NetID setup

This section provides step-by-step instructions to set up the environment to run NetID algorithm in a local computer. A Windows system is recommended.

Step 1. Software installation

Install R, Rstudio, Rtools40, ILOG CPLEX Optimization Studio (CPLEX), preferably at default location.

R (4.0.3 or later) : https://www.r-project.org/

Rstudio (: https://rstudio.com/products/rstudio/download/

Rtools40 (used for R 4.0.0 and newer): https://cran.r-project.org/bin/windows/Rtools/

CPLEX (12.10): https://www.ibm.com/academic/technology/data-science

You need to add R and Rtools40 to Environmental Variables PATH, with instruction provided at the end.

Step 2. Download code

Option 1. Using git (recommended).

(I) Install git

https://support.rstudio.com/hc/en-us/articles/200532077?version=1.3.1093&mode=desktop

(II) In Rstudio, go to File -> New project -> Version control -> Git, enter https://github.com/LiChenPU/NetID.git for URL, select a subdirectory, and create project.

(III) You should be able to see all files in place under your selected subdirectory. Use pull option to check for latest updates.

Option 2. Download files from github

Go to website https://github.com/LiChenPU/NetID, hit the green “code” button, select download zip, and unzip files.

Step 3. Package dependency installation

Most of the dependent packages can be installed by running the r script NetID_packages.R in the “get started” folder. See troubleshoot section for possible errors.

The package, cplexAPI, connecting R to CPLEX, requires additional installation steps.

(I) Go to website: https://cran.r-project.org/web/packages/cplexAPI/index.html, look for Package source, and download cplexAPI_1.4.0.tar.gz. In the same page, look for Materials, and open the INSTALL link.

(II) Unzip the folder cplexAPI to the desktop, follow the installation guide to modify the file Makevars.win.

Note: Replace "\" in the Makevars.win file into "/" in order for R to recognize the path.

For example, the -I"${CPLEX_STUDIO_DIR}\cplex\include" should be replaced with path such as

-I"C:/Program Files/IBM/ILOG/CPLEX_Studio1210/cplex/include"

The -L"${CPLEX_STUDIO_LIB}" should be replaced with path such as

-L"C:/Program Files/IBM/ILOG/CPLEX_Studio1210/cplex/bin/x64_win64"

(III) In command line, run line below to build package, change ${Username} to actual name.

R CMD build --no-build-vignettes --no-manual --md5 "C:\Users\${Username}\Desktop\cplexAPI"

A new package cplexAPI_1.4.0.tar.gz will be built under the default path (for example, C:\Users\${Username})

(IV) In command line, run line below to install package.

R CMD INSTALL --build --no-multiarch .\cplexAPI_1.4.0.tar.gz

If you see “DONE (cplexAPI)”, then the package installation is successful.

Note: if error occurs relating to “__declspec(dllimport deprecated)”, you need to go to C:\Program Files\IBM\ILOG\CPLEX_Studio1210\cplex\include\ilcplex (or other installation path), open the file “cpxconst.h”, go to the line indicated in the error message or search for “__declspec(dllimport deprecated)”, add “_” to “__declspec(dllimport deprecated)”, making it to “__declspec(dllimport_deprecated)”. Save file and repeat step (IV).

(V) To take a short venture using CPLEX in R, refer to “Package cplexAPI – Quick Start” in the link https://cran.r-project.org/web/packages/cplexAPI/index.html.

Using NetID

This section will use yeast negative-mode dataset and mouse liver negative-mode dataset as examples to walk through the NetID workflow.

Yeast negative-mode dataset

In the Sc_neg folder, file raw_data.csv is the output from Elmaven recording MS information, and is the input file for NetID. MS2 is not collected for this dataset.

(1) Running the code

(a) Open code folder -> NetID_run_script.R.

(b) In the # Setting path #### section, set work_dir as "../Sc_neg/".

(c) In the # Read data and files #### section, set filename as “raw_data.csv”, set MS2_folder as “”.

(d) Keep all other parameters as default, and run all lines.

(2) Expected output

(a) In the console, error message should not occur. If optimization step is successful, you will see messages in the following format.

"Optimization ended successfull - integer optimal, tolerance - OBJ_value = 2963.71 (bestobjective - bestinteger) / (1e-10 + |bestinteger|) = 0.000048268"

95.74 sec elapsed

(b) Three files will be generated in the Sc_neg folder.

“NetID_output.csv” contains the annotation information for each peak.

“NetID_output.RData” contains node, edge and network information. The file will be used for network visualization in Shiny R app.

".RData" records the environmental information after running codes. The file is mainly used for development and debugging.

NetID visualization in Shiny R app

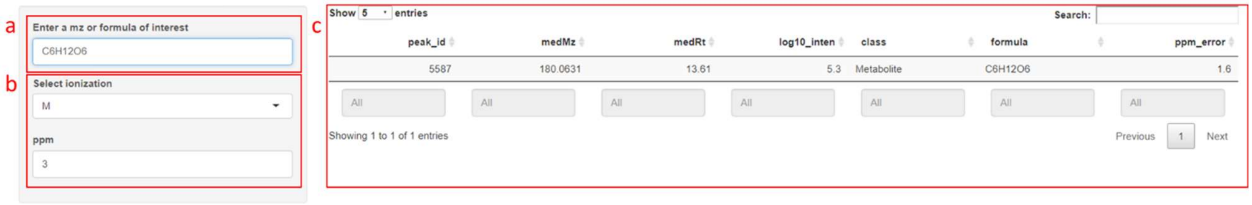
This section provides instruction to visualize and explore NetID output results in the interactive Shiny R app. A 21-inch or larger screen is recommended for best visualization.

(1) Run the Shiny R app.

(a) Open code folder -> R_shiny_App.R.
(b) In the # Read in files #### section, set datapath as "../Sc_neg/".
(c) Keep all other parameters as default, and run all lines.
(d) A Shiny app will pop up.

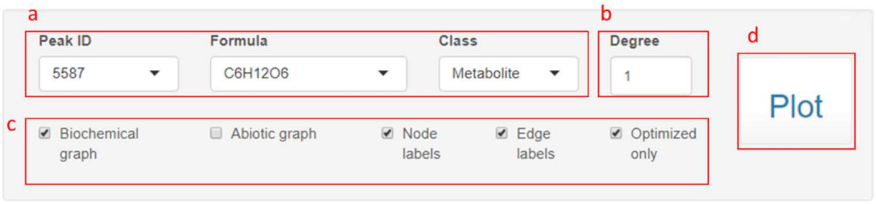
(2) Search peak of interest (top half of Shiny app).

(a) On the left panel, you can enter a m/z or a formula to search your peak of interest. For example, 180.0631 or C6H12O6 will automatically update the data table on the right. Enter 0 to restore full list for the data table.
(b) Change ionization and ppm window to adjust calculated m/z.
(c) On the right, you can explore the peak list in an interactive data table, including global text search on top right, specifying ranges for numeric column or searching text within character columns, ranking each column etc.

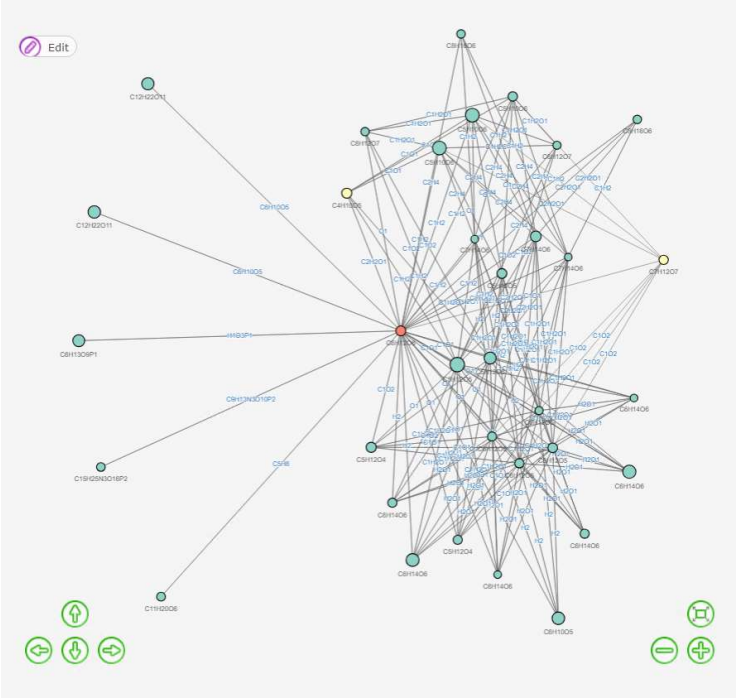


(3) Visualize network (bottom left of Shiny app).

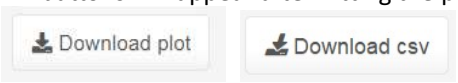
(a) Peak ID, formula and class determines the center node for the network graph. Peak ID will be automatically updated by the first line in the data table if a m/z or formula is given. Alternatively, you can manually enter Peak ID.
(b) The degree parameter controls how far the network expands from the center node. Degree 1 means only nodes directly connected to the center node will be shown and degree 2 means nodes connected to degree 1 will be shown, etc.
(c) Biochemical graph shows biochemical connections. Abiotic graph shows abiotic connections. Node labels and Edge labels determines if the graph show node or edge labels. Optimized only determines whether to show only the optimal annotations or all possible annotations in the network.
(d) When setting parameters, hit plot to see the network graph.



(e) A sample network graph is shown below (a different center node may give less complicated graph). You may edit the nodes or edges (top left), move figures with the arrow buttons (bottom left), and zoom in/out or center figure (bottom right).



(f) You can use the “Download plot” button to download a html webpage to visualize the network graph independent of the Shiny app, and the “Download csv” button to download the information of the nodes in the network. The download buttons will appear after hitting the plot button. Note: edits within the Shiny app will not go into the html file.



(4) Explore possible structures (bottom right of Shiny app).

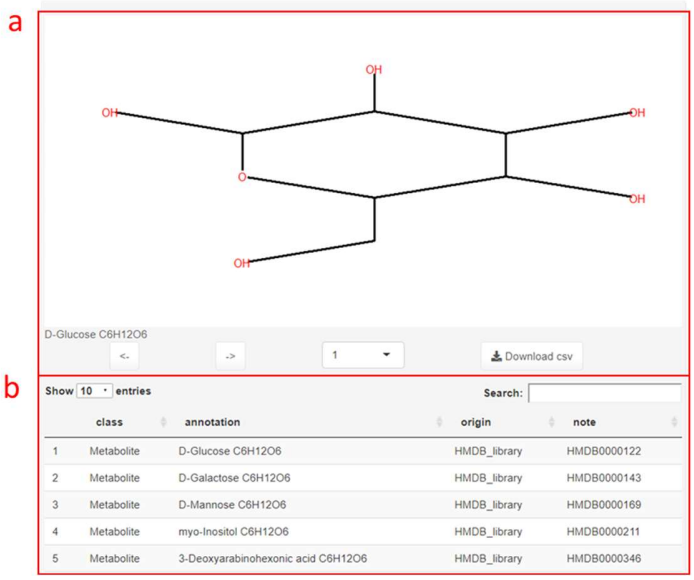
A figure + data table is provided to explore structures of the selected node in the network graph.

(a) The figure shows the chemical structure of the annotated metabolites. If the node is annotated as a putative metabolite, only the known parts of the putative metabolite will be shown.

Scroll left or right, or select the entry number, to visualize different annotations. Right click and select to save image.

(b) In the data table, class has 3 possible entries: Metabolite if it is documented in database such as HMDB library; Putative metabolite if it is transformed from a metabolite through a biotransformation edge; and Artifact if it is transformed by an abiotic edge.

Use the download button to download the data table.



	class	annotation	origin	note
1	Metabolite	D-Glucose C6H12O6	HMDB_library	HMDB0000122
2	Metabolite	D-Galactose C6H12O6	HMDB_library	HMDB0000143
3	Metabolite	D-Mannose C6H12O6	HMDB_library	HMDB0000169
4	Metabolite	myo-Inositol C6H12O6	HMDB_library	HMDB0000211
5	Metabolite	3-Deoxyarabinohexonic acid C6H12O6	HMDB_library	HMDB0000346

Troubleshooting

Failing to install package lc8

Reinstall the packages “devtools” and “digest”.

Cannot find cplexAPI even if the installation seems successful

Check R version used in RStudio to see if cplexAPI is installed under the same R version library. Which R library cplexAPI goes to depends on the R path specified in Environment Variables.

Add R to PATH

(I) Go to Environment Variables:

search PATH in windows -> open edit Environment Variables -> Environment Variables

or

control panel -> system and security -> System -> Advanced system Settings (on your left) -> Advanced -> Environment Variables

(II) In the lower Panel select the “Path Variable” and select Edit, add the R path (C:\Program Files\R\R-4.0.3\bin\x64, if installed at default location) to the Path Variable.

(III) You may need to restart computer for the R path to take effect.

Add Rtools40 to PATH

Option 1.

Add the path “C:\Rtools\bin” to the “Path Variable” in Environment Variables

Option 2.

Run the line in R

writeLines('PATH="${RTOOLS40_HOME}\\usr\\bin;${PATH}"', con = "~/.Renviron")

Use the line below in R console to check for successfully adding Rtools40

Sys.which("make")

Expected output: ## "C:\\rtools40\\usr\\bin\\make.exe"