

Package ‘dlmap’

April 17, 2009

Type Package

Title Detection Localization Mapping for QTL

Version 1.0

Date 2008-11-11

Author Emma Huang and Andrew George

Maintainer Emma Huang <Emma.Huang@csiro.au>

Description

License GPL-2

Depends qtl, ibdreg

Suggests nlme, asreml

Repository CRAN

Date/Publication 2008-11-14 16:48:43

R topics documented:

dlmap-package	2
BSdat	3
BSphe1	4
BSphe2	5
dlmap.convert.cross	6
dlmap.link.plot	9
dlmap.main	10
pvfx	14
qchibar	14

Index	15
--------------	-----------

`dlmap-package`*DLMapping for QTL detection*

Description

QTL mapping in a mixed model framework with separate detection and localization stages. The former detects the number of QTL on each chromosome based on the genetic variation due to the grouped markers on the chromosome, while the latter stage uses this information to determine the most likely QTL positions. The mixed model can accommodate general fixed and random effects, including spatial effects in field trials and random pedigree effects.

Details

Package: `dlmap`
Type: `Package`
Version: `1.0`
Date: `2008-11-11`
License: `GPL 2`

Data for the analysis must be input in three files in the following format:

- `genfile`: First row contains an identifier variable and marker names. Following rows contain genotype values for each individual
- `phefile`: First row contains names of phenotypic and environmental traits. One of these traits must be an identifier for each individual. Following rows contain values of the traits for each individual. Note: This file may contain more observations than the file containing the marker data
- `mapfile`: Contains marker names and chromosome groupings

The function `dlmap.convert.cross` is available to convert other file formats into the appropriate structure and its use is recommended to ensure that the data is properly input.

The primary function is `dlmap.asreml`, which performs the iterative algorithm to detect and position QTL on all chromosomes with significant genetic variation. This can accommodate sophisticated mixed models for phenotypic variation in addition to the genetic modeling.

Because ASReML-R is proprietary, we provide additional functions for DLMapping using the **nlme** package to fit mixed models. However, the function `dlmap.lme` does not provide as many features as that based on ASReML-R.

The vignette included in this package gives more background on the methodology, input file structure, and examples of how to use each of the important functions in the package.

Author(s)

Emma Huang and Andrew George
Maintainer: Emma Huang <Emma.Huang@csiro.au>

References

Huang, B.E. and George, A.W. Look before you leap: A new approach to QTL mapping. *Manuscript in preparation*

See Also

[read.cross](#), [asreml](#), [ibdreg](#), [lme](#)

BSdat

Simulated data for a backcross

Description

Dataset simulated according to Broman and Speed (2002).

Usage

```
data(BSdat)
```

Format

An object of class `cross`. List with two components:

- `geno` is a list with elements corresponding to chromosomes. `names(geno)` contains the names of the chromosomes. There are two components for each chromosome: `data`, a matrix whose rows are individuals and whose columns are markers, and `map`, a vector of marker positions in cM. There is no missing data, and genotypes are coded as 1=AA, 2=AB
- `pheno` is a data frame of size (250 x 2) containing the trait data. The first trait is generated from a random normal distribution with variance 1.0 and mean determined by the QTL genotypes as described below. The second trait is an ID for each individual

Details

The data was generated for a sample size of 250 from a map with 9 chromosomes. Each chromosome had length 100 cM and contained 11 equally spaced markers (spaced 10 cM apart). The background phenotypic variation was 1.0 and there was no missing data. The QTL were located as follows:

- Chr 1: 2 QTL located at 30 and 70 cM, both with effect size of 0.76
- Chr 2: 2 QTL located at 30 and 70 cM with effect size of 0.76 and -0.76
- Chr 3: 1 QTL located at 50 cM with effect size of 0.76
- Chr 4: 1 QTL located at 30 cM with effect size of 0.76
- Chr 5: 1 QTL located at 0 cM with effect size of 0.76

References

Broman, KW and Speed TP. 2002. A model selection approach for the identification of quantitative trait loci in experimental crosses. JRSS-B 64:641-656.

Examples

```
data(BSdat)
library(qtl)

# Summary of chromosomes and markers
nchr(BSdat)
nmar(BSdat)

# linkage map of data
plot.map(BSdat)

# interval mapping
BSgp <- calc.genoprob(BSdat, step=2)
BSim <- scanone(BSgp)

# composite interval mapping
BScim <- cim(BSgp, n.marcov=5, method="hk")

# LOD profile from CIM
plot(BScim)

# LOD threshold for 5 cofactors from paper
abline(h=3.56)
```

BSphe1

Simulated phenotypic data

Description

Phenotypic data simulated with BSdat which has one observation per genotype

Usage

```
data(BSphe1)
```

Format

A data frame with three columns:

ID: the ID for each individual. There is a single replicated for each of 250 individuals

phenotype: the phenotype value. It is generated from the same QTL effects as in BSdat

References

Broman, KW and Speed TP. 2002. A model selection approach for the identification of quantitative trait loci in experimental crosses. JRSS-B 64:641-656.

BSphe2

Simulated phenotypic data

Description

Phenotypic data simulated with BSdat which has multiple observations per genotype

Usage

```
data(BSphe2)
```

Format

A data frame with three columns:

ID: the ID for each individual. These are the same 250 individuals appearing in BSdat, each with 4 replicates

Block: a factor with 4 levels indicating the replicate

phenotype: the phenotype value. It is generated from the same QTL effects as in BSdat, with the additional block effect and random error

References

Broman, KW and Speed TP. 2002. A model selection approach for the identification of quantitative trait loci in experimental crosses. JRSS-B 64:641-656.

Examples

```
data(BSphe2)
```

```
boxplot(BSphe2$phenotype~BSphe2$Block)
```

```
dldmap.convert.cross
```

Convert input files to dldmap input format

Description

Reads in files from all formats supported by [read.cross](#) and converts them to dldmap input format. Creates dldmap input files in working directory.

Usage

```
dldmap.convert.cross(format = c("csv", "csvr", "csvs", "csvsr", "mm", "qtx",
"qtlcart", "gary", "karl", "rqtl"), obj, pheobj, idname="ID", dir = "",
genoutfile = "dlgenin.dat", pheoutfile = "dlphein.dat",
mapoutfile = "dlmapin.dat", file, genfile, mapfile, phefile,
chridfile, mnamesfile, pnamesfile, na.strings = c("-", "NA"),
genotypes = c("A", "H", "B", "D", "C"), alleles = c("A", "B"), ...)
```

Arguments

format	See documentation for read.cross . Also supports the input of an object of class <code>cross</code> ("rqtl" format)
obj	if format="rqtl", object of class <code>cross</code>
pheobj	Data frame or matrix containing supplementary spatial/environmental data
idname	Unique identifier variable name; will be used to match phenotypic and marker data
dir	Directory where input files are located; default is working directory
genoutfile	File with genotype data output in dldmap input format
pheoutfile	File with phenotype data output in dldmap input format
mapoutfile	File with marker position information output in dldmap input format
file	see read.cross
genfile	see read.cross
mapfile	see read.cross
phefile	see read.cross
chridfile	see read.cross
mnamesfile	see read.cross
pnamesfile	see read.cross
na.strings	see read.cross
genotypes	see read.cross
alleles	see read.cross
...	additional arguments to read.cross

Details

Use of this function will allow automatic input into `dldmap.asreml` or `dldmap.lme`. Otherwise input files will need to be constructed by hand. See below for the format if constructing input manually.

If a single set of trait values is available for each genotype, then phenotypic data will be input through the arguments `obj`, `file`, or `phefile` (depending on the file format). The argument `envobj` allows for input of phenotypic data on replicates or additional individuals which are not necessarily genotyped.

Value

Nothing returned. Three files output into names specified by `genoutfile`, `pheoutfile` and `mapoutfile` (or default values of "dlgenin.dat", "dlphein.dat", "dlmapin.dat" if these arguments are missing).

Notation:

n.gen is the number of genotyped individuals
n.ind is the number of phenotyped individuals
n.obs is the number of phenotypic observations (for a single trait)
M is the number of markers
P is the number of phenotypes
C is the number of chromosomes

Note that in general $n.gen \leq n.ind \leq n.obs$ since there may be multiple observations per individual, and more individuals may be phenotyped than genotyped. Individuals which are genotyped but not phenotyped will not be considered in analysis.

Genotypes

The genotype data file `genoutfile` has a header row with the name of the unique identifier variable and the marker names. This is followed by *n.gen* rows containing the identifier and genotypes for each individual.

ID	mrk1	...	mrkM
1	1	...	1
	
n.gen	0	...	1

Phenotypes

The phenotype data file `pheoutfile` has a header row with the name of the unique identifier variable and the phenotype names. This is followed by *n.obs* rows containing the identifier and phenotypes for each sample. If `envobj` is not input, the number of rows in this file will correspond to the number of rows in `genoutfile`.

ID	phename1	...	phenameP
1	3.7	...	2.4
	
n.ind	5.1	...	8.2

Linkage Map

The map data file `mapoutfile` has a header row with the labels shown below for the marker ID, chromosome and position in cM. The third column can be omitted, in which case the marker positions will be estimated from the data. Note: the marker names must be in the same order as in the genotype data file.

MrkID	Chr	Pos
mrk1	1	0.00
	...	
mrkM	C	100.0

Author(s)

Emma Huang and Andrew George

References

Huang, BE and George, AW. Look before you leap: A new approach to QTL mapping. *Manuscript in preparation*

See Also

[read.cross](#)

Examples

```
# load dataset
data(BSdat)
data(BSphe)

## Not run:
# convert data to dlmap format
dlmap.convert.cross(format="rqtl", obj=BSdat)

# convert data with separate phenotypic trait file
dlmap.convert.cross(format="rqtl", obj=BSdat, envobj=BSphe, idname="ID")
## End(Not run)
```

dldmap.link.plot *Plot linkage map with detected QTL*

Description

Plots the genetic linkage map for a selection of chromosomes. Indicates marker locations, marker names, and detected QTL positions and associated flanking markers obtained from a dldmap fit.

Usage

```
dldmap.link.plot(output, chr, max.dist, marker.names=TRUE, qcol="light blue", mcol="
link.map.cross(object, chr, max.dist, marker.names = TRUE, horizontal = FALSE, ...
```

Arguments

output	object of class dldmap
object	object of class cross
chr	character string naming the subset of chromosomes to plot; or, if numeric, a vector of chromosome indices
max.dist	a numerical value in cM determining the distance the genetic map should be subsetted by
marker.names	logical value. if TRUE then marker names are plotted alongside each chromosome on the left. Defaults to TRUE
qcol	colour of intervals surrounding QTL (see par for colour options)
mcol	colour of QTL flanking markers (see par for colour options)
pcol	colour of QTL positions (see par for colour options)
horizontal	logical value. If TRUE then map is plotted horizontally. Defaults to FALSE
...	arguments passed to "plot" to set up the plot region. Arguments may also be passed to "text" for the manipulation of the marker names

Details

The function `link.map.cross` was written by Julian Taylor for the `wgaim` package and is built upon here by adding QTL regions and estimated positions to the map.

The function `dldmap.plot.map` provides a neat visual display of chromosomes. If no QTL are detected, only the linkage map will be plotted; otherwise detected QTL will be placed at their estimated positions and the intervals around them (and flanking markers) will be highlighted. If a subset of chromosomes are plotted and detected QTLs exist outside that subset a warning will be given that QTLs have been omitted from the display.

The arguments `mcol`, `qcol` and `pcol` have been added for personal colour highlighting the flanking markers, QTL regions and QTL positions respectively. The procedure may also be given the usual `col` argument which will be passed on to the other markers.

In order to ensure that all marker names are displayed without vertical overlap, the default value of the "cex" parameter passed to "text" should be manipulated. For large maps with many chromosomes, marker names and adjacent chromosomes will overlap horizontally. In this case it is suggested that the user horizontally maximize the plotting window to remove overlap, or subset the chromosomes displayed.

Value

The genetic linkage map is plotted with shaded QTL regions, marked estimated positions, and highlighted flanking markers.

Author(s)

Emma Huang and Andrew George; Julian Taylor

References

Huang, BE and George, AW. Look before you leap: A new approach to QTL mapping. *submitted to Genetics*

See Also

[link.map](#)

Examples

```
# load dataset
data(BSdat)

## Not run:
dlmap.convert.cross(format="rqtl", obj=BSdat)
BSdl <- dlmap.asreml(phename="phenotype", estmap=FALSE, filestem="BS")
dlmap.link.map(BSdl)
## End(Not run)
```

dlmap.main

Perform DLMapping

Description

Fits the iterative algorithm for DLMapping. Reads in data, performs detection and localization stages and outputs summary of selected QTL effects.

Usage

```
dmap.asreml(genfile = "dlgenin.dat", phefile = "dlphein.dat",
mapfile = "dlmapin.dat", phename, baseModel, fixed = NULL,
random = NULL, rcov = NULL, sparse = NULL, pedigree,
step = 0, fixpos = 0, seed = 1, n.perm = 0, alpha = 0.05,
filestem = "dl", estmap=TRUE, ...)
```

```
dmap.lme(genfile="dlgenin.dat", phefile="dlphein.dat", mapfile="dlmapin.dat",
phename, fixed=NULL, step=0, fixpos=0, seed=1, maxit=60, alpha=.05,
filestem="dl")
```

Arguments

genfile	File with genotype data. Default filenames are output from dmap.convert.cross
phefile	File with phenotype data. Default filename is output from dmap.convert.cross
mapfile	File with marker position information. Default filename is output from dmap.convert.cross
phename	Response variable name
fixed	A formula object specifying the fixed effects part of the base model, with the terms, separated by + operators, on the right of a ~ operator. There is no left side to the ~ expression. If no fixed effect is specified, the model defaults to ~1, i.e. intercept only.
random	A formula object specifying the random effects part of the base model, with the terms, separated by + operators, on the right of a ~ operator. See asreml for more detail.
rcov	A formula object specifying the error structure of the model, with the terms, separated by + operators, on the right of a ~ operator. See asreml for more detail.
sparse	A formula object specifying the fixed effects to be absorbed, with the terms, separated by + operators, on the right of a ~ operator. See asreml for more detail.
baseModel	An alternative to specifying fixed, random, sparse, and rcov separately. If a base model has already been fit in asreml-R for the phenotypic variation, this can be input directly
pedigree	A pedigree object consisting of three columns. The first column is the individual ID, then the mother's ID and the father's ID. The name of the ID variable in the first column must match the idname variable
step	Step size for localization stage, i.e. if step=2, grid of positions spaced 2 cM apart are considered for QTL locations. If step=0 (default) positions are only located at markers.
fixpos	Alternative to specifying a step size - if fixpos=2, 2 evenly spaced positions between each marker are considered as QTL locations. If fixpos=0 (default) positions are only located at markers.
seed	Random number seed. Default=1
n.perm	Number of permutations used to get adjusted p-values at each iteration of detection. If n.perm=0 (default) the Bonferroni correction is used.

alpha	Significance level for testing
filestem	Stem to add to names of any files generated in DL Mapping process. Default="dl"
estmap	Indicator whether to re-estimate the linkage map
maxit	Maximum number of iterations to attempt for convergence of lme
...	additional arguments to asreml

Details

There are two versions of this function, which use different engines to fit the linear mixed models which form the framework of the algorithm. `dlmap.asreml` provides a much more general implementation of the DLMapping algorithm and is the preferred method of analysis. `dlmap.lme` is more restricted in its capabilities, in that it cannot model random effects or covariance structure, cannot handle more than 200 markers, and only allows for a single phenotypic observation per genotype. Also, permutation has not been implemented for this function because it is very slow. However, `dlmap.lme` will fit the basic algorithm and is useful should a license for ASReml not be available.

In `dlmap.asreml`, there are two options for specifying the model for phenotypic variation. The individual model components can either be input directly as they would be in an ASReml call, or a previous model (`baseModel`) output from ASReml can be input and the components will be retrieved from it. The latter formulation may be useful if prior phenotypic modelling has taken place. Note that in either case, variables appearing in the `rcov` statement must be ordered appropriately in the dataset. For example, if `rcov=~ar1(Column):ar1(Row)` the data must be sorted as *Row* within *Column*.

Missing values in `asreml` are replaced with zeros, so it is important to centre the covariate in question. This is done for all genotypes within the `dlmap.asreml` function. Thus individuals with phenotypic but not genotypic data, which play important roles in field trials, may be included safely. For `dlmap.lme` these individuals cannot be included, so the default behavior is to omit observations with missing values.

It is recommended that `no.perm` be set to 0 for initial exploratory analysis, as the permutation analysis may be lengthy. The Bonferroni correction is used to adjust for the number of chromosomes under consideration at each detection stage. While this is a conservative measure it seems to perform well in practice.

Two files are output with names set by the argument `filestem`, which has a default value of "dl". The file "`filestem.trace`" contains ASReml licensing and likelihood convergence output which otherwise would be dumped to the screen and possibly obscure other messages. Errors, warnings and other messages will still appear on the screen. Some warnings which appear may be passed through from an ASReml call and output on exit. These may generally be ignored. This file is not created if `dlmap.lme` is used.

The file "`filestem.det.log`" is a record of iterations in the detection stage. For each iteration the REMLRT testing for genetic variation on each chromosome is output, along with adjusted p-values, genomewide threshold and markers selected as fixed effects. The p-values are corrected for the number of chromosomes tested either by the Bonferroni correction or by permutation. If the number of permutations (`n.perm`) is greater than 0, then for the Xth iteration an additional file "`filestem.permX`" will be created which contains the test statistics for the permuted datasets. See the accompanying vignette for an example of how to interpret the ".det.log" file.

Value

zTable	Table with one row per QTL detected, columns for which chromosome the QTL is on, its position (cM), flanking markers, additive effect, Z-ratio and p-value.
no.qtl	Total number of QTL detected on all chromosomes
final.model	Object of class <code>asreml</code> for final model containing all terms in the base model, as well as effects for every QTL detected at the appropriate locations. No random effects for markers are fit
profile	If QTL are detected on C chromosomes, this is a list with C elements, each a matrix with 2 rows and a column for each position on the chromosome. The first row contains the cM position; the second row contains the Wald statistic for the model fit in the localization stage
cross	rqt1 cross object containing genotype data and linkage map
trait	name of the response fitted in the model (<code>phename</code> input argument)

Author(s)

Emma Huang and Andrew George

References

Huang, BE and George, AW. Look before you leap: A new approach to QTL mapping. *Manuscript in preparation*

See Also

[dldmap.convert.cross, asreml](#)

Examples

```
data(BSdat)
data(BSphe)

## Not run:
# Convert cross object to DL Mapping format
dldmap.convert.cross(format="rqt1", obj=BSdat)

# Analyze data
BSd1 <- dldmap.asreml(phename="phenotype", estmap=FALSE, filestem="BS")

# With additional phenotypic data
dldmap.convert.cross(format="rqt1", obj=BSdat, envobj=BSphe, idname="ID")
BSph <- dldmap.asreml(phename="phenotype", env=TRUE, random=~Block, estmap=FALSE) ## End(Not r
```

`pvfx`*Macro to compute p-value for testing variance component = 0*

Description

Given a value > 0 , computes $1 -$ (percentile of a chisquare mixture composed of $.5$ `chisq(df=0)` and $.5$ `chisq(df=1)`).

Usage`pvfx(test)`**Arguments**

<code>test</code>	Test statistic for which to compute p-value
-------------------	---

Author(s)

Emma Huang and Andrew George

See Also

[pchisq](#), [qchibar](#)

`qchibar`*Macro to compute quantile for chisquare mixture distribution*

Description

Reverses `pvfx`; i.e. given a percentile of a chisquare mixture composed of $.5$ `chisq(df=0)` and $.5$ `chisq(df=1)`, computes the corresponding quantile.

Usage`qchibar(p)`**Arguments**

<code>p</code>	probability
----------------	-------------

Author(s)

Emma Huang <Emma.Huang@csiro.au>

See Also

[pchisq](#), [pvfx](#)

Index

*Topic **datasets**

BSdat, [3](#)
BSphe1, [4](#)
BSphe2, [5](#)

asreml, [2](#), [11](#), [13](#)

BSdat, [3](#)
BSphe1, [4](#)
BSphe2, [5](#)

dlmap (*dlmap-package*), [1](#)
dlmap-package, [1](#)
dlmap.asreml, [2](#), [6](#)
dlmap.asreml (*dlmap.main*), [10](#)
dlmap.convert.cross, [2](#), [5](#), [10](#), [13](#)
dlmap.link.plot, [8](#)
dlmap.lme, [2](#), [6](#)
dlmap.lme (*dlmap.main*), [10](#)
dlmap.main, [10](#)

ibdreg, [2](#)

link.map, [10](#)
link.map.cross (*dlmap.link.plot*),
[8](#)

lme, [2](#)

pchisq, [14](#)
pvfx, [13](#), [14](#)

qchibar, [14](#), [14](#)

read.cross, [2](#), [5](#), [6](#), [8](#)