Package 'L1KProcs'

August 9, 2013

Version 0.99

Date 2013-05-07

Title L1000 data processing and analyzing pipeline

Author Jing Su author,developer <jsu@wakehealth.edu>
Chenglin Liu author,developer <cheliu@wakehealth.edu>
Xiaobo Zhou author <xizhou@wakehealth.edu>

Maintainer Chenglin Liu Developer <cheliu@wakehealth.edu>

Description L1000 data preprocessing and analyzing pipeline

License GPL 3.0, methods

URL http://ctsb.is.wfubmc.edu/itNETZ/DPPCSD.html

Depends R (>= 3.0)

Imports preprocessCore, stats, prada, gplots, SeqGSEA, L1KAnno

Enhances doParallel (>= 1.0.1), methods

BugReports http://ctsb.is.wfubmc.edu/itNETZ/AboutUs.html

R topics documented:

`AnalytePlot` *Plot Peak calling of analytes*

Description

`AnalytePlot` is to plot the distribution of a list of the raw bead analytes as well as the GMM for the peak calling. `APlot` is its inner function for plotting.

Usage

```
AnalytePlot(DataName, filename, outpath, analyteID)
APlot(DataName,analyteID,lfc,result,GeneList,plotpath)
```

Arguments

`outpath`	character. There generates a folder "WellPlots" under outpath to put the figures.
`DataName`	character. a well name.
`filename`	character. path to a well.
`analyteID`	vector. ID of analyte to be plotted.
`lfc`	vector. Readable raw data of the well extracted by readFCS of prada package.
`result`	matrix. Result(.csv) of `dPeak`
`GeneList`	charater. Gene names corresponding to the analyte IDs.
`plotpath`	charater. folder to put the figures.

Value

none.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

dPeak

Examples

```
wellname <- "CPC001_PC3_24H_X1_B3_DUO52HI53LO_A01"
filename <- file.path(system.file("test_data",package="L1KProcs"),paste(wellname,"lxb",se
##dPeak(outpath="l1kdata",filename,wellname)
##AnalytePlot(wellname, filename, "l1kdata",c(11,13))
## Same as:
dPeak(outpath="l1kdata",filename,wellname,plot=TRUE,analyteID=c(11,13))
```

CalCos *Calculate the cosine similarity of two matrix*

Description

CalCos is used to reunion the results of different iteration of csNMF factorization results.

Usage

```
CalCos(maT,maRef)
```

Arguments

maT	matrix.
maRef	matrix.

Value

A list with two elements, the median cosine values of each row of maT to maRef, the vector as the order to match maT to maRef.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Examples

```
maT <- matrix(seq(1,20),4,5)
maRef <- maT[c(1,3,2,4),]
CalCos(maT,maRef)
```

calEGEM *Calculate egem of a gene set of the SeqGeneSet object*

Description

calEGEM is an internal function to calculate the egem scores of of an object to the up and down regulated DEG gene sets as the SeqGeneSet object specified. perform.EGEM is an internal function to calculate egem scores of a list of object.

Usage

```
calEGEM(up.gene.set, down.gene.set, gene.score, weighted.type=0)
perform.EGEM(up.gene.set,down.gene.set,gene.data,nthread=1,weighted.type=0)
```

Arguments

up.gene.set	SeqGeneSet object. element "GS" is the index of up regulatedDEGs.
down.gene.set	SeqGeneSet object. element "GS" is the index of down regulated DEGs.
gene.score	vector. names of gene.score corresponding to the geneList slot of up.gene.set and down.gene.set.
gene.data	matrix. Each columns is corresponding to gene.score.
weighted.type	numeric. egem weight type.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.

Value

calEGEM returns a vector of EGEM scores of the up.gene.set and down.gene.setto the GS. perform.EGEM returns a matrix of EGEM scores.with each column is the one result of calEGEM.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

calperm.egem egem

calperm.egem *Calculate egem scores from randomized order of genes.*

Description

This is to calculate the egem scores when the order of genes are randomized.

Usage

```
calperm.egem(lstdegs,genelist,nthread=1,weighted.type=0,GSSizeMin=10,GSSizeMax=3
```

Arguments

lstdegs	list. It contains two vectors "up" and "down", with the positions of up and down regulated genes in the genelist.
genelist	vector. names of landmark genes.
weighted.type	numeric. egem weight type.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
GSSizeMin	numeric. minimum number of genes to calculate egem score.
GSSizeMax	numeric. maximum number of genes to calculate egem score.

Value

It returns a list with two vector "mean.perm", "sd.perm" as the statistics of the egem scores after randomization.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

calEGEM perform.EGEM egem

CheckData *Check if the required data exists*

Description

CheckData checks if the required data exist in the required location, including "lstNames.rda","lstPlates.rda","lstfiles.rd generated by DataStorage.

Usage

```
CheckData(outpath)
```

Arguments

outpath	character. It specifys the folder where to put all the processed data and their information. The required files should be under "data_summary" of outpath.

Details

This function is to check if the required files: "lstNames.rda","lstPlates.rda","lstfiles.rda" are generated and stored under [outpath]/data_summary.

Value

TRUE/FALSE.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

DataStorage.

Examples

```
CheckData("l1kdata")
datapath <- system.file("test_data",package="L1KProcs")
DataStorage(datapath,outpath="l1kdata")
CheckData("l1kdata")
```

ConvertM *Convert the expression of landmark genes to all genes.*

Description

ConvertM is to convert landmark gene expression to 22000 gene expression. ConvertExp is inner function of l1kpreprocs to convert a list of plates' gene expression data to all gene expresion in paralell.

Usage

```
ConvertM(inGCT=NULL,expM=NULL, refmatrix=NULL,outpath="l1kdata",
         overwrite=FALSE,check=TRUE)
ConvertExp (lstPlates=NULL,outpath="l1kdata",
            nthread=1,overwrite=FALSE,check=TRUE)
```

Arguments

lstPlates	character.a vector of the list of plates to be processed. It can be generated by function DataStorage.
outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.

overwrite	logical. if overwrite data of same name during processing. Default is FALSE.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.
expM	matrix. The matrix to be converted.
inGCT	character. path of a file name to be converted. Works when expM = NULL.
refmatrix	matrix. converting matrix from landmark genes to all genes CMatrix.

Details

ConvertExp generates files include: [outpath]/[platename]_5253_full.gct, [outpath]/[platename]_5253_LFC_full.gct, [outpath]/[platename]_5253_Raw_full.gct. [platename]s are specified by lstPlates.

Value

None.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

l1kpreprocs.

Examples

```
inGCT =
system.file("test_data",
            "CPC001_PC3_24H_X1_B3_DUO52HI53LO_A17.gct",
            package="L1KProcs")
if(!file.exists("l1kdata")) dir.create("l1kdata")
ConvertM(inGCT,outpath="l1kdata",
         overwrite=FALSE,check=TRUE)
```

---

cosine *Cosine Measure (Matrices)*

Description

Calculates the cosine measure between two vectors or between all column vectors of a matrix.

Usage

```
cosine(x, y = NULL)
```

Arguments

x	A vector or a matrix (e.g., a document-term matrix).
y	Optional: a vector with compatible dimensions to x. If 'NULL', all column vectors of x are correlated.

Details

`cosine()` calculates a similarity matrix between all column vectors of a matrix `x`. This matrix might be a document-term matrix, so columns would be expected to be documents and rows to be terms.

When executed on two vectors `x` and `y`, `cosine()` calculates the cosine similarity between them.

Value

Returns a $n * n$ similarity matrix of cosine values, comparing all n column vectors against each other. Executed on two vectors, their cosine similarity value is returned.

Note

The cosine measure is nearly identical with the pearson correlation coefficient (besides a constant factor) `cor(method="pearson")`. For an investigation on the differences in the context of textmining see (Leydesdorff, 2005).

Author(s)

Fridolin Wild <f.wild@open.ac.uk>

References

Leydesdorff, L. (2005) *Similarity Measures, Author Cocitation Analysis,and Information Theory*. In: JASIST 56(7), pp.769-772.

See Also

`cor`

Examples

```
## the cosinus measure between two vectors

vec1 = c( 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0 )
vec2 = c( 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0 )
cosine(vec1,vec2)



## the cosine measure for all document vectors of a matrix

vec3 = c( 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0 )
matrix = cbind(vec1,vec2, vec3)
cosine(matrix)
```

csNMF *csNMF method for compound signature discovery*

Description

csNMF is to find the compound signatures using csNMF method.

Usage

```
csNMF(egem.info,outpath="l1kanalysis",pNo=c(5:20),repeatNo=30,
      nthread=1, eta=-1, beta=0.01, lamda = -1,bi_conv=c(1e-3,5e-3))
```

Arguments

egem.info	list. the result of egem. It must contain three elements: egem, PNames, CNames.
outpath	character. path to store the results of the discovery.
pNo	vector. a numeric vector with potential compound numbers.
repeatNo	numeric. number of times to repeat the factorization. Default is 30.
nthread	numeric. number of cpu to be used.
eta	numeric. paramter for csNMF.
beta	numeric. paramter for csNMF.
lamda	numeric. paramter for csNMF.
bi_conv	vector. iteration termination threshold.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

csNMF.single

Examples

```
data("libEGEM",package="L1KAnno")
egem.info <- libEGEM[["MCF7_CP_24H_KD_96H"]]
egem <- egem.info[["egem"]][1:50,1:50]
PNames <- egem.info[["PNames"]][1:50]
CNames <- egem.info[["CNames"]][1:50]
pNo <- c(3:4)
repeatNo <- 2
lstcsNMF <- csNMF(egem.info,outpath="csNMFresult",pNo=pNo,repeatNo=repeatNo,nthread=4)
```

csNMF.No — *Decide the best number of signatures*

Description

This function the inner function of csNMF used to decide the signature number by consensus matrix.

Usage

```
csNMF.No(egem,lstcsNMF,outpath)
```

Arguments

egem	matrix. egem matrix.
lstcsNMF	list. intermediate result of csNMF.
outpath	character. location where to store the report.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

csNMF

csNMF.single — *csNMF method for compound signature discovery*

Description

csNMF.single is to subfunction of csNMF to factorize the egem matrix given signature number and repeat id.

Usage

```
csNMF.single(upE,downE,maPPI,outpath="l1kanalysis",k,rN,
             eta=-1, beta=0.01, lamda = -1,bi_conv=c(1e-3,5e-3),keep=FALSE)
```

Arguments

upE	matrix. egem matrix with negative values forcing to zeros.
downE	matrix. opposite of egem matrix with positive values forcing to zeros.
maPPI	matrix. PPI matrix with items are the genes of egem matrix.
outpath	character. path to store the factorization result if keep=TRUE.
k	numeric. number of signature number.
rN	numeric. Index of the repeat times.
eta	numeric. paramter for csNMF.

beta	numeric. paramter for csNMF.
lamda	numeric. paramter for csNMF.
bi_conv	vector. iteration termination threshold.
keep	logical. If keep=TRUE, the result will be not only returned, but also stored in the outpath.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

csNMF

Data *Data for preprocessing*

Description

Bead2Gene is a matrix to map analytes to genes.
CMatrix is a matrix to convert expression data of landmark genes to all genes.
lstprobeNames is a vector with gene names and probe names.
l1kControls is a matrix of control perturbagens.
PlateMap is a matrix to map plate ID to different perturbagens.
QTarget is a default target for quantile normalization.
U133A is the matrix to map probe ID to gene names.
genelist is the names of landmark genes of LINCS data of 53HI52LO type.
newPPI is a matrix of human protein protein interaction from STRING 2.0.

Examples

```
data("...",package="L1KProcs")
```

DataStorage *Save the list of well names and plate names*

Description

DataStorage saves the well and plate names.

Usage

```
DataStorage(datapath,outpath="l1kdata")
DataStorage.2(outpath="l1kdata")
```

Arguments

datapath	character. There are two options: 1. the path to the top folder where all raw data (.lxb) are stored. 2. the path to the file which lists the path to each well line by line.
outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.

Details

DataStorage saves the well and plate names by two ways. One is to to specify the top folder to store the L1000 raw data (.lxb). These data can be managed in different folders under the top folder. Another is to specify the detailed path of each well one by one. One record a line. Please use the orignal names of the raw data files, since it includes all expriment records for the following uses. DataStorage.2saves the well and plate names by detecting gct files of outpath folder. The generated data by DataStorage are used in the following processing. Generated files include: lstNames.rda, lstfiles.rda and lstPlates.rda under [outpath]/data_summary.

Value

a integer vector of number of wells and number of plates.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou

Examples

```
datapath <- system.file("test_data",package="L1KProcs")
DataStorage(datapath,outpath="l1kdata")
```

DEG *Find differential expressed genes of data*

Description

DEG is used to find the differential expressed genes(DEGs). The DEGs of genetic perturbagens are used as the feature of gene expression effect of that gene.

Usage

```
DEG(Data,th=0.5,meanD=NULL,sdD=NULL)
```

Arguments

Data	matrix. Log fold change gene expression data where to find DEGs, with rows are the genes, collums are the samples.
th	numeric. Threshold of the miminum absolute value of log fold change value.
meanD	numeric. Mean value of log fold change value of the Data. DEGs must be the ones with pvalue less than 0.05.
sdD	numeric. Standard deviation value of log fold change value in Data. DEGs must be the ones with pvalue less than 0.05.

Value

a list with two vectors, "up" and "down", which are the index of the up and down DGEs.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou

Examples

```
load(system.file("test_data","cpdata.rda",package="L1KProcs"))
lstdeg <- DEG(cpdata)
```

---

egem *Calculate egem scores based on expression data*

Description

egem is to calculate EGEM matrix based on the expression data after compound and genetic perturbagens.

Usage

```
egem(cpdata=NULL,kddata=NULL,th=0.5,meanD=NULL,sdD=NULL,
     lib.name=c("A375_96H","A549_75H",
                "HA1E_96H", "HCC515_96H",
                "HT29_96H", "MCF7_96H",
                "PC3_144H", "PC3_96H"),
     LINCS=TRUE,nthread=1,weighted.type=0,
     GSSizeMin=10,GSSizeMax=300,cpdes=NULL,kddes=NULL)
```

Arguments

cpdata	matrix. log fold change gene expression data after compound treatments. Rownames of cpdata are gene names.
kddata	matrix.log fold change gene expression data after genetic perturbagens. Rownames of cpdata are gene names.
th	numeric. Paramter used for DEG.
meanD	numeric.Paramter used for DEG.
sdD	numeric.Paramter used for DEG.
lib.name	character. Choose a dataset of DEGs from library. format is [celltype]_[time]H.
LINCS	logical. if use the database of from the LINCS project.
weighted.type	numeric. egem weight type.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
GSSizeMin	numeric. minimum number of genes to calculate egem score.
GSSizeMax	numeric. maximum number of genes to calculate egem score.
cpdes	vector. Compound names of compound perturbagens.
kddes	vector. Target genes of shRNA perturbagens.

Value

It returns a list of EGEM informations. EGEM, the matrix of egem scores; origin.EGEM, the matrix of egem scores before test; p.value, p-value of test; perm.egem, randomized egem scores;DEGs, the DEGs used as gene set for the egem score calculation.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Source

http://bioconductor.org/packages/2.12/bioc/html/SeqGSEA.html

References

Wang X, Cairns M: Gene set enrichment analysis of RNA-Seq data: integrating differential expression and splicing. BMC Bioinformatics 2013, 14(Suppl 5):S16.

See Also

calEGEM perform.EGEM calperm.egem egem.plot egem.analyze

Examples

```
load(system.file("test_data","cpdata.rda",package="L1KProcs"))
egem.info <- egem(cpdata,nthread=8,LINCS=TRUE,lib.name="HA1E_96H")
```

---

egem.analyze *Analyze the results based on EGEM matrix*

Description

egem.analyze gives gene names that have similar/reverse gene expression effects as each compounds.

Usage

```
egem.analyze(EGEM,th=0.05,outpath="l1kanalysis")
```

Arguments

EGEM	matrix. EGEM matrix which is the element "egem" of the result egem.
th	numeric. threshold to be as the element of a signature.
outpath	character. the path where the report located.

Value

a file called "SigGenes.txt".

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

egem

---

egem.plot *plot the matrix of EGEM*

Description

egem.plot gives the heatmap of EGEM matrix as well as the distributions of EGEM scores that not equal to zero.

Usage

```
egem.plot(EGEM,outpath="l1kanalysis")
```

Arguments

EGEM	matrix. EGEM matrix which is the element "egem" of the result egem.
outpath	character. the path where the plot figure located.

Value

a figure called "EGEM.png" with heatmap and histplot.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

egem

---

expLFC *Generating log fold change gene expression data.*

Description

expLFC and QN2LFC are to generate log fold change gene expression data. QN2LFC deals with one plate, while expLFC calls QN2LFC to generate data in paralell.

Usage

```
expLFC(lstPlates=NULL,lstControls=NULL,control.excludes=NULL,control.specify=NUL
outpath="l1kdata",nthread=1,overwrite=FALSE,check=TRUE)
QN2LFC(PlateName,outpath,lstControlWells=NULL,overwrite=FALSE)
```

Arguments

lstPlates	character.a vector of the list of plates to be processed. It can be generated by function DataStorage.
PlateName	character. a plate name.
lstControls	character. a list of a vector of names of control wells used for log fold change data.
lstControlWells	character. a vector of names of control wells of a plate used for log fold change data.
control.specify	character. the specific name of control perturbagen defined by users. works when lstControls==NULL, and used for FindControls.
control.excludes	character. a vector of the well names not used as controls. works when stControls==NULL, and used for FindControls.
outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
overwrite	logical. if overwrite data of same name during processing. Default is FALSE.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.

Details

expLFC generates files include: [outpath]/[platename]_5253_LFC.gct. [platename]s are specified by lstPlates.

Value

None.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

l1kpreprocs.

---

expNorm *Quantile normalization of gene expression data.*

Description

QNormgctPlate are to generate raw, normalized gene expression data of a plate. expNorm is an inner function of l1kpreprocs to process data of a list of plates in paralell.

Usage

```
QNormgctPlate(PlateName,datapath,QTarget,outpath,overwrite=FALSE)
expNorm(lstPlates=NULL,QTarget=NULL,outpath="l1kdata",
        nthread=1,overwrite=FALSE,check=TRUE)
```

Arguments

lstPlates	character.a vector of the list of plates to be processed. It can be generated by function DataStorage.
PlateName	character. a plate name.
QTarget	numeric. target vector of quantile normalization. It can be generated from dataset or package or by user specified.
outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
datapath	character. location of the gct files relating to the specified PlateName.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
overwrite	logical. if overwrite data of same name during processing. Default is FALSE.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.

Details

expNorm generates files include: [outpath]/[platename]_5253.gct, [outpath]/[platename]_5253_Raw.gct. [platename]s are specified by lstPlates.

Value

None.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

l1kpreprocs.

Examples

```
data("QTarget",package="L1KProcs")
datapath <- system.file("test_data",package="L1KProcs")
outpath <- "./"
PlateName <- "CPC001_PC3_24H_X1_B3_DUO52HI53LO"
QNormgctPlate(PlateName,datapath,QTarget,outpath)
```

FindControls *Find control wells and major control name.*

Description

It is to find control wells and major control name.

Usage

```
FindControls(PlateName,PlateMap,l1kControls,
             control.excludes=NULL, control.specify=NULL)
```

Arguments

PlateName	character. a plate name.
PlateMap	matrix. Plate mapping matrix.
l1kControls	character. l1kControls matrix.
control.excludes	character. a vector of the well names not used as controls.
control.specify	character. the specific name of control perturbagen defined by users.

Value

a list with a vector of control perturbagen names (name of the vector of control wells) and a charater of major control used for log fold change and quality control.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

PlateMap, l1kControls, l1kpreprocs.

Examples

```
data("PlateMap",package="L1KProcs")
data("l1kControls",package="L1KProcs")
FindControls("CPC001_PC3_24H_X1_B3_DUO52HI53LO",PlateMap,l1kControls)
```

FuncsNMF *csNMF method for egem matrix decomposition.*

Description

This function is vased on SNMF method, and the code is modified from the function nmf_snmf of NMF package from CRAN. It is used for egem matrix decomposition.

Usage

```
FuncsNMF(A1,A2,Pp,k,eta=-1, beta=0.01, lamda = -1, bi_conv=c(1e-5,1e-5), eps_con
fcnnls_W(A,W1,W2,H,Dp,Pp,eta, beta, lamda)
fcnnls_H(A1,A2,W1,W2,H, beta)
```

Arguments

A	matrix. Elements are positive.
A1	matrix. egem matrix with negative values forcing to zeros.
A2	matrix. opposite of egem matrix with positive values forcing to zeros.
W1	matrix. min(X1-W1xH).
W2	matrix. min(X2-W2xH)
H	matrix. min(X1-W1xH+X2-W2xH)
Dp	matrix. diagonal matrix with sum of the rows of Pp.
Pp	matrix. PPI matrix.
k	numeric. Number of colunm name of W.
eta	numeric. parameter to suppress/bound the L2-norm of H.
beta	numeric. regularisation parameter for sparsity control, which balances the trade-off between the accuracy of the approximation and the sparseness of H and W.larger beta generates higher sparseness on H (resp. W). Too large beta is not recommended.
lamda	numeric. the weight of PPI interaction impact to the model.
bi_conv	vector. parameter of the biclustering convergence test. It must be a size 2 numeric vector bi_conv=c(wminchange, hminchange), with: wminchange: the minimal allowance of change in row-clusters., hminchange: the minimal allowance of change in col-clusters.
iconv	numeric. decide convergence if row-clusters (within the allowance of wminchange) and column-clusters have not changed for iconv convergence checks.
eps_conv	threshold for the KKT convergence test.
verbose	logical. Display message information.

Details

csNMF is derived from SNMF method. The code is modified from NMF package. The output is the list with W1, W2 and H.

Value

A list with three matrix, W1, W2, and H.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Source

http://cran.r-project.org/web/packages/NMF/index.html

References

Gaujoux R, Seoighe C: A flexible R package for nonnegative matrix factorization. BMC Bioinformatics 2010, 11(1):367. Kim H, Park H: Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. Bioinformatics 2007, 23(12):1495-1502.

GCTI/O *GCT data read and write*

Description

`ReadQNData` find and read PlateName_5253.gct file. `ReadGCT` read a regular gct file. `WriteGCT` write matrix in GCT format.

Usage

```
ReadQNData(PlateName,outpath)
ReadGCT(DataName,outpath)
WriteGCT(GCTData,ofGCT)
```

Arguments

`PlateName`	character. a plate name.
`outpath`	character. path to store data.
`DataName`	character. a data name.
`GCTData`	matrix. gct format data.
`ofGCT`	character. output file path.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Examples

```
outpath <- system.file("test_data",
            package="L1KProcs")
inGCT <-ReadGCT("CPC001_PC3_24H_X1_B3_DUO52HI53LO_A17",outpath)
str(inGCT)
```

initial.class *Initialize PlateInfo classes.*

Description

initial.class is an inner function of l1kpreprocs to initialize PlateInfo classes of a list of plates.

Usage

```
initial.class(outpath="l1kdata",lstPlates=NULL,target,
              platethresh=0.5,wellthresh=0.6,check=TRUE)
```

Arguments

outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
lstPlates	character.a vector of the list of plates to be processed. It can be generated by function DataStorage.
target	logical or character. source of target. TRUE: generate target from dataset; FALSE: use package default value; [path/to/file]: a file name storing target vector.
platethresh	numeric. threshold of plate based quality control.
wellthresh	numeric. threshold of well based quality control.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.

Value

a list of PlateInfo classes with names specified by lstPlates.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

PlateInfo-class, l1kpreprocs

`l1kpreprocs` *L1K data preprocessing pipeline.*

Description

`l1kpreprocs` is to preprocessing data generate by L1000 platform.

Usage

```
l1kpreprocs(datapath=NULL,outpath="l1kdata",
            target=FALSE,plot=TRUE, ifAll=TRUE,
            Qsize=1000, RandNo=10, nthread=1,verbose=FALSE,
            pth=0.5,wth=0.6,ifqctr=TRUE,
            control.specify=NULL,overwrite=FALSE)
```

Arguments

`datapath`	character. There are two options: 1. the path to the top folder where all raw data (.lxb) are stored. 2. the path to the file which lists the path to each well line by line.
`outpath`	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
`target`	logical or character. source of target. TRUE: generate target from dataset; FALSE: use package default value; [path/to/file]: a file name storing target vector.
`plot`	logical. If TRUE, the distribution of correlations used for quality controls will be plot. Works when ifqctr=TRUE.
`ifAll`	logical. TRUE: 22000 gene expression data will be generated. FALSE: only landmark gene expression will be generated.
`Qsize`	integer. number of sample size to generate target.
`RandNo`	integer. number of repeat times to generate target. It only works when Qsize is smaller than data size.
`nthread`	positive integer. number of cpu used for parallel computing. Default is 1.
`verbose`	logical. if details display to screen.
`pth`	numeric. threshold of plate based quality control.
`wth`	numeric. threshold of well based quality control.
`ifqctr`	logical. if do quality control.
`control.specify`	character. the specific name of control perturbagen defined by users.
`overwrite`	logical. if overwrite data of same name during processing. Default is FALSE.

Details

`l1kpreprocs` is a pipeline to preprocessing data generate by L1000 platform.
It addresses three challedges for L1000 data processing: (1) Peak calling method: Guassian mixture model; (2) Experiment bias: Quantile Normalization; (3) Quality control: plate based and well based.

Functions of L1kDataProcs: (1) Peaking calling, raw data (.lxb) -> raw expression data (.csv and .gct), per well; (2) Quantile Normalization. raw, normalized, LFC expression data (.gct), per plate; (3) Coverting expression of landmark genes to 22000 genes, per plate; (4) Quality Control. Indicate bad quality plates and wells; (5) Generate heatmap of gene expression data; (6) Generate other reports including data quality.

Value

a list of PlateInfo classes with detailed information about processing.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

PlateInfo-class

Examples

```
datapath <- system.file("test_data",package="L1KProcs")
outpath <- "l1kdata"
lstPlateInfo <- l1kpreprocs(datapath,outpath,target=FALSE,
                ifAll=FALSE,nthread=6,plot=FALSE)
```

PeakCalling *Peak calling for L1000 datasets*

Description

dPeak is to detect peaks of L1000 raw data of one well. It includes two inner functions: LXB2Stats to detect peaks from raw data, and Stats2gct to map analytes to gene names.PeakCalling is an inner function of l1kpreprocs to process data of a list of wells in parallel.

Usage

```
PeakCalling(outpath="l1kdata",
            lstNames=NULL,lstfiles=NULL,
            datapath=NULL,nthread=1,overwrite=FALSE,check=TRUE)
dPeak(outpath="l1kdata",filename=NULL,
      wellname=NULL,overwrite=FALSE,check=TRUE,
      plot=FALSE,analyteID=c(11:500))
LXB2Stats(DataName,filename,outpath,overwrite=FALSE)
Stats2gct(statsDataName, outpath, overwrite=FALSE)
```

Arguments

outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
lstNames	character. a vector of the list of wells to be processed. It can be generated by function DataStorage.

lstfiles	character. a vector of the path to wells to be processed. It can be generated by function DataStorage.
datapath	character. works when lstNames==NULL | lstfiles==NULL, and used in DataStorage.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
overwrite	boolean. if overwrite data of same name during processing. Default is FALSE.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.
filename	character. path to a well.
wellname	character. a well name.
plot	logical. If TRUE, a folder " WellPlots" is generated under outpath, with figures about beads number hist plots and peak calling details of each analyte. Default is FALSE.
analyteID	vector. ID of analyte to be plotted when plot=TRUE.
DataName	character. a well name.
statsDataName	character. a well name.

Details

It is recommended to run DataStorage before this function. PeakCalling generates files include: [outpath]/[wellname].csv [outpath]/[wellname].gct.

Value

NA number or distribution of each well.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

U133A, Bead2Gene, FGMM, GMM, DataStorage, l1kpreprocs.

Examples

```
wellname <- "CPC001_PC3_24H_X1_B3_DUO52HI53LO_A01"
filename <- file.path(system.file("test_data",package="L1KProcs"),paste(wellname,"lxb",se
dPeak(outpath="l1kdata",filename,wellname,plot=TRUE,analyteID=c(11,13))
```

PlateInfo-class *'PlateInfo': a class for storing data preprocessing information.*

Description

PlateInfo class is to storing data preprocessing information.

Objects from the Class

Objects can be created by calls of the function PlateInfo.

Slots

name: Object of class "character" the name of the plate

storepath: Object of class "character" the location for expression data of the plate

target: Object of class "character" the source to generate target

wellNaNo: Object of class "numeric" number of bad analytes of each well inthe plate

controlWells: Object of class "character". control wells

majorcontrol: Object of class "character". used control well for computation

covcontrol: Object of class "matrix". Pearson Correlations of gene expression of major control wells

duplicates: Object of class "character". duplicate platenames including the plate itself

ifquality: Object of class "logical". if the plate passes the plate based quality control

covwells: Object of class "numeric". Pearson Correlations between wellls with their duplicates

goodwells: Object of class "character". wells passed well based quality control.

platethresh: Object of class "numeric". threshold of plate based quality control

wellthresh: Object of class "numeric". threshold of well based quality control

Methods

show Display information of the PlateInfo object.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

initial.class

Examples

```
new("PlateInfo",name="...",target="default")
```

PlateInfo-plot *Plot correlations used for quality control.*

Description

PlatePlot and WellPlot are to plotting correlations relating to the well based and plate based quality control.

Usage

```
PlatePlot(lstPlateInfo,outfile,lstPlates=NULL)
WellPlot(lstPlateInfo,outfile,lstPlates=NULL)
```

Arguments

lstPlateInfo	list. A list of PlateInfo classes produced by l1kpreprocs.
outfile	character. Name of the plot figure. format: png.
lstPlates	list. A list of plate names. if NULL, all plates relating to lstPlateInfo will be plotted.

Details

This is the visualization of plate based and well based quality control.

Value

None.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

PlateInfo-class, PlateInfo-class.

Examples

```
load(system.file("test_data","lstPlateInfo.rda", package="L1KProcs"))
outfile <- file.path("CorPlates.png")
PlatePlot(lstPlateInfo,outfile)
outfile <- file.path("CorWells.png")
WellPlot(lstPlateInfo,outfile)
```

QualityControl *Quality control.*

Description

QualityControl do well based and plate based quality control of the data.

Usage

```
QualityControl(outpath="l1kdata",lstControls=NULL,pth=0.5,wth=0.6,lstPlates=NULL
Quality.Cons(platename,mctrs,pth=0.5,wth=0.6,dupplates=NULL,outpath)
```

Arguments

outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
lstControls	character. a list of major controls of each plate.
pth	numeric. threshold of plate based quality control.
wth	numeric. threshold of well based quality control.
lstPlates	character.a vector of the list of plates to be processed. It can be generated by function DataStorage.
lstdupPlates	character. a list of duplicate plates of each plate, including itself.
nthread	positive integer. number of cpu used for parallel computing. Default is 1.
check	logical. check if paramters are right before processing. Please make sure it is TRUE. FALSE is only use for inner function.
platename	character. a plate name.
mctrs	character. a vector of major controls of a plate.
dupplates	character. a vector of duplicate plates of a plate.

Value

a matrix of Pearson correlations of major control wells.
a logical value to determine if passed the plate bases quality control.
a matrix with the Row 1 is the median Pearson correlations between wells with its duplicates, Row 2 is if it is passed the well based quality control.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

FindControls,l1kpreprocs.

sumM *sum of a list of matrix*

Description

It calculates the sum of a list of matrix.

Usage

```
sumM(lstT)
```

Arguments

lstT	list. a list of equal size matrix.

Value

a matrix as the sum of the input matrixes.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Examples

```
lstT <- list()
lstT[[1]] <- matrix(seq(1,10),2,5)
lstT[[2]] <- matrix(seq(11,20),2,5)
sumM(lstT)
```

TargetGenerate *Generate quantile normalization targets for plate normalization.*

Description

TargetGenerate generate targets for quantile normalization in the DataGenerate step. It generate target from all gene expression in the well name list. Quantile normalization can reduce batch effects due to exprerimental bias.

Usage

```
TargetGenerate(outpath="l1kdata",lstNames=NULL,Qsize=1000,RandNo=10,check=TRUE)
QTargetGenerate(lstTargetNames,outpath)
```

Arguments

outpath	character. It specifys the folder where to put all the processed data and their information. Default is to generate "l1kdata" under current path.
lstNames	character. a vector of the list of wells to be processed. It can be generated by function DataStorage.
Qsize	integer. number of sample size to generate target.
RandNo	integer. number of repeat times to generate target. It only works when Qsize is smaller than data size.
check	logical. check if paramters are right before processing. Please make sure it is TRUE.
lstTargetNames	character. a vector of well names to generate target for quantile normalization.

Details

Please note that this function must performed after DataStorage and PeakCalling function. The generated data QTarget.rda is required for quantile normalization in the DataGenerate step. There are three ways to provide target, including using default target of the package, providing data file of target vector, and generating new target using TargerGenerate. TargetGenerate generates files include: [outpath]/data_summary/QTarget.rda, [outpath]/data_summary/Fvalue.rda, [outpath]/data_summary/Qcorrelation.rda.

Value

None.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

See Also

QTarget, expNorm, PeakCalling, l1kpreprocs.

Util *Utility of the package*

Description

Inner functions for the package.

Usage

```
GMM(inX, inP)
FGMM(inP,inX,inY)
ParseDataName(lstDataName)
ModeShortName(DetectMode)
selectSub(vec,maxNo=4)
```

Arguments

inX	numeric. a vector of parameters.
inP	numeric.a vector of probabilities.
inY	numeric. a vector of observations.
lstDataName	character. a vector of well names.
DetectMode	character. 53HI52LO/44LO45HI/52L53HI/45HI44LO.
vec	vector.
maxNo	numeric. max number to show.

Details

They are used for inner functions.

Author(s)

Chenglin Liu, Jing Su, Xiaobo Zhou.

Examples

```
lstDataName <- "CPC001_PC3_24H_X1_B3_DUO52HI53LO_A01"
ParseDataName(lstDataName)
x <- seq(1,10)
selectSub(x,maxNo=4)
```

Index