Package 'fitTetra'

February 19, 2015

Type Package

Title fitTetra is an R package for assigning tetraploid genotype scores

Version 1.0

Date 2013-04-23

Author Roeland Voorrips and Gerrit Gort

Maintainer Gerrit Gort <gerrit.gort@wur.nl>

Suggests doMC, foreach, RSvgDevice, devEMF

Description Package fitTetra contains three functions that can be used to assign genotypes to a collection of tetraploid samples based on bialleleic marker assays. Functions fitTetra (to fit several models for one marker from the data and select the best fitting) or saveMarkerModels (calls fitTetra for multiple markers and saves the results to files) will probably be the most convenient to use. Function CodomMarker offers more control and fits one specified model for a given marker.

License GPL (>= 2.2)

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2013-05-28 22:43:11

R topics documented:

`fitTetra-package`   *Fits mixture models for genotype calling in tetraploid species*

Description

Package fitTetra contains three functions that can be used to assign genotypes to a collection of tetraploid samples based on biallelic marker assays. Functions fitTetra (to fit several models for one marker from the data and select the best fitting) or saveMarkerModels (calls fitTetra for multiple markers and saves the results to files) will probably be the most convenient to use. Function CodomMarker offers more control and fits one specified model for a given marker.

Details

Package:	fitTetra
Type:	Package
Version:	1.0
Date:	2013-04-23
License:	GPL (>= 2)
LazyLoad:	yes

Author(s)

Roeland Voorrips: <roeland.voorrips@wur.nl> and Gerrit Gort: <gerrit.gort@wur.nl>

References

Voorrips, RE, G Gort, B Vosman (2011) *Genotype calling in tetraploid species from bi-allelic marker data using mixture models* BMC Bioinformatics 12: 172.

See Also

CodomMarker fitTetra saveMarkerModels

Examples

```
data(tetra.potato.SNP)
data(diplo.potato.SNP)
SNP4 <- subset(tetra.potato.SNP, MarkerName=='PotSNP004')
# Single marker, single mixture model
rawratio <- SNP4$X_Raw/(SNP4$X_Raw+SNP4$Y_Raw)
unmix <- CodomMarker(rawratio)

# Single marker, multiple mixture models
# df.tetra <- with(tetra.potato.SNP, data.frame(MarkerName=MarkerName,
```

```
#                    SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))
# df.diplo <- with(diplo.potato.SNP, data.frame(MarkerName=MarkerName,
#                    SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))

# unmix <- fitTetra(marker=6, data=df.tetra)
# unmix <- fitTetra(marker=6, data=df.tetra, plot='fitted')
# unmix <- fitTetra(marker=6, data=df.tetra, diplo=df.diplo, plot='all')

# Multiple markers (3 are chosen), multiple mixture models
# saveMarkerModels(markers=4:6, data=df.tetra, diplo=df.diplo, plot='all',
#                  maxiter=40, logfile='output.log', modelfile='mixresults.dat',
#                  allmodelsfile='allmixresults.dat', scorefile='scores.dat')
```

CodomMarker	*Function to fit a mixture model to a vector of signal ratios of a single bi-allelic marker*

Description

This function fits a specified mixture model to a vector of signal ratios of multiple samples for a single bi-allelic marker. Returns a list with results from the fitted mixture model.

Usage

```
CodomMarker(y, ng = 5, mutype = 0, sdtype = "sd.const", ptype = "p.free",
           clus = TRUE, mu.start = NA, sd.start=rep(0.075,ng), sd.fixed = 0.075,
            p = NA, maxiter = 500, maxn.bin = 200, nbin = 200, plothist = TRUE,
            nbreaks = 40, maintitle = NULL, subtitle = NULL, xlabel = NULL,
            xaxis = "s")
```

Arguments

y	the vector of signal ratios (each value is from one sample, vector y contains the values for 1 marker). All values must be between 0 and 1 (inclusive), NAs are not allowed. The minimum length of y is 10*ng.
ng	the number of possible genotypes (mixture components) to be fitted: one more than the ploidy of the samples
mutype	an integer in 0:6. Describes how to fit the means of the components of the mixture model: with mutype=0 the means are not constrained, requiring ng degrees of freedom. With mutype in 1:6 the means are constrained based on the ng possible allele ratios according to one of 6 models; see Details.
sdtype	one of "sd.const", "sd.free", "sd.fixed". Describes how to fit the standard deviations of the components of the mixture model: with "sd.const" all standard deviations (on the transformed scale) are equal (requiring 1 degree of freedom); with "sd.free" all standard deviations are fitted separately (ng d.f.); with "sd.fixed" all sd's on the transformed scale are equal to parameter sd.fixed (0 d.f.).

ptype	one of "p.free", "p.fixed" or "p.HW". Describes how to fit the mixing proportions of the components of the mixture model: with "p.free", the proportions are not constrained (and require ng-1 degrees of freedom); with "p.fixed" the proportions given in parameter p are fixed; with "p.HW" the proportions are calculated from the overall allele frequency, requiring only 1 degree of freedom.
clus	boolean. If TRUE, the initial means and standard deviations are based on a kmeans clustering into ng groups. If false, the initial means are equally spaced on the transformed scale between the values corresponding to 0.02 and 0.98 on the original scale and the initial standard deviations are 0.075 on the transformed scale.
mu.start	vector of ng values. If present, gives the start values of mu (the means of the mixture components) on the original (untransformed) scale, must be strictly ascending (mu[i]>mu[i-1]). Overrides the start values determined by clus TRUE or FALSE.
sd.start	vector of ng values. If present, gives the start values of sd (the standard deviations of the mixture components) on the transformed scale. Overrides the start values determined by clus TRUE or FALSE.
sd.fixed	vector, recycled if less than ng values: if argument sdtype is "sd.fixed", argument sd.fixed specifies the fixed standard deviations.
p	a vector of ng elements with the initial (or fixed, if parameter ptype is "p.fixed") mixing proportions of the mixture model components.
maxiter	integer: the maximum number of times the nls function is called in CodomMarker (0 = no limit, default=500)
maxn.bin	integer, default=200: if the length of y is larger than max.nbin the values of y (after arcsine square root transformation) are binned (i.e. the range of y (0 to pi/2) is divided into nbin bins of equal width and the number of y values in each bin is used as the weight of the midpoints of each bin). This results in significant speed improvement with large numbers of samples without noticeable effects on model fitting.
nbin	integer, default=200: the number of bins(see maxn.bin)
plothist	If TRUE a histogram of y is plotted with the fitted distributions superimposed
nbreaks	number of breaks for plotting the histogram; does not have an effect on fitting the mixture model
maintitle	string, used for plotting
subtitle	string, used for plotting
xlabel	string, used for plotting
xaxis	string, used for plotting: if "n" no x-axis is plotted

Details

This function takes as input a vector of ratios of the signals of two alleles (a and b) at a genetic marker locus (ratios as a/(a+b)), one for each sample, and fits a mixture model with ng components (for a tetraploid species: ng=5 components representing the nulliplex, simplex, duplex, triplex and quadruplex genotypes). Ideally these signal ratios should reflect the possible allele ratios (for a

tetraploid: 0, 0.25, 0.5, 0.75, 1) but in real life they show a continuous distribution with a number of more or less clearly defined peaks. The arguments specify what model to fit and with what values the iterative fitting process should start. If the argument mutype is set to a value in 1:6 the means of the mixture model components are constrained based on the possible allele ratios. This constraint takes the form of one of 6 possible models, specified by mutype, as follows: 1: a basic model assuming that both allele signals have a linear response to the allele dosage; one parameter for the ratio of the slopes of the two signal responses, and two parameters for the background levels (intercepts) of both signals (total 3 parameters). 2: as 1, but with the same background level for both signals (2 parameters) 3: as 1, with two parameters for a quadratic effect in the signal responses (5 parameters) 4: as 3, but with the same background level for both signals (4 parameters) 5: as 3, but with the same quadratic parameter for both signal responses (4 parameters) 6: as 5, but with the same background level for both signals (3 parameters)

Value

A list; if an error occurs the only list component is

message   the error message

If no error occurs the list has the following components:

loglik   the optimized log-likelihood

npar   the number of fitted parameters

AIC   Akaike's Information Criterion

BIC   Bayesian Information Criterion

psi   a list with components mu, sigma and p: each a vector of length ng with the means, standard deviations and mixing proportions of the components of the fitted mixture model;the means and standard deviations are on the transformed scale

post   a matrix of ng columns and length(y) rows; each row r gives the ng probabilities that the y[r] belongs to the ng components

nobs   the number of observations in y (excluding NA's and possibly removed outliers)

iter   the number of iterations

message   an error message, "" if no error

back   a list with components mu.back and sigma.back: each a vector of length ng with the means and standard deviations of the mixture model components back-transformed to the original scale.

Author(s)

Gerrit Gort: <gerrit.gort@wur.nl>

References

Voorrips, RE, G Gort, B Vosman (2011) *Genotype calling in tetraploid species from bi-allelic marker data using mixture models* BMC Bioinformatics 12: 172.

See Also

saveMarkerModels fitTetra fitTetra-package

Examples

```
data(tetra.potato.SNP)
SNP6 <- subset(tetra.potato.SNP, MarkerName=='PotSNP006')
# Single marker, single mixture model
rawratio <- SNP6$X_Raw/(SNP6$X_Raw+SNP6$Y_Raw)
unmix <- CodomMarker(rawratio)
```

---

diplo.potato.SNP   *SNP data for diploid potato*

Description

Contains data for set of 384 SNP markers from Illumina GoldenGate arrays of 64 diploid potato varieties. SNP markers are identical to those in tetraploid dataset.

Usage

```
data(diplo.potato.SNP)
```

Format

A data frame with 24576 observations on the following 6 variables.

MarkerName a factor with levels PotSNP001 to PotSNP384

SampleName a factor with 64 levels

X_Raw a numeric vector, raw intensity of first channel

Y_Raw a numeric vector, raw intensity of second channel

Theta a numeric vector, angle in (2/pi)*radials and first polar coordinate, obtained from GenomeStudio

R a numeric vector, radius and second polar coordinate, obtained from GenomeStudio

Source

Described in Anithakumari AM, J Tang, HJ van Eck, RFG Visser, JAM Leunissen, B Vosman, CG van der Linden (2010). *A pipeline for high throughput detection and mapping of SNPs from EST databases*. Mol. Breeding 26: 65-75

Examples

```
data(diplo.potato.SNP)
```

---

fitTetra	*Function to fit multiple mixture models to signal ratios of a single bi-allelic marker.*

Description

This function takes a data frame with allele signal ratios for multiple bi-allelic markers and samples, and fits multiple mixture models to a selected marker. It returns a list, reporting on the performance of these models, selecting the best one based on the BIC criterion, optionally plotting results.

Usage

```
fitTetra(marker, data, diplo = NA, select = TRUE, diploselect = TRUE, maxiter = 40,
        maxn.bin = 200, nbin = 200, sd.threshold = 0.1, p.threshold = 0.99,
       call.threshold = 0.6, peak.threshold = 0.85, try.HW = TRUE, dip.filter = 1,
         sd.target =  NA, plot = "none", plot.type = "png", plot.dir = NA)
```

Arguments

marker	integer: specifies the marker number to analyze. "marker" is the index to the alphabetically sorted MarkerNames (see argument "data")
data	data frame for tetraploid samples, with (at least) columns "MarkerName", "SampleName", and "ratio", where ratio is the a allele signal divided by the sum of the a and b allele signals (ratio = a/(a+b)).
diplo	data frame like "data" with diploid samples. Facultative, does not affect model fitting. Diploid samples will be plotted in the plot of the best-fitting model if argument plot is "fitted" or "all". Genotypic scores for diploid samples are calculated according to the best-fitting model and are therefore from 0 (nulliplex) to 4 (quadruplex), as for the tetraploid samples.
select	boolean vector, recycled if shorter than the columns in data: indicates which rows are to be used (default: select=TRUE, i.e. keep all rows)
diploselect	as select, for diplo instead of data
maxiter	integer: the maximum number of times the nls function is called in CodomMarker (0 = no limit, default=500)
maxn.bin	integer, passed to CodomMarker, see there for explanation
nbin	integer, passed to CodomMarker, see there for explanation
sd.threshold	the maximum value allowed for the (constant) standard deviation on the arcsine - square root transformed scale, default 0.1. If the optimal model has a larger standard deviation the marker is rejected.
p.threshold	the minimum P-value required to assign a genotype to a sample; default 0.99. If the P-value for all 5 possible genotypes is less than p.threshold the sample is assigned genotype NA.
call.threshold	the minimum fraction of samples to have genotypes assigned ("called"); default 0.6. If under the optimal model the fraction of "called" samples is less than call.threshold the marker is rejected.

peak.threshold	the maximum allowed fraction of the scored samples that are in one peak; default 0.85. If any of the possible genotypes (peaks in the ratio histogram) contains more than peak.threshold of the samples the marker is rejected (because the remaining samples offer too little information for reliable model fitting)
try.HW	boolean: if TRUE (default), try models with and without a constraint on the mixing proportions according to Hardy-Weinberg equilibrium ratios. If FALSE, only try models without this constraint.
dip.filter	integer: if 1 (default), select best model only from models that do not have a dip (a lower peak surrounded by higher peaks: these are not expected under Hardy-Weinberg equilibrium or in cross progenies). If all fitted models have a dip still the best of these is selected. If 2, similar, but if all fitted models have a dip the marker is rejected. If 0, select from all fitted models including those with a dip.
sd.target	If the fitted standard deviation on the transformed scale is larger than sd.target a penalty is given (see Details); default NA i.e. no penalty is given.
plot	string, "none" (default), "fitted" or "all". If "fitted" a plot of the best fitting model and the assigned genotypes is saved with filename <marker number><marker name>.<plot.type>, preceded by "rejected_" if the marker was rejected. If "all" small images of all models are saved to files (8 per file) with filename <"plots"><marker number><A/B/C/D><marker name>. <plot.type> in addition to the plot of the best fitting model.
plot.type	string, "png" (default), "emf", "svg" or "pdf". Indicates format for saving the plots. If "emf" and the operating system is not Windows, "png" is used. If "emf" and the package "devEMF" is not installed, or if the specified format canot be produced for any other reason, "png" is used.
plot.dir	The directory where the plot files are to be saved; default NA. i.e. plot files saved in working directory.

Details

fitTetra fits a series of mixture models for the given marker by repeatedly calling CodomMarker and selects the optimal one. The models tested have four different models for the means of the mixture components: mutype 1, 2, 5 and 6 as described for CodomMarker, and one or two (depending on argument try.HW) models for the mixing proportions. These four or eight models are run using 2 or 3 different start configurations. The model with the smallest Bayesian Information Criterion (BIC) is selected, within the constraints specified by dip.filter. If sd.target is specified, the selection criterion is equal to BIC for models where (on the transformed scale) sd<sd.target, and to (sd.target/sd)*BIC where sd>sd.target (since BIC is negative, a larger sd results in a larger selection criterion which is less likely to be the minimum). The final model selected according to these criteria is then checked against call.threshold and peak.threshold and may still be rejected, in which case no fitted model is reported.

Value

a list with components:

log	a character vector with the lines of the log text
modeldata	a data frame with one row with the marker number, marker name, number of samples and (if the marker is not rejected) data of the fitted model (see below)

allmodeldata	a data frame with for each tried model one row with the marker number, marker name, number of samples and (if the marker is not rejected) data of the fitted model (see below)
scores	a data frame with the name and data for all samples (including NA's for the samples that were not selected, see parameter select): marker (same as argument marker), MarkerName, SampleName, model (a string describing the model),select (value of argument select for this data point),ratio (the given ratio from argument data), P0,P1,P2,P3,P4 (the probabilities that this sample belongs to each of the five mixture components), maxgeno (the genotype = mixture component with the highest P value), maxP (the P value for this genotype) and geno (the assigned genotype number: same as maxgeno, or NA if maxP<p.threshold). Maxgeno and geno numbers from 0 to 4: the allele dosage of the a allele.
diploscores	a data frame like scores for the samples in the data frame supplied with argument diplo. If diplo is NA also diploscores will be NA.

The modeldata and allmodeldata data frames present data on a fitted model. modeldata presents data on the selected model; allmodeldata lists all attempted. Both data frames contain the following columns:

marker	the sequential number of the marker (marker names are ordered alphabetically)
markername	the name of the marker
m	the number of the attempted or selected fit. The 8 (or 4 if try.HW is FALSE) models are tried with 2 or 3 start configurations, so m can range from 1 to 16 or 1 to 24.
model	the fitted model. Possible values are "b1", "b2", "b1,q", "b2,q", "b1 HW", "b2 HW", "b1,q HW" and "b2,q HW" where b1 and b2 indicate whether 1 or two parameters for signal background were fitted, q indicates that a quadratic term in the signal response was fitted, and HW indicates that the mixing proportions were constrained according to Hardy-Weinberg equilibrium ratios. For more details see Voorrips et al (2011)
nsamp	the number of samples for this marker for which select==TRUE, i.e. the number on which the call rate is based.
nsel	the number of these samples that have a non-NA ratio value
npar	the number of free parameters fitted
iter	the number of iterations to reach convergence
dip	0, 1 or 2, parameter passed to CodomMarker, see there for explanation.
LL	the log-likelihood of the fitted model
AIC	Akaike's Information Criterion
BIC	Bayesian Information Criterion
minsepar	a measure of the minimum peak separation. Each difference of the means of two successive mixture components is divided by the average of the standard deviations of the two components. The minimum of the four values is reported. All calculations are on the arcsine-square root transformed scale.
selcrit	The selection criterion; the model with the lowest selcrit is selected. If argument sd.target is NA selcrit is equal to BIC, else selcrit is larger than BIC, see Details for details.

meanP	For each sample the maximum probability of belonging to any mixture compo-nent is calculated. The average of these P values is reported in meanP
P80, P90, P95, P975, P99	
	the fraction of samples that have a probability of at least 0.8, 0.9, 0.95, 0.975 or 0.99 to belong to one of the five mixture components (by default a level of 0.99 is required to assign a genotype score to a sample)
muact0, muact1, muact2, muact3, muact4	
	the actual means of the samples in each of the five mixture components on the arcsine-square root transformed scale
sdact0, sdact1, sdact2, sdact3, sdact4	
	the actual standard deviations of the samples in each of the five mixture compo-nents on the transformed scale
mutrans0, mutrans1, mutrans2, mutrans3, mutrans4	
	the model means of the mixture components on the transformed scale
sdtrans0, sdtrans1, sdtrans2, sdtrans3, sdtrans4	
	the model standard deviations of the mixture components on transformed scale
P0, P1, P2, P3, P4	
	the mixing proportions of the five components
mu0, mu1, mu2, mu3, mu4	
	the model means of the five mixture components back-transformed to the origi-nal scale
sd0, sd1, sd2, sd3, sd4	
	the model standard deviations of the five mixture components back-transformed to the original scale
message	if no model was fitted, the reason is reported here

Author(s)

Roeland Voorrips: <roeland.voorrips@wur.nl>

References

Voorrips, RE, G Gort, B Vosman (2011) *Genotype calling in tetraploid species from bi-allelic marker data using mixture models* BMC Bioinformatics 12: 172.

See Also

saveMarkerModels CodomMarker fitTetra-package

Examples

```
data(tetra.potato.SNP)
data(diplo.potato.SNP)
df.tetra <- with(tetra.potato.SNP, data.frame(MarkerName=MarkerName,
                 SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))
df.diplo <- with(diplo.potato.SNP, data.frame(MarkerName=MarkerName,
                 SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))
# Single marker, multiple mixture models
```

```
unmix <- fitTetra(marker=4, data=df.tetra)
#unmix <- fitTetra(marker=4, data=df.tetra, plot='fitted')
#unmix <- fitTetra(marker=4, data=df.tetra, diplo=df.diplo, plot='all')
```

---

saveMarkerModels	*A function to fit mixture models for series of markers and save the results to files*

Description

This is a convenience function that calls fitTetra for a series of markers and saves the tabular, graphical and log output to files. Most of the arguments are identical to those of fitTetra and are directly passed through.

Usage

```
saveMarkerModels(markers = NA, data, diplo = NA, select = TRUE, diploselect = TRUE,
                 maxiter = 40, maxn.bin = 200, nbin = 200, sd.threshold = 0.1,
                 p.threshold = 0.99, call.threshold = 0.6, peak.threshold = 0.85,
                 try.HW = TRUE, dip.filter = 1, sd.target = NA, ncores = NA,
                 logfile = ", modelfile, allmodelsfile = ", scorefile,
                 diploscorefile = ", plot = "none", plot.type = "png")
```

Arguments

markers	an integer vector listing the marker numbers to be analyzed. The numbers refer to the levels of data$MarkerName. If NA (default) all markers are analyzed.
data	data frame for tetraploid samples, with (at least) columns "MarkerName", "SampleName", and "ratio", where ratio is the a allele signal divided by the sum of the a and b allele signals (ratio = a/(a+b)).
diplo	data frame like "data" with diploid samples. Facultative, does not affect model fitting. Diploid samples will be plotted in the plots of the best-fitting models if argument plot is "fitted" or "all". Genotypic scores for diploid samples are calculated according to the best-fitting models and are therefore from 0 (nulliplex) to 4 (quadruplex), as for the tetraploid samples.
select	boolean vector, recycled if shorter than the columns in data: indicates which rows are to be used (default: select=TRUE, i.e. keep all rows)
diploselect	as select, for diplo instead of data
maxiter	integer: the maximum number of times the nls function is called in CodomMarker (0 = no limit, default=500)
maxn.bin	integer, passed to CodomMarker, see there for explanation
nbin	integer, passed to CodomMarker, see there for explanation
sd.threshold	the maximum value allowed for the (constant) standard deviation on the arcsine - square root transformed scale, default 0.1. If the optimal model has a larger standard deviation the marker is rejected.

p.threshold	the minimum P-value required to assign a genotype to a sample; default 0.99. If the P-value for all 5 possible genotypes is less than p.threshold the sample is assigned genotype NA.
call.threshold	the minimum fraction of samples to have genotypes assigned ("called"); default 0.6. If under the optimal model the fraction of "called" samples is less than call.threshold the marker is rejected.
peak.threshold	the maximum allowed fraction of the scored samples that are in one peak; default 0.85. If any of the possible genotypes (peaks in the ratio histogram) contains more than peak.threshold of the samples the marker is rejected (because the remaining samples offers too little information for reliable model fitting)
try.HW	boolean: if TRUE (default), try models with and without a constraint on the mixing proportions according to Hardy-Weinberg equilibrium ratios. If FALSE, only try models without this constraint.
dip.filter	integer: if 1 (default), select best model only from models that do not have a dip (a lower peak surrounded by higher peaks: these are not expected under Hardy-Weinberg equilibrium or in cross progenies). If all fitted models have a dip still the best of these is selected. If 2, similar, but if all fitted models have a dip the marker is rejected. If 0, select from all fitted models including those with a dip.
sd.target	if the fitted standard deviation on the transformed scale is larger than sd.target a penalty is given (see Details); default NA i.e. no penalty is given.
ncores	integer: the number of processor cores that can be used for parallel processing. If NA (default) or 1 no parallelization takes place. On operating systems other than Unix / Linux, or if the packages doMC and foreach are not installed the ncores argument is ignored.
logfile	string, name of a text file. This file will contain several text lines per marker corresponding to component "log" in the result of fitTetra. If "" (default) no file is created. The directory for the plot files will be named as the log file preceded by "plots_" and without the extension ".log"; or simply "plots" if no logfile is specified.
modelfile	string, name of a text file. This file will contain one line per marker corresponding to component "modeldata" in the result of fitTetra. modelfile can be read using read.table. This argument is required and has no default value.
allmodelsfile	string, name of a text file. This file will contain 16 or 24 lines per marker, corresponding to component "allmodeldata" in the result of fitTetra. allmodelsfile can be read using read.table. If "" (default) no file is created.
scorefile	string, name of a text file. This file will contain one line per sample for every marker that could be fitted, corresponding to component "scores" in the result of fitTetra. scorefile can later be read using read.table. This argument is required and has no default value.
diploscorefile	string, name of a text file. This file will contain one line per sample in diplo for every marker that could be fitted, corresponding to component "diploscores" in the result of fitTetra. diploscorefile can later be read using read.table. If "" (default) no file is created.
plot	string, "none" (default), "fitted" or "all". Same as argument plot in fitTetra.
plot.type	string, "png" (default), "emf", "svg" or "pdf". Indicates format for saving the plots. Same as argument plot.type in fitTetra.

Details

No further details.

Value

This function does not return a value.

Author(s)

Roeland Voorrips: <roeland.voorrips@wur.nl>

References

Voorrips, RE, G Gort, B Vosman (2011) *Genotype calling in tetraploid species from bi-allelic marker data using mixture models* BMC Bioinformatics 12: 172.

See Also

[CodomMarker](#) [fitTetra](#) [fitTetra-package](#)

Examples

```
data(tetra.potato.SNP)
data(diplo.potato.SNP)

df.tetra <- with(tetra.potato.SNP, data.frame(MarkerName=MarkerName,
                 SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))
df.diplo <- with(diplo.potato.SNP, data.frame(MarkerName=MarkerName,
                 SampleName=SampleName, ratio=X_Raw/(X_Raw+Y_Raw)))

# Multiple markers (only 1 is chosen here), multiple mixture models
saveMarkerModels(markers=87:87, data=df.tetra, diplo=df.diplo, plot='fitted',
        try.HW=FALSE, modelfile = 'modelfile.dat', scorefile='scorefile.dat')
```

---

tetra.potato.SNP *SNP data for tetraploid potato*

Description

Contains data for set of 384 SNP markers from Illumina GoldenGate arrays of 224 tetraploid potato varieties covering a wide range with respect to geographic origin, year of first registration and intended application.

Usage

```
data(tetra.potato.SNP)
```

Format

A data frame with 86016 observations on the following 6 variables.

MarkerName a factor with levels PotSNP001 to PotSNP384

SampleName a factor with 224 levels

X_Raw a numeric vector, raw intensity of first channel

Y_Raw a numeric vector, raw intensity of second channel

Theta a numeric vector, angle in (2/pi)*radials and first polar coordinate, obtained from GenomeStudio

R a numeric vector, radius and second polar coordinate, obtained from GenomeStudio

Source

Described in Anithakumari AM, J Tang, HJ van Eck, RFG Visser, JAM Leunissen, B Vosman, CG van der Linden (2010). *A pipeline for high throughput detection and mapping of SNPs from EST databases*. Mol. Breeding 26: 65-75

Examples

data(tetra.potato.SNP)

Index