Package 'sonicLength'

July 23, 2025

Type Package

Title Estimating Abundance of Clones from DNA Fragmentation Data

Version 1.4.7

Date 2021-09-19

Author Charles Berry <ccberry@ucsd.edu>

Maintainer Charles Berry <ccberry@ucsd.edu>

Depends R (>= 2.14.0), splines

Description Estimate the abundance of cell clones from the distribution of lengths of DNA fragments (as created by sonication, whence `sonicLength'). The algorithm in ``Estimating abundances of retroviral insertion sites from DNA fragment length data" by Berry CC, Gillet NA, Melamed A, Gormley N, Bangham CR, Bushman FD. Bioinformatics; 2012 Mar 15;28(6):755-62 is implemented. The experimental setting and estimation details are described in detail there. Briefly, integration of new DNA in a host genome (due to retroviral infection or gene therapy) can be tracked using DNA sequencing, potentially allowing characterization of the abundance of individual cell clones bearing distinct integration sites. The locations of integration sites can be determined by fragmenting the host DNA (via sonication or fragmentase), breaking the newly integrated DNA at a known sequence, amplifying the fragments containing both host and integrated DNA, sequencing those amplicons, then mapping the host sequences to positions on the reference genome. The relative number of fragments containing a given position in the host genome estimates the relative abundance of cells hosting the corresponding integration site, but that number is not available and the count of amplicons per fragment varies widely. However, the expected number of distinct fragment lengths is a function of the abundance of cells hosting an integration site at a given position and a certain nuisance parameter. The algorithm

implicitly estimates that function to estimate the relative abundance.

License GPL (>= 2)

LazyLoad yes

Repository CRAN

Repository/R-Forge/Project soniclength

Repository/R-Forge/Revision 12

Repository/R-Forge/DateTimeStamp 2021-09-19 22:55:50

Date/Publication 2021-09-20 04:10:02 UTC

NeedsCompilation no

Contents

`sonicLength-package` *Calibration of Number of Length Variants*

Description

Estimate the number of fragments produced by sonication of integration sites in DNA (as arise in gene therapy or in retroviral infections) from tallies of the number of distinct lengths. See the Berry et al. reference where the experimental setting and algorithm are spelled out in detail.

Details

Package:	sonicLength
Type:	Package
License:	see file LICENSE distributed with this software
LazyLoad:	yes

This package has the necessary utilities to provide a function that will do calibration of the number of distinct lengths (for an integration site recovered via sonication) to estimate the number of distinct sonication fragments that underlie the observed number of lengths.

Author(s)

Charles C. Berry

Maintainer: <ccberry@ucsd.edu>

References

The reference to the algorithm is:

Estimating abundances of retroviral insertion sites from DNA fragment length data. Berry CC, Gillet NA, Melamed A, Gormley N, Bangham CR, Bushman FD. Bioinformatics; 2012 Mar 15;28(6):755-62.

Some applications that illustrate use of this method are found here:

Gillet, Nicolas A., et al. "The host genomic environment of the provirus determines the abundance of HTLV-1 infected T-cell clones." Blood 117.11 (2011): 3113-3122.

Maldarelli, F., et al. "Specific HIV integration sites are linked to clonal expansion and persistence of infected cells." Science 345.6193 (2014): 179-183.

`A1` *HTLV-1 Fragment Length Data*

Description

A sample from a patient with HTLV-1 was analyzed as described in Gillet et al (2001).

Usage

```
data(A1)
```

Format

A data frame with 6176 observations on the following 3 variables.

`locations` a factor with levels `01` `F` `101480450` `...X` `R` `99972891` indicating the genomic sites at which insertions were detected

`lengths` a numeric vector of fragment lengths

`replicates` a numeric vector giving replicate IDs

Details

For each replicate, the unique combinations of the genome location at which an insertion site was detected and the length(s) of the fragment(s) sequenced were noted. There is one such row in he data.frame for each such combination.

Source

See Gillet et al 2011 for more details.

References

Gillet, N.A., Malani, N., Melamed, A., Gormley, N., Carter, R., Bentley, D., Berry, C., Bushman, F.D., Taylor, G.P., and Bangham, C.R. (2011). The host genomic environment of the provirus determines the abundance of HTLV-1-infected T-cell clones. *Blood* 117, 3113-3122.

Examples

```
data(A1)
str(A1)
```

`estAbund` *estimate Abundances from sonicated samples*

Description

When one or more replicate samples are sonicated and the lengths for each integration site are recorded, it is of interest to estimate actual number of sonicants allowing for coincidentally equal lengths. A likelihood approach is used here.

Usage

```
estAbund(locations, lengths, replicates = NULL, jackknife = F, kmax=0,
min.length = 20, ...)
```

Arguments

`locations`	a vector of IDs for distinct locations
`lengths`	a vector of the corresponding lengths
`replicates`	an optional vector of the replicate sample ID
`jackknife`	Whether to do leave-one-out jackknife samples over the replicates
`kmax`	highest count to bother with when computing mass function for counts. All higher values are globbed together in the result. If `kmax==0` don't compute
`min.length`	Least length expected for a fragment. Shorter lengths will not be permitted
`...`	Other arguments that may be passed along to functions that do the real work.

Details

This is a wrapper function for `maxEM` so study that function to see what is going on.

Value

a list with components

theta	estimated abundances
phi	estimated bin probabilities
var.theta	variances of theta
iter	number of iterations till convergence was achieved
call	the function call
lframe	the data.frame used to construct the estimate of phi
obs	the observed counts of distinct lengths
jackknife	a list of calls to maxEM with one replicate omitted in each
data	a data.frame with columns locations,lengths, and replicates (if supplied)

Author(s)

Charles C Berry <ccberry@users.r-forge.r-project.org>

References

Estimating abundances of retroviral insertion sites from DNA fragment length data. Berry CC, Gillet NA, Melamed A, Gormley N, Bangham CR, Bushman FD. Bioinformatics; 2012 Mar 15;28(6):755-62.

See Also

maxEM phi.update.lframe The package vignette:Estimating Abundances with sonicLength

---

Ey.given.x *E-Step for Abundance Estimation*

Description

Take the E-step of the EM algorithm for estimating the number of sonicants in a sample

Usage

```
Ey.given.x(x, theta, phi)
pr.y.given.x(x, theta, phi, kmax=20 )
```

Arguments

x	a zero-one indicator matrix whose rows correspond to unique lengths with row-names indicating those lengths
theta	a vector of the abundance estimates. length(theta)==ncol(x)
phi	a vector of the probabilities of sonicant lengths. length(phi)==nrow(x)
kmax	highest count to bother with (all higher values are globbed together in the result)

Details

Supposing Poisson sampling of sonicants, `Ey.given.x(x,theta,phi)[i,j]` gives the expected value of the number of sonicants given that `x[i,j]` distinct sonicant lengths were observed. `pr.y.given.x(x,theta,phi.km` gives the probability of `k` sonicants (censored at `kmax+1`) of insertion site `j`

Value

`Ey.given.x()` is a double matrix of the same dimension as `x`. `pr.y.given.x(...,kmax)` is a double matrix of dimension `c( kmax+1, ncol(x) )`

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

Examples

```
mat <- diag(10)
mat[ ,10 ] <- 1.0
phi1 <- prop.table( rep(1,10))
theta1 <- 1:10
Ey.given.x( mat, theta1, phi1 )
pr.mat <- pr.y.given.x( mat, theta1, phi1 )
## Estimate Seen plus Unseen sites
popsize.chao <- function(tab) sum(tab)+tab[1]*(tab[1]-1)/(2*(tab[2]+1))
## evaluate at expected counts
popsize.chao( rowSums(pr.mat) )
## average randomly sampled counts
cnt.samp <- function(x) sample( seq_along(x) , 1 ,pr=x )
mean(replicate(100,popsize.chao( table(apply(pr.mat,2, cnt.samp) ))))
```

`Fragment Distribution` *Simple Fragment Length Distribution*

Description

Density, distribution function, quantile function and random generation for a simple parametric distribution for fragments lengths (given by conditioning a geometric distribution on whether the length can be recovered).

Usage

```
dfrag( x, loc=45, lscale=2.5, rate=0.02, maxx=qgeom(1-1e-7,rate) )
pfrag( q, loc=45, lscale=2.5, rate=0.02, maxx=qgeom(1-1e-7,rate), lower.tail=TRUE )
qfrag( p, loc=45, lscale=2.5, rate=0.02, maxx=qgeom(1-1e-7,rate), lower.tail=TRUE )
rfrag( n, loc=45, lscale=2.5, rate=0.02, maxx=qgeom(1-1e-7,rate) )
```

Arguments

x, q	vector of quantile (of lengths).
n	number of lengths to sample.
p	vector of probabilities
loc	vector of locations of logistic for prob of recovery.
lscale	vector of scales of logistic for prob of recovery.
rate	probability for geometric probability for fragment lengths.
maxx	integer; largest value of x to bother with.
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ otherwise, $P[X > x]$.

Details

The mass function is given by `plogis(x,loc,lscale)*dgeom(x,rate) / denom` , where `denom` scales the result to sum to 1.0 in the range `0:maxx`. The other functions all depend on this in the obvious manner. If `maxx` is not large enough a warning may be issued, but even without this warning the results may be slightly innaccurate if `pgeom(maxx,rate,lower.tail=FALSE)` is non-negligible.

Value

`dfrag` gives the mass function, `pfrag` gives the distribution function, `qfrag` gives the quantile function, and `rfrag` generates random deviates.

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

See Also

`pgeom`, `sample`

Examples

```
plot( 0:300, table(factor(rfrag(2000),0:300)) )
lines( 0:300, 2000*dfrag(0:300) )
```

maxEM *maximum likelihood estimates relative abundances*

Description

From information about the lengths of sonicants of integration sites, infer the relative abundances of different clones and the distributon of sonicant lengths

Usage

```
maxEM(slmat, theta.var=FALSE, phi.update=NULL,
            phi.deriv=NULL, lframe = NULL, glm.frm = NULL, iter.control=NULL, ... )
```

Arguments

slmat	a matrix whose rows correspond to unique lengths with rownames indicating those lengths
theta.var	logical, return variance of theta estimates?
phi.update	a function of an object like slmat that returns estimtes of phi - a default vesrion is invoked if none is provided
phi.deriv	function of theta and phi that returns derivatives of phi wrt beta (its parameters)
lframe	a data.frame which will be used to estimate phi, the supplied function phi.update must know what to do with this arg or it will be ignored
glm.frm	a formula like y ~ bs(x,knots=c(50,100)) to fit phi
iter.control	a list of default values for iteration control - see maxEM.iter.control
...	possibly other args to pass to phi.update

Details

The EM method is used to infer the relative abundances of different sites.

Value

theta	a vector of the abundance estimates
phi	a vector of the probabilities of sonicant lengths
call	the call used

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

See Also

Ey.given.x

Examples

```
mat0 <- matrix(0,nr=48,nc=140)
vals <- c(rep(1,100),2:40,100)
mat1 <- sapply( vals, function(x) as.numeric(is.element(1:200 ,rgeom(x,.02))))
mat <- rbind(mat0,mat1)
posVals <- colSums(mat) > 0
vals <- vals[ posVals ]
mat <- mat[, posVals ]
rownames(mat) <- 1:nrow(mat)
res <- maxEM(mat)
matplot( vals, cbind(res$theta, colSums(mat)), pch=1:2,
        xlab='original values', ylab='estimated values',
        main='Simulated Sonicants and Estimates')
legend( "bottomright", pch=1:2, col=1:2,
        legend=c(expression(hat(theta)[j]),expression(sum(y[ij],i))))
abline(a=0,b=1,col='gray')
```

---

maxEM.iter.control *iteration controls*

Description

Set parameters for control of iteration in maxEM

Usage

```
maxEM.iter.control(min.reps = 3, max.reps = 2000, max.abs.le = 0.1,
max.rel.le = 1e-05, phi.min=.Machine$double.eps)
```

Arguments

min.reps	a positive integer
max.reps	a positive integer
max.abs.le	a smallish double
max.rel.le	a tiny double
phi.min	a smallish double

Details

Most users should not need to tinker with these settings

Value

a list such as formals(maxEM.iter.control)

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

mstep *M-step to maximize theta*

Description

This is a utility function for maxEM

Usage

```
mstep(x, theta, phi)
```

Arguments

x	a zero-one indicator matrix whose rows correspond to unique lengths with row-names indicating those lengths
theta	a vector of the abundance estimates. length(theta)==ncol(x)
phi	a vector of the probabilities of sonicant lengths. length(phi)==nrow(x)

Details

The M-step for theta is computed. Probably, there is no need to use this function directly, but just in case it is here.

Value

a vector like the input theta

Author(s)

Charles C. Berry

See Also

maxEM

Examples

```
mat <- diag(10)
mat[ ,10 ] <- 1.0
phi1 <- prop.table( rep(1,10))
theta1 <- 1:10
sonicLength:::mstep( mat, theta1,phi1)
```

`phi.update.lframe` *estimate phi*

Description

Estimate phi in a flexible manner as guided by `lframe` - usually involving multiple strata, and allowing for formulas other than the default choice

Usage

```
phi.update.lframe(obj, return.fit = FALSE, lframe, glm.frm = NULL, ...)
```

Arguments

`obj`	a matrix whose rowSums are used to estimate phi
`return.fit`	logical - the result of the fit should be returned instead of phi
`lframe`	A `data.frame` with columns `y`, `x`, `strata`, `orig` (or matching those in `glm.frm`) and for which `rownames(lframe)[lframe$orig]` match the rownames of `obj`.
`glm.frm`	a formula. If omitted `y ~ bs( x, knots= c(50,100) )` will be used.
`...`	curently not used

Details

fitting phi - the probabilities that a sonicant lands in a particular bin - is crucial to estimating theta, then number of sonicants when more than one lans in a bin. bins may be defined by sonicant lengths when there is just one sample. When there are multiple samples, then one may wish to estimate phi separately for each one. `lframe$strata` indicates separate sample and the default `glm.frm` will fit each one separately.

Value

a vector whose sum is 1.0 with one element for each row of obj

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

See Also

`maxEM`

simSonic *Simulate Sonicated Data*

Description

Sonication and Processing of DNA containing retroviral sequences yields genomic locations of retroviral insertion sites and a set of associated fragment lengths. The simulation depends on the expected number of fragments from each site and the distribution of lengths.

Usage

```
simSonic( theta, phi )
```

Arguments

theta	a vector of expected number of fragments from each site. If the vector has unqiue names, these will be used to label the locations
phi	a vector of probabilities (or a list containing a vector named phi) whose names are like 'rep len', where 'rep' defines the replicate and 'len' defines the sonicant length . More than one replicate can be used, In any case 1-exp(-phi%o%theta) determines the probability of observations in the different locaitons, of the different lengths, and in different replicates

Details

This object can provide the arguments used by estAbund

Value

a data.frame with columns locations, lengths, and replicates. See estAbund for more details.

Author(s)

Charles C. Berry <ccberry@users.r-forge.r-project.org>

Examples

```
theta <- seq(0.5,20.5,by=0.5)
phi <- prop.table(1:10)
names(phi) <- paste( 1 , 51:60 )
res <- simSonic( theta, phi )
head(res)
tail(res)
summary(res)
```

Index