1. Installation

The following programming languages need to be installed to complete the process

R (we tested using version 3.1)

Python 2.7 (see below for list of packages)

The following files need to be located in the indicated directories (can be the same directory).

**Directory1**: processMicroarray.R, GEO CEL data (downloaded as tar file)

**Directory2**: multidatanetanalysis.py, TI\_2486\_03022014.txt, RG\_03042014.txt, TF\_03042014.txt, HIPPIE\_PPI\_04012014.txt, processed Data generated by processMicroarry.R

1. processMicroarray.R

This script is used to convert Microarray [GEO](http://www.ncbi.nlm.nih.gov/geo/) .CEL into a usable format

a. Running processMicroarray.R

Converts raw CEL data into a usable processed format (.txt)

* + 1. Open R and set directory to the path of the source code

> pd <- ‘PATH\_TO\_DIRECTORY’ # Directory containing the source code of processMicroarray.r

> setwd(pd)

> source(‘processMicroarry.r’)

> wd <- ‘PATH\_TO\_DIRECTORY’ # Directory containing RAW CEL data

> setwd(wd)

* + 1. Use readrawdata (contained in processMicroarray.R) to read in RAW.tar file using the corresponding library name (this will take 5-10min to read the data in, and another hour to annotate, note the progress bar during annotation)

> readrawdata(‘GSE30375\_RAW.tar’, ‘hgu133a.db’)

This will generate a processed file that can be used as input for the multidatanetanalysis step. Make sure that this file named Microarray\_GSEXXXXXX\_processed.txt is moved (if needed) into the same location as multidatanetanalysis.py and required files.

b. RNASeq Processing

For RNASeq data:

Normalization: RPKM

Data transformation: Ln(RPKM+1)

II. multidatanetanalysis.py

multidatanetanalysis.py is the main program that generates the AML network interactions. The following python packages are needed:

os

shutil

timeit

gzip

zipfile

numpy

scipy

matplotlib

Tkinter

tkFileDialog

zipfile

Run the require\_check.py program to determine which of the libraries need to be installed, from the command line run. You will need python version 2.7.

a. require\_check.py

This script will determine which of the necessary packages need to be installed

i. Running require\_check.py

From the command line

$ python require\_check.py

From python shell

>> import require\_check

Any packages not installed will be listed as Need to Install. To install packages follow the instructions for your operating system.

For Linux users sudo easy\_install PACKAGE\_NAME should install the package, otherwise install [pip](https://pip.pypa.io/en/latest/installing.html), and use

$ pip install PACKAGE\_NAME

Links to official sites: [numpy](http://www.numpy.org/), [matplotlib](http://matplotlib.org/downloads.html) and [scipy](http://www.scipy.org/install.html). Link to Windows [binary files](http://www.lfd.uci.edu/~gohlke/pythonlibs/)

b. multidatanetanalysis.py

Open python through command line GUI using python 2.7

Set working directory to the folder containing multidatanetanalysis

Make sure this directory contains the following files:

* TI\_2486\_03022014.txt
* RG\_03042014
* TF\_03042014.txt
* HIPPIE\_PPI\_04012014.txt

i. Running multidatanetanalysis

From the python shell

>>import multidatanetanalysis

a. Single Data Set

Run the following lines for single network/data\_file analysis

$ singledataFile = multidatanetanalysis.SingleDatasetNetworkAssembling(“FILE\_NAME”)

This step will take anywhere from 30min to 40min based on the size of the dataset and will output the cutoffs for the single data file

b. Multiple Data Sets

For multiple datasets run the following line and then select the number of files (where N is the number of files 2, 3, 4 …)

$ multidataFile = multidatanetanalysis.MultiDatasetNetworkAssembling(numFiles = N)

This step may take more than 30min for each dataset

ii. Overlap Generation:

The following steps will generate the overlap files for the inputted datasets. This may take 15min or longer depending on the number of datasets used

$ multidataFile.intgrn() # Generates the Gene Regulation Network (GRN) Overlap files (Called TFG in the corresponding paper)

$ multidataFile.intppi() # Generates the Protein-Protein Interaction (PPI) Network Overlap files

The resulting overlap\_N directories will contain the overlap files for the PPI and GRN subnetworks.