Package 'OncoPhase'

November 9, 2016

Type Package

Title SOMATIC MUTATION CELLULAR PREVALENCE COMPUTATION

Version 0.1

Date 2016-02-25

Description This package offers a direct method to accurately quantify the cellular prevalence of somatic mutations in cancer using phase information. The method utilizes three sources of information: the phasing information, the copy number variation, and the allele counts. The method is demonstrated to bring more capabilities in Cancer Genomic.

LazyData TRUE

License GPL-2

Imports limSolve

Suggests

#VignetteBuilder knitr

RoxygenNote 5.0.1

NeedsCompilation no

Author Donatien Chedom-Fotso [aut, cre],
Ahmed Ahmed [aut],
Christopher Yau [aut]

Maintainer Donatien Chedom-Fotso <donatien.chedom@well.ox.ac.uk>

R topics documented:

A-OncoPhase *OncoPhase: An R package for somatic mutations cellular prevalence quantification using haplotype phasing.*

Description

OncoPhase uses haplotype phase information when required to accurately compute mutational cellular prevalence. OncoPhase utilizes three sources of information: the phasing information, the copy number variation, and the allele counts. It takes as input a combination of phased SNV and SNP allele-specific sequence read counts and local allele-specific copy numbers to determine the proportion of cells harboring the SNV and compute specific and detailed mutation cellular prevalence for each of the following groups of cells:

Germ Germline cells having a normal genotype with no mutations and no copy number alteration at the considered locus.

Alt Cells harboring one alternative between the two somatic alterations. That is either only the SNV if C=1 (SNV occurred before SCNA) or only the SCNA if C=0 (SNV occurred after the SCNA).

Both Cells harboring both somatic alterations. That is the SNV and the SCNA

Details

OncoPhase can also compute the mutation cellular prevalence without requiring any nearby phased SNP when no phasing information is available or when explicitely specified by the choice of the mode. OncoPhase can be run under three different modes :

PhasedSNP Phasing information is required. The prevalence is computed relatively to a nearby Phased SNP whose allelic counts should be provided

SNVOnly The prevalence is computed using only the SNV information without the usage of any nearby SNP

Ultimate This is the default mode. For a given mutation, the method checks if the phasing information is required to compute an accurate cellular prevalence. If it is not, the SNVOnly mode is used. If instead the phasing information is required the mode is then set to PhasedSNP if allelic counts of a phased nearby SNP are provided. This is done by first computing the prevalence under the SNVOnly mode. If the data do not fit into this mode (hiogh residual of the linear model), then the prevalence is computed using PhasedSNP mode.

OncoPhase also infer the context establishing the temporal relationship between the SNV and the copy number lateration affecting the mutation locus. Two context exists :

C1 The SNV occured after the copy number alteration

C2 The SNV occured before the copy number alteration

The main functions of OncoPhase package are `getPrevalence` and `getSamplePrevalence`. For more detailed information on usage, see the package vignette, by typing `vignette("OncoPhase")`. All support questions should be emailed to the authors.

Author(s)

Donatien Chedom-Fotso, Ahmed Ahmed, Christopher Yau.

References

OncoPhase reference:

Chedom-Fotso Donatien, Ahmed Ashour Ahmed, and Christopher Yau. "OncoPhase: Quantification of somatic mutation cellular prevalence using phase information." bioRxiv (2016): 046631.

`getMatricesPhasedSNP` *Generate the matrices C, W and M from a set of parameters.*

Description

This is a generic function to generate the matrices of the linear system (see the paper) from the allele counts and the copy number information.

Usage

```
getMatricesPhasedSNP(varcounts_snv, refcounts_snv, major_cn, minor_cn,
  varcounts_snp, refcounts_snp, context, LocusCoverage = FALSE,
  CopyNumberFraction = FALSE)
```

Arguments

`varcounts_snv`	A count of alleles supporting the variant sequence of the somatic mutation
`refcounts_snv`	A count of alleles supporting the reference sequence of the somatic mutation
`major_cn`	major copy number at the locus of the mutation
`minor_cn`	minor copy number at the locus of the mutation
`varcounts_snp`	A count of alleles supporting the variant sequence of the Germline SNP
`refcounts_snp`	A count of alleles supporting the reference sequence of the Germline SNP
`context`	represents either the situation of a mutation which occurred after the CNV ("C1") or the context of a mutation which occurred before the CNV ("C2"). If not provided, the right context will be estimated from the input
`LocusCoverage`	when set to TRUE, the SNV locus coverage is estimated to the average coverage of the phased SNP and the variant allele fraction is the ratio of the variant allele count over the estimated locus coverage.
`CopyNumberFraction`	when set to TRUE, if the mode is PhasedSNP then copy number fraction is considered as input to the linear model is replacement of the phased SNP allele fraction.

Value

the matrices W, C and M for the linear system of prevalence computation.

Examples

```
Matrices = getMatricesPhasedSNP(3, 10,2,1,8,5,"C1")

print(Matrices)
#$context
#[1] "C1"
#
#$W
#          SNP       SNV
#SNP 0.6153846 0.0000000
#SNV 0.0000000 0.2307692
#
#$M
#    Germ Alt Both
#SNP    1   2    2
#SNV    0   0    1
#
#$C
#    Germ Alt Both
#SNP    2   3    3
#SNV    2   3    3
```

`getMatricesSNVOnly` *Generate the matrices C, W and M from a set of parameters under the mode "SNVOnly.*

Description

This is a generic function to generate the matrices of the linear system (see the paper) from the allele counts and the copy number information under the SNVOnly mode.

Usage

```
getMatricesSNVOnly(varcounts_snv, refcounts_snv, major_cn, minor_cn, context,
  sigma = NULL)
```

Arguments

`varcounts_snv`	A count of alleles supporting the variant sequence of the somatic mutation
`refcounts_snv`	A count of alleles supporting the reference sequence of the somatic mutation
`major_cn`	major Copy number at the locus of the mutation
`minor_cn`	minor copy number at the locus of the mutation
`context`	represents either the situation of a mutation which occurred after the CNV ("C1") or the context of a mutation which occurred before the CNV ("C2"). If not provided, the right context will be estimated from the input
`sigma`	Copy number of the parental chromosome harboring the mutation.

Value

the matrices W, C and M for the linear system of prevalence computation.

Examples

```
Matrices = getMatricesSNVOnly(3,10,2,1,"C1")

print(Matrices)
#$context
#[1] "C1"
#
#$W
#SNV
#SNV 0.2307692
#
#$M
#Germ Alt Both
#SNV    0   0    1
#
#$C
#Germ Alt Both
#SNV    2   3    3
```

`getPrevalence` *Computes cellular prevalence at a single mutation point*

Description

This is a generic function to compute the cellular prevalence of a somatic mutation point using OncoPhase method. The method computes the prevalence of the somatic mutation relatively to phased nearby SNPs whose prevalence are known to be 1. `getPrevalence` requires the allelic-information of the somatic mutation and the aggregated information of its Phased SNP but the function can also be run in the absence of phasing information (Ultimate mode) or nearby SNP (SNVOnly mode).

Usage

```
getPrevalence(varcounts_snv, refcounts_snv, major_cn, minor_cn,
  varcounts_snp = NULL, refcounts_snp = NULL, detail = FALSE,
  mode = "Ultimate", Trace = FALSE, LocusCoverage = FALSE,
  SomaticCountAdjust = FALSE, CopyNumberFraction = FALSE, Optimal = TRUE,
  NormalCellContamination = NULL, Context = NULL, SearchContext = TRUE,
  c2_max_residual_treshold = Inf, c1_ultimate_c2_replacing_treshold = 0.1,
  snvonly_max_treshold = 0.1)
```

Arguments

`varcounts_snv`	A count (or a vector of counts if multiple samples) of alleles supporting the variant sequence of the somatic mutation
`refcounts_snv`	A count (or a vector of counts if multiple samples) of alleles supporting the reference sequence of the somatic mutation
`major_cn`	major copy number (or a vector if multiple samples) at the locus of the mutation
`minor_cn`	minor copy number (or a vector if multiple samples) at the locus of the mutation

`varcounts_snp` A count (or a vector of counts if multiple samples) of alleles supporting the variant sequence of the Germline SNP

`refcounts_snp` A count (or a vector of counts if multiple samples) of alleles supporting the reference sequence of the Germline SNP

`detail` when set to FALSE, the function simply output the cellular prevalence of the somatic mutation. if set to TRUE, a detailed output is generated containing:

Context The inferred associated context : C1 if the SNV occurred after the copy number alteration or C2 if the SNV occurred after the CNA)

Prevalence The computed somatic mutation cellular prevalence

DetailedPrevvalence the detailed prevalence for each subpopulation of cells (germline cells (Germ), cells affected by one of the two genomic alterations (Alt), cells affected by both genomic alterations (Both)

solutionNorm The residual of the linear model representing the value of the minimized quadratic function at the solution, i.e. ||Ax-b||^2.

residualNorm Residuals from the constraints of the linear model(The sum of absolute values of solutionNorms of equalities and violated inequalities.)

Quality Quality of the prevalence calling. H if residual < 1e-05, F if residual < 1e-03 and L if residual > 1e-03

Alt_Prevalence Prevalence estimated by the model if the context were to be the alternative context

Alt_solutionNorm Residual of the linear modelunder the alternative context

Alt_residualNorm Constraints residual representing the sum of absolute values of solutionNorms of equalities and violated inequalities under the alternative context

CondensedPrevalence A colon separated list of the above fields (Context, Prevalence, Detailedprevalence and solutionNorm). The detailed prevalence are separated by "|"

lm_inputs Inputs to the linear models separated by "|" and containing the allele count supporting the variant at the SNV, Allele count supporting the reference at the SNV, major and minor copy number, Alleles counts supporting respectively the variant and the reference at the phased SNP if mode="PhasedSNP" and the context associated to the mutation.

lm_params Parameters of the linear model separated by "|" and containing the SNV allele fraction, the SNP allele fraction if mode= PhasedSNP, the copy number of the allele harboring the mutation (sigma)

lm_params Parameters of the linear model separated by "|" and containing the SNV allele fraction, the SNP allele fraction if mode= PhasedSNP, the copy number of the allele harboring the mutation (sigma)

`mode` The mode under which the prevalence is computed (default : Ultimate , alternatives modes are PhasedSNP and SNVOnly). Can also be provided as a numeric 0=SNVOnly, 1= PhasedSNP, 2=Ultimate

`Trace` if set to TRUE, print the trace of the computation.

`LocusCoverage` when set to TRUE, the SNV locus coverage is estimated to the average coverage of the phased SNP and the variant allele fraction is the ratio of the variant allele count over the estimated locus coverage.

`SomaticCountAdjust`
when set to 1, varcounts_snv and refcounts_snv might be adjusted if necessary so that they meet the rules varcounts_snv <= varcounts_snp, refcounts_snv >= refcounts_snp and varcounts_snv + refcounts_snv ~ Poiss(varcounts_snp + refcounts_snp). Not used if mode=SNVOnly.

`CopyNumberFraction`
: when set to TRUE, if the mode is PhasedSNP then copy number fraction is considered as input to the linear model is replacement of the phased SNP allele fraction.

`Optimal`
: If TRUE, the prevalence is computed under all combination of the options SomaticCountAdjust, LocusCoverage and NormalisedCount, the value with the lower residual is returned as the best prevalence

`NormalCellContamination`
: If provided, represents the rate of normal cells contaminations in the experiment.

`Context`
: if provided, the prevalence will be computed strictly under the given context, if not the prevalence is computed under both context and the one yielding the smallest solutionNorm is retained. Default : NULL

`SearchContext`
: When set to true, an optimal search of the context is done in ta region of values around the SNV allele fraction and SNP Allele fraction if mode= PhasedSNP.

`c2_max_residual_treshold`
: Maximum residual threshold under which the context C2 can be inferred. Default: INF.

`c1_ultimate_c2_replacing_treshold`
: Context C1 is inferred if its linear model residual is less than the specified threshold. Default: 0.1.

`snvonly_max_treshold`
: Maximum threshold the linear model under SNVOnly is considered valid. Is the residual is greater than the value, and PhasedSNP is considered in case the phasing information are available.

Details

Germ Cells having a germline genotype at the locus of the SNV. That is No SNV, no SCNA

Alt Cells having one alternative of the two somatic alteration. That is either the SCNA, either the SNV not both.

Both Cells having both somatic alterations. That is the SNV and the SCNA

OncoPhase can be run under three modes:

PhasedSNP Phasing information is required. The prevalence is computed relatively to a nearby Phased SNP whose allelic counts should be provided

SNVOnly The prevalence is computed using only the SNV information without the usage of any nearby SNP

Ultimate This is the default mode. For a given mutation, the method checks if the phasing information is required to compute an accurate cellular prevalence. If it is not, the SNVOnly mode is used. If instead the phasing information is required the mode is then set to PhasedSNP if allelic counts of a phased nearby SNP are provided. This is done by first computing the prevalence under the SNVOnly mode. If the data do not fit into this mode (hiogh residual of the linear model), then the prevalence is computed using PhasedSNP mode.

Value

The cellular prevalence if detail =0, a detailed output if detail = 1, and a condensed output if detail =2. See the usage of the parameter detail above.

See Also

`getPrevalence`, `getSamplePrevalence`, `getSinglePhasedSNPPrevalence`, `getSingleSNVOnlyPrevalence`

Examples

```
#Example 1
prevalence=getPrevalence(5,10,3,1,16,8)
print(prevalence)
# 0.86
#The above example under mode the mode ultimate compute the prevalence under SNVOnly.
#We can set the mode to PhasedSNP and force the usage of the phasing information.
prevalence=getPrevalence(5,10,3,1,16,8,mode="PhasedSNP")
print(prevalence)
# 0.56

#Example 2
prevalence = getPrevalence(varcounts_snv=2,refcounts_snv=8,major_cn=2,minor_cn=1,
varcounts_snp=8, refcounts_snp=6, detail=TRUE)
print(prevalence)
# Context
# [1] "C1"
#
# $Prevalence
# Both
# 0.55
#
# $DetailedPrevalence
# Germ  Alt Both
# 0.26 0.19 0.55
#
# $solutionNorm
# [1] 6.933348e-33
#
# $residualNorm
# [1] 0
#
# $Quality
# [1] "H"
#
# $Alt_Prevalence
# [1] 0.45
#
# $Alt_solutionNorm
# [1] 4.506676e-31
#
# $Alt_residualNorm
# [1] 6.661338e-16
#
# $CondensedPrevalence
# [1] "C1:0.55:0.26|0.19|0.55:6.93334779979405e-33"
#
# $lm_inputs
# [1] "2:8:2:1:C1"
#
# $lm_params
# [1] "0.2:NA:3"
#
# $Mode
# [1] "SNVOnly"
```

```
#Example: 3
prevalence =  getPrevalence(13,5,2,0,47,3,detail=TRUE)
print(prevalence)


# $Context
# [1] "C1"
#
# $Prevalence
# Both
# 0.52
#
# $DetailedPrevalence
# Germ  Alt Both
# 0.12 0.36 0.52
#
# $solutionNorm
# [1] 5.4e-32
#
# $residualNorm
# [1] 2.2e-16
#
# $Quality
# [1] "H"
#
# $Alt_Prevalence
# [1] 0.38
#
# $Alt_solutionNorm
# [1] 0.31
#
# $Alt_residualNorm
# [1] 6.7e-16
#
# $CondensedPrevalence
# [1] "C1:0.52:0.12|0.36|0.52:5.4e-32"
#
# $lm_inputs
# [1] "13:5:2:0:47:3:C1"
#
# $lm_params
# [1] "0.26:0.94:2:2"
#
# $Mode
# [1] "PhasedSNP"
#
#' # Example 4:
prevalence= getPrevalence(varcounts_snv=c(6,4,6),refcounts_snv=c(8,8,14),major_cn=c(2,2,2),
minor_cn=c(1,0,1),varcounts_snp=c(8,8,8), refcounts_snp=c(6,4,12))
print(prevalence)
#Sample_1 Sample_2 Sample_3
#1.00     0.67     0.86
#
# Example 5:
prevalence= getPrevalence(c(6,4,6),c(8,8,14),c(2,2,2),c(1,0,1),c(8,8,8), c(6,4,12), mode="PhasedSNP")
print(prevalence)
# Sample_1 Sample_2 Sample_3
```

```
#0.66     0.67     0.90
```

`getSamplePrevalence` *Somatic mutations cellular prevalence on a sample set of mutations*

Description

This function computes the cellular prevalence of a list of somatic mutations of a tumor. The function applies OncoPhase linear model to a range of mutations located at a given genomic region or at the whole genome scale. It invokes the function `getPrevalence` to compute the cellular prevalence for each mutation of the set. When phasing information are available, the method can computes the prevalence of a somatic mutation relatively to phased germline SNP under the mode "PhasedSNP". If the phasing information are not available the mode "SNVOnly" will be used to derive the cellular prevalence. as specified in `getPrevalence`.

Usage

```
getSamplePrevalence(input_df, mode = "Ultimate", nbFirstColumns = 0,
  region = NULL, detail = TRUE, LocusCoverage = FALSE,
  SomaticCountAdjust = FALSE, CopyNumberFraction = FALSE,
  NormalCellContamination = NULL, Optimal = TRUE,
  c2_max_residual_treshold = Inf, c1_ultimate_c2_replacing_treshold = 0.1,
  snvonly_max_treshold = 0.1)
```

Arguments

`input_df`	A data frame containing for each mutation the following information (columns or fields) : **varcounts_snv** Allele counts supporting the SNV **refcounts_snv** Allele counts supporting the reference at the SNV locus **major_cn** Major copy number at the SNV locus **minor_cn** Minor copy number at the SNV locus **varcounts_snp** (Optional) Allele counts supporting the nearby phased SNP. Required if mode= PhasedSNP **refcounts_snp** (Optional) Allele counts supporting the reference at the nearby phased SNP. Required if mode= PhasedSNP
`mode`	The mode under which the prevalence is computed (Default : Ultimate , alternatives methods are PhasedSNP and SNVOnly). Can also be provided as a numeric 0=SNVOnly, 1= PhasedSNP, 2=Ultimate.
`nbFirstColumns`	Number of first columns in input_df to reproduce in the output dataframe e.g: Chrom, Pos, Vartype. Columns from nbFirstColumns +1 to the last column should contains the information needed for the prevalence computation.
`region`	The region of the genome to consider for the prevalence computation in the format chrom:start-end e.g "chr22:179800-98767.
`detail`	when set to TRUE, a detailed output is generated containing, the context and the detailed prevalence for each group of cells (germline cells, cells affected by one of the two genomic alterations SNV or copy number alteration and cells affected by both copy number alteration and SNV). The residual and the linear models inputs and parameters are also reported.

`LocusCoverage` when set to TRUE, the SNV locus coverage is estimated to the average coverage of the phased SNP and the variant allele fraction is the ratio of the variant allele count over the estimated locus coverage.

`SomaticCountAdjust`
when set to TRUE, varcounts_snv and refcounts_snv might be adjusted if necessary so that they meet the reqirements varcounts_snv <= varcounts_snp, refcounts_snv >= refcounts_snp and varcounts_snv + refcounts_snv ~ Poiss(varcounts_snp + refcounts_snp). Not used if mode=SNVOnly.

`CopyNumberFraction`
when set to TRUE, if the mode is PhasedSNP then copy number fraction is considered as input to the linear model is replacement of the phased SNP allele fraction.

`NormalCellContamination`
If provided, represents the rate of normal cells contaminations in the experiment.

`Optimal` The model will be run under different configurations of the parameters LocusCoverage, SomaticCountAdjust and CopyNumberFraction, they configuration yielding the optimal residual is then selected and returned.

`c2_max_residual_treshold`
Maximum residual threshold under which the context C2 can be inferred.

`c1_ultimate_c2_replacing_treshold`
Context C1 is inferred if its linear model residual is less than the specified threshold.

`snvonly_max_treshold`
Maximum threshold the linear model under SNVOnly is considered valid. Is the residual is greater than the value, then PhasedSNP is considered in case the phasing information are available.

Value

A data frame containing :

Column 1 to NbFirstcolumn of the input data frame input_df. This will generally include the chromosome and the position of the mutation plus any other columns to report in the prevalence dataframe (e.g REF and ALL sequences, ...)

and the following information

Prevalence The Cellular Prevalence of the mutation

Germ The proportion of cells with a normal genotype

Alt The proportion of cells with only the CNA if the context C=C1 or with only the SNV if the context C=C2

Both The proportion of cells with both the SNV and the SCNA

Context Context at the mutation. If C1 then the SNV occurred after the SCNA, if C=c2 then the SNV occurred before the SCNA

solutionNorm Residual of the linear model.

residualNorm Constraints residual representing the sum of absolute values of solutionNorms of equalities and violated inequalities.

Quality Quality of the prevalence calling. H if residual < 1e-05, F if residual < 1e-03 and L if residual > 1e-03

Alt_Prevalence Prevalence estimated by the model if the context were to be the alternative context

Alt_solutionNorm Residual of the linear modelunder the alternative context

Alt_residualNorm Constraints residual representing the sum of absolute values of solution-Norms of equalities and violated inequalities under the alternative context

Mode The mode considered for the cellular prevalence computation (either SNVOnly of PhasedSNP)

Examples

```
#Example 1:

input_file=system.file("extdata","phylogeny1_d300_n80.tsv", package = "OncoPhase")
input_df<-read.table(input_file,header=TRUE)
rownames(input_df) = input_df$mutation_id
print(input_df)
#  mut_id varcounts_snv refcounts_snv major_cn minor_cn varcounts_snp refcounts_snp
#a      a      151  152        1        1      151  135
#b      b      123  176        1        1      161  150
#c      c       94  209        2        1      176  134
#d      d       23  283        1        1      155  144
#e      e       60  228        2        0      174  125

prevalence_df=getSamplePrevalence(input_df,nbFirstColumns = 1)

print(prevalence_df)
#mutation_id Prevalence   Germ    Alt   Both Context solutionNorm residualNorm Quality Alt_Prevalence
#a         a    0.9967 0.0017 0.0017 0.9967     C1 7.718183e-32 2.220446e-16      H       0.9966
#b         b    0.8230 0.0890 0.0890 0.8230     C1 1.925930e-32 0.000000e+00      H       0.8200
#c         c    0.9000 0.1000 0.0000 0.9000     C1 5.238529e-32 2.220446e-16      H       0.7300
#d         d    0.1500 0.4200 0.4200 0.1500     C1 1.972152e-31 4.440892e-16      H       0.1500
#e         e    0.4200 0.2900 0.2900 0.4200     C1 5.007418e-32 2.220446e-16      H       0.7100
#Alt_solutionNorm Alt_residualNorm         InputValues    Mode      lm_inputs lm_params
#a     1.222984e-32     0.000000e+00 151:152:1:1:151:135 SNVOnly 151:152:1:1:C1  0.5:NA:2
#b     5.623715e-32     2.220446e-16 123:176:1:1:161:150 SNVOnly 123:176:1:1:C1 0.41:NA:2
#c     6.933348e-33     0.000000e+00  94:209:2:1:176:134 SNVOnly  94:209:2:1:C1 0.31:NA:3
#d     3.081488e-33     0.000000e+00  23:283:1:1:155:144 SNVOnly  23:283:1:1:C1 0.08:NA:2
#e     5.007418e-32     2.220446e-16  60:228:2:0:174:125 SNVOnly  60:228:2:0:C1 0.21:NA:2


#'@seealso \code{\link{getPrevalence}}
```

`getSinglePhasedSNPPrevalence` *Compute the cellular prevalence of each group of cells*

Description

This is a generic function to compute the detailed prevalence of a single mutation using the linear system of the model.

Usage

```
getSinglePhasedSNPPrevalence(varcounts_snv, refcounts_snv, major_cn, minor_cn,
  varcounts_snp, refcounts_snp, context, Trace = FALSE,
  LocusCoverage = TRUE, NormalCellContamination = NULL,
  CopyNumberFraction = FALSE)
```

Arguments

varcounts_snv	A count of alleles supporting the variant sequence of the somatic mutation
refcounts_snv	A count of alleles supporting the reference sequence of the somatic mutation
major_cn	major copy number at the locus of the mutation
minor_cn	minor copy number (or a vector of copy number if multiple tumor samples)
varcounts_snp	A count of alleles supporting the variant sequence of the Germline SNP
refcounts_snp	A count of alleles supporting the reference sequence of the Germline SNP
context	represents either the situation of a mutation which occurred after the CNV ("C1") or the context of a mutation which occurred before the CNV ("C2"). If not provided, the right context will be estimated from the input
Trace	Print a trace of the eecution.
LocusCoverage	when set to TRUE, the SNV locus coverage is estimated to the average coverage of the phased SNP and the variant allele fraction is the ratio of the variant allele count over the estimated locus coverage.
NormalCellContamination	If provided, represents the rate of normal cells contaminations in the experiment.
CopyNumberFraction	when set to TRUE, if the mode is PhasedSNP then copy number fraction is considered as input to the linear model is replacement of the phased SNP allele fraction.

Value

A list of the three cellular prevalence of each of the three groups of cells

See Also

getPrevalence, getMatricesPhasedSNP

Examples

```
Prevalences = getSinglePhasedSNPPrevalence(3, 10,2,1,8,5,"C1")

 print(Prevalences)
# Germ  Alt Both
# 0.4  0.0  0.6
```

---

getSingleSNVOnlyPrevalence

Compute the cellular prevalence of each group of cells in case of SNVOnly mode

---

Description

This is a generic function to compute the detailed prevalence of a single mutation using the linear system making the model.

Usage

```
getSingleSNVOnlyPrevalence(varcounts_snv, refcounts_snv, major_cn, minor_cn,
  context, sigma = NULL, Trace = FALSE, NormalCellContamination = NULL)
```

Arguments

varcounts_snv	A count of alleles supporting the variant sequence of the somatic mutation
refcounts_snv	A count of alleles supporting the reference sequence of the somatic mutation
major_cn	major copy number at the locus of the mutation
minor_cn	minor copy number (or a vector of copy number if multiple tumor samples)
context	represents either the situation of a mutation which occurred after the CNV ("C1") or the context of a mutation which occurred before the CNV ("C2"). If not provided, the right context will be estimated from the input
sigma	The parental copy number of the chromosome harboring the mutation locus. Only needed if the context = C2. Should be either the major copy number either minor copy number
Trace	If TRUE, a trace of the execution will be printed
NormalCellContamination	If provided, represents the rate of normal cells contaminations in the experiment.

Value

A list of the three cellular prevalence of each of the three groups of cells

See Also

getPrevalence, getMatricesSNVOnly

Examples

```
Prevalences = getSingleSNVOnlyPrevalence(3,10,2,1,"C2",2)

print(Prevalences)
#Germ      Alt     Both solutionNorm
#0.60     0.31     0.09     0.00
```

Index