Package 'HiCImpute'

August 16, 2021

Type Package

Title HiCImpute: An R Package Implementing ``Bayesian Hierarchical Model for Identifying Structural Zeros and Enhancing Single-cell Hi-C Data"

Version 1.0

Date 2021-08-14

Author Qing Xie (The Ohio State University, USA), Shili Lin (The Ohio State University, USA)

Maintainer Qing Xie <xie.735@osu.edu>

Description HiCImpute carries out single cell HiC matrix imputation that takes neighborhood, similar cells, and bulk data into account.

Depends R (>= 3.5.0),

License GPL

Imports Rcpp,
parallel,
RcppArmadillo,
Rtsne,
ggpubr,
mclust,
ggplot2,
DescTools,
gridExtra

LinkingTo Rcpp,
RcppArmadillo

RoxygenNote 7.1.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

`MCMCImpute` *This is the flagship function of HiCImpute. It identifies structural zeros and imputes sampling zeros under a Bayesian framework. The outputs can be used to facilitate downstream analysis such as clustering, subtype discovery, or 3D structure construction.*

Description

This is the flagship function of HiCImpute. It identifies structural zeros and imputes sampling zeros under a Bayesian framework. The outputs can be used to facilitate downstream analysis such as clustering, subtype discovery, or 3D structure construction.

Usage

```
MCMCImpute(
  scHiC,
  bulk = bulk,
 startval = c(100, 100, 10, 8, 10, 0.1, 900, 0.2, 0, replicate(dim(single)[2], 8)),
  n,
  epsilon1 = 0.5,
  epsilon2 = 5,
  mc.cores = 1,
  cutoff = 0.5,
  niter = 30000,
  burnin = 5000
)
```

Arguments

`scHiC`	The single-cell Hi-C matrix. It can take three types of formats. The preferred format is a single-cell matrix with each column being a vector of the upper triangular matrix without including the diagonal entries of the 2D matrix of a single-cell. Another types of formats are a list with each element being a 2D single-cell contact matrix, or a 3D ($n \times n \times k$) array that has k matrices of dimension $n \times n$. HiCImpute automatically transforms these two types of input into a matrix with each column being the vector of upper triangular matrix of a single-cell. For a single-cell matrix of size $n \times n$, the length of the vector should be $n \times (n-1)/2$. We only need the upper triangular matrix because the Hi-C matrix are symmetrical.
`bulk`	The bulk data. It can take two types of formats. A 2D bulk matrix of dimension $n \times n$ or a vector of the upper triangular entries of 2D bulk matrix. It can provide information for priors settings. If bulk data is not available, simply set it to be NULL, and MCMCImpute will sum up the single-cells to construct a bulk data.

startval	The starting value for the vector of parameters $\Theta = (\alpha, \mu^{\gamma}, \beta, \mu, a, \delta, b, \pi, s, \mu_1, \cdots, \mu_K)$. See xie et al. for the details of these parameters. The default value is as set in the function.
n	Integer. The dimension of single-cell matrix.
epsilon1	The range size of δ that is used to monitor the prior mean of π_{ij}, the probability that the pair (i, j) do not interact. The default value of ϵ_1 is 0.5.
epsilon2	The range size of B that is used to monitor the prior mean of μ_{ij}, the intensity of interaction between pair (i, j). The default value of ϵ_2 is 5.
mc.cores	The number of cores to be used in mclapply function that can parallelly impute the matrix. The default value is 1 (no parallelization), but the users is advised to a higher number to increase computational speed if their computer has parallel computing capability.
cutoff	The threshold of π_{ij} that is used to define structural zeros. The default value is 0.5. That is, if the probability of being a SZ is greater than 0.5, then the pair (i, j) are labelled as not interacting due to underlying biological mechanism.
niter	The number of iterations for the MCMC run. Default is 30000.
burnin	The number of burn-in iteration. Default is 5000.

Value

A list of posterior mean of probability (SZ), the imputed data without defining SZ (Impute_All), and imputed data with SZ, using the threshold (Impute_SZ).

Examples

```
data("K562_T1_7k")
data("K562_bulk")
scHiC=K562_T1_7k
T1_7k_res=MCMCImpute(scHiC=K562_T1_7k,bulk=K562_bulk,
startval=c(100,100,10,8,10,0.1,900,0.2,0,replicate(dim(scHiC)[2],8)),n=61,mc.cores = 1,
cutoff=0.5, niter=100000,burnin=5000)
```

PTSZ95 *This function calculates PTDO when fix PTSZ=0.95.*

Description

This function calculates PTDO when fix PTSZ=0.95.

Usage

```
PTSZ95(observed, expected, result)
```

Arguments

observed	Observed single cells matrix with each column being the upper triangular of a single cell.
expected	Underline true counts from simulation.
result	Result form MCMCImpute function.

Value

A vector of PTDO and its SD when fixing PTSZ to be 0.95, and the threshold used in that case.

Examples

```
PTSZ95(observed=K562_T1_7k, expected=K562_1_true, result=T1_7k_res)
```

`scHiC_assess` *This function analyzes both simulated and real datasets, depending on the inputs of the functions.*

Description

This function analyzes both simulated and real datasets, depending on the inputs of the functions.

Usage

```
scHiC_assess(
  scHiC,
  expected = NULL,
  result = NULL,
  imputed = NULL,
  cell_index = 1,
  n,
  cell_type,
  dims = 2,
  perplexity = 10,
  seed = 1000,
  kmeans = TRUE,
  ncenters = 2
)
```

Arguments

`scHiC`	The observed data. It can take three types of formats. The preferred format is a single-cell matrix with each column being a vector of the upper triangular matrix without including the diagonal entries of the 2D matrix of a single-cell. Another types of formats are a list with each element being a 2D s ingle-cell contact matrix, or a 3D ($n \times n \times k$) array that has k matrices of dimension $n \times n$. HiCImpute automatically transforms these two types of input into a matrix with each column being the vector of upper triangular matrix of a single-cell.
`expected`	Underline true counts of the simulated data. For real data analysis, just set it as NULL.
`result`	Output of MCMCImpute.
`imputed`	The imputed data that has the same dimension as the observed data. This is needed for real data analysis. For simulated data, set it as NULL.
`cell_index`	Indicates which cell is used to draw heatmaps and scatterplot.
`n`	Dimension of 2D contact matrix.
`cell_type`	A vector of underlying true cluster.

dims	The dimension of 2D matrix.
perplexity	numeric; Perplexity parameter (should not be bigger than $3 \times perplexity < nrow(X) - 1$).
seed	Random seed for generating t-SNE data.
kmeans	Logical, whether apply K-means clustering on the t-SNE data.
ncenters	Number of centers in K-means clustering analysis.

Value

A list of accuracy measurements and plots.

Examples

```
data("K562_1_true")
options(digits = 2)
scHiC_assess(scHiC=K562_T1_7k, expected=K562_1_true, imputed=result$Impute_SZ=T1_7k_imp)
```

---

scHiC_hm *This function draws heatmap of HiC data so that we can visually compares the imputation results.*

Description

This function draws heatmap of HiC data so that we can visually compares the imputation results.

Usage

```
scHiC_hm(datvec, n, title = "Heatmap")
```

Arguments

datvec	A vector of upper triangular mamtrix.
n	Dimension of 2D matrix (i.e., the number of segments).
title	The title of the heatmap.

Value

Heatmap of the matrix.

Examples

```
.
data("K562_1_true")
scHiC_hm(K562_1_true[,1], 61, title="Expected")
```

scHiC_Kmeans *This function conduct Kmeans clustering analysis on scHi-C data.*

Description

This function conduct Kmeans clustering analysis on scHi-C data.

Usage

```
scHiC_Kmeans(
  data,
  centers,
  nstart = 50,
  iter.max = 200,
  seed = 1234,
  algorithm = c("Hartigan-Wong", "Lloyd", "Forgy", "MacQueen"),
  trace = FALSE
)
```

Arguments

data	The observed or imputed matirx, with each column being the uppertriangular of a single cell HiC matrix.
centers	Either the number of clusters, say k, or a set of initial (distinct) cluster centres. If a number, a random set of (distinct) rows in x is chosen as the initial centres.
nstart	If centers is a number, how many random sets should be chosen.
iter.max	The maximum number of iterations allowed.
seed	Random seed.
algorithm	Character: may be abbreviated. Note that "Lloyd" and "Forgy" are alternative names for one algorithm.
trace	Logical or integer number, currently only used in the default method ("Hartigan-Wong"): if positive (or true), tracing information on the progress of the algorithm is produced. Higher values may produce more tracing information.

Value

Kmeans clustering results.

Examples

```
data("GSE117874_chr1_wo_diag")
data("GSE117874_imp")
cluster=scHiC_Kmeans(GSE117874_chr1_wo_diag, centers=2, nstart=1, iter.max=1000, seed=1)
```

`scHiC_ROC` *This package draws ROC (Receiver operating characteristic) curve to visually demonstrate ability to tell SZ from DO.*

Description

This package draws ROC (Receiver operating characteristic) curve to visually demonstrate ability to tell SZ from DO.

Usage

```
scHiC_ROC(observed, expected, result)
```

Arguments

`observed`	Observed sngle cell with each column being the upper triangular of single cell.
`expected`	Underline true count of simulated data.
`result`	Result from MCMCImpute funciton.

Value

A plot of ROC curve.

Examples

```
scHiC_ROC(observed=K562_T1_7k, expected=K562_1_true, result=T1_7k_res)
```

`scHiC_simulate` *This function simulates single cells from 3D structure.*

Description

This function simulates single cells from 3D structure.

Usage

```
scHiC_simulate(
  data = str1,
  alpha_0,
  alpha_1,
  beta_l,
  beta_g,
  beta_m,
  gamma,
  eta,
  n_single
)
```

Arguments

data	3D coordinates of single cell.
alpha_0	Parameter that controls sequence depth of data.
alpha_1	Parameter that controls sequence depth of data.
beta_l	Parameter that controls effect size of covariate.
beta_g	Parameter that controls effect size of covariate.
beta_m	Parameter that controls effect size of covariate.
gamma	Quantile that is used as the threshold.
eta	Percent of structural zeros that are set to be common structural zeros among all single-cells.
n_single	Number of single cells to be generated.

Value

A list of underline true count, SZ positions, and generated single cells.

Examples

```
#Load 3d structure generated from SIMBA package
load("simba_3strs.rdata")
Set random seed
set.seed(1234)
#Generate 100 random type1 single cells
simudat <- scHiC_simulate(data=str1, alpha_0=5.6,alpha_1=-1, beta_l=0.9,beta_g=0.9,
beta_m=0.9,gamma=0.1,eta=0.8, n_single=10)
```

---

scHiC_tSNE

This function visualize scHi-C data using t-SNE (t-distributed stochastic neighbor embedding) and applying Kmeans clustering followed by xie et al. 2021.

Description

This function visualize scHi-C data using t-SNE (t-distributed stochastic neighbor embedding) and applying Kmeans clustering followed by xie et al. 2021.

Usage

```
scHiC_tSNE(
  data,
  cell_type,
  dims = 2,
  perplexity = 10,
  check_duplicates = FALSE,
  seed = 1234,
  title = NULL,
  kmeans = TRUE,
  ncenters
)
```

Arguments

data	The observed matrix, with each column being the uppertriangular of a single cell HiC matrix.
cell_type	A vector that indicates cell type.
dims	Integer. Output dimentionality. Default=2.
perplexity	Numeric; Perplexity parameter (should not be bigger than 3 * perplexity < nrow(X) - 1, see details for interpretation).
check_duplicates	Logical; Checks whether duplicates are present. It is best to make sure there are no duplicates present and set this option to FALSE, especially for large datasets (default: TRUE).
seed	Random seed.
title	Title of the plot.
ncenters	Number of clusters in kmeans clustering.

Value

A stne visualization plot.

Examples

```
scHiC_tSNE(GSE117874_chr1_wo_diag, cell_type=c(rep("GM",14),rep("PBMC",18)),
dims = 2,perplexity=10, seed=1000, title="Observed GSE117874",
kmeans = TRUE, ncenters = 2)
```

SEVI *This function generates scatterplot of expected versus imputed.*

Description

This function generates scatterplot of expected versus imputed.

Usage

```
SEVI(obsvec, expvec, impvec)
```

Arguments

obsvec	A vector of observed single cell.
expvec	A vector of expected single cell.
impvec	A vector of imputed single cell.

Value

The scatterplot of expected versus imputed, with read dots being the observed zero pairs.

Examples

```
SEVI(obsvec=K562_T1_7k[,1], expvec=K562_1_true[,1], impvec=T1_7k_imp[,1] )
```

SOVI *This function generates scatterplot of observed versus imputed for nonzero observed counts.*

Description

This function generates scatterplot of observed versus imputed for nonzero observed counts.

Usage

```
SOVI(obsvec, impvec)
```

Arguments

obsvec	A vector of observed single cell.
impvec	A vector of imputed single cell.

Value

The scatterplot of observed versus imputed.

Examples

```
data("GSE117874_imp")
data("GSE117874_chr1_wo_diag")
SOVI(obsvec = GSE117874_chr1_wo_diag[,1], impvec = GSE117874_imp[,1])
```

Index