Package 'DeMixT'

November 5, 2026

Title Cell type-specific deconvolution of heterogeneous tumor samples with two or three components using expression data from RNAseq or microarray platforms

Version 2.1.0

Date 2026-03-20

Description DeMixT is a software package that performs deconvolution on transcriptome data from a mixture of two or three components.

LazyData TRUE

Depends R (>= 4.0.0), parallel, Rcpp (>= 1.0.0), SummarizedExperiment

Imports matrixStats, stats, truncdist, base64enc, ggplot2, KernSmooth, matrixcalc, sva, dendextend, fitdistrplus, pbapply, psych, magrittr, graphics, grDevices, S4Vectors

Suggests knitr, rmarkdown, calibrate, BiocStyle

LinkingTo Rcpp

NeedsCompilation yes

VignetteBuilder knitr

biocViews Software, StatisticalMethod, Classification, GeneExpression, Sequencing, Microarray, TissueMicroarray, Coverage

License GPL-3

RoxygenNote 7.3.3

Encoding UTF-8

git_url https://git.bioconductor.org/packages/DeMixT

git_branch devel

git_last_commit 7874c43

git_last_commit_date 2026-04-28

Repository Bioconductor 3.24

Date/Publication 2026-11-05

Author Zeya Wang [aut],
Shaolong Cao [aut],
Liyang Xie [aut],
Ruonan Li [cre],
Wenyi Wang [aut]

Maintainer Ruonan Li <RLi10@mdanderson.org>

Contents

batch_correction . . . 2
DeMixNB . . . 3
DeMixNB_preprocessing . . . 4
DeMixT . . . 5
DeMixT_DE . . . 9
DeMixT_GS . . . 11
DeMixT_preprocessing . . . 14
DeMixT_S2 . . . 15
detect_suspicious_sample_by_hierarchical_clustering_2comp . . . 17
gene_selection_DE . . . 18
Optimum_KernelC . . . 20
plot_dim . . . 22
scale_normalization_75th_percentile . . . 23
simulate_2comp . . . 24
simulate_3comp . . . 25
subset_sd . . . 26
subset_sd_gene_remaining . . . 28
test.data.2comp . . . 29
test.data.3comp . . . 30

Index **32**

`batch_correction` *batch_correction*

Description

Batch effect correction for multiple batches of tumor samples using ComBat

Usage

```
batch_correction(count.matrix, batch_labels)
```

Arguments

`count.matrix`	A matrix of raw expression count with G by (My), where G is the number of genes, My is the number of mixed tumor samples. Row names are genes column names are tumor sample ids.
`batch_labels`	Factor of tumor samples from different batches

Value

Batch effect corrected count matrix for tumor samples

Examples

```
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- paste0("S", 1:20)
rownames(mat) <- paste0("G", 1:100)
batch_labels <- factor(c(rep("batch1", 10), rep("batch2", 10)))
result <- batch_correction(mat, batch_labels)
```

DeMixNB *Deconvolution of heterogeneous tumor samples with two components using expression data from RNAseq or microarray platforms*

Description

DeMixNB is a variant of DeMixT such that the expressions are assumed to follow the negative binomial function

Usage

```
DeMixNB(
  data.Y,
  data.N1,
  niter = 30,
  nspikein = 200,
  tol = 1e-08,
  nthread = 1,
  seeds = c(123, 456, 789),
  output.more.info = FALSE,
  consistency_threshold = 0.1,
  data.scale = 1
)
```

Arguments

data.Y	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
niter	The maximum number of iterations used in the algorithm of iterated conditional modes. A larger value better guarantees the convergence in estimation but increases the running time. The default is 30.
nspikein	The number of spikes in normal reference used for proportion estimation. The default value is $min(200, My)$, where My the number of mixed samples. If it is set to 0, proportion estimation is performed without any spike in normal reference.
tol	The convergence criterion. The default is 10^(-8).
nthread	The number of cores to use for parallelization. The default is 1, i.e. no parallelization at the top level. Note that OpenMP is used under the hood when calculating the complete likelihood.
seeds	A numeric vector of random seeds to use for each run. The DeMixNB algorithm will be run once for each seed. Default is c(123, 456, 789).
output.more.info	A boolean flag controlling whether to return the estimations in each iteration. Default is False.

`consistency_threshold`
: The maximum allowed difference (range) between pi estimates across runs for a spot to be considered "consistent". Default is 0.1.

`data.scale`
: The constant scale to be multiplied to the data. Default is 1.

Value

`pi_t_summary`
: A data frame containing the estimated tumor proportions for each run and a consistency flag at each sample indicating whether the runs agree with each other based on the maximum difference.

`pi_n_summary`
: A data frame containing the estimated non-tumor proportions for each run.

`all_run_results`
: A list containing all the detailed outputs of each run, including the mu and phi estimates.

Author(s)

Liyang Xie

Examples

```
# Example: estimate proportions for simulated two-component data
  data(test.data.2comp)
# res.NB = DeMixNB(data.Y = test.data.2comp$data.Y,
#                    data.N1 = test.data.2comp$data.N1,
#                    niter = 10, tol = 10^(-5))
```

DeMixNB_preprocessing *DeMixNB_preprocessing*

Description

DeMixNB preprocessing in one go

Usage

```
DeMixNB_preprocessing(
  count.matrix,
  normal.id,
  tumor.id,
  selected.genes = 9000,
  cutoff_normal_range = c(0.1, 1),
  cutoff_tumor_range = c(0, 2.5),
  cutoff_step = 0.2
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids
selected.genes	A integer number indicating the number of genes selected before running DeMixNB
cutoff_normal_range	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_tumor_range	A vector of two numeric values, indicating the lower and upper bounds to search standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_step	A scatter value indicating the step size of changing cutoff_normal_range and cutoff_tumor_range to find a suitable subset of count matrix for downstream analysis

Value

processed count matrix

Examples

```
set.seed(123)
mat <- matrix(rpois(10000, lambda = 50), nrow = 500, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:500)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- DeMixNB_preprocessing(mat, normal.id, tumor.id,
                                selected.genes = 100,
                                cutoff_normal_range = c(0.1, 1.0),
                                cutoff_tumor_range = c(0, 2.5),
                                cutoff_step = 0.1)
```

DeMixT *Deconvolution of heterogeneous tumor samples with two or three components using expression data from RNAseq or microarray platforms*

Description

DeMixT is a software that performs deconvolution on transcriptome data from a mixture of two or three components.

Usage

```
DeMixT(
  data.Y,
  data.N1,
  data.N2 = NULL,
  niter = 10,
  nbin = 50,
  if.filter = TRUE,
  filter.sd = 0.5,
  ngene.selected.for.pi = NA,
  mean.diff.in.CM = 0.25,
  nspikein = NULL,
  gene.selection.method = "GS",
  ngene.Profile.selected = NA,
  tol = 10^(-5),
  output.more.info = FALSE,
  pi01 = NULL,
  pi02 = NULL,
  nthread = parallel::detectCores() - 1
)
```

Arguments

data.Y	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
data.N2	A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. Component 2 is needed only for running a three-component model.
niter	The maximum number of iterations used in the algorithm of iterated conditional modes. A larger value better guarantees the convergence in estimation but increases the running time. The default is 10.
nbin	The number of bins used in numerical integration for computing complete likelihood. A larger value increases accuracy in estimation but increases the running time, especially in a three-component deconvolution problem. The default is 50.
if.filter	The logical flag indicating whether a predetermined filter rule is used to select genes for proportion estimation. The default is TRUE.
filter.sd	The cut-off for the standard deviation of lognormal distribution. Genes whose log transferred standard deviation smaller than the cut-off will be selected into the model. The default is 0.5.
ngene.selected.for.pi	The percentage or the number of genes used for proportion estimation. The difference between the expression levels from mixed tumor samples and the known component(s) are evaluated, and the most differential expressed genes are selected, which is called DE. It is enabled when if.filter = TRUE. The default

is $min(1500, 0.3 * G)$, where G is the number of genes. Users can also try using more genes, ranging from $0.3 * G$ to $0.5 * G$, and evaluate the outcome.

`mean.diff.in.CM`
: Threshold of expression difference for selecting genes in the component merging strategy. We merge three-component to two-component by selecting genes with similar expressions for the two known components. Genes with the mean differences less than the threshold will be selected for component merging. It is used in the three-component setting, and is enabled when if.filter = TRUE. The default is 0.25.

`nspikein`
: The number of spikes in normal reference used for proportion estimation. The default value is $min(200, 0.3 * My)$, where My the number of mixed samples. If it is set to 0, proportion estimation is performed without any spike in normal reference.

`gene.selection.method`
: The method of gene selection used for proportion estimation. The default method is 'GS', which applies a profile likelihood based method for gene selection. If it is set to 'DE', the most differential expressed genes are selected.

`ngene.Profile.selected`
: The number of genes used for proportion estimation ranked by profile likelihood. The default is $min(1500, 0.1 * G)$, where G is the number of genes. This is enabled only when gene.selection.method is set to 'GS'.

`tol`
: The convergence criterion. The default is 10^(-5).

`output.more.info`
: The logical flag indicating whether to show the estimated proportions in each iteration in the output.

`pi01`
: Initialized proportion for first kown component. The default is $Null$ and pi01 will be generated randomly from uniform distribution.

`pi02`
: Initialized proportion for second kown component. pi02 is needed only for running a three-component model. The default is $Null$ and pi02 will be generated randomly from uniform distribution.

`nthread`
: The number of threads used for deconvolution when OpenMP is available in the system. The default is the number of whole threads minus one. In our no-OpenMP version, it is set to 1.

Value

`pi`
: A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings, and each column corresponds to one sample.

`pi.iter`
: Estimated proportions in each iteration. It is a $niter * My * p$ array, where p is the number of components. This is enabled only when output.more.info = TRUE.

`ExprT`
: A matrix of deconvolved expression profiles corresponding to T-component in mixed samples for a given subset of genes. Each row corresponds to one gene and each column corresponds to one sample.

`ExprN1`
: A matrix of deconvolved expression profiles corresponding to N1-component in mixed samples for a given subset of genes. Each row corresponds to one gene and each column corresponds to one sample.

ExprN2	A matrix of deconvolved expression profiles corresponding to N2-component in mixed samples for a given subset of genes in a three-component setting. Each row corresponds to one gene and each column corresponds to one sample.
Mu	A matrix of estimated Mu of log2-normal distribution for both known ($MuN1, MuN2$) and unknown component (MuT). Each row corresponds to one gene.
Sigma	Estimated $Sigma$ of log2-normal distribution for both known ($SigmaN1, SigmaN2$) and unknown component ($SigmaT$). Each row corresponds to one gene.
gene.name	The names of genes used in estimating the proportions. If no gene names are provided in the original data set, the genes will be automatically indexed.

Author(s)

Zeya Wang, Wenyi Wang

References

Wang Z, Cao S, Morris J S, et al. Transcriptome Deconvolution of Heterogeneous Tumor Samples with Immune Infiltration. iScience, 2018, 9: 451-460.

See Also

http://bioinformatics.mdanderson.org/main/DeMixT

Examples

```
# Example 1: simulated two-component data by using GS(gene selection method)
  data(test.data.2comp)
# res <- DeMixT(data.Y = test.data.2comp$data.Y,
#               data.N1 = test.data.2comp$data.N1,
#               data.N2 = NULL, nspikein = 50,
#               gene.selection.method = 'GS',
#               niter = 10, nbin = 50, if.filter = TRUE,
#               ngene.selected.for.pi = 150,
#               mean.diff.in.CM = 0.25, tol = 10^(-5))
# res$pi
# head(res$ExprT, 3)
# head(res$ExprN1, 3)
# head(res$Mu, 3)
# head(res$Sigma, 3)
#
# Example 2: simulated two-component data by using DE(gene selection method)
# data(test.data.2comp)
# res <- DeMixT(data.Y = test.data.2comp$data.Y,
#               data.N1 = test.data.2comp$data.N1,
#               data.N2 = NULL, nspikein = 50, g
#               ene.selection.method = 'DE',
#               niter = 10, nbin = 50, if.filter = TRUE,
#               ngene.selected.for.pi = 150,
#               mean.diff.in.CM = 0.25, tol = 10^(-5))
#
# Example 3: three-component mixed cell line data applying
# component merging strategy
# data(test.data.3comp)
# res <- DeMixT(data.Y = test.data.3comp$data.Y,
#               data.N1 = test.data.3comp$data.N1,
```

```
#               data.N2 = test.data.3comp$data.N2,
#               if.filter = TRUE)
#
# Example: convert a matrix into the SummarizedExperiment format
# library(SummarizedExperiment)
# example <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2, ncol = 3, byrow = TRUE)
# example.se <- SummarizedExperiment(assays = list(counts = example))
```

`DeMixT_DE` *Estimates the proportions of mixed samples for each mixing component*

Description

This function is designed to estimate the deconvolved expressions of individual mixed tumor samples for unknown component for each gene.

Usage

```
DeMixT_DE(
  data.Y,
  data.N1,
  data.N2 = NULL,
  niter = 10,
  nbin = 50,
  if.filter = TRUE,
  filter.sd = 0.5,
  ngene.selected.for.pi = NA,
  nspikein = NULL,
  mean.diff.in.CM = 0.25,
  tol = 10^(-5),
  pi01 = NULL,
  pi02 = NULL,
  nthread = parallel::detectCores() - 1
)
```

Arguments

`data.Y`	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
`data.N1`	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
`data.N2`	A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. Component 2 is needed only for running a three-component model.

niter — The maximum number of iterations used in the algorithm of iterated conditional modes. A larger value better guarantees the convergence in estimation but increases the running time. The default is 10.

nbin — The number of bins used in numerical integration for computing complete likelihood. A larger value increases accuracy in estimation but increases the running time, especially in a three-component deconvolution problem. The default is 50.

if.filter — The logical flag indicating whether a predetermined filter rule is used to select genes for proportion estimation. The default is TRUE.

filter.sd — The cut-off for the standard deviation of lognormal distribution. Genes whose log transferred standard deviation smaller than the cut-off will be selected into the model. The default is 0.5.

ngene.selected.for.pi — The percentage or the number of genes used for proportion estimation. The difference between the expression levels from mixed tumor samples and the known component(s) are evaluated, and the most differential expressed genes are selected, which is called DE. It is enabled when if.filter = TRUE. The default is $min(1500, 0.3*G)$, where G is the number of genes. Users can also try using more genes, ranging from $0.3 * G$ to $0.5 * G$, and evaluate the outcome.

nspikein — The number of spikes in normal reference used for proportion estimation. The default value is $min(200, 0.3 * My)$, where My the number of mixed samples. If it is set to 0, proportion estimation is performed without any spike in normal reference.

mean.diff.in.CM — Threshold of expression difference for selecting genes in the component merging strategy. We merge three-component to two-component by selecting genes with similar expressions for the two known components. Genes with the mean differences less than the threshold will be selected for component merging. It is used in the three-component setting, and is enabled when if.filter = TRUE. The default is 0.25.

tol — The convergence criterion. The default is 10^(-5).

pi01 — Initialized proportion for first kown component. The default is $Null$ and pi01 will be generated randomly from uniform distribution.

pi02 — Initialized proportion for second kown component. pi02 is needed only for running a three-component model. The default is $Null$ and pi02 will be generated randomly from uniform distribution.

nthread — The number of threads used for deconvolution when OpenMP is available in the system. The default is the number of whole threads minus one. In our no-OpenMP version, it is set to 1.

Value

pi — A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings, and each column corresponds to one sample.

pi.iter — Estimated proportions in each iteration. It is a $niter * Ny * p$ array, where p is the number of components. This is enabled only when output.more.info = TRUE.

gene.name — The names of genes used in estimating the proportions. If no gene names are provided in the original data set, the genes will be automatically indexed.

Author(s)

Zeya Wang, Wenyi Wang

References

Wang Z, Cao S, Morris J S, et al. Transcriptome Deconvolution of Heterogeneous Tumor Samples with Immune Infiltration. iScience, 2018, 9: 451-460.

See Also

http://bioinformatics.mdanderson.org/main/DeMixT

Examples

```
# Example 1: estimate proportions for simulated two-component data
# with spike-in normal reference
  data(test.data.2comp)
# res.DE = DeMixT_DE(data.Y = test.data.2comp$data.Y,
#                    data.N1 = test.data.2comp$data.N1,
#                    niter = 10, nbin = 50, nspikein = 50,
#                    if.filter = TRUE,
#                    mean.diff.in.CM = 0.25, ngene.selected.for.pi = 150,
#                    tol = 10^(-5))
#
# Example 2: estimate proportions for simulated two-component data
# without spike-in normal reference
# data(test.data.2comp)
# res.DE = DeMixT_DE(data.Y = test.data.2comp$data.Y,
#                    data.N1 = test.data.2comp$data.N1,
#                    niter = 10, nbin = 50, nspikein = 0,
#                    if.filter = TRUE,
#                    mean.diff.in.CM = 0.25, ngene.selected.for.pi = 150,
#                    tol = 10^(-5))
#
# Example 3: estimate proportions for simulated three-component
# mixed cell line data
# data(test.data.3comp)
# res.DE <- DeMixT_DE(data.Y = test.data.3comp$data.Y,
#                     data.N1 = test.data.3comp$data.N1,
#                     data.N2 = test.data.3comp$data.N2,
#                     if.filter = TRUE)
```

`DeMixT_GS` *Estimates the proportions of mixed samples for each mixing component using profile likelihood gene selection*

Description

This function is designed to estimate the proportions of all mixed samples for each mixing component with a new proposed profile likelihood based gene selection, which can select most identifiable genes as reference gene sets to achieve better model fitting quality. We first calculated the Hessian matrix of the parameter spaces and then derive the confidence interval of the profile likelihood of

each gene. We then utilized the length of confidence interval as a metric to rank the identifiability of genes. As a result, the proposed gene selection approach can improve the tumor-specific transcripts proportion estimation.

Usage

```
DeMixT_GS(
  data.Y,
  data.N1,
  data.N2 = NULL,
  niter = 10,
  nbin = 50,
  if.filter = TRUE,
  filter.sd = 0.5,
  ngene.Profile.selected = NA,
  ngene.selected.for.pi = NA,
  mean.diff.in.CM = 0.25,
  nspikein = NULL,
  tol = 10^(-5),
  pi01 = NULL,
  pi02 = NULL,
  nthread = parallel::detectCores() - 1
)
```

Arguments

`data.Y`	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
`data.N1`	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
`data.N2`	A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. Component 2 is needed only for running a three-component model.
`niter`	The maximum number of iterations used in the algorithm of iterated conditional modes. A larger value better guarantees the convergence in estimation but increases the running time. The default is 10.
`nbin`	The number of bins used in numerical integration for computing complete likelihood. A larger value increases accuracy in estimation but increases the running time, especially in a three-component deconvolution problem. The default is 50.
`if.filter`	The logical flag indicating whether a predetermined filter rule is used to select genes for proportion estimation. The default is TRUE.
`filter.sd`	The cut-off for the standard deviation of lognormal distribution. Genes whose log transferred standard deviation smaller than the cut-off will be selected into the model. The default is TRUE.
`ngene.Profile.selected`	The number of genes used for proportion estimation ranked by profile likelihood. The default is $min(1500, 0.1 * G)$, where G is the number of genes.

`ngene.selected.for.pi`
: The percentage or the number of genes used for proportion estimation. The difference between the expression levels from mixed tumor samples and the known component(s) are evaluated, and the most differential expressed genes are selected, which is called DE. It is enabled when if.filter = TRUE. The default is $min(1500, 0.3 * G)$, where G is the number of genes. Users can also try using more genes, ranging from $0.3 * G$ to $0.5 * G$, and evaluate the outcome.

`mean.diff.in.CM`
: Threshold of expression difference for selecting genes in the component merging strategy. We merge three-component to two-component by selecting genes with similar expressions for the two known components. Genes with the mean differences less than the threshold will be selected for component merging. It is used in the three-component setting, and is enabled when if.filter = TRUE. The default is 0.25.

`nspikein`
: The number of spikes in normal reference used for proportion estimation. The default value is $min(200, 0.3 * My)$, where My the number of mixed samples. If it is set to 0, proportion estimation is performed without any spike in normal reference.

`tol`
: The convergence criterion. The default is 10^(-5).

`pi01`
: Initialized proportion for first kown component. The default is $Null$ and pi01 will be generated randomly from uniform distribution.

`pi02`
: Initialized proportion for second kown component. pi02 is needed only for running a three-component model. The default is $Null$ and pi02 will be generated randomly from uniform distribution.

`nthread`
: The number of threads used for deconvolution when OpenMP is available in the system. The default is the number of whole threads minus one. In our no-OpenMP version, it is set to 1.

Value

`pi`
: A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings, and each column corresponds to one sample.

`pi.iter`
: Estimated proportions in each iteration. It is a $niter * My * p$ array, where p is the number of components. This is enabled only when output.more.info = TRUE.

`gene.name`
: The names of genes used in estimating the proportions. If no gene names are provided in the original data set, the genes will be automatically indexed.

Note

A Hessian matrix file will be created in the working directory and the corresponding Hessian matrix with an encoded name from the mixed tumor sample data will be saved under this file. If a user reruns this function with the same dataset, this Hessian matrix will be loaded to in place of running the profile likelihood method and reduce running time.

Author(s)

Shaolong Cao, Zeya Wang, Wenyi Wang

References

Gene Selection and Identifiability Analysis of RNA Deconvolution Models using Profile Likelihood. Manuscript in preparation.

See Also

http://bioinformatics.mdanderson.org/main/DeMixT

Examples

```
# Example 1: estimate proportions for simulated two-component data
# with spike-in normal reference
  data(test.data.2comp)
# res.GS = DeMixT_GS(data.Y = test.data.2comp$data.Y,
#                    data.N1 = test.data.2comp$data.N1,
#                    niter = 10, nbin = 50, nspikein = 50,
#                    if.filter = TRUE, ngene.Profile.selected = 150,
#                    mean.diff.in.CM = 0.25, ngene.selected.for.pi = 150,
#                    tol = 10^(-5))
#
# Example 2: estimate proportions for simulated two-component data
# without spike-in normal reference
# data(test.dtat.2comp)
# res.GS = DeMixT_GS(data.Y = test.data.2comp$data.Y,
#                    data.N1 = test.data.2comp$data.N1,
#                    niter = 10, nbin = 50, nspikein = 0,
#                    if.filter = TRUE, ngene.Profile.selected = 150,
#                    mean.diff.in.CM = 0.25, ngene.selected.for.pi = 150,
#                    tol = 10^(-5))
```

DeMixT_preprocessing *DeMixT_preprocessing*

Description

DeMixT preprocessing in one go

Usage

```
DeMixT_preprocessing(
  count.matrix,
  normal.id,
  tumor.id,
  selected.genes = 9000,
  cutoff_normal_range = c(0.1, 1),
  cutoff_tumor_range = c(0, 2.5),
  cutoff_step = 0.2
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids
selected.genes	A integer number indicating the number of genes selected before running DeMixT with the GS (Gene Selection) method
cutoff_normal_range	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_tumor_range	A vector of two numeric values, indicating the lower and upper bounds to search standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_step	A scatter value indicating the step size of changing cutoff_normal_range and cutoff_tumor_range to find a suitable subset of count matrix for downstream analysis

Value

processed count matrix

Examples

```
set.seed(123)
mat <- matrix(rpois(10000, lambda = 50), nrow = 500, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:500)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- DeMixT_preprocessing(mat, normal.id, tumor.id,
                               selected.genes = 100,
                               cutoff_normal_range = c(0.1, 1.0),
                               cutoff_tumor_range = c(0, 2.5),
                               cutoff_step = 0.1)
```

DeMixT_S2 *Deconvolves expressions of each individual sample for unknown component*

Description

This function is designed to estimate the deconvolved expressions of individual mixed tumor samples for unknown component for each gene.

Usage

```
DeMixT_S2(
  data.Y,
  data.N1,
  data.N2 = NULL,
  givenpi,
  nbin = 50,
  nthread = parallel::detectCores() - 1
)
```

Arguments

data.Y	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
data.N2	A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. Component 2 is needed only for running a three-component model.
givenpi	A vector of proportions for all mixed tumor samples. In two-component analysis, it gives the proportions of the unknown reference component, and in three-component analysis, it gives the proportions for the two known components.
nbin	Number of bins used in numerical integration for computing complete likelihood. A larger value increases accuracy in estimation but increases the running time, especially in a three-component deconvolution problem. The default is 50.
nthread	The number of threads used for deconvolution when OpenMP is available in the system. The default is the number of whole threads minus one. In our no-OpenMP version, it is set to 1.

Value

decovExprT	A matrix of deconvolved expression profiles corresponding to T-component in mixed samples for a given subset of genes. Each row corresponds to one gene and each column corresponds to one sample.
decovExprN1	A matrix of deconvolved expression profiles corresponding to N1-component in mixed samples for a given subset of genes. Each row corresponds to one gene and each column corresponds to one sample.
decovExprN2	A matrix of deconvolved expression profiles corresponding to N2-component in mixed samples for a given subset of genes in a three-component setting. Each row corresponds to one gene and each column corresponds to one sample.
decovMu	A matrix of estimated Mu of log2-normal distribution for both known ($MuN1, MuN2$) and unknown component (MuT). Each row corresponds to one gene.
decovSigma	Estimated $Sigma$ of log2-normal distribution for both known ($SigmaN1, SigmaN2$) and unknown component ($SigmaT$). Each row corresponds to one gene.

Author(s)

Zeya Wang, Wenyi Wang

References

Wang Z, Cao S, Morris J S, et al. Transcriptome Deconvolution of Heterogeneous Tumor Samples with Immune Infiltration. iScience, 2018, 9: 451-460.

See Also

http://bioinformatics.mdanderson.org/main/DeMixT

Examples

```
# Example 1: two-component deconvolution given proportions
  data(test.data.2comp)
  givenpi <- c(t(as.matrix(test.data.2comp$pi[-2,])))
  res.S2 <- DeMixT_S2(data.Y = test.data.2comp$data.Y,
                      data.N1 = test.data.2comp$data.N1,
                      data.N2 = NULL,
                      givenpi = givenpi,
                      nbin = 50)
#
# Example 2: three-component deconvolution given proportions
# data(test.data.3comp)
# givenpi = c(t(test.data.3comp$pi[-3,]))
# res <- DeMixT_S2(data.Y = test.data.3comp$data.Y,
#                  data.N1 = test.data.3comp$data.N1,
#                  data.N2 = test.data.3comp$data.N2,
#                  givenpi = givenpi,
#                  nbin = 50)
```

---

detect_suspicious_sample_by_hierarchical_clustering_2comp

detect_suspicious_sample_by_hierarchical_clustering_2comp

Description

Detect suspicious samples by a hierarchical clustering

This function is designed to evaluate the separation of tumor samples and normal samples in a PCA space. If some normal samples appear in the tumor-sample dominated cluster, these normal samples are likely to be tumor samples and they are supposed to be filtered out before downstream analysis. But for those tumor samples appearing in the normal-sample dominated cluster, we do not remove them since they may be the ones with low tumor purity.

Plot the standard deviation of log2 raw expression

Usage

```
detect_suspicious_sample_by_hierarchical_clustering_2comp(
  count.matrix,
  normal.id,
  tumor.id
)

plot_sd(count.matrix, normal.id, tumor.id)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids

Value

list object

None (creates plots)

Examples

```
set.seed(123)
mat <- matrix(rpois(10000, lambda = 50), nrow = 500, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:500)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- detect_suspicious_sample_by_hierarchical_clustering_2comp(
  mat, normal.id, tumor.id)
```

---

gene_selection_DE *gene_selection_DE*

Description

Perform gene selection by first normalizing the log-transformed data and doing the differential expressions analysis.

Usage

```
gene_selection_DE(
  count.matrix,
  normal.id,
  tumor.id,
  ngene.selected.for.normalization = 9000,
  cutoff_normal_range = c(0.1, 1),
  cutoff_tumor_range = c(0, 2.5),
```

```
  cutoff_step = 0.2,
  filter.sd = NA,
  ngene.selected.for.pi = NA
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids
ngene.selected.for.normalization	A integer number indicating the number of genes selected before running DeMixNB
cutoff_normal_range	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.1, 0.6)
cutoff_tumor_range	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the tumor samples to subset. Default is c(0.2, 0.8)
cutoff_step	A scatter value indicating the step size of changing cutoff_normal_range and cutoff_tumor_range to find a suitable subset of count matrix for downstream analysis
filter.sd	The cut-off for the standard deviation of lognormal distribution. Genes whose log transferred standard deviation smaller than the cut-off will be selected into the model. The default is TRUE.
ngene.selected.for.pi	The percentage or the number of genes used for proportion estimation. The difference between the expression levels from mixed tumor samples and the known component(s) are evaluated, and the most differential expressed genes are selected, which is called DE. It is enabled when if.filter = TRUE. The default is $min(1500, 0.3*G)$, where G is the number of genes. Users can also try using more genes, ranging from $0.3 * G$ to $0.5 * G$, and evaluate the outcome.

Value

The selected genes

Examples

```
set.seed(123)
mat <- matrix(rpois(10000, lambda = 50), nrow = 500, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:500)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
selected.genes <- gene_selection_DE(mat, normal.id, tumor.id,
                                    ngene.selected.for.normalization = 100,
                                    ngene.selected.for.pi = 50)
```

`Optimum_KernelC` *Kernel function for optimizing parameters and hidden variables in DeMixT*

Description

This function is invoked by DeMixT_GS or DeMixT_DE and DeMixT_S2 to finish parameter estimation by iterated conditional mode algorithm and reconstitute gene expression profile of all components.

Usage

```
Optimum_KernelC(
  inputdata,
  groupid,
  nspikein,
  setting.pi,
  givenpi,
  givenpiT,
  niter,
  ninteg,
  tol,
  sg0 = 0.5^2,
  mu0 = 0,
  pi01 = NULL,
  pi02 = NULL,
  nthread = 1
)
```

Arguments

`inputdata`	A matrix of expression data (e.g gene expressions) from reference (e.g. normal) and mixed samples (e.g. mixed tumor samples). It is a $G * M$ matrix where G is the number of genes and M is the number of samples including reference and mixed samples. Samples with the same tissue type should be placed together in columns (e.g. cbind(normal amples, mixed tumor samples).
`groupid`	A vector of indicators to denote if the corresponding samples are reference samples or mixed tumor samples. DeMixT is able to deconvolve mixed tumor samples with at most three components. We use 1 and 2 to denote the samples referencing the first and the second known component in mixed tumor samples. We use 3 to indicate mixed tumor samples prepared to be deconvolved. For example, in two-component deconvolution, we have c(1,1,...,3,3) and in three-component deconvolution, we have c(1,1,...,2,2,...,3,3).
`nspikein`	The number of spikes in normal reference used for proportion estimation. The default value is $min(200, 0.3 * My)$, where My the number of mixed tumor samples. If it is set to 0, proportion estimation is performed without any spike in normal reference.
`setting.pi`	If it is set to 0, then deconvolution is performed without any given proportions; if set to 1, deconvolution with given proportions for the first and the second known component is run; if set to 2, deconvolution is run with given tumor proportions.

This option helps to perform deconvolution in different settings. In estimation of component-specific proportions, we use a subset of genes ; so when it is required to deconvolve another subset of genes, we just easily plug back our estimated proportions by setting this option to 1. In our two-step estimation strategy in a three-component setting, this option is set to 2 to implement the second step.

`givenpi` ST-Vector of proportions. Given the number of mixed tumor samples is $My(My < M)$, ST is set to $2 * My$ in a three-component setting and My in a two-component setting. When setting.pi is 1, it is fixed with the given proportions for the first and the second known component of mixed tumor samples, or for one unknown component when there is just one type of reference tissues. It has the form of Vector $PiN1-1$, $PiN1-2$, ..., $PiN1-My$, $PiN2-1$, $PiN2-2$, ..., $PiN2-My$.

`givenpiT` ST-Vector of proportions. When setting.pi is set to 2, givenpiT is fixed with given proportions for unknown component of mixed tumor samples. This option is used when we adopt a two-step estimation strategy in deconvolution. It has the form of Vector $PiT-1$, $PiT-2$, ..., $PiT-My$. If option is not 2, this vector can be given with any element.

`niter` The number of iterations used in the algorithm of iterated conditional modes. A larger value can better guarantee the convergence in estimation but increase the computation time.

`ninteg` The number of bins used in numerical integration for computing complete likelihood. A larger value can increase accuracy in estimation but also increase the running time. Especially in three-component deconvolution, the increase of number of bins can greatly lengthen the running time.

`tol` The convergence criterion. The default is 10^(-5).

`sg0` Initial value for σ^2. The default is 0.5^2.

`mu0` Initial value for μ. The default is 0.

`pi01` Initialized proportion for first kown component. The default is $Null$ and pi01 will be generated randomly from uniform distribution.

`pi02` Initialized proportion for second kown component. pi02 is needed only for running a three-component model. The default is $Null$ and pi02 will be generated randomly from uniform distribution.

`nthread` The number of threads used for deconvolution when OpenMP is available in the system. The default is the number of whole threads minus one. In our no-OpenMP version, it is set to 1.

Value

`pi` Matrix of estimated proportions for each known component. The first row corresponds to the proportion estimate of each sample for the first known component (groupid = 1) and the second row corresponds to that for the second known component (groupid = 2).

`decovExpr` A matrix of deconvolved expression profiles corresponding to unknown (e.g tumor) component in mixed samples for a given subset of genes. Each row corresponds to one gene and each column corresponds to one sample.

`decovMu` Estimated Mu of log2-normal distribution for tumor component.

`decovSigma` Estimated $Sigma$ of log2-normal distribution for tumor component.

`pi1` An $My * I$ matrix of estimated proportions for each iteration, where I is the number of iteration, for the first known component.

pi2 An $My * I$ matrix of estimated proportions for each iteration, where I is the number of iteration, for the second known component.

Author(s)

Zeya Wang, Wenyi Wang

References

Wang Z, Cao S, Morris J S, et al. Transcriptome Deconvolution of Heterogeneous Tumor Samples with Immune Infiltration. iScience, 2018, 9: 451-460.

See Also

http://bioinformatics.mdanderson.org/main/DeMixT

Examples

```
# Example 1: simulated two-component data
  data(test.data.2comp)
# data.N1 <- SummarizedExperiment::assays(test.data.2comp$data.N1)[[1]]
# data.Y <- SummarizedExperiment::assays(test.data.2comp$data.Y)[[1]]
# inputdata <- cbind(data.N1, data.Y)
# groupid <- c(rep(1, ncol(data.N1)), rep(3, ncol(data.Y)))
# nspikein <- 0
# Optimum_KernelC(inputdata, groupid,
#                 nspikein = nspikein, setting.pi = 0,
#                 givenpi = rep(0, 2 * ncol(data.y)),
#                 niter = 10, ninteg = 30, tol = 10^(-4))
```

---

plot_dim *plot_dim*

Description

Plot the distribution of tumor and normal samples in a 2D PCA space based on their expressions

Usage

```
plot_dim(
  count.matrix,
  labels,
  legend.position = "bottomleft",
  legend.cex = 1.2
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
labels	A vector of sample labels for plotting
legend.position	Position of legend in the plot. Default is bottomleft.
legend.cex	Character expansion factor relative to current par("cex"). Default = 1.2

Value

None (create plots)

Examples

```
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:100)
labels <- c(rep("Normal", 10), rep("Tumor", 10))
plot_dim(mat, labels)
```

---

scale_normalization_75th_percentile

scale_normalization_75th_percentile

Description

Quantile normalization for the raw count matrix of tumor and normal reference using the 0.75 quantile scale normalization

Quantile normalization for the raw count matrix of tumor and normal reference using the 0.75 quantile scale normalization

Usage

```
scale_normalization_75th_percentile(count.matrix)

scale_normalization_75th_percentile(count.matrix)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.

Value

the scale normalized count matrix

the scale normalized count matrix

Examples

```
mat <- matrix(rpois(200, lambda = 50), nrow = 20, ncol = 10)
colnames(mat) <- paste0("S", 1:10)
rownames(mat) <- paste0("G", 1:20)
result <- scale_normalization_75th_percentile(mat)
mat <- matrix(rpois(200, lambda = 50), nrow = 20, ncol = 10)
colnames(mat) <- paste0("S", 1:10)
rownames(mat) <- paste0("G", 1:20)
result <- scale_normalization_75th_percentile(mat)
```

---

simulate_2comp *Function to simulate two-component test data*

Description

Function to simulate two-component test data for DeMixT.

Usage

```
simulate_2comp(G = 500, My = 100, M1 = 100, output.more.info = FALSE)
```

Arguments

G	Number of genes for simulation.
My	Number of mixture tumor samples for simulation.
M1	Number of normal reference for simulation.
output.more.info	The logical flag indicating wheter to show True.data.T and True.data.N1 in the output. The default is FALSE.

Value

pi	A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings. Each column corresponds to one sample.
Mu	Simulated Mu of log2-normal distribution for both known ($MuN1$) and unknown component (MuT).
Sigma	Simulated $Sigma$ of log2-normal distribution for both known ($SigmaN1$) and unknown component ($SigmaT$).
data.Y	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.

`True.data.T` A SummarizedExperiment object of simulated tumor expression data. It is a G by My matrix, where G is the number of genes and My is the number of mixed samples. This is enabled only when output.more.info = TRUE.

`True.data.N1` A SummarizedExperiment object of simulated true expression data for reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1. This is enabled only when output.more.info = TRUE.

Examples

```
test.data = simulate_2comp(G = 500, My = 100, M1 = 100)
test.data$pi
test.data$Mu
test.data$Sigma
```

---

simulate_3comp *Function to simulate three-component mixed cell line test data*

Description

Function to simulate three-component mixed cell line test data used in DeMixT function.

Usage

```
simulate_3comp(
  G1 = 675,
  G2 = 25,
  My = 20,
  M1 = 100,
  M2 = 100,
  output.more.info = FALSE
)
```

Arguments

`G1` Number of genes, where μ_{N1} is close to μ_{N2}.

`G2` Number of genes, where μ_{N1} is not close to μ_{N2}.

`My` Number of mixture tumor samples for simulation.

`M1` Number of first known reference for simulation.

`M2` Number of second known reference for simulation.

`output.more.info` The logical flag indicating wheter to show True.data.T, True.data.N1 and True.data.N2 in the output. The default is FALSE.

Value

`pi` A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings. Each column corresponds to one sample.

Mu	Simulated Mu of log2-normal distribution for both known ($MuN1, MuN2$) and unknown component (MuT).
Sigma	Simulated $Sigma$ of log2-normal distribution for both known ($SigmaN1, SigmaN2$) and unknown component ($SigmaT$).
data.Y	A SummarizedExperiment object of simulated expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of simulated expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
data.N2	A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2.
True.data.T	A SummarizedExperiment object of simulated tumor expression data. It is a G by My matrix, where G is the number of genes and My is the number of mixed samples.This is enabled only when output.more.info = TRUE.
True.data.N1	A SummarizedExperiment object of simulated true expression data for reference component 1 (e.g., stroma). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1. This is enabled only when output.more.info = TRUE.
True.data.N2	A SummarizedExperiment object of simulated true expression data for reference component 2 (e.g., immue). It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. This is enabled only when output.more.info = TRUE.

Examples

```
test.data = simulate_3comp(G1 = 675, G2 = 25, My = 20, M1 = 100, M2 = 100)
test.data$pi
test.data$Mu
test.data$Sigma
```

---

subset_sd *subset_sd*

Description

Subset a count matrix given the the ranges of the standard deviations of the log2 expressions from the tumor and normal samples.

Subset a count matrix given the the ranges of the standard deviations of the log2 expressions from the tumor and normal samples

Usage

```
subset_sd(
  count.matrix,
  normal.id,
  tumor.id,
```

```
  cutoff_normal = c(0.1, 0.6),
  cutoff_tumor = c(0.2, 0.8)
)

subset_sd(
  count.matrix,
  normal.id,
  tumor.id,
  cutoff_normal = c(0.1, 0.6),
  cutoff_tumor = c(0.2, 0.8)
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids
cutoff_normal	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.1, 0.6)
cutoff_tumor	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the tumor samples to subset. Default is c(0.2, 0.8)

Value

A subset of the count matrix

A subset of the count matrix

Examples

```
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:100)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- subset_sd(mat, normal.id, tumor.id,
                    cutoff_normal = c(0.1, 0.6),
                    cutoff_tumor = c(0.1, 0.8))
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:100)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- subset_sd(mat, normal.id, tumor.id,
                    cutoff_normal = c(0.1, 0.6),
                    cutoff_tumor = c(0.1, 0.8))
```

subset_sd_gene_remaining *subset_sd_gene_remaining*

Description

Find the cutoffs to filter out genes with large standard deviations of log2 expressions in both normal and tumor samples

Find the cutoffs to filter out genes with large standard deviations of log2 expressions in both normal and tumor samples

Usage

```
subset_sd_gene_remaining(
  count.matrix,
  normal.id,
  tumor.id,
  cutoff_normal_range = c(0.2, 0.6),
  cutoff_tumor_range = c(0.2, 0.8),
  cutoff_step = 0.2
)

subset_sd_gene_remaining(
  count.matrix,
  normal.id,
  tumor.id,
  cutoff_normal_range = c(0.2, 0.6),
  cutoff_tumor_range = c(0.2, 0.8),
  cutoff_step = 0.2
)
```

Arguments

count.matrix	A matrix of raw expression count with G by $(My+M1)$, where G is the number of genes, My is the number of mixed samples and $M1$ is the number of normal samples. Row names are genes column names are sample ids.
normal.id	A vector of normal sample ids
tumor.id	A vector of tumor sample ids
cutoff_normal_range	A vector of two numeric values, indicating the lower and upper bounds of standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_tumor_range	A vector of two numeric values, indicating the lower and upper bounds to search standard deviation of log2 count matrix from the normal samples to subset. Default is c(0.2, 0.6)
cutoff_step	A scatter value indicating the step size of changing cutoff_normal_range and cutoff_tumor_range to find a suitable subset of count matrix for downstream analysis

Value

A numeric vector of the optimal cutoff values for normal and tumor standard deviations

A numeric vector of the optimal cutoff values for normal and tumor standard deviations

Examples

```
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:100)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- subset_sd_gene_remaining(mat, normal.id, tumor.id,
                                   cutoff_normal_range = c(0.1, 0.6),
                                   cutoff_tumor_range = c(0.1, 0.8),
                                   cutoff_step = 0.1)
set.seed(123)
mat <- matrix(rpois(2000, lambda = 50), nrow = 100, ncol = 20)
colnames(mat) <- c(paste0("N", 1:10), paste0("T", 1:10))
rownames(mat) <- paste0("G", 1:100)
normal.id <- paste0("N", 1:10)
tumor.id <- paste0("T", 1:10)
result <- subset_sd_gene_remaining(mat, normal.id, tumor.id,
                                   cutoff_normal_range = c(0.1, 0.6),
                                   cutoff_tumor_range = c(0.1, 0.8),
                                   cutoff_step = 0.1)
```

---

test.data.2comp *Simulated two-component test data*

Description

A list of simulated two-component test data used in DeMixT function. Expression data with 500 genes and 100 samples are simulated.

Usage

```
test.data.2comp
```

Format

An object of class `list` of length 5.

Value

A list with 5 elements (2 more elements when output.more.info = TRUE), which are

`pi`	A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings. Each column corresponds to one sample.
`Mu`	Simulated Mu of log2-normal distribution for both known ($MuN1$) and unknown component (MuT).

Sigma	Simulated $Sigma$ of log2-normal distribution for both known ($SigmaN1$) and unknown component ($SigmaT$).
data.Y	A SummarizedExperiment object of expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.
data.N1	A SummarizedExperiment object of expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.
True.data.T	A SummarizedExperiment object of simulated tumor expression data. It is a G by My matrix, where G is the number of genes and My is the number of mixed samples. This is shown only when output.more.info = TRUE.
True.data.N1	A SummarizedExperiment object of simulated true expression data for reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1. This is shown only when output.more.info = TRUE.

---

test.data.3comp *Simulated three-component mixed cell line test data*

Description

A list of simulated three-component mixed cell line test data used in DeMixT function. Expression data with 700 genes and 20 samples are simulated, where 675 genes' $MuN1$ is close to $MuN2$.

Usage

```
test.data.3comp
```

Format

An object of class `list` of length 6.

Value

A list with 6 elements (3 more elements when output.more.info = TRUE), which are

pi	A matrix of estimated proportion. First row and second row corresponds to the proportion estimate for the known components and unkown component respectively for two or three component settings. Each column corresponds to one sample.
Mu	Simulated Mu of log2-normal distribution for both known ($MuN1, MuN2$) and unknown component (MuT).
Sigma	Simulated $Sigma$ of log2-normal distribution for both known ($SigmaN1, SigmaN2$) and unknown component ($SigmaT$).
data.Y	A SummarizedExperiment object of simulated expression data from mixed tumor samples. It is a G by My matrix where G is the number of genes and My is the number of mixed samples. Samples with the same tissue type should be placed together in columns.

`data.N1` A SummarizedExperiment object of simulated expression data from reference component 1 (e.g., normal). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1.

`data.N2` A SummarizedExperiment object of expression data from additional reference samples. It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2.

`True.data.T` A SummarizedExperiment object of simulated tumor expression data. It is a G by My matrix, where G is the number of genes and My is the number of mixed samples.This is shown only when output.more.info = TRUE.

`True.data.N1` A SummarizedExperiment object of simulated true expression data for reference component 1 (e.g., stroma). It is a G by $M1$ matrix where G is the number of genes and $M1$ is the number of samples for component 1. This is shown only when output.more.info = TRUE.

`True.data.N2` A SummarizedExperiment object of simulated true expression data for reference component 2 (e.g., immue). It is a G by $M2$ matrix where G is the number of genes and $M2$ is the number of samples for component 2. This is shown only when output.more.info = TRUE.

Index

∗ **DeMixT_DE**
DeMixT_DE, 9
∗ **DeMixT_GS**
DeMixT_GS, 11
∗ **DeMixT_S2**
DeMixT_S2, 15
∗ **DeMixT**
DeMixT, 5
∗ **Optimum_KernelC**
Optimum_KernelC, 20
∗ **datasets**
test.data.2comp, 29
test.data.3comp, 30
∗ **simulate_3comp**
simulate_3comp, 25

batch_correction, 2

DeMixNB, 3
DeMixNB_preprocessing, 4
DeMixT, 5
DeMixT_DE, 9
DeMixT_GS, 11
DeMixT_preprocessing, 14
DeMixT_S2, 15
detect_suspicious_sample_by_hierarchical_clustering_2comp, 17

gene_selection_DE, 18

Optimum_KernelC, 20

plot_dim, 22
plot_sd *(*detect_suspicious_sample_by_hierarchical_clustering_2comp*)*, 17

scale_normalization_75th_percentile, 23
simulate_2comp, 24
simulate_3comp, 25
subset_sd, 26
subset_sd_gene_remaining, 28

test.data.2comp, 29
test.data.3comp, 30