

Package ‘Ppurple’

January 23, 2018

Title P(robabilistic)PU(rity)PL(oidy)E(stimation)
Operations

Version 0.2.0

Description Purity and ploidy inference from WGS, panel, WES from total read density and hets.

Depends R (>= 3.1.0),
GenomicRanges (>= 1.18),
gUtils,
bamUtils,
data.table (>= 1.9)

Imports IRanges (>= 2.0),
S4Vectors (>= 0.4),
GenomeInfoDb (>= 1.2),
parallel,
BiocGenerics(>= 0.12),
methods,
Matrix,
stringr

Suggests BSgenome.Hsapiens.UCSC.hg19,
testthat,
rtracklayer

License GPL-2

BugReports <http://github.com/mskilab/Ppurple>

LazyData true

RoxygenNote 6.0.1.9000

R topics documented:

hapseg	2
lfactorial	2
llnorm	3
llpois	3
log.sum.exp	4
ppem	4
ppemgrid	5
ppurple	5
ssegment	6
Index	7

hapseg	<i>Internal function doing simple implementation of Carter, Getz 2011 hapseg</i>
--------	--

Description

Phases grouped (i.e. presegmented) hets and computes cluster centers (lambdas) and expected sufficient statistics for joint poisson per group (i.e. segment)

Internal function

Usage

```
hapseg(hets)
```

Author(s)

Marcin Imielinski

lfactorial	<i>Internal function doing approximate log factorial</i>
------------	--

Description

Internal function

Usage

```
lfactorial(x)
```

Arguments

x	integer vector for which to compute approximate log factorial
---	---

Value

log factorial

Author(s)

Marcin Imielinski

`llnorm`*Internal function doing product of normal log likelihoods*

Description

Internal function

Usage

```
llnorm(x, sos, n, mu, sd)
```

Author(s)

Marcin Imielinski

`llpois`*Internal function doing product of poisson log likelihoods*

Description

Internal function

Usage

```
llpois(slfx, sx, n, lambda)
```

Arguments

<code>slfx</code>	numeric vector of sum log factorials of data
<code>sx</code>	numeric vector of sum of data
<code>n</code>	numeric vector of data ns
<code>lambda</code>	numeric vector of lambda parameters

Value

log poisson joint likelihood

Author(s)

Marcin Imielinski

<code>log.sum.exp</code>	<i>Internal function doing log sum exp</i>
--------------------------	--

Description

Internal function

Usage

```
## S3 method for class 'sum.exp'
log(x)
```

Arguments

`x` vector of log probabilities

Value

`log.sum.exp` of input

Author(s)

Marcin Imielinski

ppem	<i>ppem</i>
------	-------------

Description

Utility function taking in `segs = grid of total copy number segments x alpha x tau x k -> y, mu, sd, sos, sos.k, nbins` `segs.h = data.table storing grid of "high" and "low" haplotype segments x alpha x tau x k -> y.high, y.low, high.slfx, high.sx, low.slfx, low.sx, nbins`

and returns locally optimal solution with tolerance $\leq \text{tol}$ or after `max_iter` iterations initialized with `sd0` and copy number prior with `tau.sd = 1` the output is list with fields, each containing the following data.tables `$pp` = posterior probability of purity ploidy combo `alpha tau` `$segs` = grid of segments with final output `$segs.h` = grid of haplotype segments with final output `$pi.k` = mixing probability across total copy states for each `alpha tau` `$pi.kh` = mixing probability across haplotype specific copy states for each `alpha tau`

Usage

```
ppem(segs.grid, segs = NULL, segs.h = NULL,
      sd0 = segs.grid[!duplicated(j), sqrt(var(y))], sd0.k = 3,
      use.tot = TRUE, use.het = TRUE, use.uniform = FALSE, tol = 1,
      fix.sd = NULL, max_iter = 100, verbose = FALSE)
```

Author(s)

Marcin Imielinski

ppemgrid

ppemgrid internal function

Description

Given data.table of segs and segs.h populated with summary stats sets up grid across purities and ploidies and K copy states and deploys ppem on it returns posterior probability across purities x ploidy combos given data and optimal params

Usage

```
ppemgrid(purities = NULL, ploidies = NULL, pp = NULL, segs, segs.h,
         rho = 1, rho.h = 1, k.dist = 3, K = 20, verbose = TRUE)
```

Author(s)

Marcin Imielinski

ppurple

Probabilistic purity ploidy estimation

Description

Computes posterior probability of purity ploidy for data with coverage cov (granges with value \$y specifying coverage) hets with \$talt, \$tref, \$nref, \$nalt specifying tumor and normal alt and ref allelic counts, and (optional) segments via EM algorithm.

If segments not specified, then it is inferred from the segmentation of cov via CBS (using DNA copy)

EM begins with initial (user specified) grid of values and then refines at 10 fold (or value of refine arg, >1) around the modes, meaning that it reruns the calculation with a finer grid completely surrounding the modes to output the final solution which is a data.table mapping \$purity and \$ploidy values to a posterior probability \$p

Usage

```
ppurpl(cov, hets = NULL, segs = NULL, purities = seq(0, 1, 0.1),
       ploidies = seq(1, 5, 0.2), refine = 10, K = 20, verbose = TRUE,
       min.p = 1e-04, mc.cores = 1, numchunks = mc.cores)
```

Author(s)

Marcin Imielinski

`ssegment`*Internal function utilizing DNACopy to segment a coverage profile*

Description

Internal function utilizing DNACopy to segment a coverage profile

Usage

```
ssegment(tcov)
```

Arguments

`tcov` GRanges of binned genome-wide coverage

Value

GRanges of genomewise segments of piecewise constant coverage

Author(s)

Marcin Imielinski

Index

`hapseg`, [2](#)

`lfactorial`, [2](#)

`llnorm`, [3](#)

`llpois`, [3](#)

`log.sum.exp`, [4](#)

`ppem`, [4](#)

`ppemgrid`, [5](#)

`ppurpl` (*ppurple*), [5](#)

`ppurple`, [5](#)

`ssegment`, [6](#)