This document is served as the manual to build a new version of FI network

**Set up Eclipse and Load FINetworkBuild Project**

The project used to build the FI network is programmed in Java. Eclipse has been used for all Java related programming. For this project, we have not built a project that can run independent of Eclipse. So you need to download Eclipse from http://www.eclipse.org, set it up and download the FINetworkBuild project from the CVS repository at reactomecurator.oicr.on.ca.

After you connect to the CVS repository, find the FINetworkBuild module, and use “Check Out Project As…” popup menu to create a Java project.

You can find an updated document in the doc folder after you check out the CVS module. Use the updated document.

**Notes:**

1. The configuration file (file name: configuration.prop) in the resources folder is used to configure files and some parameters used by the project. Make sure these three values in the file are correct: YEAR, RESULT\_DIR, and DATA\_SET\_DIR.
2. YEAR is the year when the FI network is built.
3. RESULT\_DIR is used to hold results generated from the project.
4. DATA\_SET\_DIR is used to hold the downloaded files and other source files for the project.

**Overview of Workflow**

The workflow to build a functional interaction (FI) network is described in Figure 1. Some changes may be made version by version. So the following Figure may be updated periodically.

Macintosh HD:Users:gwu:Documents:EclipseWorkspace:caBigR3:doc:FIConstructionWorkFlow.pdf

*Figure 1. Workflow to build the Reactome FI network.*

Two types of data sources are used: Data sources for predicted FIs and data sources for annotated FIs. The data sources for predicted FIs should be downloaded from protein-protein interaction databases: IntAct, HPRD, and BioGrid, pFam (for domain interactions), and from various journals (see below for details). The data sources for annotated FIs are several human curated pathway databases.

**Data Sources for id mapping**

**UniProt**

UniProt is used as the protein reference database in the FI network construction. We need to download the latest version of UniProt so other data sets can be mapped to it. The download site for UniProt is: [ftp.uniprot.org](ftp://ftp.uniprot.org). You don’t need any registration. We only need to download the human data set.

1. After log into [ftp.uniprot.org](ftp://ftp.uniprot.org), download two files from current\_release/knowledgebase/taxonomic\_divisions/: uniprot\_sprot\_human.data.gz and uniprot\_trembl\_huma.dat.gz. You may also download relnotes.txt or README files for future reference. After download these files, unzip them, and place them into a sub-directory in the dataset folder. Change the configuration value in the configuration file for UNIPROT\_DIR to point to this uniprot directory.

# Used to control the UniProt version

UNIPROT\_DIR=${DATA\_SET\_DIR}/UniProt/release\_2013\_11/

1. Besides these files in step 1, you also need to download the fasta file for Isoform sequences to be used in normalizing features for NBC training and predictions from this web site: <http://www.uniprot.org/downloads>: Isoform sequences: the file name should be called “uniprot\_sprot\_varsplic.fasta” after downloaded.

**IPRO\_CLASS**

IPROCLASS\_HUMAN\_FILE

A mapping file is needed to map entrez gene id to UniProt accession number. We need to download an original mapping file from the PIR web site, and will do some pre-process using a Java method later on.

1. Download the mapping file from this ftp site: [ftp.pir.georgetown.edu/databases/](ftp://ftp.pir.georgetown.edu/databases/). Go to idmapping/mapping\_by\_sp, and download file h\_sapiens.tb. Remember where you place your file.
   1. The ftp address may change. Check <https://proteininformationresource.org/pirwww/download/> to find out the address. Under Mac OS, don’t enter any password after enter ‘anonymous’.
2. Modify the following two values in the configuration file:

# This file is used to map Entrez id to UniProt accession number

IPROCLASS\_HUMAN\_FILE = ${DATA\_SET\_DIR}/iproclass/120513/h\_sapiens.tb

ENTREZ\_TO\_UNIPROT\_MAP\_FILE\_NAME=${DATA\_SET\_DIR}/iproclass/120513/EntrezToUniProt.txt

**Data Sources for Annotated FIs**

Annotated FIs are FIs extracted from human curated pathways. Multiple pathway databases have been used. Pathways in other non-Reactome databases are converted to curator tool project first, and then dumped into a modified gk\_central Reactome databases. Different file formats are used for different databases.

**Reactome:**

Reactome is used as the foundation for the FI network build. Public released Reactome database have augmented predicted objects from other non-human species. To minimize the effects of these predicted non-human objects, we use a slice of gk\_central, which is used as the seed for a new public Reactome release.

1. Create a mysqldump for the latest slice database @reactomerelease.oicr.on.ca: Log into reactomerelease, and generate a dump by using the following command:

*mysqldump –u{user\_name} –p test\_slice\_{release\_number} (this db name should be changed for the latest version) > test\_slice\_{release\_number}.sql*

You will be asked to enter your mysql password. Replace text in {} by correct values.

If you want to do build at your local computer, zip the generated sql dump as following:

*jar –Mcvf test\_slice\_{release\_number}.sql }.sql.zip test\_slice\_{release\_number}.sql*

Notes: 1). Tables in the databse are in MyISAM format (use SQL command show table status to make sure). They will be converted into InnoDB using a script below (step 4.a).

2). If you are not sure about the slice database, please ask Lisa Matthew in the Reactome project.

3). For archiving purpose, copy the zipped file to the results folder.

4). Note added on July 6, 2020: Since we will use both human and mouse pathways, use the released database by downloading from the Reactome download site as the base.

1. Create a Reactome database: Copy the above zipped dump file into your build computer (e.g. your desktop machine), unzip it using the following command:

*jar –xvf test\_slice\_{release\_number}.sql.zip*

Log into your mysql database using a user account that can create a database. Create a database as following:

*mysql> create database reactome\_{release\_number}\_plus\_i;*

Here release\_number should be replaced by actual release number (e.g. 79). The release number should be the number in the slice database. Note: Reactome has a quarterly release schedule.

After the empty database is created, run the following two commands to load the generated slice database dump:

*mysql> use reactome\_{release\_number}\_plus\_i;*

*mysql> source test\_slice\_{release\_number}.sql*

Note: In order to make sure mysql can find the dump file, start mysql from the directory containing the unzipped mysqldump file. Or you have to use absolute path to your dump file (no quotation marks)

1. Make changes to the configuration file in the source folder for these three properties: REACTOME\_SOURCE\_DB\_NAME, DB\_USER and DB\_PWD, so that the expanded Reactome database can be used in the future as follows:

# local database that is used as the data source

REACTOME\_SOURCE\_DB\_NAME=reactome\_55\_plus\_i // This should be updated for each build

DB\_USER=root

DB\_PWD=macmysql01

1. The FI network may need the whole set of human SwissProt records. However, the slice database contains a subset of them only. Following steps 1 and 2, get a copy of gk\_central at reactomecurator.oicr.on.ca into your local desktop. The database name has been assumed as following in the configuration file:

REACTOME\_GK\_CENTRAL\_DB\_NAME=gk\_central\_120413

1. Modify the loaded Reactome slice database by running JUnit methods in class org.reactome.data. ReactomeDatabaseModifier:
   1. Before running any Java method, create an empty text file for keeping all logging output in the RESTULT directory. The file name should be called “Combined\_Logging.txt”.
   2. For transaction protection, we need to use InnoDB. Run JUnit method changeMyISAMToInnodb() in the class. You should see logging output as following:

2013-12-05 10:53:37,084 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Drop full text index: Affiliation.address

2013-12-05 10:53:37,318 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Drop full text index: Affiliation\_2\_name.name

2013-12-05 10:53:37,442 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Drop full text index: Book.ISBN

……

2013-12-05 10:55:35,218 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Alter Table to InnoDB: \_InstanceBeforeChange

2013-12-05 10:55:35,424 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Alter Table to InnoDB: \_InstanceBeforeChange\_2\_attributeValuesBeforeChange

2013-12-05 10:55:35,574 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Alter Table to InnoDB: \_Release

* 1. **(Don't’ forget to run this)** Copy human ReferenceGeneProducts that are not in the slice database into the generated Reactome\_plus\_i database by running method, copyHumanReferenceGeneProducts(). (Note: assign 12G as –Xmx). Some of outputs are copied below:

2015-12-07 15:49:41,602 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Total human ReferenceGeneProduct in the source database: 41585

2015-12-07 15:49:41,937 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceGeneProduct:244824] UniProt:Q8NDT2 RBM15B

2015-12-07 15:49:42,542 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:797706] UniProt:B6A8C7-1 TARM1

2015-12-07 15:49:42,739 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:410879] UniProt:Q8IX30-1 SCUBE3

……

2015-12-07 16:07:56,382 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceGeneProduct:385012] UniProt:Q3ZLR7 SUPT20HL1

2015-12-07 16:07:56,396 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:402121] UniProt:O95263-1 PDE8B

2015-12-07 16:07:56,416 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:246797] UniProt:Q59EK9-4 RUNDC3A

2015-12-07 16:07:56,434 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:356555] UniProt:Q2Y0W8-6 SLC4A8

2015-12-07 16:07:56,452 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Copy [ReferenceIsoform:355421] UniProt:Q2KHT4-4 GSG1

2015-12-07 16:08:07,491 [main] INFO org.reactome.data.ReactomeDatabaseModifier - Total copied: 32655

1. Modify Reactome database schema: the regular Reactome database schema needs to be expanded for the FI network. A new attribute called “dataSource” should be added to the top-level class, “DatabaseObject”. Two new classes, Interaction and TargetedInteraction should be created. (TO\_DO: Modify the final database schema for Perl API: need a new Protégé project)

*mysql> source {absolute\_path\_to}/SchemaModification.sql*

Note: You should find a copy of SchemaModifcation.sql in resources folder.

After a successful run of the above sql, check if two new classes (Interaction and TargetedInteraction), and new attribute dataSource for DatabaseObject have been created by connecting your curator tool to your newly created database. Check the database schema view to make sure these three items are there.

In order to use this new expanded Reactome schema for future procedures, we need to export the new schema from the database into a local computer. In the curator tool, after connecting to the modified database, do the following:

In the database schema view, choose “Export Schema” in the File menu, and save the schema into the resource folder in your project folder. This schema file will be needed by the Reactome Java API, which is used extensively in the following procedures.

Note: the file name has to be “schema” without any extension. Don’t change it!

**KEGG**

KEGG has many good pathway diagrams and many disease pathways. We import KEGG pathways using its KGML export. The KEGG KGML is downloaded from KEGG ftp folder. You need to have a license first in order to access its ftp site.

1. Download the following files from the KEGG ftp site: [ftp.bioinformatics.jp](ftp://ftp.bioinformatics.jp)
   1. kegg/xml/kgml/non-metabolic/organisms/hsa.tar.gz: we used human non-metabolic pathways only. Reactome has enough metabolic pathways. Untar the downloaded file before step c since step 3 will download a file with the same name.
   2. kegg/pathway/pathway.list: the whole list of pathways. From this list, we can choose what pathways we want to use. This file is optional.
   3. Also the file map\_title.tab in the /kegg/pathway folder, and hsa.tar.gz in the /kegg/pathway/organisms folder, which will be used for generating gene to pathway mapping. hsa.tar.gz should be unzipped and leave unzipped files as they are.
   4. Note: Make sure unzipped files from two hsa.tar.gz files are in the same folder “hsa”!
   5. kegg/genes/links/genes\_uniprot.list.gz: a mapping file from kegg ids to UniProt ids. After unzipping, this file is around 487 Mb (Dec 22, 2018). You can use the following command to get a human only mapping file for easy view and quick processing (Mac or Linux only):

*grep hsa: genes\_uniprot.list > hsa\_genes\_uniprot.list*

The output file hsa\_genes\_uniprot.list is around 500 K.

1. Specify values in the configuration file for the KEGG data set:

# These parameters are related to KEGG pathways

KEGG\_DIR=${DATA\_SET\_DIR}/KEGG/120513/

# Unzipped human KGML files should be in this directory

KEGG\_HSA\_KGML\_DIR=${KEGG\_DIR}/hsa/

# The converted KEGG pathways is saved in this project file

KEGG\_CONVERTED\_FILE=${KEGG\_DIR}/KEGG.rtpj

# This file is used to map KEGG gene ids to UniProt ids

KEGG\_ID\_TO\_UNIPROT\_MAP\_FILE=${KEGG\_DIR}/hsa\_genes\_uniprot.list

Note: For 2017 version, during the FI network construction, the account to KEGG was not valid. Therefore, new pathways (9 in total) were collected from KEGG’s web site and some code was changed to generate pathway to gene mapping. The 2021 version used the data download in 2020 because of the license issue.

**NCI-PID**

There are two kinds of pathways available in this database: pathways curated by NCI-PID curators and pathways imported from BioCarta and Reactome. Indeed, we don’t need to import Reactome pathways here. BioCarta pathways have not been updated since they were imported in this database. But because the changes of the Reactome schema, these pathways should be re-imported each time if we want to use them.

1. Download pathways from NCI-PID: The data format we are going to use is BioPAX level 2. Download these two BP2 files in its download site, <http://pid.nci.nih.gov/download.shtml>: NCI-Nature Curated Data (BioPAX Level 2), and BioCarta data (BioPAX Level 2). After unzipping, the file names should be:
   1. NCI-Nature\_Curated.bp2.owl
   2. BioCarta.bp2.owl

Note: Since June, 2012, the BP2 file for curated pathways include pathways imported from both Reactome and BioCarta. So this file cannot be used any more. Since there are not many new pathways added, the file downloaded in January, 2012 is used (If you don’t have these files, ask Guanming). This issue needs to be monitored. However, an email using the feedback link was bounced back!!!

Note: On December 15, 2014, the last update is still in December, 2012. So the old files were still used for the 2014 version and afterwards of the FI network.

1. Modified the following configurations in the configuration file:

# Used for the Nature-PID database files

NATURE\_PID\_DIR=${DATA\_SET\_DIR}/NCI-Pathways/011612/

NATURE\_PID\_CURATED=${NATURE\_PID\_DIR}/NCI-Nature\_Curated.bp2.owl

NATURE\_PID\_CURATED\_CONVERTED=${NATURE\_PID\_DIR}/NCI-Nature\_Curated.bp2.rtpj

NATURE\_PID\_BIOCARTA=${NATURE\_PID\_DIR}/BioCarta.bp2.owl

NATURE\_PID\_BIOCARTA\_CONVERTED=${NATURE\_PID\_DIR}/BioCarta.bp2.rtpj

1. Find the BioPAX to Reactome mapping file: resources/BioPAXToReactomeMappers.xml. Make sure the following XML element is **NOT** commented out so that we can use NciPIDBToRPostProcessor class in package org.reactome.b2rPostProcessor during mapping:

<!-- This PostProcessor is for NCI Pathways -->

<postProcessor class=*"org.reactome.b2rPostProcessor.NciPIDBToRPostProcessor"* />

**Panther**

Pathways in the Panther database have not been updated since December, 2007, though the version number has been changed from 2.5 to 3.0.1. Basically the annotation of pathways in Panther has stopped for quite a while. However, because our Reactome schema has been evolved continually, we still need to re-convert pathways from Panther into the Curator Tool project in order to merge with other pathways.

1. Download Pathways: download pathways from Panther’s ftp site, [ftp.pantherdb.org/pathway/current\_release](ftp://ftp.pantherdb.org/pathway/current_release). Please download these two files: SBML\_{version}.zip and SequenceAssociationPathway3.0.1.txt. We use the SBML files, which are the file format used by the Panther annotation. The second file is used to map pathway component ids used in pathways into UniProt accession numbers.

Notes: 1). Annotation at Panther is very slow. However, some pathways have been added into their pathway databases. As of December, 2013, the latest release is 3.2.1 containing 176 pathways (2520 pathway components). The 3.0.1 has 165 pathways (2408 pathway components). It will be nicer to check the latest version of panther pathways, if significant new pathways have been added, probably a new download should be used.

2). One pathway component used in panther pathways can be mapped to multiple UniProt identifiers because these components are actually based on HMM families constructed by Panther. In order to control the quality of imported pathways, we use mappings based on the following reliable confidence only:

"IGI",

"IPI",

"IDA",

"IEP",

"TAS",

"IC",

"IMP",

"RCA"

Notes: 1). Since there is nothing really new added to Panther, the 2014 version of the FI network still used the old downloaded files in directory, Version3.0.1. The generated files have different file names, containing a date (121514). Same for the 2015-2018 version of the FI network and so on.

1. Modify the following values in the configuration file related to Panther:

# Used for the panther database files

# Download on Jan 18, 2011

PANTHER\_DIR=${DATA\_SET\_DIR}/Panther/Version3.0.1/

PANTHER\_FILES\_DIR=${PANTHER\_DIR}/SBML/

PANTHER\_MAPPING\_FILE=${PANTHER\_DIR}/SequenceAssociationPathway3.01.txt

PANTHER\_CONVERTED\_FILE=${PANTHER\_DIR}/Panther\_3\_0\_1\_120513.rtpj

*Note: Some part of Panther converted was developed by Andreas Hoelzlwimmer* [*andreash@ebi.ac.uk*](mailto:andreash@ebi.ac.uk) *during his internship at EBI. Needs to give him an acknowledgement in next FI related publication.*

Note: A file called “SpeciesToChEBIId.txt” in the resources folder has been used for converting. This file is old and has not been updated. This may be fine considering all Panther pathways have not been updated for quite a while.

**TRED**

TRED provides us transcription factors (TF) and their targets interactions. Many of interactions in this database are computationally predicted. We extract only manual curated interactions, which have been supported by literatures if not considering curation errors. The content in the TRED database should not be updated any more. What we need to do is just convert the data in the TRED database into a Curator tool project using the latest Reactome data model. To fetch the data from the TRED database, we use a customized hibernate based API. So have to make sure you have set up your class path correctly in order to use this hibernate API.

1. Install the TRED database into your local computer: If you are building the FI network at your local computer, you may need to install the TRED database for quick performance. If you don’t want to install it, you can access this database at reactomedev.oicr.on.ca. The database name is test\_TRED. To install this database, do a mysqldump from reactomedev, and load the mysqldump into your local mysql database.

Note: There is a back-up file in data\_archive/tred/TRED.sql.zip, which can be used to load the database.

1. Configure the hibernate configuration file: Find file TREDHibernate.cfg.xml in the resources folder. Make changes to these three properties so that they point to the correct values:

<property name=*"connection.url"*>jdbc:mysql://localhost:3306/TRED</property>

<property name=*"connection.username"*>root</property>

<property name=*"connection.password"*>macmysql01</property>

1. Make changes to these two values in the configuration file:

# For TRED files

TRED\_DIR=${DATA\_SET\_DIR}/TRED/

TRED\_CONVERTED\_FILE=${TRED\_DIR}/TRED\_030112.rtpj

**ENCODE**

Encode TF/target interactions were generated by the Geistein’s group in Yale. The interaction files we have used are based on a Nature publication (TODO: add a link to the nature paper), and downloaded from http://archive.gersteinlab.org/proj/encodenets/ (This link is not accessible right now). For our network built, a pre-processed file by combing both proximity and distal interaction together has been generated and stored in our Git repository: data\_archive/encode/tf-targets.txt.

1. Synchronize the above interaction file into your local project.
2. Created a new directory called “encode” in the {DATA\_SET\_DIR} directory, and create a link to the above tf-targets.txt file.
3. Make changes (or Add) the following configurations related to ENCODE:

# ENCODE data

ENCODE\_DIR=${DATA\_SET\_DIR}/encode

ENCODE\_TFF\_FILE=${ENCODE\_DIR}/tf-targets.txt

ENCODE\_TFF\_CONVERTED\_FILE=${ENCODE\_DIR}/tf-targets\_120513.rtpj

Note: There are many TF/Target interactions in the original ENCODE project release. However, many of them may not be related to actual biological functions since they are physical interactions basically. We use a simple filter to pick up TF/Target interactions that are supported by gene co-expression and/or GO BP annotation sharing.

**Data sources for predicted FIs**

**Ensembl-Compara: this is an extremely slow step and should be performed as early as possible.**

Ensembl-compara is used to map PPIs from non-human species to human. We need to have a local compara database to do this mapping by downloading some files from the Ensembl-Compara database:

1. Download files: In this web page, <http://www.ensembl.org/info/data/ftp/index.html>, choose Comparative/MySQL to its ftp site. From the ftp site, choose ensemble\_compara\_xx (xx for release number, 74 as of December, 2013). Download member.txt.gz (changed to seq\_member.txt.gz as of Dec 2014), family.txt.gz, and family\_member.txt.gz. Also download the database schema file: ensembl\_compara\_74.sql.gz. Unzip all files.

Notes: a). The ftp site may be slow. You may consider to check the directory structure out using the following rsync command: rsync -r rsync://ftp.ensembl.org/ensembl/pub/release-94/mysql/ensembl\_compara\_94

b). Or use <ftp://ftp.ensembl.org/ensembl/pub/release-101/mysql/ensembl_compara_101> (direct to the comapara folder for quick list of files).

c). However, it still took too much time to download these files using the local macOS computer. Trying the OHSU linux machine, RWS00061, or find any linux box to download these files using ftp directly. It was much faster.

1. Create ensemble\_compara database: log into a local mysql database, create an ensembl\_compara database by using, “create database ensembl\_compara\_xx” (xx should be the release number for compara). After the database is created, load the schema by using ensembl\_compara\_xx.sql with command: source ensembl\_compara\_74.sql, assuming you start mysql from the directory containing this sql file. Otherwise, provide the absolute path to it.
2. Load data into the database: log out from the mysql database, and run the following command:

mysqlimport –u{mysql\_db\_user} -p --local ensembl\_compara\_62 family.txt family\_member.txt seq\_member.txt

Notes: a). Make sure you use correct mysql user name and password

b). The above import may take a very long time (~hours). Make sure to arrange your time to avoid being stuck with this step.

c). Make sure the following query should return a number around 6200 for yeast proteins.

// 559292 is for Saccharomyces cerevisiae S288c, which is used for protein interaction data.

mysql> select count(\*) from seq\_member where taxon\_id = 559292 and source\_name like 'UniProt%';

+----------+

| count(\*) |

+----------+

| 6282 |

+----------+

1 row in set (0.03 sec)

1. Make changes to the following values in the configuration file:

# For ensembl related files

ENSEMBL\_DIR=${DATA\_SET\_DIR}/Ensembl/release\_74/

ENSEMBL\_COMPARA\_DATABASE=ensembl\_compara\_74

ENSEMBL\_PROTEIN\_FAMILIES=${ENSEMBL\_DIR}/ProteinFamilies.txt

1. Note: we may also connect to the ensembl public mysql databases. However, it is much faster to load the above tables into a local mysql database.

**Protein-Protein Interactions (PPIs)**

Protein protein interactions are downloaded from iRefIndex, a comprehensive protein-protein interaction databases by grouping many popular protein interaction databases together, including IntAct, BioGrid, HPRD, BIND, and others, and creating a non-redundant interaction database using customized hash keys. One problem using this database is that we need to make sure this data source is updated regularly. As of now (February, 2012, release 9.0), it is updated pretty regularly.

1. Download interaction files: the PPIs in iRefIndex are provided in PSIMI-TAB format. The following files (these files names are for release 9.0. Names may have different date in future releases) should be downloaded from its ftp site via this web page: <http://irefindex.org/wiki/index.php?title=iRefIndex>, and unzip them.

// 4932.mitab.10182011.txt.zip (baker’s yeast): this file is not used!

6239.mitab.10182011.txt.zip (worm)

7227.mitab.10182011.txt.zip (fly)

9606.mitab.10182011.txt.zip (human)

10090.mitab.10182011.txt.zip (mouse)

559292.mitab.10182011.txt.zip (yeast S288c)

Note in 2016: The latest version is 14, released in April, 2015. We may need to switch to use another source for PPIs next year? For version 2016 and 2017, this release is used. For version 2018, release 15, created in January, 2018, was used. However, human interactions from Reactome were integrated. The code was changed to remove interactions from Reactome (only human Reactome PPIs are imported) when loading PPIs in the pipeline. There is no need to run method filterReactomeInterations() in class org.reactome.data.IRefIndexMITTabAnalyzer for all human PPIs. This method can be used for test.

1. Based on the above data files, make changes to the configuration data file:

# Constants for iRefIndex data files

IREFINDEX\_DIR=${DATA\_SET\_DIR}/iRefIndex/9.0/

IREFINDEX\_HUMAN\_FILE=${IREFINDEX\_DIR}/9606.mitab.10182011.txt

IREFINDEX\_HUMAN\_PPI\_FILE=${IREFINDEX\_DIR}/HumanPPIsInUniProt022712.txt

IREFINDEX\_YEAST\_FILE=${IREFINDEX\_DIR}/559292.mitab.10182011.txt

IREFINDEX\_YEAST\_PPI\_FILE=${IREFINDEX\_DIR}/YeastPPIsInUniProt022812.txt

IREFINDEX\_YEAST\_TO\_HUMAN\_PPI\_FILE=${IREFINDEX\_DIR}/HumanPPIsFromYeastInUniProt030112.txt

IREFINDEX\_FLY\_FILE=${IREFINDEX\_DIR}/7227.mitab.10182011.txt

IREFINDEX\_FLY\_PPI\_FILE=${IREFINDEX\_DIR}/FLyPPIsInUniProt022812.txt

IREFINDEX\_FLY\_TO\_HUMAN\_PPI\_FILE=${IREFINDEX\_DIR}/HumanPPIsFromFLyInUniProt030112.txt

IREFINDEX\_WORM\_FILE=${IREFINDEX\_DIR}/6239.mitab.10182011.txt

IREFINDEX\_WORM\_PPI\_FILE=${IREFINDEX\_DIR}/WormPPIsInUniProt022812.txt

IREFINDEX\_WORM\_TO\_HUMAN\_PPI\_FILE=${IREFINDEX\_DIR}/HumanPPIsFromWormInUniProt030112.txt

IREFINDEX\_MOUSE\_FILE=${IREFINDEX\_DIR}/10090.mitab.10182011.txt

IREFINDEX\_MOUSE\_PPI\_FILE=${IREFINDEX\_DIR}/MousePPIsInUniProt031412.txt

IREFINDEX\_MOUSE\_TO\_HUMAN\_PPI\_FILE=${IREFINDEX\_DIR}/HumanPPIsFromMouseInUniProt031412.txt

# For storing normalized PPIs mapped from non-human species

HUMAN\_PPIS\_FROM\_YEAST\_FILE=${RESULT\_DIR}/HumanPPIsFromYeast122218.txt

HUMAN\_PPIS\_FROM\_WORM\_FILE=${RESULT\_DIR}/HumanPPIsFromWorm122218.txt

HUMAN\_PPIS\_FROM\_FLY\_FILE=${RESULT\_DIR}/HumanPPIsFromFly122218.txt

HUMAN\_PPIS\_FROM\_MOUSE\_FILE=${RESULT\_DIR}/HumanPPIsFromMouse122218.txt

**Gene-Expression Correlations**

Two gene expression data sets have been used as NBC features: Lee’s Gene Expression and Prieto’s Gene Expression. The original data files were downloaded from their paper’s web sites, and should not be changed any more. However, we need to update protein-pairs to latest version of UniProt.

Note: The original downloaded data files contain gene pairs only. In order to be used as features in NBC, we have to map gene names to UniProt accession numbers. We use downloaded UniProt data file to do this mapping. We use the swissprot part of UniProt data for doing the mapping for the Prieto data file. For Lee’s data file, we use the original downloaded mapping file and normalize with the latest UniProt data.

1. Get the archived source files from CVS: synchronize data\_archive/data\_archive.zip. Unzip this zipped file. You should find two files: GeneExpWith3FromPavlidis.txt under the LeeGeneExp folder, and union60.txt in the PrietoGeneExp folder.
2. Make sure these two constants in the configuration file have been set correctly (usually you don’t need to make a change):

# Two Gene expression data files

LEE\_GENE\_EXP\_FILE\_SOURCE=${DATA\_SET\_DIR}/microarray/Pavlidis/GeneExpWith3FromPavlidis.txt;

LEE\_GENE\_EXP\_FILE=${RESULT\_DIR}/LeeGeneExp.txt

PRIETO\_PAIRS\_FILE=${DATA\_SET\_DIR}/microarray/PrietoCarlos/union60.txt

PRIETO\_GENE\_EXP\_FILE=${RESULT\_DIR}/PrietoGeneExp.txt

**Gene Ontology Annotation**

The sharing of GO annotation (Biological Process only) is used as another feature in our NBC training and prediction. In our usage, we check a simple sharing only without considering any parent-child relationships among GO terms.

1. Download GO annotation file for homo sapiens from the gene ontology’s web site: <http://www.geneontology.org/GO.downloads.annotations.shtml>. The file name should be: gene\_association.goa\_human.
   1. Note: As of 2016, the file name has been changed to goa\_human.gaf. However, in the web application configuration file, gene\_association.goa\_human, is still used. Added a symbol link to folder 122816 to avoid changing other configurations.
   2. Reactome annotation may have been included in the annotation. As of the 2018 version of the FI network, these annotations have been excluded. However, they are still included for GO enrichment analysis as expected.
2. Another file, GO.terms\_and\_ids, will be used in converting BioPAX for BioCarta pathways, and be used for GO enrichment analysis for FI-plug-in. Download this file from this URL: <http://www.geneontology.org/doc/GO.terms_and_ids> (use save as to create a file in the GO directory, see step 3. Note: the link cannot work anymore on Dec 29, 2021). The file name should be “GO.terms\_and\_ids.txt” (about 2.1 Mb).
   1. Note: It was found that this terms to ids mapping file was created in 2012 and not updated since then. For the 2016 version of the Reactome FI network and after, go.obo will be used instead. This obo file can be downloaded via <http://current.geneontology.org/ontology/index.html> (choose go.obo and save as…). Classes related to this change has been updated. The Go.terms\_and\_ids.txt file will be generated later on. There is no need to handle this right now.
3. Modify the following value in the configuration file to point to the correct directory that holds the above downloaded file:

# For GO related files

GO\_DIR=${DATA\_SET\_DIR}/GO/122217/

GOA\_FILE\_NAME=${GO\_DIR}/goa\_human.gaf

**Domain-Domain Interactions**

We use domain-domain interactions from the pFam database.

1. Download domain interaction file from pFam: go to pFam release ftp site, <ftp://ftp.ebi.ac.uk/pub/databases/Pfam>, go to the releases folder, the latest release, and database\_files. Download three files: pfamA.txt.gz and pfamA\_interactions.txt.gz, and version.txt.gz, unzip them.
2. Specify the following in the configuration file as being used in step 2:

# Directory for pFam

PFAM\_DIR\_NAME=${DATA\_SET\_DIR}/Pfam/27.0/

**Functional Interaction Network Build**

Run the following methods in order in class org.reactome.fi.FINetworkBuilder

Notes: 1). It is suggested to copy all logging output from Eclipse in a file for future reference. A file Combined\_Logging.txt is created for this purpose.

2). Make sure all directories configured should be there in the dataset directory. Otherwise, exceptions may be thrown.

1. **prepareMappingFile()**

Example logging from running this method like this:

2012-04-17 13:38:07,578 [main] INFO org.reactome.fi.FINetworkBuilder - Running UniProtAnalyzer.generateUniProtIDsMap()...

total entries in uniprot: 178240

total entries in uniprot: 79584

2012-04-17 13:38:13,368 [main] INFO org.reactome.fi.FINetworkBuilder - Running UniProtAnalyzer.generateEntrezGeneToUniProt()...

Counter: 70246

2012-04-17 13:38:14,364 [main] INFO org.reactome.fi.FINetworkBuilder - Running UniProtAnalyzer.generateUniToPfamMap()...

After this method running, 5 files should be generated as following:

Uni2Pfam.txt in the UniProt directory

SwissProtACIDMap.txt in the UniProt directory

ACIDMap.txt in the UniProt directory

ENTREZ\_TO\_UNIPROT\_MAP\_FILE\_NAME as configured (e.g. ${DATA\_SET\_DIR}/iproclass/122816/EntrezToUniProt.txt)

GO.terms\_and\_ids.txt in the configured GO directory

1. **convertPathwayDBs()**

Convert Pathways in KEGG, NCIPID, and Panther, and TF/Target interactions into individual curator tool project files. You may need to open these converted project files in the curator tool to see how they look like and make sure they are correct.

Notes: -1). It is found that new JDK 1.8 cannot work with old jar files used by the Protégé API. A workable version id jdk8u212-b04-jre, which can be downloaded from <https://adoptopenjdk.net>. Note for Version 2021: Apparently adoptopenjdk-11.jdk works fine.

0). In order to keep the logging file, make sure log4j.prop is configured to write output to a file.

1). Make sure to assign enough memory for running this method (e.g. –Xmx10240m).

2). On the KEGG download in January, 2012, there are 26 KEGG ids cannot be mapped to UniProts. Check to ID mapping in UniProt, find three mappings. These unmapped KEGG ids are left without further processing since the number is very small. The unmapped ids are listed in file UnMappedKeggIds.txt (this file was created manually from the output in the Eclipse console). (173 KEGG ids in December, 2013; 177 KEGG ids in December, 2014 couldn’t be mapped; 209 in 2015)

3). In the version of FI network constructed in 2009, one KEGG id in pathway has been mapped to one UniProt only. This has been changed now. A KEGG id can be mapped to multiple UniProt via DefinedSet. So a DefinedSet converted from KEGG may contain another DefinedSet.

4). ReferenceIsoform instances may have been fetched out from the database during converting. However, our converting cannot take use of isoforms yet. So the Isoform instances should be treated as the top-level ReferenceGeneProduct instance!

5). We use KGML to import KEGG pathways. However, many complexes have not been described in KGML files. So interactions from these complexes cannot be extracted. For example, see complex HAP1-HTT-Dyneim-Dunactin in Huntington’s disease pathway: <http://www.genome.jp/kegg/pathway/hsa/hsa05016.html>. It seems there is no good file that can be used to extract these complexes. Probably a future import can use map files directly (e.g. hsadd05016.conf).

6). Because of the above problem, the pathway to protein/gene mapping file extracted for KEGG uses KGML files directly, instead of imported Reactome pathways (see below).

7). For converting the BioCarta pathways in the NCI-PID pathway, a file called GO.terms\_and\_ids.txt in the resource folder has been used, which is required by BioPAXToReactomeConverter in another project, PathwayExchange. This file will be copied from the GO directory to the resources directory so that an updated version can be used by the method automatically (Note: This step is handled by the program, there is no need to copy it manually).

8). For converting pathways in the Panther database, a file called “SpeciesToChEBIId.txt” in the resources folder is used. This file is old and has not been updated. This may be fine considering all Panther pathways have not been updated for quite a while.

9). Starting with Version 2020, literature references used by Panther are not processed any more since the NCBI EUtil API used by auto-fetching detailed pubmed information is just too slow. Also the detailed information for imported panther pathways are not really investigated.

10). For TRED database, we use gene names, which are used in the TRED database, in our target Reactome database directly to map to UniProt. Some of gene names in TRED cannot be mapped. See below (partial list):

……

2013-12-06 15:01:15,278 [main] WARN org.reactome.convert.common.PostProcessTemplate - PWCR1 for EWAS -4791 cannot be mapped to UniProt!

2013-12-06 15:01:16,685 [main] WARN org.reactome.convert.common.PostProcessTemplate - NYD-SP28 for EWAS -5071 cannot be mapped to UniProt!

2013-12-06 15:01:18,931 [main] WARN org.reactome.convert.common.PostProcessTemplate - HMG-I for EWAS -5546 cannot be mapped to UniProt!

2013-12-06 15:01:21,482 [main] WARN org.reactome.convert.common.PostProcessTemplate - MIG-6 for EWAS -6256 cannot be mapped to UniProt!

2013-12-06 15:01:21,648 [main] WARN org.reactome.convert.common.PostProcessTemplate - LOC339524 for EWAS -6278 cannot be mapped to UniProt!

2013-12-06 15:01:25,102 [main] WARN org.reactome.convert.common.PostProcessTemplate - FLJ23356 for EWAS -7005 cannot be mapped to UniProt!

2013-12-06 15:01:26,128 [main] WARN org.reactome.convert.common.PostProcessTemplate - col4a1 for EWAS -7393 cannot be mapped to UniProt!

2013-12-06 15:01:26,128 [main] WARN org.reactome.convert.common.PostProcessTemplate - col4a2 for EWAS -7397 cannot be mapped to UniProt!

2013-12-06 15:01:26,882 [main] WARN org.reactome.convert.common.PostProcessTemplate - RPS3a for EWAS -7709 cannot be mapped to UniProt!

2013-12-06 15:01:30,853 [main] WARN org.reactome.convert.common.PostProcessTemplate - LOC285847 for EWAS -8409 cannot be mapped to UniProt!

2013-12-06 15:01:31,805 [main] WARN org.reactome.convert.common.PostProcessTemplate - TRD@ for EWAS -8742 cannot be mapped to UniProt!

2013-12-06 15:01:31,944 [main] INFO org.reactome.convert.common.PostProcessTemplate - Total unmapped names: 48

10). For the Encode interaction, we got much more unmapped gene names:

……

2013-12-06 15:09:17,791 [main] WARN org.reactome.convert.common.PostProcessTemplate - LOC100289673 for EWAS -52795 cannot be mapped to UniProt!

2013-12-06 15:09:18,127 [main] WARN org.reactome.convert.common.PostProcessTemplate - LOC283314 for EWAS -52847 cannot be mapped to UniProt!

2013-12-06 15:09:19,042 [main] WARN org.reactome.convert.common.PostProcessTemplate - SAA2 for EWAS -52939 cannot be mapped to UniProt!

2013-12-06 15:09:20,498 [main] WARN org.reactome.convert.common.PostProcessTemplate - SNORD59A for EWAS -53527 cannot be mapped to UniProt!

2013-12-06 15:09:22,622 [main] WARN org.reactome.convert.common.PostProcessTemplate - SNORD59B for EWAS -54534 cannot be mapped to UniProt!

2013-12-06 15:09:25,656 [main] INFO org.reactome.convert.common.PostProcessTemplate - Total unmapped names: 602

11). We may get a rough idea how many proteins can be merged into the original Reactome from the output in the last section like the following (ids for UniProt accession numbers):

Total ids from Reactome: 7447 // This number should be bigger in the latest release.

Total ids from KEGG: 7897

Total ids after merging: 11525 (0.5163986323920476, 4078)

Total SwissProt ids after merging: 8534 (0.421473725800079)

Total ids from Nature-PID: 2573

Total ids after merging: 11984 (0.17839098328799066, 459)

Total SwissProt ids after merging: 8977 (0.44335242986961676)

Total ids from Biocarta-PID: 1526

Total ids after merging: 12785 (0.5249017038007864, 801)

Total SwissProt ids after merging: 9076 (0.4482418016594232)

Total ids from TRED: 1142

Total ids after merging: 12951 (0.14535901926444833, 166)

Total SwissProt ids after merging: 9242 (0.4564401422362703)

Total ids from ENCODE: 9418

Total ids after merging: 17576 (0.48948821405818643, 4610)

Total SwissProt ids after merging: 13943 (0.6889855215694026)

Total ids from Panther: 1423

Total ids after merging: 17684 (0.0758959943780745, 108)

Total SwissProt ids after merging: 14051 (0.6943222809704995)

Total ids from iRefIndex PPIs: 13864

Total ids after merging: 21643

Total SwissProt ids after merging: 16504 (0.8155358995898602)

Note: You may not get the last section about iRefIndex in the above output with an exception thrown (FileNotFoundException). That should be fine since you have not created the needed PPI file.

Notes for the 2014 version: 1). There is un-documented type=”other” in the entry elements for pathway, hsa03320.xml, which broke the code. However, these types didn’t exist in the 2013 version of hsa03320.xml and other XML files in the current version. Copied the old 2013 version into the 2014 folder to avoid this problem. 2). Same in the 2015 version. However, these other types have been converted into OtherEntity type. They are groups of drugs.

1. **dumpPathwayDBs()**

Dump the converted curator tool projects into the extended Reactome database created before, and generate FI files from these converted projects.

Notes: 0). It is prudent to dump the reactome\_plus\_i database first before running this method just in case there is anything wrong to save time.

1). You are encouraged to compare the numbers from different pathways to ones in the previous reactome\_plus\_i database to make sure there is no big surprise there. You can use the dataSource attribute to get a list of converted Event instances for a specific database (e.g. KEGG, Panther) with the following data sources: pantherdb (2019: 3066), Pathway Interaction Database (2019: 7616), TRED (2019: 3096), KEGG (2019: 9165), BioCarta – Imported by PID (2907), ENCODE (45328).

2). Some of UniProt identifiers that cannot be mapped to the original ReferenceGeneProducts from different databases may be duplicated in the database. Just leave them as they are. But have to remember to merge them when counting how many UniProt accessions in the database. Avoid checking the numbers based on ReferenceGeneProduct instances only.

1. **dumpPathwayFIs()**

Notes: 1). For the 2014 version of the FI network, hasCandidate values are not considered any more, which reduces the number of FIs extracted from Reactome about 11% from the 2013 version (aka from 144733 to 127382).

2). The current implementation will not extract TF/Target interactions annotated in Reactome using BlackboxEvent (e.g. DB\_ID = 452894) because the extraction checks only ReferenceGeneProducts. For the time being, this should be fine since Reactome doesn’t have many TF/Target interactions. We should get enough from TRED and ENCODE. Most likely these interactions will be extracted later on. Here are numbers for the 2014 FI network: if TF/target interactions are considered: 128,020; 127,382 if not considered (0.5%).

3). The 2017 version of the FI network got less FIs from complexes than 2016 from Reactome directly. This is a little bit surprising though more FIs were extracted from reactions.

The dumped pathway FIs should have similar numbers or more as follows (July 2012):

Total interactions from reactions: 130397

Time for looping: 19913

Total interactions from Reactome: 171024

After filtering: 146823

Before sequence consolidating: 146807

After sequence consolidating: 146414

Done data source: Reactome

Total interactions from reactions: 14514

Time for looping: 3950

Total interactions from Reactome: 15381

After filtering: 15381

Before sequence consolidating: 15381

After sequence consolidating: 15381

Done data source: pantherdb

Total interactions from reactions: 15000

Time for looping: 50670

Total interactions from Reactome: 15797

Total interactions from Interaction Event: 0

After filtering: 15716

Before sequence consolidating: 15694

After sequence consolidating: 15694

Done data source: Pathway Interaction Database

Total interactions from reactions: 5701

Time for looping: 21278

Total interactions from Reactome: 6763

Total interactions from Interaction Event: 0

After filtering: 6763

Before sequence consolidating: 6763

After sequence consolidating: 6761

Done data source: BioCarta - Imported by PID

Total interactions from reactions: 0

Time for looping: 226

Total interactions from Reactome: 2551

Total interactions from Interaction Event: 58807

After filtering: 60923

Before sequence consolidating: 60923

After sequence consolidating: 50415

Done data source: Kegg

Total interactions from reactions: 0

Time for looping: 0

Total interactions from Reactome: 0

Total interactions from Interaction Event: 2856

After filtering: 2856

Before sequence consolidating: 2856

After sequence consolidating: 2856

Done data source: TRED

Total interactions from reactions: 0

Time for looping: 0

Total interactions from Reactome: 0

Total interactions from Interaction Event: 41376

After filtering: 41376

Before sequence consolidating: 41376

After sequence consolidating: 41370

Done data source: ENCODE

(Note: the numbers for ENOCDE for not-filtered results).

1. **prepareNBCFeatures()**

Try to set up a large Xmx for this method (e.g. -Xmx12G, as large as possible). Output from the above running:

2012-04-17 14:47:23,100 [main] INFO org.reactome.fi.FINetworkBuilder - Running EnsemblAnalyzer.dumpProteinFamilies()...

Size of familyToProteins(): 44475

Total families before filtering: 44475

Total families after filtering: 11202

2012-04-17 14:47:26,823 [main] INFO org.reactome.fi.FINetworkBuilder - Running IRefIndexMITTabAnalyzer.loadHumanPPIs()...

Total PPIs: 112639

Total ids: 20722

in SwissProt: 12154

in UniProt: 14728

Total ids: 13864

in SwissProt: 11538

in UniProt: 13864

Total time: 7245

2012-04-17 14:47:34,068 [main] INFO org.reactome.fi.FINetworkBuilder - Running IRefIndexMITTabAnalyzer.loadFlyPPIs()...

Total PPIs: 38940

Total time: 2634

2012-04-17 14:47:36,703 [main] INFO org.reactome.fi.FINetworkBuilder - Running IRefIndexMITTabAnalyzer.loadWormPPIs()...

Total PPIs: 11626

Total time: 701

2012-04-17 14:47:37,404 [main] INFO org.reactome.fi.FINetworkBuilder - Running IRefIndexMITTabAnalyzer.loadYeastPPIs()...

Total PPIs: 214848

Total time: 9350

2012-04-17 14:47:46,755 [main] INFO org.reactome.fi.FINetworkBuilder - Running IRefIndexMITTabAnalyzer.loadMousePPIs()...

Total PPIs: 12787

Total time: 784

2012-04-17 14:47:47,539 [main] INFO org.reactome.fi.FINetworkBuilder - Running PsiMiOrthologyAnalyzer.generateHumanPPIsFromYeastInUniProt()...

Total yeast proteins: 1272

Mapped human PPIs from other species: 214848 -> 1368910

After filtering: 1368910

Before sequence consolidating: 1368910

After sequence consolidating: 1304755

Before normalization: 1368910, after: 1304755

2012-04-17 14:48:07,954 [main] INFO org.reactome.fi.FINetworkBuilder - Running PsiMiOrthologyAnalyzer.generateHumanPPIsFromWormInUniProt()...

Mapped human PPIs from other species: 11626 -> 129619

After filtering: 129619

Before sequence consolidating: 129619

After sequence consolidating: 122547

Before normalization: 129619, after: 122547

2012-04-17 14:48:21,018 [main] INFO org.reactome.fi.FINetworkBuilder - Running PsiMiOrthologyAnalyzer.generateHumanPPIsFromFlyInUniProt()...

Mapped human PPIs from other species: 38940 -> 394801

After filtering: 394801

Before sequence consolidating: 394801

After sequence consolidating: 376781

Before normalization: 394801, after: 376781

2012-04-17 14:48:35,018 [main] INFO org.reactome.fi.FINetworkBuilder - Running PsiMiOrthologyAnalyzer.generateHumanPPIsFromMouseInUniProt()...

Total mouse proteins: 41655

Mapped human PPIs from other species: 12787 -> 854318

After filtering: 854318

Before sequence consolidating: 854318

After sequence consolidating: 770385

Before normalization: 854318, after: 770385

2012-04-17 14:48:51,936 [main] INFO org.reactome.fi.FINetworkBuilder - Checking odds ratio...

File: datasets/iRefIndex/9.0//HumanPPIsInUniProt022712.txt

Total checked pairs: 95959

Total: 140095

Mapped to ppi: 8542 (0.060973)

Mapped to random: 141 (0.001006)

Mapped to random: 152 (0.001085)

Mapped to random: 154 (0.001099)

Mapped to random: 139 (0.000992)

Mapped to random: 128 (0.000914)

Mapped to random: 145 (0.001035)

Mapped to random: 160 (0.001142)

Mapped to random: 147 (0.001049)

Mapped to random: 135 (0.000964)

Mapped to random: 149 (0.001064)

Average odds ratio: 62.919799976046136 +- 4.227475795333164 (from 10 tests)

File: results/2012/HumanPPIsFromYeast030112.txt

Total checked pairs: 1304755

Total: 140095

Mapped to ppi: 2210 (0.015775)

Mapped to random: 206 (0.001470)

Mapped to random: 216 (0.001542)

Mapped to random: 200 (0.001428)

Mapped to random: 163 (0.001163)

Mapped to random: 178 (0.001271)

Mapped to random: 226 (0.001613)

Mapped to random: 167 (0.001192)

Mapped to random: 177 (0.001263)

Mapped to random: 172 (0.001228)

Mapped to random: 187 (0.001335)

Average odds ratio: 11.988223137514396 +- 1.326957494425449 (from 10 tests)

File: results/2012/HumanPPIsFromWorm030112.txt

Total checked pairs: 122547

Total: 140095

Mapped to ppi: 593 (0.004233)

Mapped to random: 33 (0.000236)

Mapped to random: 26 (0.000186)

Mapped to random: 30 (0.000214)

Mapped to random: 38 (0.000271)

Mapped to random: 26 (0.000186)

Mapped to random: 41 (0.000293)

Mapped to random: 27 (0.000193)

Mapped to random: 31 (0.000221)

Mapped to random: 31 (0.000221)

Mapped to random: 37 (0.000264)

Average odds ratio: 19.04322216358146 +- 3.001409423056792 (from 10 tests)

File: results/2012/HumanPPIsFromFly030112.txt

Total checked pairs: 376781

Total: 140095

Mapped to ppi: 1944 (0.013876)

Mapped to random: 65 (0.000464)

Mapped to random: 75 (0.000535)

Mapped to random: 75 (0.000535)

Mapped to random: 71 (0.000507)

Mapped to random: 75 (0.000535)

Mapped to random: 74 (0.000528)

Mapped to random: 70 (0.000500)

Mapped to random: 80 (0.000571)

Mapped to random: 71 (0.000507)

Mapped to random: 79 (0.000564)

Average odds ratio: 26.897110981818983 +- 1.6644923160996432 (from 10 tests)

File: results/2012/HumanPPIsFromMouse031412.txt

Total checked pairs: 770385

Total: 140095

Mapped to ppi: 4645 (0.033156)

Mapped to random: 125 (0.000892)

Mapped to random: 117 (0.000835)

Mapped to random: 166 (0.001185)

Mapped to random: 131 (0.000935)

Mapped to random: 141 (0.001006)

Mapped to random: 139 (0.000992)

Mapped to random: 129 (0.000921)

Mapped to random: 147 (0.001049)

Mapped to random: 142 (0.001014)

Mapped to random: 127 (0.000907)

Average odds ratio: 35.49923808754484 +- 3.4322625312095267 (from 10 tests)

2012-04-17 14:48:58,373 [main] INFO org.reactome.fi.FINetworkBuilder - Running MicroarrayDataAnalyzer.normalizeLeeGeneExp()...

Pairs before normalizing: 209883

After filtering: 205936

Before sequence consolidating: 205903

After sequence consolidating: 205903

Pairs after normalizing: 205903

2012-04-17 14:49:11,354 [main] INFO org.reactome.fi.FINetworkBuilder - Checking its odds ratio...

Total Co-expression: 205903

Total checked pairs: 205903

Total: 140095

Mapped to ppi: 5853 (0.041779)

Mapped to random: 339 (0.002420)

Mapped to random: 348 (0.002484)

Mapped to random: 354 (0.002527)

Mapped to random: 349 (0.002491)

Mapped to random: 359 (0.002563)

Mapped to random: 343 (0.002448)

Mapped to random: 321 (0.002291)

Mapped to random: 351 (0.002505)

Mapped to random: 319 (0.002277)

Mapped to random: 344 (0.002455)

Average odds ratio: 17.804913094911186 +- 0.7129963491801969 (from 10 tests)

2012-04-17 14:49:12,476 [main] INFO org.reactome.fi.FINetworkBuilder - Running MicroarrayDataAnalyzer.generatePrietoCarlosGeneExpFile()...

Total gene pairs: 15841

Total time for loading: 1225

Size of geneNameToUnAcces: 37951

Total unmapped: 2442

Total UniProt pairs before filtering: 19205

After filtering: 19205

Before sequence consolidating: 19205

After sequence consolidating: 19204

After filtering: 19204

2012-04-17 14:49:26,458 [main] INFO org.reactome.fi.FINetworkBuilder - Check its odds ratio...

Co-expressed genes: 19204

Total checked pairs: 19204

Total: 140095

Mapped to ppi: 1477 (0.010543)

Mapped to random: 23 (0.000164)

Mapped to random: 28 (0.000200)

Mapped to random: 26 (0.000186)

Mapped to random: 25 (0.000178)

Mapped to random: 26 (0.000186)

Mapped to random: 28 (0.000200)

Mapped to random: 30 (0.000214)

Mapped to random: 22 (0.000157)

Mapped to random: 37 (0.000264)

Mapped to random: 31 (0.000221)

Average odds ratio: 55.20614231655218 +- 8.106345475156546 (from 10 tests)

2012-04-17 14:49:27,598 [main] INFO org.reactome.fi.FINetworkBuilder - Running PfamAnalyzer.convertIntToPfamIDs()...

Cannot be mapped: 83 10359

Cannot be mapped: 83 10367

Cannot be mapped: 1670 10194

Cannot be mapped: 8683 8683

Cannot be mapped: 10163 10163

Cannot be mapped: 10194 1670

Cannot be mapped: 10359 83

Cannot be mapped: 10359 10359

Cannot be mapped: 10359 10367

Cannot be mapped: 10367 83

Cannot be mapped: 10367 10359

Cannot be mapped: 10367 10367

2012-04-17 14:49:27,946 [main] INFO org.reactome.fi.FINetworkBuilder - Checking its odds ratio...

Total: 140095

Mapped to ppi: 26214 (0.187116)

Mapped to random: 2385 (0.017024)

Mapped to random: 2438 (0.017402)

Mapped to random: 2518 (0.017974)

Mapped to random: 2506 (0.017888)

Mapped to random: 2426 (0.017317)

Mapped to random: 2390 (0.017060)

Mapped to random: 2387 (0.017038)

Mapped to random: 2492 (0.017788)

Mapped to random: 2517 (0.017966)

Mapped to random: 2585 (0.018452)

Average odds ratio: 12.86455345326133 +- 0.3638005629354243 (from 10 tests)

2012-04-17 14:49:30,551 [main] INFO org.reactome.fi.FINetworkBuilder - Checking GO features...

GO BP:

Total: 140095

Mapped to ppi: 83248 (0.594225)

Mapped to random: 12389 (0.088433)

Mapped to random: 12002 (0.085670)

Mapped to random: 12287 (0.087705)

Mapped to random: 12243 (0.087391)

Mapped to random: 12353 (0.088176)

Mapped to random: 12202 (0.087098)

Mapped to random: 12196 (0.087055)

Mapped to random: 12298 (0.087783)

Mapped to random: 12131 (0.086591)

Mapped to random: 12258 (0.087498)

Average odds ratio: 15.303754902615841 +- 0.1542690878940338 (from 10 tests)

GO MF:

Total: 140095

Mapped to ppi: 70072 (0.500175)

Mapped to random: 29013 (0.207095)

Mapped to random: 29045 (0.207324)

Mapped to random: 29404 (0.209886)

Mapped to random: 28955 (0.206681)

Mapped to random: 28887 (0.206196)

Mapped to random: 29213 (0.208523)

Mapped to random: 29037 (0.207266)

Mapped to random: 29047 (0.207338)

Mapped to random: 28900 (0.206289)

Mapped to random: 28959 (0.206710)

Average odds ratio: 3.8260116856207893 +- 0.02579312511474479 (from 10 tests)

GO CP:

Total: 140095

Mapped to ppi: 96108 (0.686020)

Mapped to random: 48300 (0.344766)

Mapped to random: 48418 (0.345608)

Mapped to random: 48146 (0.343667)

Mapped to random: 48362 (0.345209)

Mapped to random: 48440 (0.345765)

Mapped to random: 48444 (0.345794)

Mapped to random: 48144 (0.343653)

Mapped to random: 48293 (0.344716)

Mapped to random: 48430 (0.345694)

Mapped to random: 48385 (0.345373)

Average odds ratio: 4.147761248626852 +- 0.014979309329591423 (from 10 tests)

Notes: 1). For gene expression, we use the SwissProt part of UniProt data only in order to map gene names to UniProt accessions. This is used to control the size of the final UniProt pairs. There are just too many mappings if we use STREMBL too: many of HLA proteins. Since our FI network focuses on the SwissProt proteins, this should be fine.

2). The odds ratio for GO BP is much higher than others. This is the reason why we have not chosen other two GO aspects. In 2018, we excluded GO annotations imported from the Reactome, therefore, the odds ratio has reduced from the 2017 version, which was expected.

3). The odds ratio for domain-domain interaction should be around 10.

4). It seems that yeast proteins that can be mapped to human are reducing gradually: 1272 (2012), 821 (2014), and 539 (2015), but back to 2145 in Year 2016. This may be caused by changes in compara?

5). In 2018, there is a mouse PPI, P01899-P01900, each of two mouse proteins in this PPI can be mapped to over 20,000 human protein Ids, resulting in an explosion of the human PPIs. Therefore, this PPI is excluded in the 2018 version. Need to consider to rewrite the use of PPIs without generating mapping. The same was done for P01901-P97484, which is mapped to 11,962,548 human PPIs. Also to control the mapping, only map size < 2,000 are used (e.g. a yeast protein that is mapped to more than 2,000 human proteins is not considered). In 2020, it was found that PPIs collected from BioGrid in iRefIndex caused an abnormal odds ratio for mapped worm PPIs (~ 120) much higher than other mapped PPIs and human PPIs. To control this abnormal feature, all BioGrid PPIs in iRefIndex are filtered out. Most likely Reactome PPIs were included in some PPIs collected BioGrid for worm.

6). To build the 2019 version of the FI network, the latest version of iRefIndex was 16.0, released in May, 2019. However, it seems that there is some error in the download file, especially corrupted file in 6239.mitab.05-29-2019.txt. Also many complexes were included in these files. Therefore, the older, 2018, version 15.0 was still used for the 2019 version of the FI network. The 2020 FI version uses the latest version of 17.0 without any issue. However, a large odds ratio increase for the mapped worm PPI is noted: from 39 +- 10 to 121 +- 31!

1. **trainNBC()**
   1. Before running trainNBC() (use –Xmx12G), make sure the following methods are correct in class org.reactome.weka.FeatureHandlerForV3:
      1. In method getFeatureList(), make sure all features to be used by NBC have been listed there.
      2. In method loadFatureToPairs(), make sure all PPIs features have point to correct file names. Usually this should be correct if you have set up correct configurations in the configuration.prop file.
   2. Set the following two properties in the configuration file:

# The following propertied are used during NBC training

ROC\_CURVE\_FILE=${RESULT\_DIR}/ROC\_100\_041712.txt

BP\_DOMAIN\_SHARED\_PAIRS=${RESULT\_DIR}/BP\_Domain\_Shared\_Pairs.txt

Output from this method should be similar as follows (edited, not all):

2012-04-17 16:15:57,479 [main] INFO org.reactome.fi.FINetworkBuilder - Running NBCAnalyzer.calculateNBCBasedOnReactome()...

Total pairs: 140095

After filtering: 92342

Prior probability: 0.0054197030050645596

Total negative: 14009500

test fis before filtering: 80521

Removing fis in the training data set: 51196

Total test fis: 19316

Cutoff False\_Positive\_Rate True\_Positive\_Rate

0.0 1.0 1.0

0.1 0.007751132592107361 0.23389935804514392

0.2 0.007594250208700323 0.2263926278732657

0.30000000000000004 0.007010356834695319 0.20066266307724168

0.4 0.001490382642366867 0.11539656243528681

0.5 0.001331941824687573 0.10669910954648995

0.6000000000000001 9.76618810613353E-4 0.08764754607579209

0.7000000000000001 4.77400365400889E-4 0.05161524125077656

0.8 4.722055844933712E-4 0.051149306274591015

0.9 3.942838708806037E-4 0.0479913025471112

1.0 0.0 0.0

One feature contribution:

pavlidisGeneExp: 0.006976872361487769

mousePPI: 0.014120356643917708

celPPI: 0.0073210054704194935

humanInteraction: 0.02392075252844076

pfamDomainInt: 0.0071525124710044765

carlosGeneExp: 0.01915376111547854

dmePPI: 0.010067552716758948

scePPI: 0.004875978648523508

goBPSharing: 0.024191346728450212

Learning probabilties:

carlosGeneExp\_false|false: 0.999721617473857

carlosGeneExp\_false|true: 0.9840051114335838

carlosGeneExp\_true|false: 2.7838252614297444E-4

carlosGeneExp\_true|true: 0.01599488856641615

celPPI\_false|false: 0.9997069131660659

celPPI\_false|true: 0.9935782200948647

celPPI\_true|false: 2.930868339341161E-4

celPPI\_true|true: 0.006421779905135258

dmePPI\_false|false: 0.9992931225240016

dmePPI\_false|true: 0.9789478243919343

dmePPI\_true|false: 7.068774759984297E-4

dmePPI\_true|true: 0.02105217560806567

goBPSharing\_false|false: 0.8900556765052285

goBPSharing\_false|true: 0.09848173095666113

goBPSharing\_true|false: 0.1099443234947714

goBPSharing\_true|true: 0.9015182690433389

humanInteraction\_false|false: 0.998610514293872

humanInteraction\_false|true: 0.9074960473024193

humanInteraction\_true|false: 0.0013894857061279846

humanInteraction\_true|true: 0.09250395269758073

mousePPI\_false|false: 0.9987644098647347

mousePPI\_false|true: 0.9496978622945139

mousePPI\_true|false: 0.0012355901352653557

mousePPI\_true|true: 0.050302137705486125

pavlidisGeneExp\_false|false: 0.9967881794496591

pavlidisGeneExp\_false|true: 0.9366160576985554

pavlidisGeneExp\_true|false: 0.00321182055034084

pavlidisGeneExp\_true|true: 0.06338394230144463

pfamDomainInt\_false|false: 0.9819248367179414

pfamDomainInt\_false|true: 0.7161205085443243

pfamDomainInt\_true|false: 0.018075163282058604

pfamDomainInt\_true|true: 0.28387949145567565

scePPI\_false|false: 0.9983287769013883

scePPI\_false|true: 0.9760672283467978

scePPI\_true|false: 0.0016712230986116563

scePPI\_true|true: 0.023932771653202227

……

2012-04-17 16:18:44,757 [main] INFO org.reactome.fi.FINetworkBuilder - Running NBCAnalyzer.checkSharedBPPairAndDomainPairs()...

Total GO ids: 29701

Time for looping: 237994

Total shared: 942953

2012-04-17 16:22:44,603 [main] INFO org.reactome.fi.FINetworkBuilder - Running checkCutoffValueForPredictedFIs()...

Total pairs: 3693723

Cutoff: 0.1

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 1307694 (27683)

SwissProt IDs: 12690

SwissProt Coverage: 0.6267285657842749

FIs merged: 1501595 (29794)

SwissProt IDs: 13440

SwissProt Coverage: 0.6637692611615962

Cutoff: 0.2

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 1260342 (26405)

SwissProt IDs: 12610

SwissProt Coverage: 0.6227775582773607

FIs merged: 1454243 (28588)

SwissProt IDs: 13376

SwissProt Coverage: 0.6606084551560648

Cutoff: 0.30000000000000004

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 1021057 (22947)

SwissProt IDs: 12480

SwissProt Coverage: 0.6163571710786251

FIs merged: 1214958 (25397)

SwissProt IDs: 13269

SwissProt Coverage: 0.655323982615567

Cutoff: 0.4

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 72977 (10893)

SwissProt IDs: 8456

SwissProt Coverage: 0.4176214934808376

FIs merged: 266878 (15276)

SwissProt IDs: 10897

SwissProt Coverage: 0.538176610035559

Cutoff: 0.5

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 67892 (10366)

SwissProt IDs: 8024

SwissProt Coverage: 0.3962860529435006

FIs merged: 261793 (14884)

SwissProt IDs: 10591

SwissProt Coverage: 0.523064006321612

Cutoff: 0.6000000000000001

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 51293 (9169)

SwissProt IDs: 7141

SwissProt Coverage: 0.3526768075859344

FIs merged: 245194 (14155)

SwissProt IDs: 10124

SwissProt Coverage: 0.5

Cutoff: 0.7000000000000001

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 37767 (6663)

SwissProt IDs: 4761

SwissProt Coverage: 0.23513433425523508

FIs merged: 231668 (12933)

SwissProt IDs: 9013

SwissProt Coverage: 0.4451303832477282

Cutoff: 0.8

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 34390 (6442)

SwissProt IDs: 4751

SwissProt Coverage: 0.2346404583168708

FIs merged: 228291 (12740)

SwissProt IDs: 9010

SwissProt Coverage: 0.4449822204662189

Cutoff: 0.9

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 25230 (5735)

SwissProt IDs: 4532

SwissProt Coverage: 0.223824575266693

FIs merged: 219131 (12157)

SwissProt IDs: 8885

SwissProt Coverage: 0.43880877123666534

Notes: 1). You should also run NBCGUITest.main() to study the results using different combinations of feature selections. Here is a screenshot from this application:



2). ROC curve and AUC should be plotted and calculated by using the generated file in the results directory, ROC\_100\_031512.txt. In order to do these, use a R script, ROCCurveDrawing.R, in the RSource folder. Change the value fileName to the above generated ROC point file. Run this R script in R. You should get the ROC curve similar to the following:



Based on the above results, calculate AUC using R as following. In order to call this function, the ROC package should be installed as described in this web page: <http://www.bioconductor.org/packages/release/bioc/html/ROC.html>.

> calculate.AUC(rocData)

[1] "AUC: 0.858892192875038"

3). Based on results from above, choose a reasonable cutoff value for high specificity. Usually cutoff value 0.50 should be chosen though you may get a pretty low true positive rate (recall rate).

1. **predictFIs()**
   1. Before running the method, set the cutoff value in the configuration file (usually it should be 0.50) and assign at least 12G for the method:

CUT\_OFF\_VALUE=0.50d

PREDICTED\_FI\_FILE=${RESULT\_DIR}/PredictedFIs\_031512.txt

Output may be like the following:

2012-04-17 16:37:28,524 [main] INFO org.reactome.fi.FINetworkBuilder - Running NBCAnalyzer.generatePredictedFIs()...

FIs from pathways: 193901 (9609)

SwissProt IDs: 7378

SwissProt Coverage: 0.3643816673251679

FIs from prediction: 67892 (10366)

SwissProt IDs: 8024

SwissProt Coverage: 0.3962860529435006

FIs merged: 261793 (14884)

SwissProt IDs: 10591

SwissProt Coverage: 0.523064006321612

1. **buildFIDb()**
   1. Before running the method, create an empty mysql database named as “FI\_yyyy” (yyyy should the year the FI network is building).
   2. Before running the method, make sure the following values in the hibernate configuration file, resources/funcIntHibernate.cfg.xml , are correct:

<property name=*"connection.url"*>jdbc:mysql://localhost:3306/FI\_2012</property>

<property name=*"connection.username"*>root</property>

<property name=*"connection.password"*>macmysql01</property>

* 1. Make sure –Xmx12G is set before running the method.

Output from the method running (logging information from hibernate has been removed):

2012-04-17 16:47:20,257 [main] INFO org.reactome.fi.FINetworkBuilder - Running FIDBBuilder.generateSchema()...

……

2012-04-17 16:47:22,447 [main] INFO org.hibernate.tool.hbm2ddl.SchemaExport - schema export complete

Interaction should be empty: 0

2012-04-17 16:47:22,548 [main] INFO org.reactome.fi.FINetworkBuilder - Running FIDBBuilder.dump()...

Total empty interactions: 0

Total empty interactions: 0

……

Total empty interactions: 505

……

Total empty interactions: 199

total interactions: 193901

Total time to extract: 130507

……

Total time for saving: 144991

2012-04-17 16:51:58,046 [main] INFO org.reactome.fi.FINetworkBuilder - Running FIDBBuilder.dumpPredicted()...

Total predicted FIs: 67892

Total protein from db: 9609

Total FIs will be added to the FI database: 67892

Save proteins: 5275

Save evidences: 67892

1. **generateCytoscapePlugInFiles()**
   1. Before running the method, the following properties should be set up in the configuration file:

# The minimum size used for a pathway

MINIMUM\_PAHTWAY\_SIZE = 3; // Change to 1 now for Pathway Enrichment Analysis to take care of the filtering.

GENE\_FI\_BIG\_COMP\_FILE\_NAME=${RESULT\_DIR}/FIsInGene\_121013\_BigComp.txt

GENE\_FI\_FILE\_NAME=${RESULT\_DIR}/FIsInGene\_121013.txt

GENE\_FI\_PATHWAY\_FILE\_NAME=${RESULT\_DIR}/FIsInGene\_Pathway\_121013.txt

GENE\_FI\_PREDICTED\_FILE\_NAME=${RESULT\_DIR}/FIsInGene\_Predicted\_121013.txt

# File for mapping accession to names dumped from the FI network

PROTEIN\_ACCESSION\_TO\_NAME\_FILE = ${RESULT\_DIR}/ProteinAccessionToName\_121013.txt

# We need a flattened list of pathways from Reactome for enrichment analysis

REACTOME\_PATHWAYS = ${RESULT\_DIR}/ReactomePathways121013.txt

# As of Decemember, 2013, we also want to dump all Reactome pathways in order

# to do a hierarhy based pathway enrichment analysis

PROTEIN\_ID\_TO\_REACTOME\_PATHWAYS = ${RESULT\_DIR}/ProteinIdToReactomePathways121013.txt

GENE\_TO\_REACTOME\_PATHWAYS = ${RESULT\_DIR}/ProteinNameToReactomePathways121013.txt

# Gene sets based on pathways

PROTEIN\_ID\_TO\_TOPIC = ${RESULT\_DIR}/ProteinIdToTopic121013.txt

GENE\_TO\_TOPIC = ${RESULT\_DIR}/ProteinNameToTopics121013.txt

REACTOME\_GMT\_FILE\_NAME = ${RESULT\_DIR}/ReactomePathways\_020720.gmt

Output from the method running should be similar to the following:

2012-04-17 17:33:41,044 [main] INFO org.reactome.fi.FINetworkBuilder - Running HibernateFIReader.generateFIFileInGeneInHiberante()...

……

Total interactions from prediction: 67892

Total interactions from pathways: 193901

Time for getting interactions: 5318

Total predicted FIs: 38450

Total pathway FIs:135432

Total FIs: 172235

Total predicted proteins: 8102

Total pathway proteins: 7402

Total proteins: 10696

2012-04-17 17:33:47,972 [main] INFO org.reactome.fi.FINetworkBuilder - Running FIGraphAnalyzer.analyzeComponents()...

Total interactions: 172235

Total components: 81

0: 10501

……

2012-04-17 17:33:49,860 [main] INFO org.reactome.fi.FINetworkBuilder - Running HiberanteFIReader.generateAccessionToProteinNameMap()...

2012-04-17 17:33:50,353 [main] INFO org.reactome.fi.FINetworkBuilder - Running ReactomeAnalyzer.generateListOfPathways()...

Total Pathways: 141

2012-04-17 17:34:12,681 [main] INFO org.reactome.fi.FINetworkBuilder - Running PathwayGeneSetGenerator.generateProteinNameToPathwayMap()...

……

* 1. This step has been disabled in 2020 since HotNet is not supported any more: The following two method calls at this step are related to create a matrix and heat kernel using R at the OICR cluster. The second method should not be run at the first time:

HotNetMatrixCalculator hotnetMatrixCalculator = **new** HotNetMatrixCalculator();

hotnetMatrixCalculator.testCalculateHeatKernel();

// The following method should be called after the kernel file was generated from using R

// hotnetMatrixCalculator.generateSerializedMatrixFile();

The generated matrix file, HotNet\_L\_matrix\_{year}.txt, should be zipped and moved to the cluster and processed by the script, runHeatKernel\_R.sh, in directory, ~/home/caBigR3. The script should be modified based on the actual file names, and submit using this: qsub –l h\_vmem=16G runHeatKernel\_R.sh. The results should be copied back and then ran the above method (second method).

* 1. The web application for ReactomeFIViz needs the annotated FI file, FIsInGene\_xxxx\_with\_annotations.txt too. This file is generated by using the method org.reactome.r3.fi.InteractionAnnotator.annotateAllFIs() in the FIVizWS\_corews project. There are some interactions may not be annotated, which may be mixed with known types and should be manually fixed.
     1. Make sure the InteractionTypeMapper.xml file is updated to the file having the same name in the FINetworkBuild project. This file should be copied into the webapp/WEB-INF folder.
     2. About 10 extracted FIs have no types because of gene name changes during sequence-based normalization (e.g. CALM1 nd CALM3 have the same sequence, causing ENCODE TargetedInteractions for CALM1 swapped). You may copy annotations from the previous year’s version or manually annotate them based on the original types in the database.
     3. This is a long process, taking about 2.5 hours.

1. **CompareFilesToPreviousVersion**:
   1. This method was added later on to check how similar the new version of the FINetwork, along with files generated for the web application, to previous one
   2. Output from this method run is something like this (partial results):

2018 2017

BP\_Domain\_Shared\_Pairs.txt 5600374 4252882

CGISurvivalAnalysis.R 5731 5731

Combined\_Logging.txt 2463140 39094755

FI\_2017.sql NA 37300517

FI\_2017\_Numbers.pptx NA 54591

FI\_2018.sql 54204498 NA

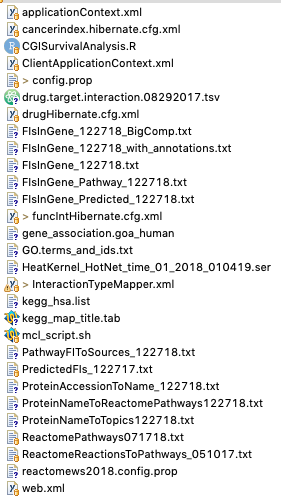
FI\_2018\_Numbers.pptx 54200 NA

FIsInGene\_071718.txt NA 2972316

……

1. **Set up Cytoscape Web Application and Plug-in**

Note: The ant script may need to be tested more and updated. The current way to generate war file is to use Eclipse: All needed files should be manually copied from the result folder to FIVizWS\_corews/WebContent/WEB-INF. All configurations should be edited manually. Here is a list of files needed by this web application:



The web application for the Reactome FI plug-in app is managed by another independent Java Web project, caBigR3WebApp. In this project, there is an ant build file, buil.xml, run this build.xml file to create a new war file and deploy this war file to the tomcat webapps folder in the deployment machine.

Notes: 1). These two values in the build.xml should be modified before running ant, version and buildDate.

2). There is a bug in the current version of the build code. The generated mysqldump file for the \_*plus*\_i database has corrupted the Ontology table. You may have to fix this table by re-install a correct version of Ontology table from gk\_central.

3). In order to view instances for newly added classes (Interaction and TargettedInteraction), the Ontology table needs to be modified by using a Perl script called updateDatabase.pl in the script folder. However, in order to use this script, a new Protégé project should be created by downloading a project file from gk\_central in the web site, adding a new attribute dataSource, and two new classes (Interaction and TargettedInteraction). Please follow the database model in the curator tool to make changes. (This is not needed any more since a simple view is used for instance).

4). For this web application and providing the download file, we need to add annotations to the FIsInGene\_XXX.txt file. To generate this file, call this method, annotateAllFIs() in class org.reactome.r3.fi.InteractionAnnotator, which is in the project caBigR3WebApp. Because these two genes, TRAPPC2 and TRAPPC2P1, have the same sequence, the following FI cannot be annotated, which is imported from ENCODE:

TRAPPC2P1 ZBTB33 unknown null 1.00

Based on the original data source, the above should be changed into the following manually in the file:

TRAPPC2P1 ZBTB33 expression regulated by <- 1.00

5). GSEA-based pathway enrichment analysis is provided by another WS service. Don’t forgot to copy the newly generated Reactome human gmt file (e.g. ReactomePathways\_122718.gmt) to that WS servlet and then restart it.

6). PathwyDiagramsFactorGraphs.xml.zip should be copied into the Cytoscape folder. All other files should be in the WEB-INF folder except two GMT files.

7). Don’t forget to enable file-based logging for all servlets when deploying the applications.

*======The END======*