MoNET is an R package providing network enrichment analysis of -omics findings. It is built on top of a manually integrated network, including metabolite-protein interactions, protein interactions and TFBS-SNP relationships. MoNET has 6 functional modules, including network construction, path search, subnetwork extraction, node annotation, pathway enrichment and subnetwork visualization. Below, we demonstrate the usage of these 6 modules. All the files included can be downloaded from <https://github.com/Feng-GitHub-Feng/MoNET>. For each function, detailed information for all parameters can be found through ‘help’ function in R.

1. **Network Construction**

Function BuildNet() helps build a new network by removing the edges from the integrated multi-omic network whose protein interaction score is less than the given threshold. The threshold can be any value between 0 and 1, with 1 as most confident and 0 as not confident at all.

> MulOmicNet<-BuildNet(threshold=0)

[1] "The Multi-Omic Network with gene interaction score larger than 0.000 has been constructed."

1. **Path search**

Given a list of SNPs, genes/proteins or metabolites, paths to other -omics layers will be generated by functions SNPquery(), Genequery() and Metaquery() respectively. A text file containing all paths will be generated, where each line represents a path from one SNP/gene/protein to one metabolite.

Taking SNPquery() as an example, a list of SNPs is read from the text file “snp\_test.txt”. Inside the file, each row is a SNP, indicated either by rs ID or chromosome:position.

>snplist<-read.delim("IGAP\_snp.txt ",header = FALSE)

> head(snplist,10)

V1

1 10:59940129

2 6:32436350

3 6:32512457

4 6:32512459

5 6:32524983

6 6:32524985

7 6:32525208

8 6:32525244

9 6:32527297

10 rs10000546

In addition to a list of SNPs, you need to specify which path to search, from SNP to proteins or SNP to metabolites. Here, *node* is set as "metabolite” to search for paths from SNPs to metabolites. You can also control minimum and maximum length of paths with *minpath* and *maxpath*.

>PathList<-SNPquery(MulOmicNet=MulOmicNet,snplist=snplist,node="metabolite",minpath=3,maxpath=4)

[1] "Identification of paths from 257 snps to 4901 metabolites are about to start."

[1] 12

[1] 13

[1] 18

[1] 26

.......

[1] 256

[1] 257

[1] "End of searching. Writing paths with length between 3 and 4 into text file."

Function PathList() generated a list of paths from query SNPs to metabolites, which is automatically written into a text file “Path.txt”. Each row is a path from one SNP to one metabolite, including IDs for SNPs (rs- or chr:pos), genes/proteins (numbers), reactions (R-) and metabolites (M-).

> Paths<-read.delim(file="Paths.txt",header = FALSE)

> head(Paths,5)

V1

1 rs11039149,10062,3295,R4076,M44

2 rs11039149,10062,1962,R7092,M44

3 rs11039149,10062,3295,R7092,M44

4 rs11039149,10062,1962,R7078,M44

5 rs11039149,10062,51,R7078,M44

Since genes and metabolites both have multiple IDs, function ListFilters() allows users to look up different IDs for genes and metabolites.

> ListFilters("gene")

[1] "Gene\_Entrez\_ID, Gene\_Name, Ensembl\_Gene\_ID, Enzyme\_IDs, RefSeq\_IDs"

> ListFilters("metabolite")

[1] "Metabolites, Names, Formulas, HMDB\_ID, KEGG\_ID, PubChem\_ID, CHEBI\_ID"

1. **Subnetwork extraction**

Function SubNetRFW() helps generate a subnetwork given a list of specified paths. It generates a text file “SubNet.txt” to save the subnetwork where each line representing one interaction among SNPs, genes/proteins, reactions and metabolites.

>SubNetRFW(MulOmicNet=MulOmicNet,PathList=PathList,Anno=FALSE,enrich=FALSE,simplot=FALSE)

[1] "Writing subnetwork file."

> SubNet <- read.delim(file="SubNet.txt")

> head(SubNet,5)

from to

1 rs11039149 10062

2 rs11039297 10658

3 rs11607981 6688

4 rs1303621 2206

5 rs1441586 2206

1. **Node Annotation.**

For function SubNetRFW(), when *Anno* is set to true, one can get annotation files of all SNPs, genes/proteins and metabolites in the final subnetwork.

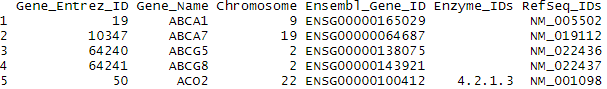
>SubNetRFW(MulOmicNet=MulOmicNet,PathList=PathList,Anno=TRUE,enrich=FALSE,simplot=FALSE)

[1] "Writing subnetwork file."

[1] "Writing annotation file."

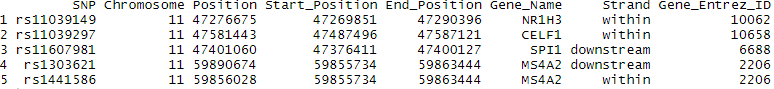
> SubGeneAnno <-read.delim(file="SubGeneAnno.txt")

> head(SubGeneAnno,5)



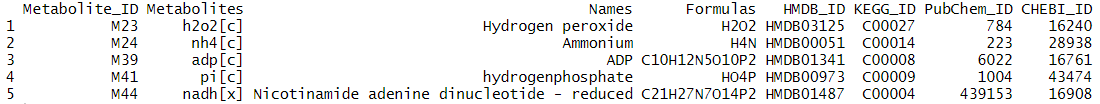
> SubSNPAnno <-read.delim(file="SubSNPAnno.txt")

> head(SubSNPAnno,5)



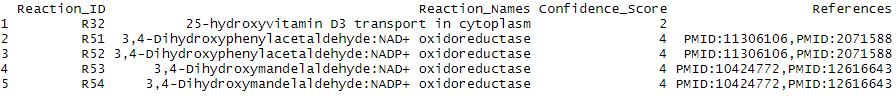
> SubMetaAnno <-read.delim(file="SubMetaAnno.txt")

> head(SubMetaAnno,5)



> SubReacAnno <-read.delim(file="SubReacAnno.txt")

> head(SubReacAnno,5)



1. **Pathway Enrichment.**

For function SubNetRFW(), when *enrich is* set to true, pathway enrichment analysis will be performed using EnrichR to identify pathways enriched by the genes involved in the subnetwork. By default, we use 2019 KEGG human pathway database. Enrichment results will be saved into the text file “Enrichemnt.txt”.

>SubNetRFW(MulOmicNet=MulOmicNet,PathList=PathList,Anno=FALSE,enrich=TRUE,simplot=FALSE)

[1] "Writing subnetwork file."

[1] "Writing enrichment file."

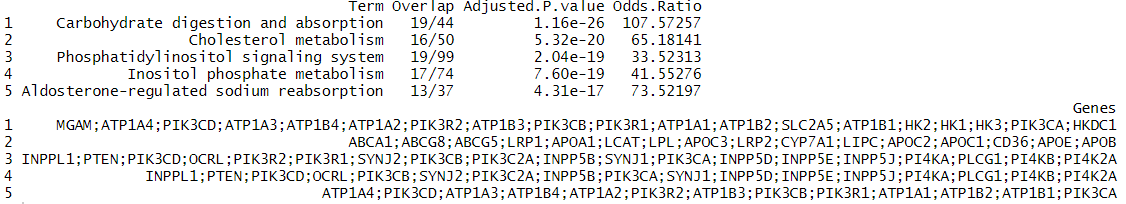
Uploading data to Enrichr... Done.

Querying KEGG\_2019\_Human... Done.

Parsing results... Done.

> Enrichment <- read.delim(file="Enrichment.txt")

> head(Enrichment,5)



1. **Subnetwork Visualization**

For function SubNetRFW(), when *simplot* is set to true, the subnetwork including all the specified paths will be plotted as a graph in PDF format using the force-directed layout algorithm.

>SubNetRFW(MulOmicNet=MulOmicNet,PathList=PathList,Anno=FALSE,enrich=FALSE,simplot=TRUE)

[1] "Writing subnetwork file."

[1] "Plotting subnetwork."

null device

1

