Package 'EstMix'

September 13, 2018

Type Package

Title Tumor Clones Percentage Estimations

Version 1.0.1

Author Xuan You <youxuan90@gmail.com>, Yichen Cheng <ycheng11@gsu.edu>

Maintainer Xuan You <youxuan90@gmail.com>

Description Includes R functions for the estimation of tumor clones percentages for both snp data and (whole) genome sequencing data. See Cheng, Y., Dai, J. Y., Paulson, T. G., Wang, X., Li, X., Reid, B. J., & Kooperberg, C. (2017). Quantification of multiple tumor clones using gene array and sequencing data. The Annals of Applied Statistics, 11(2), 967-991, <doi:10.1214/17-AOAS1026> for more details.

License GPL (>= 2)

Encoding UTF-8

LazyLoad yes

LazyData yes

Imports Rcpp (>= 0.12.12), PSCBS

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-09-13 04:20:02 UTC

R topics documented:

est_mixture	*It is a function that takes the LRR obtained from SNP array data and returns the estimated tumor and normal proportions. Currently, the function can performs the proportion estimations by assuming the number of tumor clones to be 1 or 2 or 3. The normalization step is not required and the normalization constant will be returned by this function. The function will output two sets of solutions corresponding to the top 2 optimal solutions based on the posterior distribution. You can choose according to your expertise the one that is more reasonable.*

Description

It is a function that takes the LRR obtained from SNP array data and returns the estimated tumor and normal proportions. Currently, the function can performs the proportion estimations by assuming the number of tumor clones to be 1 or 2 or 3. The normalization step is not required and the normalization constant will be returned by this function. The function will output two sets of solutions corresponding to the top 2 optimal solutions based on the posterior distribution. You can choose according to your expertise the one that is more reasonable.

Usage

```
est_mixture(BAF, LRR, chr, x, GT, seg_raw = "NA", num_tumor = 1)
```

Arguments

BAF	a numeric vector containing the B Allele Frequency for the sample, corresponding to the location (chr, x).
LRR	a numveric vector containing the Log R ratio for the sample, corresponding to the location (chr, x). In practice, the LRR values you include should be the raw LRR output devided by 0.55.
chr	a factor vector containing the chromosome.
x	a numeric vector containing the location on the chromosome, measured by base pair.
GT	a factor vector containing the genotype. Possible values are "AA", "AB", "BB" and NA.
seg_raw	Optional. A dataframe containing the segmentaiton results. If not supplied, function `segmentByPairedPSCBS` from package PSCBS will be used to obtain the segmentation. You can also use the `segmentByPairedPSCBS` function to preprocess your data set and obtain the segmentation results and use that and the input. (On examples about how to obtain the segmentation results beforehand, please see the examples section below.)
num_tumor	1 or 2 or 3, indicating the number of tumor clones. 1 indicates a mixture for a normal and one tumor clone. 2 indicates a mixture for a normal and 2 tumors and so on. Default value is set to be 1.

Value

sol1_pct	the estimated percentages for all tumor clones for optimal solution 1. Each value is between 0 and 100.
sol1_scale	a scaler that provide the normalization constant for LRR for optimal solution 1. That is 2*2^LRR/scale will be on the same scale as the copy number.
sol1_cn1	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 1 for the optimal solution.
sol1_cn2	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 2 for the optimal solution.
sol1_pscn1	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 1 for the optimal solution.
sol1_pscn2	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 2 for the optimal solution.
sol2_pct	the estimated percentages for all tumor clones for optimal solution 2. Each value is between 0 and 100.
sol2_scale	a scaler that provide the normalization constant for LRR for optimal solution 2. That is 2*2^LRR/scale will be on the same scale as the copy number.
sol2_cn1	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 1 for the second optimal solution.
sol2_cn2	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 2 for the second optimal solution.
sol2_pscn1	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 1 for the second optimal solution.
sol2_pscn2	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 2 for the second optimal solution.

Examples

```
############################################################
##
## short example
##
############################################################
## first load the data
BAF <- example_data$BAF
LRR <- example_data$LRR ## In practice, the orignal LRR should be devided by 0.55
chr <- example_data$chr
loc <- example_data$x
GT <- example_data$GT
gt = (GT=='BB')*2+(GT=='AB')*1.5+(GT=='AA')-1;gt[gt==(-1)]=NA

## then perform segmentation
gaps = PSCBS::findLargeGaps(x=loc,minLength=5e6,chromosome=chr)
if(!is.null(gaps)) knownSegments = PSCBS::gapsToSegments(gaps)
p <- 0.0001
fit <- PSCBS::segmentByPairedPSCBS(CT=2*2^LRR,betaT=BAF,muN=gt,chrom=chr,
```

```
knownSegments=knownSegments,tbn=FALSE,x=loc,seed=1, alphaTCN=p*.9,alphaDH=p*.1)
seg_eg = fit$output

## then perform tumor mixture estimation by assuming 1 tumor clones
out = est_mixture(BAF, LRR, chr, loc, GT, num_tumor = 1, seg_raw = seg_eg)
out$sol1_pct
out$sol1_scale
## References: Quantification of multiple tumor clones using gene array and sequencing data.
## Y Cheng, JY Dai, TG Paulson, X Wang, X Li, BJ Reid, C Kooperberg.
## Annals of Applied Statistics 11 (2), 967-991
## Segmentation-based detection of allelic imbalance and loss-of-heterozygosity
## in cancer cells using whole genome SNP arrays.
## J Staaf, D Lindgren, J Vallon-Christersson, A Isaksson, H Goransson, G Juliusson,
## R Rosenquist, M H, A Borg, and M Ringner
```

`est_mixture_wgs` *It is a function that takes the count data obtained from whole genome sequencing (WGS) data and returns the estimated tumor and normal proportions. Currently, the function can performs the proportion estimations by assuming the number of tumor clones to be 1 or 2. The normalization step is not required and the normalization constant will be returned by this function. The function will output two sets of solutions corresponding to the top 2 optimal solutions based on the posterior distribution. You can choose according to your expertise the one that is more reasonable.*

Description

It is a function that takes the count data obtained from whole genome sequencing (WGS) data and returns the estimated tumor and normal proportions. Currently, the function can performs the proportion estimations by assuming the number of tumor clones to be 1 or 2. The normalization step is not required and the normalization constant will be returned by this function. The function will output two sets of solutions corresponding to the top 2 optimal solutions based on the posterior distribution. You can choose according to your expertise the one that is more reasonable.

Usage

```
est_mixture_wgs(exp_data, normal_snp, tumor_snp, f_path, num_tumor = 1)
```

Arguments

`exp_data`	a string. It provides the file name of interval. exp_data.intervals should be the name of the interval file. For the format of this file, please see the example section. The file should contain 6 and only 6 columns with each column corresponds to "ID","chrm","start","end","tumorCount" and "normalCount". It is very important to keep the order of the columns the same as listed.
`normal_snp`	a string. It provides the file name of WGS count data for a normal sample or a control sample.

tumor_snp	a string. It provides the file name of WGS count data for the tumor sample.
f_path	a string. It provides the absolute path of the folder that contains the files above.
num_tumor	1 or 2, indicating the number of tumor clones. 1 indicates a mixture for a normal and one tumor clone. 2 indicates a mixture for a normal and 2 tumors and so on. Default value is set to be 1.

Value

sol1_pct	the estimated percentages for all tumor clones for optimal solution 1. Each value is between 0 and 100.
sol1_scale	sol1_scale: a scaler that provide the normalization constant for LRR for optimal solution 1. That is 2*tumor_count/normal_count will be on the same scale as the copy number.
sol1_cn1	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 1 for the optimal solution.
sol1_cn2	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 2 for the optimal solution.
sol1_pscn1	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 1 for the optimal solution.
sol1_pscn2	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 2 for the optimal solution.
sol2_pct	the estimated percentages for all tumor clones for optimal solution 2. Each value is between 0 and 100.
sol2_scale	sol1_scale: a scaler that provide the normalization constant for LRR for optimal solution 2. That is 2*tumor_count/normal_count will be on the same scale as the copy number.
sol2_cn1	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 1 for the second optimal solution.
sol2_cn2	a vector of length S, where S is the number of segments. It is the estimated copy number for tumor 2 for the second optimal solution.
sol2_pscn1	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 1 for the second optimal solution.
sol2_pscn2	a vector of length S, where S is the number of segments. It is the estimated parent specifit copy number for tumor 2 for the second optimal solution.

Examples

```
exp_data = "data_exp_eg" ## exp_data.intervals should be the file name of the segments.
## For the format of the input files, you can use the example code below.
normal_snp = "snp_norm_eg" ## snp_norm_eg.txt should be the count file name for the normal sample.
tumor_snp = "snp_tum_eg" ## snp_tum_eg.txt should be the count file name for the tumor sample.
f_path = system.file("extdata",package="EstMix")
## f_path should be the absolute path of folder that contains the txt and interval files.
out_wgs = est_mixture_wgs(exp_data, normal_snp, tumor_snp,f_path,num_tumor = 1)
out_wgs$sol1_pct
out_wgs$sol1_scale
```

```
## for the format of the input files, please see the following code
data_exp_path = file.path(f_path, paste("/", exp_data, ".intervals", sep=""))
snp_norm_path = file.path(f_path, paste("/",normal_snp, ".txt", sep=""))
snp_tumor_path = file.path(f_path, paste("/",tumor_snp, ".txt", sep=""))
data_exp = read.table(data_exp_path);
colnames(data_exp) = c("ID","chrm","start","end","tumorCount","normalCount")
snp_norm = read.table(snp_norm_path)
snp_tum = read.table(snp_tumor_path)

## References: Quantification of multiple tumor clones using gene array and sequencing data.
## Y Cheng, JY Dai, TG Paulson, X Wang, X Li, BJ Reid, C Kooperberg.
## Annals of Applied Statistics 11 (2), 967-991
```

Index