

Package ‘WebGestaltR’

May 11, 2017

Type Package

Title The R Version of WebGestalt

Version 0.1.1

Date 2017-05-02

Author Jing Wang <jingwang.uestc@gmail.com>

Maintainer Jing Wang <jingwang.uestc@gmail.com>

Description The web version WebGestalt <<http://www.webgestalt.org>> supports 12 organisms, 324 gene identifiers and 150,937 function categories. Users can upload the data and functional categories with their own gene identifiers. In addition to the Over-Representation Analysis, WebGestalt also supports Gene Set Enrichment Analysis. The user-friendly output interface allow interactive and efficient exploration of enrichment results. The WebGestaltR package not only supports all above functions but also can be integrated into other pipeline or simultaneous analyze multiple gene lists.

License LGPL

LazyLoad yes

Depends R (>= 3.3)

Imports methods, data.table (>= 1.10.0), parallel (>= 3.3.2), doParallel (>= 1.0.10), foreach (>= 1.4.0), PythonInR (>= 0.1-3), pkgmaker (>= 0.22), rjson (>= 0.2.15)

Collate zzz.R WebGestaltR.R WebGestaltR_batch.R IDMapping.R
GOSlimSummary.R listGeneSet.R listIDType.R listOrganism.R
readGMT.R listReferenceSet.R listArchiveURL.R formatCheck.R

SystemRequirements Java, Python

NeedsCompilation no

Repository CRAN

Date/Publication 2017-05-11 06:00:43 UTC

R topics documented:

WebGestaltR-package	2
formatCheck	3

GOSlimSummary	4
IDMapping	5
listArchiveURL	6
listGeneSet	7
listIDType	8
listOrganism	8
listReferenceSet	9
readGMT	10
WebGestaltR	10
WebGestaltR_batch	14

Index **18**

WebGestaltR-package *The R version of WebGestalt*

Description

WebGestaltR is the R version of our well-known web application tool WebGestalt (www.webgestalt.org). The advantage of this R package is it can be integrated to other pipeline or simultaneous analyze multiple gene lists.

Details

Package:	WebGestaltR
Type:	Package
License:	LGPL
LazyLoad:	yes

Author(s)

Jing Wang Maintainer: Jing Wang <jing.wang@bcm.edu>

References

WEB-based GENE SeT AnaLysis Toolkit (WebGestalt): update 2013. Nucleic Acids Res, 41 (Web Server issue), W77-83.

See Also

[WebGestaltR](#)

formatCheck	<i>Check the format of the uploaded gene list or ranked gene list file or object.</i>
-------------	---

Description

The formatCheck function can check the format of the gene list or ranked gene list file or object uploaded to the WebGestaltR for the analysis.

Usage

```
formatCheck(dataType="list", inputGeneFile=NULL, inputGene=NULL)
```

Arguments

dataType	Currently, the formatCheck function supports 2 data type: 1. "list" means the uploaded file or data is a gene list 2. "rnk" means the uploaded file or data is a ranked gene list with two columns (genes and scores).
inputGeneFile	The uploaded data file. If the dataType is "list", the file extension should be ".txt" and have only one column. If the dataType is "rnk", the file extension should be ".rnk" and have two columns (genes and scores).
inputGene	The uploaded R object. If the dataType is "list", the R object should be a vector. If the dataType is "rnk", the R object should be a data.frame and have two columns (genes and scores).

Value

The formatCheck function will return error if the data format is incorrect. Otherwise, it will return the processed data for the analysis.

Author(s)

Jing Wang

Examples

```
geneFile<-system.file("extdata", "interestingGenes.txt", package="WebGestaltR")
interestGene<-formatCheck(dataType="list", inputGeneFile=geneFile, inputGene=NULL)
```

GOSlimSummary	<i>Summary the gene list based on the GO (Gene Ontology) Slim data sets</i>
---------------	---

Description

This function can summary the gene list based on the biological process, cellular component and molecular function ontologies of the GO Slim data sets. The summary result will be plotted as three bar plots and outputted to the PDF file.

Usage

```
GOSlimSummary(organism, genelist, outputFile, outputType, hostName)
```

Arguments

organism	Currently, GOSlimSummary supports 12 organisms. Users can use the function <code>listOrganism</code> to check the available organisms.
genelist	An R vector object containing a gene list. GOSlimSummary only supports NCBI EntrezGene ID for the summary. For other ID types, please first use <code>IDMapping</code> function to map to the EntrezGene ID.
outputFile	The output file name.
outputType	The output file extension that can be pdf, png, or bmp.
hostName	The server URL for accessing the data. User can use <code>listArchiveURL</code> function to get all archive version URL.

Value

GOSlimSummary function will return a high-resolution image containing the summary information of the uploaded gene list based on the biological process (red bar plot), cellular component (blue bar plot) and molecular function (green bar plot) ontologies.

Note

Because the GOSlimSummary function will read the GO Slim data from the server, the running time for this function will be also based on the internet speed. Generally, it will take around 20 seconds to perform the summary analysis.

Author(s)

Jing Wang

Examples

```
geneFile<-system.file("extdata", "GOSlimExample.txt", package="WebGestaltR")
geneList<-read.table(geneFile, header=FALSE, sep="\t", stringsAsFactors=FALSE)
geneList<-as.vector(as.matrix(geneList))
outputFile<-paste(getwd(), "GOSlimSummary", sep="")
#GOSlimSummary(organism="hsapiens", geneList=geneList,
#outputFile=outputFile, outputType="pdf")
```

IDMapping

*ID Mapping***Description**

The IDMapping function can map one ID type supported by the WebGestaltR to any other ID type supported by the WebGestaltR. This function can perform the ID mapping for three types of data: a gene list, a gene list with the scores and a gmt file.

Usage

```
IDMapping(organism="hsapiens", dataType="list", inputGeneFile=NULL,
inputGene=NULL, sourceIdType, targetIdType, collapseMethod="mean",
is_outputFile=FALSE, outputFileName="", methodType="R",
hostName="http://www.webgestalt.org/")
```

Arguments

organism	Currently, WebGestaltR supports 12 organisms. Users can use the function <code>listOrganism</code> to check the available organisms.
dataType	The IDMapping function can perform the ID mapping for three types of data: <code>list</code> (a gene list), <code>rnk</code> (a ranked gene list) and <code>gmt</code> (a gmt file).
inputGeneFile	Three types of the files are supported for uploading the data: a <code>txt</code> file for a gene list, a <code>rnk</code> file for a gene list with scores (separated by tab), a <code>gmt</code> file (first column is category id, second one is external link of the category and others are the annotated genes. all columns are separated by tab).
inputGene	Two types of the R objects are supported for uploading the data: an R vector object for a gene list and an R data.frame object for a gene list with scores.
sourceIdType	The ID type of the uploaded data. The supported ID type of the WebGestaltR for the selected organism can be found by the function <code>listIDType</code> .
targetIdType	The target ID type for ID mapping. The supported ID type of the WebGestaltR for the selected organism can be found by the function <code>listIDType</code> .
collapseMethod	The method to collapse the duplicate ids for a gene list with scores before ID mapping. <code>mean</code> , <code>median</code> , <code>min</code> and <code>max</code> represent the mean, median, minimum and maximum of scores for the duplicate ids.
is_outputFile	If <code>is_outputFile</code> is TRUE, the mapping results will be outputted to a file.

outputFileName	The output file name. No extension in the file name and the function will add the extension based on the input data type.
methodType	For the large ID mapping table (e.g. dbSNP mapping table), Users can use R or Python function to read it. Sometimes Python code is faster than the R code. If users select to use Python code to read the mapping table, users should first install python and the module pandas in the computer.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.

Value

The IDMapping function will output an R data.frame object with three types of structure based on the three types of the input data. If the targetIdType is one of entrezgene, genesymbol and genename, the output object will contain four columns for a gene list (userid, genesymbol, genename and entrezgene), five columns for a gene list with scores (userid, genesymbol, genename, entrezgene and score) and six columns for a gmt file (geneset, link, userid, genesymbol, genename and entrezgene). If the targetIdType is other ID type, the data.frame object will add one more column targetid.

Note

Because the IDMapping function will read the mapping tables from the server, the running time for the WebGestaltR function will be also based on the internet speed. Generally, it will take around 20 seconds to perform the ID mapping.

Author(s)

Jing Wang

Examples

```
geneFile<-system.file("extdata","interestingGenes.txt",package="WebGestaltR")
#idmap <- IDMapping(organism="hsapiens",dataType="list",
#inputGeneFile=geneFile, sourceIdType="genesymbol",targetIdType="entrezgene",
#is_outputFile=FALSE,outputFileName="",methodType="R",
#hostName="http://www.webgestalt.org/")
```

listArchiveURL	<i>List the server URL of each version of the data update</i>
----------------	---

Description

The listArchiveURL function can list the server URL of each version of the data update. Users can select different server URL as the hostName to perform the enrichment analysis.

Usage

```
listArchiveURL()
```

Value

The listArchiveURL function can list the server URL of each version of the data update.

Author(s)

Jing Wang

Examples

```
archiveURL <- listArchiveURL()
```

listGeneSet	<i>List gene sets</i>
-------------	-----------------------

Description

The listGeneSet function can list all available gene sets for the selected organism in the WebGestaltR.

Usage

```
listGeneSet(organism="hsapiens",hostName="http://www.webgestalt.org/")
```

Arguments

organism	Currently, the listGeneSet function supports 12 organisms. Users can use the function listOrganism to check the available organisms.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.

Value

The listGeneSet function can list all available gene sets for the selected organism in the WebGestaltR.

Author(s)

Jing Wang

Examples

```
geneSet <- listGeneSet(organism="hsapiens",hostName="http://www.webgestalt.org/")
```

listIDType	<i>List ID type</i>
------------	---------------------

Description

The listIDType function can list all available ID types for the selected organism in the WebGestaltR.

Usage

```
listIDType(organism="hsapiens", hostName="http://www.webgestalt.org/")
```

Arguments

organism	Currently, the listIDType function supports 12 organisms. Users can use the function listOrganism to check the available organisms.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.

Value

The listIDType function can list all available ID types for the selected organism in the WebGestaltR.

Author(s)

Jing Wang

Examples

```
idType <- listIDType(organism="hsapiens", hostName="http://www.webgestalt.org/")
```

listOrganism	<i>List organisms</i>
--------------	-----------------------

Description

The listOrganism function can list all supported organisms in the WebGestaltR.

Usage

```
listOrganism(hostName="http://www.webgestalt.org/")
```

Arguments

hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.
----------	---

Value

The listOrganism function can list all supported organisms in the WebGestaltR.

Author(s)

Jing Wang

Examples

```
organism <- listOrganism(hostName="http://www.webgestalt.org/")
```

listReferenceSet	<i>List reference set</i>
------------------	---------------------------

Description

The listReferenceSet function can list all existing reference gene sets for the selected organism in the WebGestaltR.

Usage

```
listReferenceSet(organism="hsapiens", hostName="http://www.webgestalt.org/")
```

Arguments

organism	Currently, the referenceSet function supports 12 organisms. Users can use the function listOrganism to check the available organisms.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.

Value

The listReferenceSet function can list all existing reference gene sets for the selected organism in the WebGestaltR.

Author(s)

Jing Wang

Examples

```
referenceSet <- listReferenceSet(organism="hsapiens",  
hostName="http://www.webgestalt.org/")
```

readGMT	<i>Read GMT file</i>
---------	----------------------

Description

The readGMT function can read the GMT file and transform to an R matrix object containing three columns: category ID, external link of the category and the annotated genes.

Usage

```
readGMT(gmtFile)
```

Arguments

gmtFile The GMT file with the extension gmt.

Value

This function will return an R matrix object containing three columns: category ID, external link of the category and the annotated genes.

Author(s)

Jing Wang

Examples

```
gmtFile <- system.file("extdata", "exampleGMT.gmt", package="WebGestaltR")
#gmtData <- readGMT(gmtFile)
```

WebGestaltR	<i>Comprehensive R function for the enrichment analysis</i>
-------------	---

Description

WebGestaltR function can perform two popular enrichment analyses: ORA (Over-Representation Analysis) and GSEA (Gene Set Enrichment Analysis). Based on the user uploaded gene list or gene list with scores (for GSEA method), WebGestaltR function will first map the gene list to the entrez gene ids and then summary the gene list based on the GO (Gene Ontology) Slim. After performing the enrichment analysis, WebGestaltR function also returns an user-friendly HTML report containing the ID mapping table, GO Slim summary result and the enrichment analysis result. If the functional categories have the DAG (directed acyclic graph) structure, the structure of the enriched categories can also be visualized in the report.

Usage

```
WebGestaltR(enrichMethod="ORA", organism="hsapiens",
enrichDatabase="geneontology_Biological_Process",enrichDatabaseFile=NULL,
enrichDatabaseType=NULL,enrichDatabaseDescriptionFile=NULL,interestGeneFile=NULL,
interestGene=NULL,interestGeneType=NULL,collapseMethod="mean",referenceGeneFile=NULL,
referenceGene=NULL,referenceGeneType=NULL,referenceSet=NULL, minNum=10, maxNum=500,
fdrMethod="BH",sigMethod="fdr",fdrThr=0.05,topThr=10,dNum=20,perNum=1000,
lNum=20,is.output=TRUE,outputDirectory=getwd(),projectName=NULL,keepGSEAFolder=FALSE,
methodType="R",dagColor="binary",hostName="http://www.webgestalt.org/")
```

Arguments

- enrichMethod** WebGestaltR supports two enrichment analysis methods: ORA (Over-Representation Analysis) and GSEA (Gene Set Enrichment Analysis).
- organism** Currently, WebGestaltR supports 12 organisms. Users can use the function `listOrganism` to check the available organisms. Users can also input others to perform the enrichment analysis based on other organisms not supported by WebGestaltR. For the other organisms, users need to upload the enrichment categories, interesting list and reference list (for ORA method). Because WebGestaltR does not perform the ID mapping for the other organisms, the above uploaded data should have the same ID type.
- enrichDatabase** The functional categories for the enrichment analysis. Users can use the function `listGeneset` to check the available functional databases for the selected organism. Users can also input others to upload the functional database not supported by WebGestaltR for the selected organism.
- enrichDatabaseFile**
If users set `organism` as others or set `enrichDatabase` as others, users need to upload a GMT file as the functional categories for the enrichment analysis. The extension of the file should be `gmt` and the first column of the file is the category ID, the second one is the external link for the category. Genes annotated to the category are from the third column. All columns are separated by tab.
- enrichDatabaseType**
If users set `enrichDatabase` as others, WebGestaltR will also perform ID mapping for the uploaded GMT file. Thus, users need to set the ID type of the genes in the `enrichDatabaseFile`. If users set `organism` as others, users do not need to set this ID type because WebGestaltR will not perform ID mapping for other organisms. The supported ID type of the WebGestaltR for the selected organism can be found by the function `listIDType`.
- enrichDatabaseDescriptionFile**
Users can also upload a description file for the uploaded `enrichDatabaseFile`. The extension of the description file should be `des`. The description file contains two columns: the first column is the category ID that should be exactly the same as the category ID in the uploaded `enrichDatabaseFile` and the second column is the description of the category. All columns are separated by tab.
- interestGeneFile**
If `enrichMethod` is ORA, the extension of the `interestGeneFile` should be `txt` and the file can only contain one column: the interesting gene list. If

	enrichMethod is GSEA, the extension of the interestGeneFile should be <code>rnk</code> and the file should contain two columns separated by tab: the gene list and the corresponding scores.
<code>interestGene</code>	Users can also use the R object as the input. If <code>enrichMethod</code> is ORA, <code>interestGene</code> should be an R vector object containing the interesting gene list. If <code>enrichMethod</code> is GSEA, <code>interestGene</code> should be an R <code>data.frame</code> object containing two columns: the gene list and the corresponding scores.
<code>interestGeneType</code>	The ID type of the interesting gene list. The supported ID type of the WebGestaltR for the selected organism can be found by the function <code>listIDType</code> . If the organism is <code>others</code> , users do not need to set this parameter.
<code>collapseMethod</code>	The method to collapse the duplicate ids for the GSEA method. <code>mean</code> , <code>median</code> , <code>min</code> and <code>max</code> represent the mean, median, minimum and maximum of scores for the duplicate ids.
<code>referenceGeneFile</code>	For ORA method, the users need to upload the reference gene list. The extension of the <code>referenceGeneFile</code> should be <code>txt</code> and the file can only contain one column: the reference gene list.
<code>referenceGene</code>	For ORA method, users can also use the R object as the reference gene list. <code>referenceGene</code> should be an R vector object containing the reference gene list.
<code>referenceGeneType</code>	The ID type of the reference gene list. The supported ID type of the WebGestaltR for the selected organism can be found by the function <code>listIDType</code> . If the organism is <code>others</code> , users do not need to set this parameter.
<code>referenceSet</code>	Users can directly select the reference set from the existing platform in the WebGestaltR and do not need to upload the reference set. All existing platform supported in the WebGestaltR can be found by the function <code>listReferenceSet</code> . If <code>referenceGeneFile</code> and <code>referenceGene</code> are <code>NULL</code> , WebGestaltR will use the <code>referenceSet</code> as the reference gene set. Otherwise, WebGestaltR will use the user uploaded reference set for the enrichment analysis.
<code>minNum</code>	WebGestaltR will exclude the categories with the number of annotated genes less than <code>minNum</code> for the enrichment analysis. The default is 10.
<code>maxNum</code>	WebGestaltR will exclude the categories with the number of annotated genes larger than <code>maxNum</code> for the enrichment analysis. The default is 500.
<code>fdrMethod</code>	For the ORA method, WebGestaltR supports five FDR methods: <code>holm</code> , <code>hochberg</code> , <code>hommel</code> , <code>bonferroni</code> , <code>BH</code> and <code>BY</code> . The default is <code>BH</code> .
<code>sigMethod</code>	Two significant methods are available in the WebGestaltR: <code>fdr</code> and <code>top</code> . <code>fdr</code> means the enriched categories are identified based on the FDR and <code>top</code> means all categories are ranked based on FDR and then selected top categories as the enriched categories. The default is <code>fdr</code> .
<code>fdrThr</code>	The significant level for the <code>fdr</code> method. The default is 0.05.
<code>topThr</code>	The threshold for the <code>top</code> method. The default is 10.
<code>dNum</code>	The number of enriched categories visualized in the final report. The default is 20 and the maximum is 100. A larger <code>dNum</code> will increase the running time.

perNum	The number of permutations for the GSEA method. The default is 1000.
lNum	The number of categories with the output leading edge genes for the GSEA method. The default is 20. Note: GSEA first ranks the categories based on NES (normalized enrichment score) instead of FDR and then outputs the leading edge genes for top lNum categories. Because NES does not necessarily decrease with the increase of the FDR, using sigMethod defined in WebGestaltR to identify the significant categories may cause some categories with outputted leading edge genes are not included in the final result even if the number of significant categories is larger than lNum.
is.output	If is.output is TRUE, WebGestaltR will create a folder named by the projectName and save the mapping results, GO slim summary, enrichment results and an user-friendly HTML report in the folder. Otherwise, WebGestaltR will only return an R data.frame object containing the enrichment results. If hundreds of gene list need to be analyzed simultaneous, it is better to set is.output as FALSE.
outputDirectory	The output directory for the results.
projectName	The name of the project. If projectName is Null, WebGestaltR will use time stamp as the project name.
keepGSEAFolder	If keepGSEAFolder is TRUE, WebGestaltR will keep all folders generated from GSEA tool that contain all figures and tables related to the GSEA analysis.
methodType	For the large ID mapping table (e.g. dbSNP mapping table), Users can use R or Python function to read it. Sometimes Python code is faster than the R code. If users use Python code to read the mapping table, users should first install python and the module pandas in the computer.
dagColor	If dagColor is binary, the significant terms in the DAG structure will be colored by red for ORA method or red (positive related) and blue (negative related) for GSEA method. If dagColor is continuous, the significant terms in the DAG structure will be colored by the red gradient for ORA method or red (positive related) and blue (negative related) gradient for GSEA method based on the corresponding FDR.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.

Value

The WebGestaltR function not only outputs the user-friendly HTML report containing the ID mapping table, GO Slim summary result and the enrichment analysis result but also outputs an R object containing the enrichment analysis result.

Note

Because WebGestaltR will read the data from the server for the ID mapping and GO Slim summary, the running time for the WebGestaltR function will be also based on the internet speed. Generally, it will take around one minute to perform the whole analysis. Because of the huge number of the Gene ontology categories, running GSEA analysis for these categories may take one to six minutes based on the different size of the uploaded gene rank list. Decreasing the parameter maxNum can reduce the running time.

If the user has any problem for the Network Visualization in the HTML report, please follow the instruction in the <http://cytoscapeweb.cytoscape.org/tutorial>.

Author(s)

Jing Wang

Examples

```
#####ORA example#####
geneFile<-system.file("extdata","interestingGenes.txt",package="WebGestaltR")
refFile<-system.file("extdata","referenceGenes.txt",package="WebGestaltR")
outputDirectory<-getwd()
#enrichResult<-WebGestaltR(enrichMethod="ORA",organism="hsapiens",
#enrichDatabase="pathway_KEGG",interestGeneFile=geneFile,
#interestGeneType="genesymbol",referenceGeneFile=refFile,
#referenceGeneType="genesymbol",is.output=TRUE,
#outputDirectory=outputDirectory,projectName=NULL)

#####GSEA example#####
#rankFile<-system.file("extdata","GeneRankList.rnk",package="WebGestaltR")
#outputDirectory<-getwd()
#enrichResult<-WebGestaltR(enrichMethod="GSEA",organism="hsapiens",
#enrichDatabase="pathway_KEGG",interestGeneFile=rankFile,
#interestGeneType="genesymbol",collapseMethod="mean",
#is.output=TRUE,outputDirectory=outputDirectory)
```

WebGestaltR_batch

Batch analysis for the multiple gene or ranked gene lists

Description

The WebGestaltR_batch function can perform the batch analysis for multiple gene or ranked gene lists.

Usage

```
WebGestaltR_batch(interestGeneFolder=NULL,interestGeneType=NULL,enrichMethod="ORA",
organism="hsapiens",enrichDatabase="geneontology_Biological_Process",
enrichDatabaseFile=NULL,enrichDatabaseType=NULL,enrichDatabaseDescriptionFile=NULL,
collapseMethod="mean",referenceGeneFile=NULL,referenceGene=NULL,referenceGeneType=NULL,
referenceSet=NULL,minNum=10,maxNum=500,fdrMethod="BH",sigMethod="fdr",fdrThr=0.05,
topThr=10,dNum=20,perNum=1000,lNum=20,is.output=TRUE,outputDirectory=getwd(),
keepGSEAFolder=FALSE,methodType="R",hostName="http://www.webgestalt.org/",
is_parallel=FALSE,nThreads=3)
```

Arguments

interestGeneFolder	The folder containing multiple interesting gene files. If enrichMethod is ORA, the extension of all files should be txt and each file can only contain one column: the interesting gene list. If enrichMethod is GSEA, the extension of each file should be rnk and the file should contain two columns separated by tab: the gene list and the corresponding scores.
interestGeneType	The ID type of the lists in all files. The supported ID type of the WebGestaltR for the selected organism can be found by the function listIDType. If the organism is others, users do not need to set this parameter. NOTE: the ID type in all files should be the same.
enrichMethod	WebGestaltR supports two enrichment analysis methods: ORA (Over-Representation Analysis) and GSEA (Gene Set Enrichment Analysis).
organism	Currently, WebGestaltR supports 12 organisms. Users can use the function listOrganism to check the available organisms. Users can also input others to perform the enrichment analysis based on other organisms not supported by WebGestaltR. For the other organisms, users need to upload the enrichment categories, interesting list and reference list (for ORA method). Because WebGestaltR does not perform the ID mapping for the other organisms, the above uploaded data should have the same ID type.
enrichDatabase	The functional categories for the enrichment analysis. Users can use the function listGeneset to check the available functional databases for the selected organism. Users can also input others to upload the functional database not supported by WebGestaltR for the selected organism.
enrichDatabaseFile	If users set organism as others or set enrichDatabase as others, users need to upload a GMT file as the functional categories for the enrichment analysis. The extension of the file should be gmt and the first column of the file is the category ID, the second one is the external link for the category. Genes annotated to the category are from the third column. All columns are separated by tab.
enrichDatabaseType	If users set enrichDatabase as others, WebGestaltR will also perform ID mapping for the uploaded GMT file. Thus, users need to set the ID type of the genes in the enrichDatabaseFile. If users set organism as others, users do not need to set this ID type because WebGestaltR will not perform ID mapping for other organisms. The supported ID type of the WebGestaltR for the selected organism can be found by the function listIDType.
enrichDatabaseDescriptionFile	Users can also upload a description file for the uploaded enrichDatabaseFile. The extension of the description file should be des. The description file contains two columns: the first column is the category ID that should be exactly the same as the category ID in the uploaded enrichDatabaseFile and the second column is the description of the category. All columns are separated by tab.
collapseMethod	The method to collapse the duplicate ids for the GSEA method. mean, median, min and max represent the mean, median, minimum and maximum of scores for the duplicate ids.

referenceGeneFile	For ORA method, the users need to upload the reference gene list. The extension of the referenceGeneFile should be txt and the file can only contain one column: the reference gene list.
referenceGene	For ORA method, users can also use the R object as the reference gene list. referenceGene should be an R vector object containing the reference gene list.
referenceGeneType	The ID type of the reference gene list. The supported ID type of the WebGestaltR for the selected organism can be found by the function listIDType. If the organism is others, users do not need to set this parameter.
referenceSet	Users can directly select the reference set from the existing platform in the WebGestaltR and do not need to upload the reference set. All existing platform supported in the WebGestaltR can be found by the function listReferenceSet. If referenceGeneFile and refereneceGene are \code{NULL}, WebGestaltR will use the referenceSet as the reference gene set. Otherwise, WebGestaltR will use the user uploaded reference set for the enrichment analysis.
minNum	WebGestaltR will exclude the categories with the number of annotated genes less than minNum for the enrichment analysis. The default is 10.
maxNum	WebGestaltR will exclude the categories with the number of annotated genes larger than maxNum for the enrichment analysis. The default is 500.
fdrMethod	For the ORA method, WebGestaltR supports five FDR methods: holm, hochberg, hommel, bonferroni, BH and BY. The default is BH.
sigMethod	Two significant methods are available in the WebGestaltR: fdr and top. fdr means the enriched categories are identified based on the FDR and top means all categories are ranked based on FDR and then selected top categories as the enriched categories. The default is fdr.
fdrThr	The significant level for the fdr method. The default is 0.05.
topThr	The threshold for the top method. The default is 10.
dNum	The number of enriched categories visualized in the final report. The default is 20 and the maximum is 100. A larger dNum will increase the running time.
perNum	The number of permutations for the GSEA method. The default is 1000.
lNum	The number of categories with the output leading edge genes for the GSEA method. The default is 20. Note: GSEA first ranks the categories based on NES (normalized enrichment score) instead of FDR and then outputs the leading edge genes for top lNum categories. Because NES does not necessarily decrease with the increase of the FDR, using sigMethod defined in WebGestaltR to identify the significant categories may cause some categories with outputted leading edge genes are not included in the final result even if the number of significant categories is larger than lNum.
is.output	If is.output is TRUE, WebGestaltR will create a folder named by the projectName and save the mapping results, GO slim summary, enrichment results and an user-friendly HTML report in the folder. Otherwise, WebGestaltR will only return an R data.frame object containing the enrichment results. If hundreds of gene list need to be analyzed simultaneous, it is better to set is.output as FALSE.

outputDirectory	The output directory for the results.
keepGSEAFolder	If keepGSEAFolder is TRUE, WebGestaltR will keep all folders generated from GSEA tool that contain all figures and tables related to the GSEA analysis.
methodType	For the large ID mapping table (e.g. dbSNP mapping table), Users can use R or Python function to read it. Sometimes Python code is faster than the R code. If users use Python code to read the mapping table, users should first install python and the module pandas in the computer.
hostName	The server URL for accessing the data. User can use listArchiveURL function to get all archive version URL.
is_parallel	If is_parallel is TRUE, WebGestaltR_batch will use parallel computing to simultaneously analyze the lists in all files.
nThreads	The number of cores used for parallel computing.

Value

If `is.output` is TRUE, each enriched result will be saved in a folder with the name containing the input file name under the `outputDirectory`. Otherwise, the `WebGestaltR_batch` function will return a list object containing all results.

If there are errors during the calculation, error message can also be found in the returned list object.

Author(s)

Jing Wang

Examples

```
###interestGeneFolder contains multiple .txt files for ORA analysis
refFile<-system.file("extdata","referenceGenes.txt",package="WebGestaltR")
outputDirectory<-getwd()
#enrichResult<-WebGestaltR_batch(interestGeneFolder=interestGeneFolder,
#interestGeneType="genesymbol",enrichMethod="ORA",organism="hsapiens",
#enrichDatabase="pathway_KEGG",referenceGeneFile=refFile,
#referenceGeneType="genesymbol",is.output=TRUE,
#outputDirectory=outputDirectory,is_parallel=FALSE)
```

Index

*Topic **methods**

- formatCheck, 3
- GOSlimSummary, 4
- IDMapping, 5
- listArchiveURL, 6
- listGeneSet, 7
- listIDType, 8
- listOrganism, 8
- listReferenceSet, 9
- readGMT, 10
- WebGestaltR, 10
- WebGestaltR_batch, 14

*Topic **package**

- WebGestaltR-package, 2

formatCheck, 3

GOSlimSummary, 4

IDMapping, 5

listArchiveURL, 6

listGeneSet, 7

listIDType, 8

listOrganism, 8

listReferenceSet, 9

readGMT, 10

WebGestaltR, 2, 10

WebGestaltR-package, 2

WebGestaltR_batch, 14