Package 'qtlDesign'

February 20, 2015

Title Design of QTL experiments

Version 0.941

Date 03 September 2010

Author Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

Description Tools for the design of QTL experiments

Maintainer Saunak Sen <sen@biostat.ucsf.edu>

License GPL

URL http://www.biostat.ucsf.edu/sen/

Repository CRAN

Repository/R-Forge/Project qtldesign

Repository/R-Forge/Revision 18

Date/Publication 2012-10-29 08:59:32

NeedsCompilation no

R topics documented:

Confidence interval expected widths
Calculating expected QTL confidence interval widths

Description

Provides expected confidence interval widths for QTL location when we have dense markers.

Usage

```
ci.length(cross,n,effect,p=0.95,sigma2=1,env.var,gen.var,bio.reps=1)
```

Arguments

cross	String indicating cross type which is "bc", for backcross, "f2" for intercross, and "ri" for recombinant inbred lines.
n	Sample size
p	Confidence level for desired confidence interval
effect	The QTL effect we want to detect. For powercalc and samplesize this is a numeric (vector). For detectable it specifies the relative magnitude of the additive and dominance components for the intercross. The specification of effect depends on the cross. For backcross, it is the difference in means the heterozygote and homozygote. For RI lines it is half the difference in means of the homozygotes, for intercross, it is a two component vector of the form c(a,d), where a is the additive effect (half the difference between the homozygotes), and d is the dominance effect (difference between the heterozygote and the average of the homozygotes). The genotype means will be -a-d/2, d/2, and a-d/2. For detectable, optionally for the intercross, one can use a string to specify the QTL effect type. The strings "add" or "dom" are used to denote an additive or dominant model respectively for the phenotype. It may be it can be a numerical vector of the form c(a,d) indicating the relative magnitudes of the additive and dominance components (as defined above). The default is "add".
sigma2	Error variance; if this argument is absent, env.var and gen.var must be specified.
env.var	Environmental (within genotype) variance
gen.var	Genetic (between genotype) variance due to all loci segregating between the parental lines.
bio.reps	Number of biological replicates per unique genotype. This is usually 1 for backcross and intercross, but may be larger for RI lines.

Details

With dense markers, the log likelihood follows a compound process. Approximate expected confidence intervals can be calculated by pretending the log likelihood decays linearly with a drift rate that depends on the effect size and cross type.

Value

Returns the expected confidence interval width in cM assuming dense markers.

Author(s)

Saunak Sen

References

Dupuis J and Siegmund D (1999) Statistical methods for mapping quantitative trait loci from a dense set of markers. Genetics 151:373-386.

Darvasi A (1998) Experimental strategies for the genetic dissection of complex traits in animal models. Nature Genetics 18:19-24.

Kong A and Wright FA (1994) Asymptotic theory for gene mapping. Proceedings of the National Academy of Sciences of the USA 91:9705-9709.

See Also

[powercalc](#).

Examples

```
ci.length(cross="bc",n=400,effect=5,p=0.95,sigma2=1)
```

Information	*Information under null hypothesis of equal means*

Description

Functions to calculate the information under the null hypothesis of no effect. Functions for discount factors for incomplete genotyping.

Usage

```
info(sel.frac,theta=0,cross)
info.bc(sel.frac,theta=0)
info.f2(sel.frac,theta=0)
deflate(theta,cross)
deflate.bc(theta)
deflate.f2(theta)
nullinfo(sel.frac)
```

Arguments

cross	Cross type, either "bc" for backcross, or "f2" for intercross.
sel.frac	Selection fraction; proportion of extremes genotyped
theta	Recombination fraction between flanking markers

Details

The `nullinfo` function calculates the information content per observation for any contrast between genotype means when densely genotyping an `sel.frac` fraction of the extreme phenotypic individuals. The information content is calculated under the null hypothesis of no difference between the genotype means. For small differences in genotype means, the information content will be approximately equal to the null, but in general, the information estimate under the null is the lower bound.

The `info` function calculates the information per observation for backcross, and F2 intercross under the null hypothesis of equal gentoype means. The information is calculated for a point in the middle of an interval spanned by markers separated by a recombination fraction `theta`. The function `deflate` calculates a deflation factor for the information attenuation in the middle of a marker interval relative to a completely typed location.

Value

Information per individual for information functions, and the discount factor for the discount functions.

Note

Information is calculated under the equal means assumption. This approximation is very good in practice, and is slightly conservative. If the difference between the means is large, these functions will underestimate the information. For power calculations, that is okay.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

Examples

```
nullinfo(0.5)
info(0.5,cross="bc")
info(0.5,cross="f2")
info(0.5,0.1,cross="bc")
info(0.5,0.1,cross="f2")
deflate(0.1,"bc")
deflate(0.1,"f2")
```

Information-cost functions

Functions to calculate information-cost ratios

Description

Functions to calculate information cost-ratios.

Usage

```
info2cost(sel.frac,cost,d,G=NULL,cross)
info2cost.bc(sel.frac,cost,d,G=NULL)
info2cost.f2(sel.frac,cost,d,G=NULL)
```

Arguments

sel.frac	Selection fraction; proportion of individuals genotyped
cost	Genotyping cost in units of raising an individual. When d=0 (dense genotyping), it is the cost of genotyping an individual. When d!=0, it is the cost of a single marker genotype in an individual.
d	Marker spacing in centiMorgans
G	Genome size in Morgans
cross	Cross type, "bc"or "f2"

Details

The information calculations are done under the null hypothesis of no QTL effect.

Value

For d!=0 it calculates the ratio of information in the middle of a marker interval of length d cM to the cost of genotyping the cross. For d=0, it calculates the ratio of information at any locus to the cost of genotyping the cross.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

See Also

info

Examples

```
info2cost.bc(0.5,1)
info2cost.bc(0.5,1,10,1450)
```

K1   *Calculate scores for minimum moment abberations.*

Description

Calculate the MMA K1, K12, and the standardized dissimilarity score (eff1).

Usage

```
Kstat(genomat, type = 1)
K1(genomat)
K12(genomat)
eff1(n, nmark, s1)
```

Arguments

genomat	Genotype matrix.
n	Desired sample size.
type	Type of dissimilarity measure desired (first or second moment).
nmark	Number of markers.
s1	Dissimilarity score from K1 or K12.

Value

Score or standardized score based on selected marker list. K1 and K12 call Kstat with type = 1 and 2, respectively. Kstat computes the minimum moment abberation score. eff1 computes the standardized genetic dissimilarity.

Author(s)

Brian S. Yandell (mailto:byandell@wisc.edu)

References

Jin C, Lan H, Attie AD, Churchill GA, Bulutuglo D, Yandell BS (2004) Selective phenotyping for increased efficiency in genetic mapping studies. Genetics 168: 2285-2293.

See Also

[mma](mma), read.cross

---

mma   *Selective phenotyping with similarity measure 2*

Description

Selective phenotyping with similarity measure 2 to select the most dissimilar subset of individuals.

Usage

```
mma(genof, p, sequent = FALSE, exact = FALSE, dismat = FALSE)
```

Arguments

genof	Genotype matrix.
p	Sample size to select.
sequent	Perform sequential optimization if TRUE (see below).
exact	Count allele differences if FALSE; binary 0 = same number of alleles, 1 = different if TRUE.
dismat	Return dissimilarity matrix if TRUE.

Details

Sequentially minimize 1st moment and then 2nd moment, swapping one subject at a time. op finds all the samples with same 1st moment similarity with mma results. op2 finds all the samples with the same 1st moment similarity with every list from op result. A combination of op and op2 comes very close to exhaustive search in practice. moment2 find the best list with minimum 2nd moments from the output of op2. Note that some warnings occurs accompanying our return statement. The results are not affected though.

This function combines several functions in Jin's original code. mma(genof,p,sequent=TRUE is identical to the depricated mmasequent(genof,p. mma(genof,p,exact=TRUE is identical to the depricated mmaM1(genof,p (actually, mma uses dissimilarity while mmaM1 used similarity = 1 - dissimilarity).

Value

A list containing cList, dismat if that option is TRUE and further optimized lists (op, op2, moment2) if sequent is TRUE. vector as the first item. The list of items includes:

cList	vector of selected subjects by function mma
op	list containing vector of selection and update flag from function op
op2	matrix of selection by function op2
moment2	vector of second moment calculations
dismat	dissimilarity matrix

Author(s)

Brian S. Yandell (mailto:byandell@wisc.edu)

References

Jin C, Lan H, Attie AD, Churchill GA, Bulutuglo D, Yandell BS (2004) Selective phenotyping for increased efficiency in genetic mapping studies. Genetics 168: 2285-2293.

See Also

K1, read.cross

---

mma.level   *MMA utility*

Description

This routine is for internal use. It sets 3 levels to 0,1,2.

Usage

```
mma.level(mat)
```

Arguments

mat   input matrix

Details

Converts matrix to levels between 0 and 2.

Value

Matrix of genotype levels between 0 and 2.

Author(s)

Brian S. Yandell (mailto:byandell@wisc.edu)

References

Jin C, Lan H, Attie AD, Churchill GA, Bulutuglo D, Yandell BS (2004) Selective phenotyping for increased efficiency in genetic mapping studies. Genetics 168: 2285-2293.

See Also

mma, read.cross

Optimal marker spacing

Optimal marker spacing

Description

Functions to find optimal marker spacing given cost.

Usage

```
optspacing(cost,G=NULL,sel.frac,cross)
optspacing.bc(cost,G=NULL,sel.frac)
optspacing.f2(cost,G=NULL,sel.frac)
optspacing(cost,G=NULL,sel.frac=NULL,cross)
optspacing.bc(cost,G=NULL,sel.frac=NULL)
optspacing.f2(cost,G=NULL,sel.frac=NULL)
```

Arguments

cost	Cost of genotyping in units of raising an individual
sel.frac	Selection fraction; proportion of individuals genotyped
G	Genome size in centiMorgans
cross	Cross type, "bc" or "f2"

Details

The function optim is used to search for the optima.

Value

In the first form, with the selection fraction specified, the spacing in centiMorgans that maximizes the information to cost ratio in the middle of the marker interval. In the second form, with the selection fraction unspecified, it returns the value of (spacing,sel.frac) which maximizes the information to cost ratio in the middle of the marker interval.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

See Also

optim, optimize

Examples

```
optspacing(cost=0.1,G=1440,sel.frac=0.5,cross="bc")
optspacing(cost=30/3000,G=1440,sel.frac=NULL,cross="f2")
```

Optimal selection fraction

Optimal selection fraction

Description

Functions to find optimal selection fractions given cost.

Usage

```
optselection(cost,d=0,G=NULL,cross)
optselection.bc(cost,d=0,G=NULL)
optselection.f2(cost,d=0,G=NULL)
```

Arguments

cost	Cost per genotype in units of raising individual
d	Marker spacing in Morgans
G	Genome size in Morgans
cross	Cross type, "bc" or "f2"

Details

The function `optimize` is used to search for the optima.

Value

The optimal selection fraction.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

See Also

[optimize](#)

Examples

```
optselection(1,cross="bc")
optselection(0.001,10,1450,cross="bc")
optselection(0.001,10,1450,cross="f2")
```

Package version   *Version of qtlDesign package*

Description

Returns the version number for the qtlDesign package.

Usage

```
version.qtlDesign()
```

Value

The version number.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

Power calculations   *Power, sample size, and detectable effect size calculations*

Description

Power, sample size, and minimum detectable effect size calculations are performed for backcross, F2 intercross, and recombinant inbred (RI) lines.

Usage

```
powercalc(cross,n,effect,sigma2,env.var,gen.var,thresh=3,sel.frac=1,
          theta=0,bio.reps=1)
detectable(cross,n,effect=NULL,sigma2,env.var,gen.var,power=0.8,thresh=3,
          sel.frac=1,theta=0,bio.reps=1)
samplesize(cross,effect,sigma2,env.var,gen.var,power=0.8,thresh=3,
          sel.frac=1,theta=0,bio.reps=1)
```

Arguments

`cross`	String indicating cross type which is "bc", for backcross, "f2" for intercross, and "ri" for recombinant inbred lines.
`n`	Sample size
`sigma2`	Error variance; if this argument is absent, `env.var` and `gen.var` must be specified.
`env.var`	Environmental (within genotype) variance
`gen.var`	Genetic (between genotype) variance due to all loci segregating between the parental lines.
`effect`	The QTL effect we want to detect. For `powercalc` and `samplesize` this is a numeric (vector). For `detectable` it specifies the relative magnitude of the additive and dominance components for the intercross. The specification of `effect` depends on the cross. For backcross, it is the difference in means the heterozygote and homozygote. For RI lines it is half the difference in means of the homozygotes, for intercross, it is a two component vector of the form `c(a,d)`, where a is the additive effect (half the difference between the homozygotes), and d is the dominance effect (difference between the heterozygote and the average of the homozygotes). The genotype means will be `-a-d/2`, `d/2`, and `a-d/2`. For `detectable`, optionally for the intercross, one can use a string to specify the QTL effect type. The strings "add" or "dom" are used to denote an additive or dominant model respectively for the phenotype. It may be it can be a numerical vector of the form `c(a,d)` indicating the relative magnitudes of the additive and dominance components (as defined above). The default is "add".
`power`	Proportion indicating power desired
`thresh`	LOD threshold for declaring significance
`sel.frac`	Selection fraction
`theta`	Recombination fraction corresponding to a marker interval
`bio.reps`	Number of biological replicates per unique genotype. This is usually 1 for backcross and intercross, but may be larger for RI lines.

Details

These calculations are done assuming that the asymptotic chi-square regimes apply. A warning message is printed if the effective sample size is less than 30 and either `sel.frac` is less than 1 or `theta` is greater than 0. First we calculate the effective sample size using the width of the marker interval and the selection fraction. The QTL is assumed to be in the middle of the marker interval. Then we use the fact that the non-centrality parameter of the likelihood ration test is $m * \delta^2$, where m is the effctive sample size and δ is the QTL effect measured as the deviation of the genotype means from the overall mean. The chi-squared approximation is used to calculate the power. The minimum detectable effect size is obtained by solving the power equation numerically using `uniroot`. The theory behind the information calculations is described by Sen et. al. (2005).

A key input is the error variance, `sigma2` which is generally unknown. The user can enter the error variance directly, or estimate it using `env.var` and `gen.var`. The function `error.var` is used to the error variance using estimates of the environmental variance and genetic variance. Another key input is the effect segregating in a cross, which can be calculated using `gmeans2model`.

Value

For powercalc the power is returned, along with the proportion of variance explained. For detectable the effect size detectable is returned, along with the proportion of variance explained. For backcross and RI lines this is the effect of an allelic substitution. For F2 intercross the additive and dominance components are returned. For samplesize the sample size (rounded up to the nearest integer) is returned along with the proportion of variance explained.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

See Also

uniroot. error.var, gmeans2effect.

Examples

```
powercalc("bc",100,5,sigma2=1,sel.frac=1,theta=0)
powercalc(cross="ri",n=30,effect=5,env.var=64,gen.var=25,bio.rep=6)
detectable("bc",100,sigma2=1)
detectable(cross="ri",n=30,env.var=64,gen.var=25,bio.rep=8)
samplesize(cross="f2",effect=c(5,0),env.var=64,gen.var=25)
```

Thresholds and tail probabilities

Calculating thresholds and tail probabilities for genome scans

---

Description

Provides genome-wide thresholds and tail probabilities for the maxima of genome scans using Poisson approximations.

Usage

```
tailprob(t,G,cross,type="1",d=0.01,cov.dim=0)
thresh(G,cross,type="1",p=c(0.10,0.05,0.01),d=0.01,cov.dim=0,
        interval=c(1,40))
```

Arguments

G	Genome size in centiMorgans.
t	LOD value for which tail probability is desired.
p	Vector giving the genome-wide Type I error for which thresholds are desired.
cross	String indicating cross type which is "bc", for backcross, "f2" for intercross.
type	Type of LOD score for which threshold is desired. Right now the only option is "1", but more options will be added in the future.
d	Marker spacing in centiMorgans.
cov.dim	Dimension of interacting covariate. Set to 0 right now.
interval	Interval over which to search for LOD threshold.

Details

The tail probabilities are calculated using the method of Dupuis and Siegmund (1999). The thresholds are calculated by solving the tail probability equation using uniroot. At this time only one-dimensional thresholds are calculated, but this function will be extended in the future.

Value

The function tailprob returns the probability that the genome-wide maximum LOD score exceeds a particular value. The function thresh returns genome-wide LOD thresholds corresponding to a particular Type I error rate.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Dupuis J and Siegmund D (1999) Statistical methods for mapping quantitative trait loci froma dense set of markers. Genetics 151:373-386.

See Also

[uniroot](.).

Examples

```
tailprob(t=3,G=1440,cross="f2",d=10)
thresh(G=1440,cross="bc",d=10)
```

Utility   *Utility functions*

Description

Utility functions

Usage

```
recomb(d)
genetic.dist(theta)
```

Arguments

d	Genetic distance in Morgans
theta	Recombination fraction

Value

recomb returns the recombination fraction corresponding to a genetic distance in Morgans. genetic.dist returns the genetic distance in Morgans for a recombination fraction.

Note

We assume Haldane mapping function for the genetic distance.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

Examples

```
recomb(0.1)
genetic.dist(0.1)
```

```
Variance and effect size
```
Effect size, proportion variance explained, and error variance calcu-
lations

Description

The function `error.var` estimates the error variance using estimates of the environmental variance
and genetic variance. The effect segregating at a locus, can be calculated using `gmeans2effect`
These are key inputs for power calculations. The function `prop.var` calculates the proportion of
variance explained by a locus given the effect size and error variance.

Usage

```
error.var(cross,env.var=1,gen.var=0,bio.reps=1)
gmeans2effect(cross,means)
prop.var(cross,effect,sigma2)
```

Arguments

`cross`	String indicating cross type which is "bc", for backcross, "f2" for intercross, and "ri" for recombinant inbred lines.
`env.var`	Environmental (within genotype) variance
`gen.var`	Genetic (between genotype) variance due to all loci segregating between the parental lines.
`bio.reps`	Number of biological replicates per unique genotype. This is usually 1 for back-cross and intercross, but may be larger for RI lines.
`means`	Vector of genotype means in the form `c(a,h,b)`, where a is the mean of the "AA" homozygotes, h is the mean of the "AB" heterozygotes, and b is the mean of the "BB" homozygotes.
`effect`	The QTL effect which depends on the cross. For backcross, it is the difference in means the heterozygote and homozygote. For RI lines it is half the difference in means of the homozygotes, for intercross, it is a two component vector of the form `c(a,d)`, where a is the additive effect (half the difference between the ho-mozygotes), and d is the dominance effect (difference between the heterozygote and the average of the homozygotes). The genotype means will be `-a-d/2`, `d/2`, and `a-d/2`.
`sigma2`	Error variance.

Details

The function `error.var` estimates the error variance segregating in a cross using estimates of the
environmental (within genotype) variance, and the genetic (between genotype variance). The envi-
ronmental variance is assumed to be invariant between cross types. The genetic variance segregating

in RI lines is assumed to be double that in F2 intercross, and four times that of the backcross. This assumption holds if all loci are additive. The error variance at a locus of interest is aproximately

$$\sigma_G^2/c + \sigma_E^2/m,$$

where

$$\sigma_G^2$$

is the genetic variance (gen.var), c is a constant depending on the cross type (1, for RI lines, 1/2 for F2 intercross, and 1/4 for backross),

$$\sigma_E^2$$

is the environmental variance (env.var), and m is the number of biological replicates per unique genotype (bio.reps).

The function gmeans2effect calculates the QTL effects from genotype means depending on the cross.

The function prop.var calculates the proportion of variance attributable to a locus given the effects size(s) and the error variance. The definition of effect size is in the output of gmeans2effect (see below).

Value

For error.var the value is the estimated error variance based on the assumptions mentioned above. For gmeans2effect the value depends on the type of cross. For RI lines it is simply the additive effect of the QTL which is half the difference between the homozygote means. For intercross, it is a vector giving the additive and dominance components. The additive component is half the difference between the homozygote means, and the dominance component is the difference between the heterozygotes and the average of the homozygotes. For the backcross, it is a vector of length 2, c(a-h,h-b), which is the effect of an allelic substitution of an "A" allele in the backcrosses to each parental strain.

Author(s)

Saunak Sen, Jaya Satagopan, Karl Broman, and Gary Churchill

References

Sen S, Satagopan JM, Churchill GA (2005) Quantitative trait locus study design from an information perspective. Genetics, 170:447-64.

See Also

powercalc

Examples

```
error.var(cross="bc",env.var=1,gen.var=1,bio.reps=1)
error.var(cross="f2",env.var=1,gen.var=1,bio.reps=1)
error.var(cross="ri",env.var=1,gen.var=1,bio.reps=1)
error.var(cross="ri",env.var=1,gen.var=1,bio.reps=10)
gmeans2effect(cross="f2",means=c(0,1,2))
```

```
gmeans2effect(cross="f2",means=c(0,1,1))
gmeans2effect(cross="bc",means=c(0,1,1))
gmeans2effect(cross="ri",means=c(0,1,1))
prop.var(cross="bc",effect=5,sigma2=1)
prop.var(cross="f2",effect=c(5,0),sigma2=1)
prop.var(cross="ri",effect=5,sigma2=1)
```

Index