2, minrepl = 3,  
  max.minor.freq = 3/200, winsize = 5e+05, min.minor.freq = 0,  
  min.lib.frac = 0.75, win.scale = "bp", pos.cM = NULL)
```

Arguments

<code>chr</code>	character vector specifying the chromosome name for each genome-wide SNP
<code>pos</code>	numeric vector specifying the chromosomal position for each genome-wide SNP
<code>base.freq</code>	numeric vector specifying the frequency of the minor allele polarized in experimental starting population for each genome-wide SNP
<code>lib.freqs</code>	matrix specifying the frequencies of all genome-wide SNPs (rows) for all different libraries (time points and replicates, columns).
<code>pop.ident</code>	numeric vector specifying the identity of each library in terms of replicate ID
<code>pop.generation</code>	numeric vector specifying the time point of the respective library
<code>use.libs</code>	logical vector specifying which libraries should be used for haplotype-block reconstruction. The choice taken here determines SNP filtering as parameters <code>minfreqchange</code> and <code>minrepl</code> depend on the choice of the data set here. For visualization of marker frequencies, however, the remaining libraries will also be available.
<code>minfreqchange</code>	numeric specifying the minimum frequency change required in 'minrepl' replicates required to include the SNP in the analysis
<code>minrepl</code>	numeric specifying the number of replicates, in which the 'minfreqchange' is required to include the SNP in the analysis
<code>max.minor.freq</code>	numeric specifying the maximum frequency of the minor allele (polarized in the experimental starting population) to be included in the analysis
<code>winsize</code>	numeric specifying the window size on which to perform the analysis
<code>min.minor.freq</code>	numeric specifying the minimum frequency of the minor allele (polarized in the experimental starting population) to be included in the analysis (default=0).
<code>min.lib.frac</code>	minimum fraction of non-NA values for a SNP across libraries (only using libraries specified in <code>use.libs</code>) (default=0.75).
<code>win.scale</code>	character string specifying which genome-wide distance measure is used for window definition. Options are "bp" (base pairs) or "cM" (centi Morgan). cM distances can only be used if genetic positions are provided in 'pos.cM' (default="Mb").
<code>pos.cM</code>	numeric vector corresponding to SNP positions in <code>col.info</code> with genetic positions in cM.

Details

The function takes as input genome-wide frequencies of SNPs polarized for the minor frequency allele in the experimental starting population for multiple time points and replicates. SNP positions are filtered for a maximum frequency in the experimental starting population and a minimum frequency change in at least one time point for a specified number of replicates. The initialized data is returned as a `SNP_time_series` object that is required as input for the function `reconstruct_hb` to reconstruct unknown haplotype-blocks from the experimental starting population.

Value

an object of the class `SNP_time_series` data

Author(s)

Susanne U. Franssen

References

Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)

See Also

[ex_dat SNP_time_series](#)

inspect_window,hbr-method

Haplotype-block marker inspection for a window

Description

Plots correlations and groupings of haplotype-markers for a given window

Usage

```
## S4 method for signature 'hbr'  
inspect_window(object, window, colCluster = T)
```

Arguments

object	object of class hbr containing the results of reconstructed haplotype-blocks
window	number of the window, which should be inspected
colCluster	boolean value indicating whether columns in the output correlation plot should be clustered or the original positions of the markers reflecting genomic positions should be kept.

Details

The plotting method operates on [hbr](#) objects. For a specified window the correlation matrix for identified haplotype-block markers in this window is visualized also indicating haplotype-block identity.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#)
[number_hbr](#)

 inspect_window_avLink,hbr-method

Hierachical clustering of haplotype-block marker for a window

Description

Performs average linkage clustering for all SNPs in a window after filtering (i.e. [initialize_SNP_time_series](#)).

Usage

```
## S4 method for signature 'hbr'
inspect_window_avLink(object, window, min.cl.cor = 0,
  plotDendro = T, plotCluster = T)
```

Arguments

object	object of class hbr containing the results of reconstructed haplotype-blocks
window	number of the window, which should be inspected
min.cl.cor	minimum cluster correlation, if none is provided, the min.cl.cor of the provided hbr object is taken.
plotDendro	Boolean indicating if dendrogram using average linkage clustering should be plotted.
plotCluster	Boolean indicating if clusters identified with average linkage clustering should be visualized in PCs.

Details

Performs average linkage clustering on the time series data of the filtered SNPs and the selected time points (filtered and selected through use.libs in [initialize_SNP_time_series](#)) in a given window. The horizontal red line indicates the correlation (1-correlation) threshold on which clusters were originally identified. Note: Clusters with less than min.cl.cor markers are discarded. If no horizontal line is shown this indicates that all SNPs were assigned to the same cluster.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_dbScan](#)
[number_hbr](#)

```
inspect_window_dbScan,hbr-method
```

Clustering with the dbscan algorithm

Description

Performs clustering with the `dbscan` (Density-based spatial clustering of applications with noise) for all SNPs in a window after filtering (i.e. `initialize_SNP_time_series`) and visualizes found clusters based on principal components. NOTE: The clusters identified here are not necessarily identical with the clusters identified with average linkage clustering. Therefore if haplotype reconstruction was done using average linkage clustering the clusters shown here can be different from clusters of identified haplotype blocks.

Usage

```
## S4 method for signature 'hbr'
inspect_window_dbScan(object, window,
  minPts = object@min.cl.size, eps, plotkNN = T, plotCluster = T)
```

Arguments

<code>object</code>	object of class <code>hbr</code> containing the results of reconstructed haplotype-blocks
<code>window</code>	Number of the window, which should be inspected
<code>minPts</code>	Number of minimum points in the <code>eps</code> region (for core points). (default <code>min.cl.size</code> used for haplotype block reconstruction)
<code>eps</code>	The size of the epsilon neighborhood. Defines the distance between samples that built up a core cluster, for details see <code>?dbscan</code> of the <code>dbscan</code> package. (This value is only used for the <code>plotCluster</code> functionality.)
<code>plotkNN</code>	Boolean indicating if the kNN distance plot should be printed, for details see <code>?kNNdistplot</code> of the <code>dbscan</code> package. Briefly, the y-value where a bent in the curve is visible is a good indicator of the <code>eps</code> value to choose for the given <code>k = minPts</code> .
<code>plotCluster</code>	Boolean indicating if clusters identified with average linkage clustering should be visualized in PCs.

Details

Performs clustering with `dbscan` (Density-based spatial clustering of applications with noise) for all SNPs in a window after filtering (i.e. `initialize_SNP_time_series`) and visualizes found clusters based on principal components. NOTE: The clusters identified here are not necessarily identical with the clusters identified with average linkage clustering. Therefore if haplotype reconstruction was done using average linkage clustering the clusters shown here can be different from clusters of identified haplotype blocks. This method is rather indented for inspection and getting an idea if haplotype reconstruction should rather be run using `dbscan` instead of average linkage clustering.

This package used the `dbscan` implementation of the package `dbscan` (Michael Hahsler) originally described by Ester et al. (1996).

Author(s)

Susanne U. Franssen

See Also

[hbr ex_dat summary](#) [hbr plot](#) [hbr plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map rev_map markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#)
[number_hbr](#)

`inspect_window_PCA,hbr-method`

PCA of haplotype-block marker for a window

Description

Plots all filtered SNPs for a given window in terms of their the first two principal components of the respective time series data.

Usage

```
## S4 method for signature 'hbr'
inspect_window_PCA(object, window)
```

Arguments

<code>object</code>	object of class hbr containing the results of reconstructed haplotype-blocks
<code>window</code>	number of the window, which should be inspected

Details

Performs principal component analysis (PCA) on the time series data of the filtered SNPs and the selected time points (filtered and selected through `use.libs` in [initialize_SNP_time_series](#)) in a given window. SNPs are shown in terms of the first two principal components and colored according to the haplotype-block they were assigned to. Black colored points were not assigned to any cluster.

Author(s)

Susanne U. Franssen

See Also

[hbr ex_dat summary](#) [hbr plot](#) [hbr plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map rev_map markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_avLink](#) [inspect_window_dbScan](#)
[number_hbr](#)

map, hbr-method	<i>Map from reconstructed haplotype-blocks to windows</i>
-----------------	---

Description

A method returning information which windows are contain in which reconstructed haplotype-block.

Usage

```
## S4 method for signature 'hbr'  
map(object)
```

Arguments

object object of class [hbr](#) containing the results of reconstructed haplotype-blocks

Details

The method operates on [hbr](#) objects and returns a list summarizing the mapping information from reconstructed haplotype-blocks to contained windows. NOTE: A window is only listed for a haplotype-block when at least `min.inter` many markers of the block are in that window.

Value

list with elements corresponding to all reconstructed haplotype-blocks (id for all blocks without filtering based on the number of markers). Each element contains a vector of window indices that are contained in the block

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#)
[inspect_window_dbScan](#) [number_hbr](#)

markers,hbr-method *Markers of the specified reconstructed haplotype-block*

Description

Returns the genomic positions of the markers for the specified reconstructed haplotype-block

Usage

```
## S4 method for signature 'hbr'  
markers(object, hbr_id)
```

Arguments

object object of class [hbr](#) containing the results of reconstructed haplotype-blocks
hbr_id index of the haplotype-block of interest

Value

numeric vector with marker positions of the specified reconstructed haplotype-block

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map](#) [rev_map](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#)
[inspect_window_dbScan](#) [number_hbr](#)

number_hbr,hbr-method *The number of reconstructed haplotype-block*

Description

Returns the number of reconstructed haplotype-blocks

Usage

```
## S4 method for signature 'hbr'  
number_hbr(object)
```

Arguments

object object of class [hbr](#) containing the results of reconstructed haplotype-blocks

Value

returns the number of reconstructed haplotype-blocks

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#)
[inspect_window_dbScan](#)

plot,hbr-method

Method to visualize reconstructed haplotype-blocks

Description

The method visualizes reconstructed haplotype-blocks for a chromosome.

Usage

```
## S4 method for signature 'hbr'
plot(x, min.marker = 1, xlab = "Genomic position [Mb]",
     ylab = "Reconstructed haplotype-block", main = "default", col = "black",
     lwd = 4, hline = 10, indicate_shared = F, addPoints = F,
     hbr_plot = NULL, ...)
```

Arguments

x	object of class hbr containing the results of reconstructed haplotype blocks for visualization.
min.marker	numeric specifying the minimum number of markers a haplotype-block should contain in order to be visualized.
xlab	Label of the x-axis with the default value 'Genomic position [MB]'.
ylab	Label of the y-axis with the default value 'Reconstructe haplotype-block'.
main	Plot title (default: "Chromosome XX").
col	Color of the lines representing the haplotype-blocks (default: "black").
lwd	Line width of the lines representing the haplotype-blocks (default: 4).
hline	Distance between horizontal lines plotted for orientation (default: 10).
indicate_shared	logical value specifying if "spurious" markers that are identical between pairs of haplotype-blocks should be indicated. This function is usefull for inspecting results and deciding whether all identified blocks are all independent or maybe reconstruction parameters should be changed.

<code>addPoints</code>	logical value indicating if for each reconstructed block markers should additionally be indicated.
<code>hbr_plot</code>	boolean vector of length the number of reconstructed blocks indicating which ones should be plotted
<code>...</code>	arguments of the generic plot method.

Details

The method operates on [hbr](#) objects and visualizes location of reconstructed haplotype-blocks with respect to its genomic position.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#) [map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScannum](#)

`plot_cluster_trajectories,hbr-method`

Method to visualize the trajectories for all identified clusters in a window

Description

Method to visualize the trajectories for all identified clusters in a window

Usage

```
## S4 method for signature 'hbr'
plot_cluster_trajectories(object, window, ylim = c(0, 1),
  p.median = T, tp.cor = T)
```

Arguments

<code>object</code>	object of class <code>hbr</code>
<code>window</code>	numeric specifying the number of the window, for which trajectories should be plotted.
<code>ylim</code>	numeric vector with two elements specifying the limits of the y-axis.
<code>p.median</code>	boolean specifying if the median trajectory for each cluster should be added to the plot (default: TRUE).
<code>tp.cor</code>	Boolean indicating if only the time points used for calculating correlations <code>use.libs</code> are shown (<code>tp.cor=T</code>) or all time points present in the data set are shown (<code>tp.cor=F</code>).

Details

The method operates on [hbr](#) objects and plots the trajectories of all clusters and replicates in the specified window. Cluster trajectories are visualized by trajectories of all markers contained in the respective clusters and median trajectories for a cluster are visualized in a different color. For each cluster it is indicated to which haplotype-block it was later assigned. NOTE: It is possible that to clusters are assigned to the identical haplotype-block. This can happen when a cluster in an overlapping window shares markers with two clusters of the "focal" window.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_marker_trajectories](#) [map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#) [number_hbr](#)

plot_hbr_freq,hbr-method

Plots frequencies of a reconstructed haplotype-block along the chromosome

Description

Plots frequencies of a reconstructed haplotype-block along the chromosome

Usage

```
## S4 method for signature 'hbr'
plot_hbr_freq(object, hbr_id = 1, replicate, timepoint,
  window = 1, cols = NULL, add = F, sumstat = "mean", cex = 0.7,
  xlab = "Genomic position [Mb]", ylab = "Marker frequency", xlim = NULL,
  ylim = c(0, 1), pch = 20, lwd = 2)
```

Arguments

object	object of class hbr containing the results of reconstructed haplotype-blocks
hbr_id	id (integer) of the haplotype-block to be plotted
replicate	numeric vector of integers specifying the replicates for which results should be plotted
timepoint	numeric vector of time points for which the results should be plotted
window	window size over which frequencies are averaged (results are plotted for windows overlapping by windows/2)
cols	vector of colors with the length of the specified libraries
add	logical specifying if a new plot should be created or haplotype-block frequencies are added to an existing plot.

sumstat	Summary statistics used for the y-values in a window. Either specify "mean" (the default) or "median".
cex	scaling of the point size in the output plot
xlab	x-label in the output plot
ylab	y-label in the output plot
xlim	vector of the limits on the x-axis in the output plot
ylim	vector of the limits on the y-axis in the output plot
pch	option to specify symbols to use when plotting points in the output plot
lwd	line width in the output plot

Details

The plotting method operates on [hbr](#) objects and returns a plot containing frequencies of a respective haplotype-block along the chromosome for specified time points and replicates.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#)
[map](#) [rev_map](#) [markers](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#)
[number_hbr](#)

`plot_marker_trajectories,hbr-method`

Method to visualize the trajectories for all markers in a haplotype-block

Description

Method to visualize the trajectories for all markers in a haplotype-block

Usage

```
## S4 method for signature 'hbr'
plot_marker_trajectories(object, hbr_id, ylim = c(0, 1),
  loc.col = T, tp.cor = T)
```


Arguments

object	object of class hbr
hbr_id	the id of the haplotype-block, for which the marker trajectories should be plotted.
ylim	numeric vector with two elements specifying the limits of the y-axis.
loc.col	boolean indicating if trajectories should be coloured ranging from red over blue to yellow according to their location on the chromosome (default: TRUE).
tp.cor	Boolean indicating if only the time points used for calculating correlations use.libs are shown (tp.cor=T) or all time points present in the data set are shown (tp.cor=F).

Details

The method operates on [hbr](#) objects and plots the trajectories of all markers in a haplotype-block in all replicates. Note: As blocks can span a wide range along the genome, it is not expected that trajectories within a block stay very similar. Changing location along the genome can be indicated with `loc.col=T`.

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [summary.hbr](#) [plot.hbr](#) [plot_cluster_trajectories](#) [map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#) [number_hbr](#)

reconstruct_hb,SNP_time_series-method

Reconstruction of haplotype-blocks

Description

This function reconstructs haplotype-blocks from a `SNP_time_series` object based on correlated allele frequency changes across time points and replicates

Usage

```
## S4 method for signature 'SNP_time_series'
reconstruct_hb(object, chrom, min.cl.size = 4,
  min.cl.cor = 0.8, min.inter = 2, single.win = F, transf = T,
  scaleSNP = T, pos.cor = F, clusterM = "avLink", eps = 0.3)
```

Arguments

object	object of the class SNP_time_series data
chrom	a string specifying the name of the chromosome present in the object for which haplotype-block should be reconstructed
min.cl.size	numeric specifying the minimum number of correlated markers in a window for haplotype-block reconstruction
min.cl.cor	numeric specifying the correlation between marker SNPs required using average linkage clustering for markers to be assembled in one cluster
min.inter	numeric specifying the minimum number of markers in two clusters of overlapping windows required to be identical for cluster elongation across windows to build haplotype-blocks
single.win	boolean specifying that are supported by only a cluster in one window are also included in the markers of reconstructed blocks. If FALSE only hbrs are returned that span at least two windows and only markers being present in the intersection between overlapping windows are included.
transf	boolean indicating if frequency data should be square root transformed prior to calculation pairwise correlations with Pearson's correlation coefficient.
scaleSNP	boolean indicating whether time series allele frequency data for each SNP should be scaled (scaling to a mean of zero and standard deviation of 1).
pos.cor	boolean indicating if negative correlations should be set to zero prior to clustering.
clusterM	indicating the clustering method, choose from: "avLink" (average linkage clustering) and " dbscan ".
eps	size of the epsilon neighborhood when clustering with dbscan . For details please refer to the dbscan package.

Details

This function reconstructs haplotype-blocks via markers that show correlated frequency changes across multiple time-points and replicates. In a sliding window based approach SNPs are clustered by average linkage clustering based on correlations of their frequencies across time points and replicates. Clusters are build for each window using the parameters `min.cl.size` and `min.cl.cor`. Clustered in overlapping windows (window size/2) are elongated to haplotype-blocks if two adjacent clusters have `min.inter` identical markers. Reconstructed haplotype blocks were constructed from at least two overlapping windows and consist of markers that are present in two overlapping window clusters.

Value

an object of the class `hbr`

Author(s)

Susanne U. Franssen

References

Franssen et al. 2016, [Reconstruction of haplotype-blocks selected during experimental evolution](#), [MBE](#)

See Also

[ex_dat initialize_SNP_time_series SNP_time_series](#)

rev_map,hbr-method *Map from windows to reconstructed haplotype-blocks*

Description

A method returning information which reconstructed haplotype-blocks are present in which windows.

Usage

```
## S4 method for signature 'hbr'  
rev_map(object)
```

Arguments

object object of class [hbr](#) containing the results of reconstructed haplotype-blocks

Details

The method operates on [hbr](#) objects and returns a list summarizing the mapping information from windows to reconstructed haplotype-blocks.

Value

list with elements corresponding to all windows that contain a reconstructed haplotype-block segment. Each element contains a vector of haplotype-block indices that overlap the respective window.

Author(s)

Susanne U. Franssen

See Also

[hbr ex_dat summary.hbr plot.hbr plot_cluster_trajectories plot_marker_trajectories map markers plot_hbr_freq inspect_window inspect_window_PCA inspect_window_avLink inspect_window_dbScan number_hbr](#)

show,hbr-method *Display of a hbr object*

Description

This function displays the summarized features of a hbr object

Usage

```
## S4 method for signature 'hbr'  
show(object)
```

Arguments

object object of the class hbr data

Details

This function displays the summarized features of a hbr object

Author(s)

Susanne U. Franssen

References

Franssen et al. 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)

See Also

[summary.hbr](#) [ex_dat](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#) [map_rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#) [number_hbr](#)

show,SNP_time_series-method *Display of a SNP_time_series object*

Description

This function displays the summarized features of a SNP_time_series object

Usage

```
## S4 method for signature 'SNP_time_series'  
show(object)
```

Arguments

`object` object of the class `SNP_time_series` data

Details

This function displays the summarized features of a `SNP_time_series` object

Author(s)

Susanne U. Franssen

References

Franssen et al. 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)

See Also

[ex_dat initialize_SNP_time_series SNP_time_series](#)

`SNP_time_series-class` *An S4 class storing time series data*

Description

An S4 class representing the input data for haplotype-block reconstruction

Details

Time-series data is initialized to be used for haplotype-block reconstruction.

An S4 class storing the initialized input data for haplotype-block reconstruction with `reconstruct_hb`. Genome-wide time series data for multiple time points and replicates is stored along with replicate and time point information of the time series data. The stored data is filtered for SNPs with a maximum minor frequency in the experimental starting population and a minimum frequency change `minfreqchange` that is required for `minrepl` many replicates. An object of the class `SNP_time_series` can only be created with the function `initialize_SNP_time_series`.

Slots

`col.info` `data.table` with columns 'chr', 'pos', 'base.freq' and window. Each row corresponds to a SNP position that fullfills the filtering criteria.

`lib.freqs` `data.table` with columns for the different libraries (time points and replicates) and rows for all SNP positions that fullfill the filtering criteria.

`pos.cM` numeric vector corresponding to SNP positions in `col.info` with genetic positions in cM.

`pop.ident` numeric vector specifying the identity of each library in terms of replicate ID

`pop.generation` numeric vector specifying the time point of the respective library

`use.libs` logical vector specifying which libraries should be used for haplotype-block reconstruction

`winsize` numeric specifying the window size on which to perform the analysis

`win.scale` character string specifying which genome-wide distance measure is used for window definition. Options are "bp" (base pairs) or "cM" (centi Morgan). cM distances can only be used if genetic positions are provided in 'pos.cM' (default="Mb").

`min.minor.freq` numeric specifying the minimum frequency of the minor allele (polarized in the experimental starting population) to be included in the analysis (default=0).

`max.minor.freq` numeric specifying the maximum frequency of the minor allele (polarized in the experimental starting population) to be included in the analysis (default=3/113).

`min.lib.frac` minimum fraction of non-NA values for a SNP across libraries (only using libraries specified in `use.libs`) (default=0.75).

`minfreqchange` numeric specifying the minimum frequency change required in 'minrepl' replicates required to include the SNP in the analysis

`minrepl` numeric specifying the number of replicates, in which the 'minfreqchange' is required to include the SNP in the analysis

Author(s)

Susanne U. Franssen

References

Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution](#), *MBE*

See Also

[ex_dat initialize_SNP_time_series reconstruct_hb](#)

summary,hbr-method *Method to summarize information of reconstructed haplotype-blocks*

Description

The method summarizes information of the reconstructed haplotype-blocks for a chromosome.

Usage

```
## S4 method for signature 'hbr'
summary(object, min.marker = 1)
```

Arguments

object	object of class hbr containing the results of reconstructed haplotype blocks to be summarized.
min.marker	numeric specifying the minimum number of markers a haplotype-block to be reported. NOTE: IDs of haplotype-blocks in the provided summary are identical to the IDs of the previously identified blocks.

Details

The method operates on [hbr](#) objects and summarizes information of reconstructed haplotype-blocks

Value

data.table with row entries for each reconstructed haplotype-block and columns for: chr (chromosome), id (numbering of selected blocks), n.marker (number of markers in the reconstructed block), spos, epos (starting and end position of the block), len.pos (length of the block in bp), MperMb (number of markers per Mb), swin, ewin (starting and end window of the block)

Author(s)

Susanne U. Franssen

See Also

[hbr](#) [ex_dat](#) [plot.hbr](#) [plot_cluster_trajectories](#) [plot_marker_trajectories](#) [map](#) [rev_map](#) [markers](#) [plot_hbr_freq](#) [inspect_window](#) [inspect_window_PCA](#) [inspect_window_avLink](#) [inspect_window_dbScan](#) [number_hbr](#)

sync_to_frequencies *Data input from a sync file*

Description

Reads in SNP time series data from a file with .sync format.

Usage

```
sync_to_frequencies(file, base.pops, header, mincov = 15)
```

Arguments

file	the name of the ".sync" file where the data should be read from. Sync files are specified in Kofler et al. (2011). Sync files contain 3 + n columns with; col 1: chromosome (reference contig), col 2: position (in the reference contig), col 3: reference allele, col >3: sync entries for allele frequencies for all populations in the form A-count:T-count:C-count:G-count:N-count:deletion-count. Sync files originally don't have a header but headers are accepted when specified with header=T.
------	--

base.pops	logical vector with the same length as the number of libraries present in the sync file. Libraries indicated with TRUE will be used for identification on the two main alleles (minor and major allele). Allele frequencies of all libraries will subsequently be polarized for the minor allele in this specified subset.
header	logical value specifying whether a header is present in the provided sync file.
mincov	minimum coverage to calculate allele frequencies. If the sum of allele counts of the minor and major allele are below this threshold the respective frequency will be encoded as NA (default=15).

Details

Time series data from a file with sync format are read in. The sync format is specified in Kofler et al. 2011 (PoPoolation2: identifying differentiation between populations using sequencing of pooled DNA samples (Pool-Seq)). Allele counts are read in for each library and SNP and transformed to allele frequencies. Allele frequencies are polarized for the minor and major allele of a specifies (sub-)set of libraries, i.e. libraries of the experimental founder population. Frequencies are determined only based on the counts of the two most common alleles in the specified base populations base.pops. Please note: This procedure does not substitute a proper SNP calling. Provided sync files are expected only to contain positions of previously called SNPs and at least two alleles should be present in the specified base populations.

Value

a data.table with 6 plus N columns with; col 1: chr (chromosome), col 2: pos (position on respective chromosome), col 3: ref (reference allele), col 4: minallele (minor allele across all specified base populations), col 5: majallele (major allele across all specified base populations), col 6: weighted mean frequency of all specified base populations polarized for the minor allele, col >6: allele frequency of the minor allele for each library

Author(s)

Susanne U. Franssen

References

Franssen, Barton & Schloetterer 2016, [Reconstruction of haplotype-blocks selected during experimental evolution, MBE](#)

Index

- ex_dat, [2](#), [5](#), [7](#), [8](#), [10–17](#), [19–23](#)

- hbr, [7–19](#), [23](#)
- hbr (hbr-class), [4](#)
- hbr-class, [4](#)

- initialize_SNP_time_series, [5](#), [8–10](#), [19](#), [21](#), [22](#)
- inspect_window, [5](#), [8](#), [10–17](#), [19](#), [20](#), [23](#)
- inspect_window
 - (inspect_window, hbr-method), [7](#)
- inspect_window, hbr-method, [7](#)
- inspect_window.hbr
 - (inspect_window, hbr-method), [7](#)
- inspect_window_avLink, [5](#), [7](#), [10–17](#), [19](#), [20](#), [23](#)
- inspect_window_avLink
 - (inspect_window_avLink, hbr-method), [8](#)
- inspect_window_avLink, hbr-method, [8](#)
- inspect_window_avLink.hbr
 - (inspect_window_avLink, hbr-method), [8](#)
- inspect_window_dbScan, [5](#), [7](#), [8](#), [10–17](#), [19](#), [20](#), [23](#)
- inspect_window_dbScan
 - (inspect_window_dbScan, hbr-method), [9](#)
- inspect_window_dbScan, hbr-method, [9](#)
- inspect_window_dbScan.hbr
 - (inspect_window_dbScan, hbr-method), [9](#)
- inspect_window_PCA, [5](#), [7](#), [8](#), [10–17](#), [19](#), [20](#), [23](#)
- inspect_window_PCA
 - (inspect_window_PCA, hbr-method), [10](#)
- inspect_window_PCA, hbr-method, [10](#)
- inspect_window_PCA.hbr
 - (inspect_window_PCA, hbr-method), [10](#)

- map, [5](#), [7](#), [8](#), [10](#), [12–17](#), [19](#), [20](#), [23](#)
- map (map, hbr-method), [11](#)
- map, hbr-method, [11](#)
- map.hbr (map, hbr-method), [11](#)
- markers, [5](#), [7](#), [8](#), [10](#), [11](#), [13–17](#), [19](#), [20](#), [23](#)
- markers (markers, hbr-method), [12](#)
- markers, hbr-method, [12](#)
- markers.hbr (markers, hbr-method), [12](#)

- number_hbr, [5](#), [7](#), [8](#), [10–12](#), [14–17](#), [19](#), [20](#), [23](#)
- number_hbr (number_hbr, hbr-method), [12](#)
- number_hbr, hbr-method, [12](#)
- number_hbr.hbr (number_hbr, hbr-method), [12](#)

- plot, hbr-method, [13](#)
- plot.hbr, [5](#), [7](#), [8](#), [10–13](#), [15–17](#), [19](#), [20](#), [23](#)
- plot.hbr (plot, hbr-method), [13](#)
- plot_cluster_trajectories, [5](#), [7](#), [8](#), [10–14](#), [16](#), [17](#), [19](#), [20](#), [23](#)
- plot_cluster_trajectories
 - (plot_cluster_trajectories, hbr-method), [14](#)
- plot_cluster_trajectories, hbr-method, [14](#)
- plot_cluster_trajectories.hbr
 - (plot_cluster_trajectories, hbr-method), [14](#)
- plot_hbr_freq.hbr
 - (plot_hbr_freq, hbr-method), [15](#)
- plot_hbr_freq, [5](#), [7](#), [8](#), [10–15](#), [17](#), [19](#), [20](#), [23](#)
- plot_hbr_freq
 - (plot_hbr_freq, hbr-method), [15](#)
- plot_hbr_freq, hbr-method, [15](#)
- plot_marker_trajectories, [5](#), [7](#), [8](#), [10–16](#), [19](#), [20](#), [23](#)
- plot_marker_trajectories
 - (plot_marker_trajectories, hbr-method), [16](#)

plot_marker_trajectories, hbr-method, 16

plot_marker_trajectories.hbr (plot_marker_trajectories, hbr-method), 16

reconstruct_hb, 4, 22

reconstruct_hb (reconstruct_hb, SNP_time_series-method), 17

reconstruct_hb, SNP_time_series-method, 17

reconstruct_hb.SNP_time_series (reconstruct_hb, SNP_time_series-method), 17

rev_map, 5, 7, 8, 10–17, 20, 23

rev_map (rev_map, hbr-method), 19

rev_map, hbr-method, 19

rev_map.hbr (rev_map, hbr-method), 19

show (show, hbr-method), 20

show, hbr-method, 20

show, SNP_time_series-method, 20

show.hbr (show, hbr-method), 20

show.SNP_time_series (show, SNP_time_series-method), 20

SNP_time_series, 7, 19, 21

SNP_time_series (SNP_time_series-class), 21

SNP_time_series-class, 21

summary, hbr-method, 22

summary.hbr, 5, 7, 8, 10–17, 19, 20

summary.hbr (summary, hbr-method), 22

sync_to_frequencies, 2, 23