Met2Graph

Introduction

Met2Graph implements a flexible process flow to build graphs starting from a genome-scale metabolic model and can be easily integrated with user-customized functions. It allows the creation of three different types of graphs, based on selection of nodes, edges, and attributes. This document provides a general overview of the functionalities presented in the Met2Graph package. Below, a tutorial on how to use the package functions and get different metabolic graphs is provided.

To report bugs and arising issues, please visit https://github.com/cds-group/Met2Graph

Installation Instructions

System Prerequisites
Met2Graph needs the package sybilSBML, which in turn depends on libSBML to import and read SBML files.
Running functions could fail if this prerequisite library is not available.
Please read through the instructions provided at
https://github.com/cran/sybilSBML/blob/master/inst/INSTALL.

R Package dependencies
The devtools package is needed to install Met2Graph directly from github.

devtools
Package devtools is available at CRAN. For Windows, this seems to depend on having Rtools for Windows installed. You can download and install this from:
http://cran.r-project.org/bin/windows/Rtools/

To install R package devtools call:
```r
   install.packages("devtools")

```

Met2Graph Installation
From GitHub using devtools:
In R console, type:

```r
   library(devtools)
   install_github(repo="Met2Graph", username="cds-group")
```

Getting Started
First, load the library.

```r
library(Met2Graph)
```

Metabolites-based graphs
The metabolites are the nodes, connected by reactions and, if present, by the relative catalyzing enzymes.

Two metabolites are connected if, in one or multiple reactions, one is a substrate and the other one is a product.

The edges can be weighted by the expression values of the corresponding enzymes.

The example below creates a Metabolites-based graph for each sample of expression data of 3 kidney tumor samples downloaded from the Genomic Data Commons data portal (https://portal.gdc.cancer.gov) in the form of FPKM (fragments per kilobase per million reads mapped) normalized read counts (the directory containing the expression files is indicated by "exprDir" argument). The network is created from the metabolic model of kidney tissue downloaded from https://metabolicatlas.org/ in SBML format ("infile").

The parameter "catalyzed"" is TRUE, and only reactions with associated enzymes are kept.

Recurring metabolites (e.g., ATP, CO2, NADP) can be removed to avoid unrealistic paths by setting the parameter "rmMets"==TRUE and using the list of these compounds contained in the data of the package by default or one provided by the user through the parameter "recMets".

metName	metID
ADP[c]	m01285c
ADP[g]	m01285g
ADP[l]	m01285l
ADP[m]	m01285m
ADP[n]	m01285n
ADP[p]	m01285p

The gene-protein-reaction (GPR) associations are automatically extracted from the model. If not present, those extracted from the human generic GEM are used.
Below, Human generic GEM gpr rules downloaded from https://metabolicatlas.org/.

Rxn.name	Gene.reaction.association
HMR_3905	ENSG00000147576 or ENSG00000172955 or ENSG00000180011 or ENSG00000187758 or ENSG00000196344 or ENSG00000196616 or ENSG00000197894 or ENSG00000198099 or ENSG00000248144

| HMR_3907 | ENSG00000117448
|
| HMR_4097 | ENSG00000131069
|
| HMR_4099 | ENSG00000111058 or ENSG00000154930
|
| HMR_4108 | ENSG00000131069
|
| HMR_4133 | ENSG00000131069
|
| HMR_4133 | ENSG00000131069

The resulting graph can be obtained with multiple edges corresponding to the multiple enzymes connecting the two metabolites or can be simplified to single edges. The choice is indicated by the parameter "simpl", where a logical value is required.

Several methods of simplification are proposed, and can be indicated through the parameter "GPRparse".
For this parameter three options are possible:
1) "meanSum": for each edge, expression values of enzymes acting in the same reaction are averaged, and these averages are then summed up for enzymes acting in different reactions;
2) "minMax": according to the GPR associations, enzymes catalyzing the same reaction are associated by "AND" or "OR" logical relationships based on complex or mutual exclusive activity, respectively. For each edge, the weight is given by the minimum expression value of enzymes related by "AND" and the maximum value for "OR". Again, the values of enzymes acting in different reactions are summed up;
3) "minSum": according to the GPR associations, enzymes catalyzing the same reaction are associated by "AND" or "OR" logical relationships based on complex or mutual exclusive activity respectively. For each edge, the weight is given by the minimum expression value of enzymes related by "AND" and the summed value for "OR". Finally, the values of enzymes acting in different reactions are summed up.

In the example multiple edges are simplified by "meanSum" (simpl=TRUE, GPRparse="meanSum").

The parameter "add_rev_rxn" is by default set to FALSE, otherwise the edges will be assigned considering both directions of reversible reactions.

The resulting graph can be built directed or not. Given the rules behind the definition of connections, the networks is naturally directed and the "directed" parameter is by default set to TRUE. Nonetheless the user can choose to not consider this aspect, setting the "directed" parameter to FALSE.
Below, the obtained graphs are weighted and directed.

"outDir" indicates the directory to store the networks and "outFormat" the format in which save the files (see igraph documentation for the other formats).

```{r, echo=TRUE, eval=FALSE}
infile <- system.file("extdata", "kidney.xml", package = "Met2Graph")
```

```
exprDir <- system.file("extdata/expression/", package = "Met2Graph")
outDir <- system.file("extdata/output/", package = "Met2Graph")
Met2MetGraph(infile, catalyzed = TRUE, rmMets = TRUE, exprDir=exprDir, simpl = TRUE ,
GPRparse = "meanSum", outDir= outDir, outFormat = "ncol")
```

Output files:

1) A graph for each sample in the desired format ("ncol" in this case).

2) In case of simplification a matrix with simplified weights of all samples for each metabolites couple.

from	to	TCGA-2Z-A9J5-01A	TCGA-2Z-A9JT-01A	TCGA-4A-A93X-01A
m00001c	m00019c	1.788813202	1.394664028	2.205575381
m00001c	m01450l	3.198852563	2.439335552	2.534157038
m00001c	m01597c	1.788813202	1.394664028	2.205575381
m00001c	m02646l	3.198852563	2.439335552	2.534157038
m00002c	m01510r	2.011966205	1.978748806	1.95790483

If the simplification of edges is not applied, a matrix with not simplified expression values for each enzyme, corresponding metabolites couple and reactions is provided.

gene	TCGA-2Z-A9J5-01A	TCGA-2Z-A9JT-01A	TCGA-4A-A93X-01A	from	to	Reaction
ENSG00000000938	3.662998244	2.585743491	0.877386152	m02593g	m01285c	HMR_7617
ENSG00000000938	3.662998244	2.585743491	0.877386152	m02593g	m00218c	HMR_7617
ENSG00000001036	5.48596813	5.313031446	5.794687005	m00026m	m02237s	HMR_7373
ENSG00000001036	5.48596813	5.313031446	5.794687005	m00026m	m01159s	HMR_7373
ENSG00000001036	5.48596813	5.313031446	5.794687005	m00049p	m01159l	HMR_7375

Reactions-based graphs
Reactions are the nodes connected by edges represented by metabolites.

Two reactions are connected if one produces a metabolite consumed by the other one.

In the example below, the metabolic model of kidney, as above, is given by "infile" argument.

Recurring metabolites (e.g. ATP, CO2, NADP) can be removed to avoid unrealistic paths setting the parameter "rmMets"==TRUE and using the list of these compounds contained in the data of the package by default or one provided by the user through parameter "recMets".

The parameter "add_rev_rxn" is by default set to FALSE, otherwise the edges will be assigned considering both directions of reversible reactions.

"outDir" indicates the directory to store the networks and "outFormat" the format in which save the files (see igraph documentation for the other formats).

The obtained graphs are unweighted and directed.

```{r, echo=TRUE, eval=FALSE}
infile <- system.file("extdata", "kidney.xml", package = "Met2Graph")
outDir <- system.file("extdata/output/", package = "Met2Graph")
Met2RxnGraph(infile, rmMets = TRUE, outDir=outDir, outFormat = "ncol")
```

Output files:
1) Graph in the desired format ("ncol" in this case) and a tabular file containing reaction nodes and metabolite edges.

from	to	Metabolite
HMR_8596	HMR_3552	m00001c
HMR_8596	HMR_0256	m00001c
HMR_9267	HMR_3625	m00002s
HMR_8600	HMR_3625	m00002s
HMR_0686	HMR_1460	m00003c

Enzymes-based graphs
Enzymes are the nodes connected by edges represented by metabolites.

Two enzymes are connected if they catalyze two reactions which produce and consume a metabolite, respectively.

Enzymes for each reaction are annotated according to GPR rules. Enzymes related by AND are complexes and considered as a single node, while OR relationships are split as different nodes. The GPR associations are needed. They are automatically extracted form the model. If not present in the model, the one from the human generic GEM is used.

Recurring metabolites (e.g., ATP, CO2, NADP) can be removed to avoid unrealistic paths by setting the parameter "rmMets"==TRUE and using the list of these compounds contained in the data of the package by default or one provided by the user through the parameter "recMets".

The parameter "add_rev_rxn" is by default set to FALSE, otherwise the edges will be assigned considering both directions of reversible reactions.

The resulting graph can be built directed or not. Given the rules behind the definition of connections, the networks is naturally directed and the "directed" parameter is by default set to

TRUE. Nonetheless the user can choose to not consider this aspect, setting the "directed" parameter to FALSE.
Below, the obtained graphs are unweighted and directed.

"outDir" indicates the directory to store the networks and "outFormat" the format in which save the files (see igraph documentation for the other formats).

```{r, echo=TRUE, eval=FALSE}
infile <- system.file("extdata", "kidney.xml", package = "Met2Graph")
outDir <- system.file("extdata/output/", package = "Met2Graph")
Met2EnzGraph(infile, rmMets=TRUE, outDir=outDir, outFormat="ncol")
```

Output files:
1) Graph in the desired format ("ncol" in this case) and a tabular file containing enzyme nodes and metabolite edges.

from	to	Metabolite
ENSG00000138109	ENSG00000107798	m00001c
ENSG00000138109	ENSG00000170835	m00001c
ENSG00000138109	ENSG00000097021	m00001c
ENSG00000138109	ENSG00000119673	m00001c
ENSG00000138109	ENSG00000136881	m00001c

Plotting graphs
Plot_graphs plots the graphs to pdf. Input directory containing the graphs in the specific "format" needs to be indicated ("inDir"), as well as the path where to save the pdf ("outDir").
Additional plotting parameters can be set (In the example below, e.g., "layout", "vertex.size", "edge.arrow.size", "vertex.label"). See igraph.plotting for the complete list.

```{r, echo=TRUE, eval=FALSE}
inDir <- system.file("extdata/output/MetGraphs/", package = "Met2Graph")
outDir <- system.file("extdata/output/", package = "Met2Graph")
Plot_graphs(inDir=inDir, outDir=outDir, format="ncol", layout=layout_with_graphopt, vertex.size=1,edge.arrow.size=0.1,vertex.label=NA)
```

Import the metabolic model and get the lists of reactant and product metabolites

To give the possibility to explore the model and have a look to the list of reactants and products for each reaction used to create the networks above, the function "MetExtract" is available. It imports and reads a metabolic model in SBML format to get lists of reactants and products for each reaction. In the example below, the metabolic model of kidney tissue downloaded from https://metabolicatlas.org/ is imported. The output objects, consisting of model and two dataframes corresponding to reactants and products, are stored in a list object.

```{r, echo=TRUE, eval=FALSE}
out <- MetExtract(system.file("extdata", "kidney.xml", package = "Met2Graph"))
```

Citation
If you use the Met2Graph package, please cite the following paper:

Granata, I., Manipur, I., Giordano, M., Maddalena, L., Guarracino, M.R. TumorMet: A repository of tumor metabolic networks derived from context-specific Genome-Scale Metabolic Models, Scientific Data, submitted (2022)