Package 'MetaQC'

July 2, 2014

Type Package

Title MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis

Version 0.1.13

Author Don Kang <donkang75@gmail.com> and George Tseng <ctseng@pitt.edu>

Maintainer Don Kang <donkang75@gmail.com>

Description MetaQC implements our proposed quantitative quality control measures: (1) internal homogeneity of co-expression structure among studies (internal quality control; IQC); (2) external consistency of co-expression structure correlating with pathway database (external quality control; EQC); (3) accuracy of differentially expressed gene detection (accuracy quality control; AQCg) or pathway identification (AQCp); (4) consistency of differential expression ranking in genes (consistency quality control; CQCg) or pathways (CQCp). (See the reference for detailed explanation.) For each quality control index, the p-values from statistical hypothesis testing are minus log transformed and PCA biplots were applied to assist visualization and decision. Results generate systematic suggestions to exclude problematic studies in microarray meta-analysis and potentially can be extended to GWAS or other types of genomic meta-analysis. The identified problematic studies can be scrutinized to identify technical and biological causes (e.g. sample size, platform, tissue collection,preprocessing etc) of their bad quality or irreproducibility for final inclusion/exclusion decision.

Depends R (>= 2.10.0), proto, foreach, iterators

Suggests doMC, doSNOW, FactoMineR, matrixStats, gdata, gtools,survival

License GPL-2

URL https://github.com/donkang75/MetaQC

LazyLoad yes

Collate MetaQC.R requireAll.R functions.R runQC.R cleanup.R

Date 2012-12-21

Repository CRAN

Date/Publication 2012-12-22 07:39:42

NeedsCompilation no

R topics documented:

`MetaQC-package` *MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis*

Description

MetaQC implements our proposed quantitative quality control measures: (1) internal homogeneity of co-expression structure among studies (internal quality control; IQC); (2) external consistency of co-expression structure correlating with pathway database (external quality control; EQC); (3) accuracy of differentially expressed gene detection (accuracy quality control; AQCg) or pathway identification (AQCp); (4) consistency of differential expression ranking in genes (consistency quality control; CQCg) or pathways (CQCp). (See the reference for detailed explanation.) For each quality control index, the p-values from statistical hypothesis testing are minus log transformed and PCA biplots were applied to assist visualization and decision. Results generate systematic suggestions to exclude problematic studies in microarray meta-analysis and potentially can be extended to GWAS or other types of genomic meta-analysis. The identified problematic studies can be scrutinized to identify technical and biological causes (e.g. sample size, platform, tissue collection, preprocessing etc) of their bad quality or irreproducibility for final inclusion/exclusion decision.

Details

Package:	MetaQC
Type:	Package
Version:	0.1.13
Date:	2012-12-21

License: GPL-2
LazyLoad: yes

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    brainQC
    plot(brainQC)

    ## For parallel computation with only 2 cores
## R >= 2.11.0 in windows to use parallel computing
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE, nCores=2)
    #B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)

    ## For parallel computation with all cores
## In windows, only 2 cores are used if not specified explicitly
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)

## Real Example which is used in the paper
#download the brainFull file
```

```
#from https://github.com/downloads/donkang75/MetaQC/brainFull.rda
load("brainFull.rda")
    brainQC <- MetaQC(brainFull, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=TRUE,
verbose=TRUE, isParallel=TRUE)
    runQC(brainQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt") #B was 1e5 in the paper
    plot(brainQC)

## Survival Data Example
#download Breast data
#from https://github.com/downloads/donkang75/MetaQC/Breast.rda
load("Breast.rda")
    breastQC <- MetaQC(Breast, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=FALSE,
verbose=TRUE, isParallel=TRUE, resp.type="Survival")
    runQC(breastQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt")
    breastQC
    plot(breastQC)

## End(Not run)
```

`brain` *7 brain cancer studies*

Description

7 brain cancer studies comparing Anaplastic Astrocytoma (AA) and Glioblastoma multiforme (GBM) samples.

Data Name	Published Year	Array Platform	Sample Size	GEO Accession ID
Freije	2004	HG-U133A,B	85	GSE4412
Phillips	2006	HG-U133A,B	100	GSE4271
Sun	2006	HG-U133 Plus 2	100	GSE4290
Yamanaka	2006	Agilent	29	GSE4381
Petalidis	2008	HG-U133A	58	GSE1993
Gravendeel	2009	HG-U133 Plus 2	175	GSE16011
Sun	2010	HG-U133 Plus 2	42	GSE19578

Usage

```
brain
```

Format

A list containing 7 matrices. Each matrix is gene expression data after gene filtering.

Source

Gene Expression Omnibus (GEO)

References

Freije W, Castro-Vargas F, Fang Z, Horvath S, Cloughesy T, et al. (2004) Gene expression profiling of gliomas strongly predicts survival. Cancer research 64: 6503.

Phillips H, Kharbanda S, Chen R, Forrest W, Soriano R, et al. (2006) Molecular subclasses of high-grade glioma predict prognosis, delineate a pattern of disease progression, and resemble stages in neurogenesis. Cancer Cell 9: 157-173.

Sun L, Hui A, Su Q, Vortmeyer A, Kotliarov Y, et al. (2006) Neuronal and glioma-derived stem cell factor induces angiogenesis within the brain. Cancer cell 9: 287-300.

Yamanaka R, Arao T, Yajima N, Tsuchiya N, Homma J, et al. (2006) Identification of expressed genes characterizing long-term survival in malignant glioma patients. Oncogene 25: 5994-6002.

Petalidis L, Oulas A, Backlund M, Wayland M, Liu L, et al. (2008) Improved grading and survival prediction of human astrocytic brain tumors by artificial neural network analysis of gene expression microarray data. Molecular cancer therapeutics 7: 1013.

Gravendeel L, Kouwenhoven M, Gevaert O, de Rooi J, Stubbs A, et al. (2009) Intrinsic gene expression profiles of gliomas are a better predictor of survival than histology. Cancer research 69: 9065.

Paugh B, Qu C, Jones C, Liu Z, Adamowicz-Brice M, et al. (2010) Integrated molecular genetic profiling of pediatric high-grade gliomas reveals key differences with the adult disease. Journal of Clinical Oncology 28: 3061.

`cleanup` *Cleaning up resources.*

Description

It is to shutdown the workers used for parallel processing and release resources. It is only necessary in windows. (Deprecated)

Usage

```
cleanup(QC)
```

Arguments

`QC`	A proto R object which obtained by MetaQC function.

Value

NA

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) Meta-QC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

See Also

MetaQC

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    ## For parallel computation with only 2 cores
## R >= 2.11.0 in windows to use parallel computing
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE, nCores=2)
    #B is recommended to be >= 1e4 in real application
runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)
    cleanup(brainQC) #neccessary for windows after using parallel processing


## End(Not run)
```

MetaQC *MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis*

Description

MetaQC implements our proposed quantitative quality control measures: (1) internal homogeneity of co-expression structure among studies (internal quality control; IQC); (2) external consistency of co-expression structure correlating with pathway database (external quality control; EQC); (3) accuracy of differentially expressed gene detection (accuracy quality control; AQCg) or pathway identification (AQCp); (4) consistency of differential expression ranking in genes (consistency quality control; CQCg) or pathways (CQCp). (See the reference for detailed explanation.) For each quality control index, the p-values from statistical hypothesis testing are minus log transformed and PCA biplots were applied to assist visualization and decision. Results generate systematic suggestions to exclude problematic studies in microarray meta-analysis and potentially can be extended to GWAS or other types of genomic meta-analysis. The identified problematic studies can be scrutinized to identify technical and biological causes (e.g. sample size, platform, tissue collection, preprocessing etc) of their bad quality or irreproducibility for final inclusion/exclusion decision.

Usage

```
MetaQC(DList, GList, isParallel = FALSE, nCores = NULL,
useCache = TRUE, filterGenes = TRUE,
maxNApctAllowed=.3, cutRatioByMean=.4, cutRatioByVar=.4, minNumGenes=5,
verbose = FALSE, resp.type = c("Twoclass", "Multiclass", "Survival"))
```

Arguments

DList	Either a list of all data matrices (Case 1) or a list of lists (Case 2); The first case is simplified input data structure only for two classes comparison. Each data name should be set as the name of each list element. Each data should be a numeric matrix that has genes in the rows and samples in the columns. Row names should be official gene symbols and column names be class labels. For the full description of input data, you can use the second data format. Each data is represented as a list which should have x, y, and geneid (geneid can be replaced to row names of matrix x) elements, representing expression data, outcome or class labels, and gene ids, respectively. Additionally, in the survival analysis, censoring.status should be set.
GList	The location of a file which has sets of gene symbol lists such as gmt files. By default, the gmt file will be converted to list object and saved with the same name with ".rda". Alternatively, a list of gene sets is allowed; the name of each element of the list should be set as a unique pathway name, and each pathway should have a character vector of gene symbols.
isParallel	Whether to use multiple cores in parallel for fast computing. By default, it is false.
nCores	When isParallel is true, the number of cores can be set. By default, all cores in the machine are used in the unix-like machine, and 2 cores are used in windows.
useCache	Whether imported gmt file should be saved for the next use. By default, it is true.
filterGenes	Whether to use gene filtering (recommended).
maxNApctAllowed	Filtering out genes which have missing values more than specified ratio (Default .3). Applied if filterGenes is TRUE.
cutRatioByMean	Filtering out specified ratio of genes which have least expression value (Default .4). Applied if filterGenes is TRUE.
cutRatioByVar	Filtering out specified ratio of genes which have least sample wise expression variance (Default .4). Applied if filterGenes is TRUE.
minNumGenes	Mininum number of genes in a pathway. A pathway which has members smaller than the specified value will be removed.
verbose	Whether to print out logs.
resp.type	The type of response variable. Three options are: "Twoclass" (unpaired), "Multiclass", "Survival." By default, Twoclass is used

Value

A proto R object. Use RunQC function to run QC procedure. Use Plot function to plot PCA figure. Use Print function to view various information. See examples below.

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

See Also

runQC

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    brainQC
    plot(brainQC)

    ## For parallel computation with only 2 cores
## R >= 2.11.0 in windows to use parallel computing
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE, nCores=2)
    #B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)

    ## For parallel computation with all cores
## In windows, only 2 cores are used if not specified explicitly
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)
```

```
## Real Example which is used in the paper
#download the brainFull file
#from https://github.com/downloads/donkang75/MetaQC/brainFull.rda
load("brainFull.rda")
    brainQC <- MetaQC(brainFull, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=TRUE,
verbose=TRUE, isParallel=TRUE)
    runQC(brainQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt") #B was 1e5 in the paper
    plot(brainQC)

## Survival Data Example
#download Breast data
#from https://github.com/downloads/donkang75/MetaQC/Breast.rda
load("Breast.rda")
    breastQC <- MetaQC(Breast, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=FALSE,
verbose=TRUE, isParallel=TRUE, resp.type="Survival")
    runQC(breastQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt")
    breastQC
    plot(breastQC)

## End(Not run)
```

`plot.proto` *Plot MetaQC results.*

Description

It draws a PCA biplot which shows the four QC measures. CQCg and AQCg are combined to be CAQCg, and CQCp and AQCp are combined to be CAQCp to reduce the dominance of CQC and AQC due to their greater correlation.

Usage

```
## S3 method for class 'proto'
plot(x, ...)
```

Arguments

`x`	A proto R object which obtained by MetaQC function.
`...`	Further arguments to print function.

Value

NA

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

See Also

MetaQC

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)


## End(Not run)
```

---

print.proto *Print MetaQC results.*

Description

It prints out the results of all QC measures and standardized mean rank of each study. CQCg and AQCg are combined to be CAQCg, and CQCp and AQCp are combined to be CAQCp to reduce the dominance of CQC and AQC due to their greater correlation.

Usage

```
## S3 method for class 'proto'
print(x, ...)
```

Arguments

x	A proto R object which obtained by MetaQC function.
...	Further arguments to print function.

Value

NA

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

See Also

MetaQC

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    brainQC


## End(Not run)
```

---

requireAll — *MetaQC: Quantitative Quality Assessment for Inclusion/Exclusion Criteria of Genomic Meta-Analysis*

Description

requireAll description

Usage

```
requireAll(packages)
```

Arguments

packages	A character vector of required packages. Unavailable packages are going to be installed.

Value

None

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

Examples

```
## Not run:
    libs <- c("proto", "foreach", ifelse(.Platform$OS.type == "unix",
"doMC", "doSNOW"))
    requireAll(libs)

## End(Not run)
```

---

runQC *Command to execute quality control procedures.*

Description

It it a utility function to RunQC method in MetaQC object.

Usage

```
runQC(QC, nPath=NULL, B=1e4, pvalCut=.05,
pvalAdjust=FALSE, fileForCQCp="c2.all.v3.0.symbols.gmt")
```

Arguments

QC	A proto R object which obtained by MetaQC function.
nPath	The number of top pathways which would be used for EQC calculation. The top pathways are automatically determined by their mean rank of over significance among given studies. It is important that gene sets used for EQC are expected to have higher correlation than background. For better performance, this should be set as a reasonably small number.
B	The number of permutation tests used for EQC calculation. More than 1e4 is recommended.
pvalCut	P-value threshold used for AQC calculation.
pvalAdjust	Whether to apply p-value adjustment due to multiple testing (B-H procedure is used).
fileForCQCp	Gene set used for CQCp calculation. Usually larger gene set is used than EQC calculation.

Value

A data frame showing a summary of each quality control score.

Author(s)

Don Kang (donkang75@gmail.com) and George Tseng (ctseng@pitt.edu)

References

Dongwan D. Kang, Etienne Sibille, Naftali Kaminski, and George C. Tseng. (Nucleic Acids Res. 2012) MetaQC: Objective Quality Control and Inclusion/Exclusion Criteria for Genomic Meta-Analysis.

See Also

MetaQC

Examples

```
## Not run:
    requireAll(c("proto", "foreach"))

   ## Toy Example
    data(brain) #already hugely filtered
    #Two default gmt files are automatically downloaded,
#otherwise it is required to locate it correctly.
    #Refer to http://www.broadinstitute.org/gsea/downloads.jsp
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    brainQC
    plot(brainQC)

    ## For parallel computation with only 2 cores
## R >= 2.11.0 in windows to use parallel computing
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE, nCores=2)
    #B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)

    ## For parallel computation with all cores
## In windows, only 2 cores are used if not specified explicitly
    brainQC <- MetaQC(brain, "c2.cp.biocarta.v3.0.symbols.gmt",
filterGenes=FALSE, verbose=TRUE, isParallel=TRUE)
#B is recommended to be >= 1e4 in real application
    runQC(brainQC, B=1e2, fileForCQCp="c2.all.v3.0.symbols.gmt")
    plot(brainQC)

## Real Example which is used in the paper
#download the brainFull file
```

```
#from https://github.com/downloads/donkang75/MetaQC/brainFull.rda
load("brainFull.rda")
    brainQC <- MetaQC(brainFull, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=TRUE,
verbose=TRUE, isParallel=TRUE)
    runQC(brainQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt") #B was 1e5 in the paper
    plot(brainQC)

## Survival Data Example
#download Breast data
#from https://github.com/downloads/donkang75/MetaQC/Breast.rda
load("Breast.rda")
    breastQC <- MetaQC(Breast, "c2.cp.biocarta.v3.0.symbols.gmt", filterGenes=FALSE,
verbose=TRUE, isParallel=TRUE, resp.type="Survival")
    runQC(breastQC, B=1e4, fileForCQCp="c2.all.v3.0.symbols.gmt")
    breastQC
    plot(breastQC)

## End(Not run)
```

Index