Package 'LPmerge'

October 12, 2022

Title Merging Linkage Maps by Linear Programming

Version 1.7

Author Jeffrey Endelman

Maintainer Jeffrey Endelman <endelman@wisc.edu>

Description Creates a consensus genetic map by merging linkage maps from different populations. The software uses linear programming (LP) to efficiently minimize the mean absolute error between the consensus map and the linkage maps. This minimization is performed subject to linear inequality constraints that ensure the ordering of the markers in the linkage maps is preserved. When marker order is inconsistent between linkage maps, a minimum set of ordinal constraints is deleted to resolve the conflicts.

Depends Rglpk, Matrix

License GPL-3

URL

NeedsCompilation no

Repository CRAN

Date/Publication 2018-06-07 17:43:50 UTC

R topics documented:

LPmerge   *Merging linkage maps by linear programming.*

Description

This package creates a consensus genetic map by merging linkage maps from different popula-
tions. The software uses linear programming (LP) to efficiently minimize the mean absolute error
between the consensus map and the linkage maps. This minimization is performed subject to lin-
ear inequality constraints that ensure the ordering of the markers in the linkage maps is preserved.
When marker order is inconsistent between linkage maps, a minimum set of ordinal constraints is
deleted to resolve the conflicts.

Usage

```
LPmerge(Maps, max.interval = 1:3, weights = NULL)
```

Arguments

Maps	List of T maps for a single linkage group. Each genetic map is a data frame with two columns: the first column contains the marker names, the second column contains the map position.
max.interval	A whole number (K in formula below) specifying the maximum interval size between bins to include in the objective function. An array of numbers can be passed to test different values (one consensus map is produced for each value in the array).
weights	Optional vector of length T containing the weights for each map in the objective function (see details). If not passed, the maps are given equal weight.

Details

Map positions in the ith linkage map are denoted by y_i, and consensus map positions are denoted by
x. Within linkage map i, the markers are ordered from j = 1 to M_i, and the map distance between the
jth and (j+q)th markers is $y_i(j+q) - y_i(j)$. Letting $u(j;i)$ denote the consensus map bin containing
marker j from map i, the corresponding distance in the consensus map is $x(u(j+q;i)) - x(u(j;i))$.
The total error across T maps with maximum interval size K is

$$\sum_{i=1}^{T} W_i N_i^{-1} \sum_{q=1}^{K} \sum_{j=1}^{M_i} |x(u(j+q;i)) - x(u(j;i)) - [y_i(j+q) - y_i(j)]|$$

where $N_i = \sum_{q=1}^{K} \sum_{j=1}^{M_i} 1$ is the number of error terms for map i, and W_i are the weights (equal
to 1 by default). At the end of the linkage map, where the sum j+q exceeds M_i, the expression is
evaluated as if the map were circular rather than linear. These "wrap-around" error terms keep the
total consensus map length commensurate with the average length of the linkage maps.

Linear inequality constraints are used to ensure the marker order in the consensus map is consistent
with the order in the linkage maps. When the linkage maps are not consistent, a minimum set
of constraints is deleted to resolve the conflicts. See the reference for more details. The deleted
constraints are printed to the standard output.

One way to select the maximum interval size K is based on the principle of minimizing the root
mean-squared error (RMSE) between the consensus map and the linkage maps. The RMSE for each
linkage map, and the overall mean, is displayed for convenience. Since the consensus map length
can vary with K, this is another factor to consider when selecting this parameter. See the tutorial at
http://potatobreeding.cals.wisc.edu/software for a detailed example.

Value

A list with length equal to the length of the max.interval parameter. Each entry in the list is a data frame containing the consensus map and the component linkage maps.

References

Endelman, JB, and C Plomion. 2014. LPmerge: An R package for merging genetic maps by linear programming. Bioinformatics 30:1623-1624.

Examples

```
mapI <- data.frame(marker=c("A","B","C","D","E","F","G"),position=0:6)
mapII <- data.frame(marker=c("A","C","B","D","E","F","G"),position=0:6)
mapIII <- data.frame(marker=c("A","B","C","D","E","G","F"),position=0:6)
mapIV <- data.frame(marker=c("B","A","C","D","E","F","G"),position=0:6)

maps <- list(I=mapI,II=mapII,III=mapIII,IV=mapIV)
ans <- LPmerge(maps)
```

Index

LPmerge,