The Inter-Tools User Manual

Hannah Catabia
hannahcatabia@gmail.com
June 15, 2017

Supplementary materials for:
Inter-Tools: A toolkit for interactome research
by Hannah Catabia, Caren Smith and José Ordovás

Contents

1 Introduction

Inter-Tools is a Python command line toolkit for buidling interactome databases, as well as creating GEXF files of gene sets within the interactome. It contains inter-build.py, a program that currates and combines protein-protein interaction (PPI) data from different MITAB files. It also contains inter-map.py, which contextualizes a user-input gene set on the interactome.

1.1 Publication

This user manual accompanies the publication **Inter-Tools: a toolkit for interactome research** by Hannah Catabia, Caren Smith, and Jose Ordovás. This paper was submitted to bioRXiv on June 15, 2017. Inter-Tools software is open-source under the MIT License. When using Inter-Tools as part of another project, please cite it.

CITATION

1.2 Intended usage

Inter-Tools is meant to be used as a resource for creating and visualizing interactome datasets. Of course, scientifially valid interactome analysis cannot be done exclusively visually; conclusive interactome analysis must be specifically designed for the question that the research is trying to answer. With this fact in mind, please note that Interact-Tools is geared toward interactome-based hypothesis generation and preliminary investigation. We hope that you find it useful!

2 Getting Started

2.1 Download

Inter-Tools is available for free download at: `www.githbub.com/catabia/inter-tools`

It is easiest to use Inter-Tools if all program and data files are kept in the same directory, such as a file folder on your desktop.

2.2 Using the Command Line

If you have never used the command line before, please take this opportunity to learn the basics. There are several excellent tutorials available online. Here's one from Django Girls [1]: `https://tutorial.djangogirls.org/en/intro_to_command_line/`

The command line inputs listed in this user guide will work as written only if all Inter-Tools files are in the same directory, and you have navigated to that directory in the Terminal.

3 System Requirements

Interact-Tools requires a version of Python 2.7.9 or higher to run. [2] Download the latest version of Python 2.7 at: `www.python.org/downloads`

The following Python libraries need to be installed: pandas [3], NetworkX [4], Matplotlib [5], Matplotlib_venn [6], NumPy [7], rpy2 [8], MyGene [9].

If these libraries are not currently installed on your machine, you may run inter-install.py through the command line in order to install them:

```
python inter-install.py
```

Alternatively, you may install these packages using pip or another package manager: `https://packaging.python.org/installing/`

Inter-Tools also requires R 3.3.3 to be installed on your computer. [10] Download R 3.3.3 at: `www.r-project.org` The inter-build.py script automatically installs the R library OntoCAT. [11]

4 inter-build.py

This program currates and combines protein-protein interaction (PPI) data from different MITAB files into a single database. It produces a species-specific interactome file for use with inter-map.py.

4.1 About the MITAB format

Many high-quality PPI datasets are available in MITAB format. MITAB format specifications are maintained by the Human Proteome Organization (HUPO) Proteomics Standards Initiative at `https://github.com/HUPO-PSI/`

miTab. [12] MITAB files are tab-delimited, and can be opened as spreadsheets in programs such as Microsoft Excel.

Many popular interactome datasets available for download in MITAB format include:

1. BioGRID [13]: https://thebiogrid.org/
2. IntAct [14]: http://www.ebi.ac.uk/intact/
3. DIP [15]: http://dip.doe-mbi.ucla.edu/dip/Main.cgi

There are many other high-quality PPI database sources available.

4.2 Required Inputs

There is only one required command-line input for inter-build.py: a listing of MITAB files to be currated and combined.

To run the program, first save the MITAB files you would like to combine in the same directory as inter-build.py. Next, open terminal, navigate to the directory containing your files, and type the following:

```
python inter-build.py -m mitab1.txt mitab2.txt mitab3.txt
```

(Here, mitab1.txt, mitab2.txt, and mitab3.txt are example file names. Replace them with the names of the MITAB files you would like to combine.) The program will accept as many MITAB files as you would like, but there must be at least one. All MITAB files should be written after the command -m and be separated by spaces.

4.3 Optional Inputs

There are several optional inputs that may be used to further customize the final interactome dataset.

4.3.1 Taxonomy IDs

Users may enter NCBI Taxonomy IDs to restrict the interactome to a particular species or species group. A list of NCBI Taxonomy IDs may be found at: https://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/. [16] Here's a command-line example for creating a dataset with interactions found in zebrafish (7955), mice (10090), and humans (9606):

```
python inter-build.py -m mitab1.txt -t 7955 10090 9606
```

Inputs should be written in integer form after the command -t and be separated by spaces. If no taxonomy identifier is specified by the user, 9606 (*homo sapiens*) is used as a default.

4.3.2 Interaction Detection Methods

Not all methods for discovering PPIs are equivalently reliable. Indeed, some databases contain interactions that have been inferred, but have not actually been experimentally verified in a lab. For this reason, inter-build.py has the ability to include only interactions that have been verfied by the user's prefered interaction detection method. Users may enter the code for the interaction detection method from PSI-MI ontology [12], which may be browsed here:

`https://www.ebi.ac.uk/ols/ontologies/mi/terms?iri=http://purl.obolibrary.org/obo/MI_0001`.

Interactions found using this method, as well all children of this method, will be included in the final dataset. Here is a commnd-line example for creating a dataset containing interactions detected by two-hybrid (MI-0018) studies:

```
python inter-build.py -m mitab1.txt mitab2.txt -i 0018
```

The input should be written in integer form after the command -i. The program can only accept one interaction detection method at a time. If no taxonomy identifier is specified by the user, 0045 (*experimental detection method*) is used as a default. PSI-MI ontologies are accessed with ontoCAT via R.

4.3.3 Diameter

The diameter of a network is defined as the longest distance in the set of all shortest distances between all pairs of nodes in the network. Knowing the diameter of an interactome network can be useful for judging what types of analysis to perform. However, calculating it takes time, and is not done by default. To switch it on, type -d into the command line:

```
python inter-build.py -m mitab1.txt mitab2.txt -t 10090 -d
```

The diameter will be reported in the BUILD-RESULTS.txt file assembled by inter-build.py. Please expect this calculation to take up to one hour long, depending on the size of the interaction network.

4.3.4 Format

The program automatically saves the interactome dataset as a CSV file. However, if the user would prefer having a TSV file, they may indicate this in the command line after -f:

```
python inter-build.py -m mitab1.txt mitab2.txt -t 10090 -f TSV
```

4.4 Outputs

This program produces four separate files:

4.4.1 Interactome dataset

A file called interactome_DATE-TIME.csv, which contains the curated, compiled interactome dataset. It may be viewed as a spreadsheet, and it will contain the columns for: Entez ID for Interactor A, Entrez ID for Interactor B, Uniprot A, Uniprot B, Gene Symbol A, Gene Symbol B, Interaction Detection Method, Publication Identifiers, Taxonomy ID for Interactor A, Taxonomy ID for Interactor B, Interaction Type, Source Database, From Files.

4.4.2 Results text file

BUILD-RESULTS.txt contains some basic information about the interactome dataset, such as the number of unique genes and interactions, the size of the largest connected component (LCC), and the median number of interactions per gene. If the diameter is calculated, it will be listed here.

4.4.3 Venn diagram

A Venn diagram representing the interactions sources from the three largest MITAB files entered into the program.

4.4.4 Histogram

A histogram representing the number of interactions per gene in log scale (base 10).

5 inter-map.py

This program maps a user-input set of genes onto an interactome network. It produces two GEXF graph files and a results text file.

5.1 Required Inputs

This program requires three inputs:

5.1.1 Interactome

This program requires an interactome dataset as a CSV or TSV file in which each row represents a particular PPI. Ideally, this dataset should be created with inter-build.py, but it is possible to use other interactome datasets, such as BioPlex 2.0. [21] However, any such dataset must be reformatted to have the following *exact* 13 column headers:

1. **entrez_A**
2. **entrez_B**
3. **uniprot_A**
4. **uniprot_B**
5. **symbol_A**
6. **symbol_B**
7. interaction_detection_methods
8. publication_identifiers
9. taxid_A
10. taxid_B

11. interaction_type
12. source_database
13. from_files

The bold column headers must have entries for each PPI in order for inter-map.py to funtion properly; the non-bold headers may be left blank without affecting the program's ability to run. Even if a column is left completely blank, it must still nevertheless have the appropriate column header or inter-map.py will reject it.

Here is a command-line example of how to input the interactome file name in to inter-map.py after typing -i:

```
python inter-map.py -i your_interactome_file.csv -g genes.tsv -u
```

Please note that if you do not input an interactome file, the program will automatically look for and use a file named 'interactome.csv' or 'interactome.tsv' in the program's directory.

5.1.2 Gene set

This is a CSV or TSV file with a set of genes that the user wishes to investigate. The gene list should occupy the first column of the file and be identified by either Entrez IDs or Uniprot KB. Here is a command line usage example:

```
python inter-map.py -i interactome.tsv -g your_gene_set.csv -e
```

5.1.3 Entrez/Uniprot

The user may decide whether to identify genes in the set with either Entrez IDs or UniProt KB. However, the user must also tell the program what kind of ID to expect. Enter -e into the command line if using Entrez IDs, and -u if using UniProt KB. Here is a command-line example for gene sets identified with Entrez IDs:

```
python inter-map.py -i interactome.csv -g gene_set.tsv -e
```

And an example for UniProt KB:

```
python inter-map.py -i interactome.csv -g gene_set.tsv -u
```

5.2 Optional Inputs

There is only one optional input:

5.2.1 Self-interactions

Typing -s into the command line keeps PPIs in which a gene interacts with itself in the interactome dataset. Otherwise, they are eliminated by default. Here is a command-line example:

```
python inter-map.py -i interactome.csv -g gene_set.tsv -u -s
```

5.3 Outputs

This program produces three separate files. Two are GEXF files, which may be opened, viewed, an analyzed with graph visualization software such as Cytoscape or Gephi. In these files, genes are represented as nodes and have the following attributes:

1. Entrez ID
2. UniProt KB
3. Symbol
4. Taxonomy ID
5. gene_set
6. Interactome Degree (the number of interactions that the gene has with other genes in the interactome)
7. Gene Set Degree (the number of direct interactions that the gene has with other members of the gene set)
8. Weight

The weight attribute is the ratio of the interactome degree with the gene set degree. It reveals the proportion of a gene's interactions that occur with other members of the gene set. It may be used to highlight genes that are particularly active in the gene set network as opposed to generally promiscuous in the interactome. Genes not in the gene set have a weight of 0.

Interactions are represented as edges with the following attributes:

1. Interaction detection method
2. Publication identifiers
3. Interaction type
4. Source database
5. From files

5.3.1 Gene set network

The file gene_set_network.gexf contains only the genes listed in the gene set that were found in the interactome, as well as the interactions between them. It is useful for seeing how members of the gene set interact directly with each other.

5.3.2 Full interactome network

The file full_interactiome_network.gexf contains all of the genes in the interactome, with the nodes representing the gene set having the attribute 'gene_set = True'. Depending on the size of the interactome, this graph is most likely too busy to parse visually. However, it may be useful for conducting graph analyses within Gephi or Cytoscape.

5.3.3 MAP-RESULTS.txt

BUILD-RESULTS.txt contains some basic information about the portion of the gene set found within the interactome, such the gene set largest connected component (LCC) and mean shortest distance (MSD). In this instance, we define the mean shortest distance the same way that Menche, *et. al.* does: as the average of the shortest distance from each gene in the gene set to any other gene in the gene set. [22]

6 Illustrative Example

6.1 Human Interactome Dataset

We used inter-build.py to assemble an example human interactome dataset. On April 24, 2017, we downloaded the most current available MITAB files

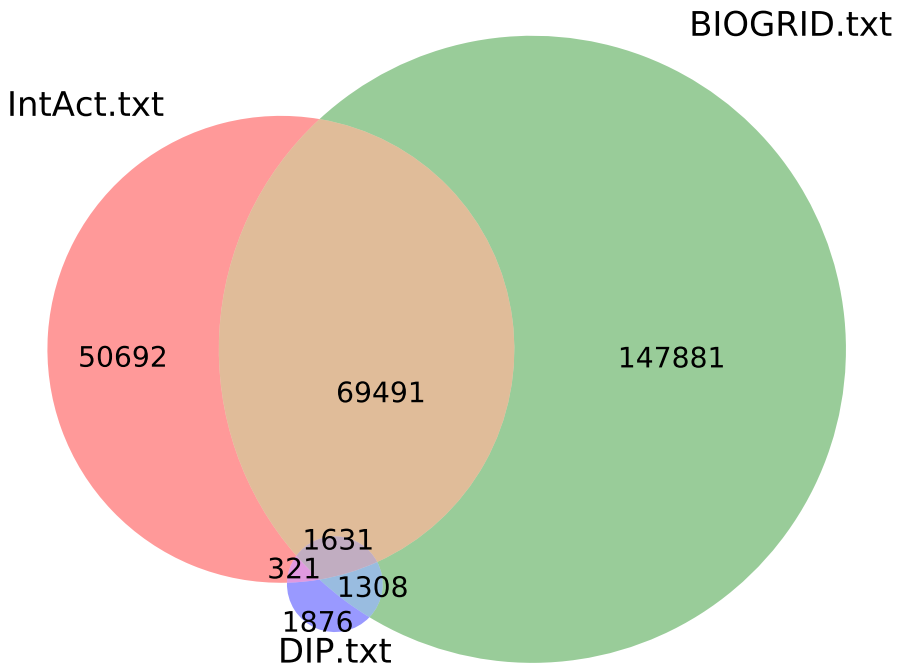
from three PPI repositories:

Database	No. Interactions	Gene ID	Species	Last Updated
BioGRID [13]	1,381,963	Entrez ID	All	25 Mar 2017
IntAct [14]	720,712	UniProt KB	All	9 Apr 2017
DIP [15]	7,794	UniProt KB	Human Only	5 Feb 2017

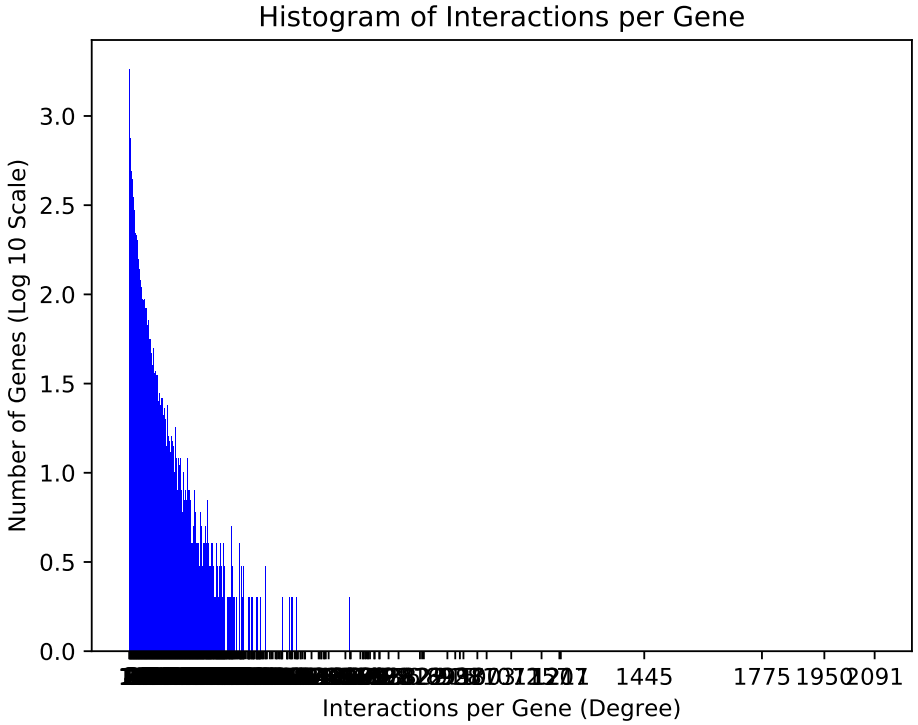
We chose to utilize the default taxonomy ID (*homo sapiens*: 9606) and interaction detection method (experimental detection method: 0045). Our command line input was:

```
python inter-build.py -g BIOGRID.txt IntAct.tx DIP.txt -e
```

The resulting interactome database had 16,972 unique genes and 173,200 unique interactions, of which 3,115 (1.1%) were self-interactions. The sources of these iteractions are shown in the Venn diagram produced by inter-build.py:



This human interactome dataset has a diameter of 8. The median number of interactions per gene is 13, but the most promiscuous gene in the network has 2,091 interactions. Here is the histogram (in log-10 scale) of the number of interactions per gene produced by inter-build.py:



You may download this example human interactome database at `www.githbub.com/catabia/inter-tools`. It is titled 'human-interactome.csv', and information about it is stored in 'BUILD-RESULTS-human-interactome.txt'.

6.2 Global Lipids Consortium Gene Set

Next, we used inter-map.py to explore a set of 157 genes identified by the Global Lipids Consortium to be associated with high density lipoprotein cholesterol (HDL), low density lipoprotein cholesterol (LDL), triglycerides, and total cholesterol concentration in humans. [23] We used MyGene [9] to translate the gene symbol names listed in the publication into Entrez IDs. (The CSV file with the list of these Entrez IDs is available at `github.com/catabia/inter-tools` as the file example-gene-set.csv.) Utilizing the

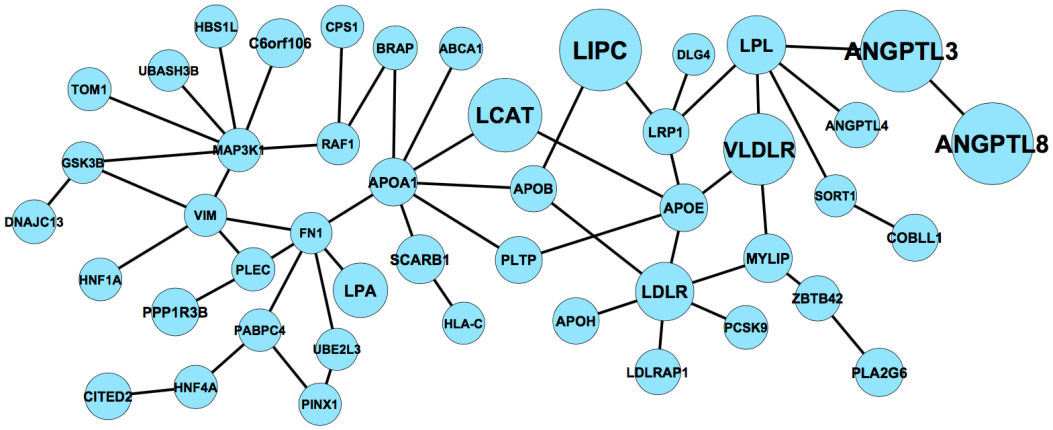
human interactome we built with inter-build.py, we ran inter-map.py with the following command line input:

```
python intermap.py -i human-interactome.csv -g example_gene_set.
csv -e
```

145 (92.4%) of the 157 genes listed in the paper were found in the interactome. Of those found, 54 (34.1%) were connected to at least one other gene in the set, and 46 (31.7%) formed the largest connected component. The mean shortest distance of the gene set was 1.63. This instance of inter-map.py created one text file and two GEXF files, all of which are available for download at `github.com/catabia/inter-tools` as:

1. MAP-RESULTS-example-gene-set.txt
2. example-full-network.gexf
3. example-gene-set-network.gexf

The following visualization of the largest connected component of the gene set was created with example-gene-set-network.gexf and Gephi [20]. Each node is sized proportionally to its weight, which is the ratio of its degree in the gene set to its degree in the interactome.



7 Acknowledgements

Many thanks to the Nutrition and Genomics Laboratory at the Jean Mayer USDA Human Nutrition Research Center on Aging of Tufts Univeristy. In

particular, a warm thank you to José Ordovás, Caren Smith, Chao-Qiang Lai, Laurence Parnell, and Martin Obin for your support and guidance.

8 Contact

Please contact Hannah at hannahcatabia@gmail.com with any questions, comments, or bug reports.

References

[1] "Introduction to the command-line interface." *Django Girls*. N.p., n.d. Web. 06 June 2017. `https://tutorial.djangogirls.org/en/intro_to_command_line/`

[2] Python Software Foundation. *Python Language Reference*. version 2.7. Available at http://www.python.org

[3] McKinney W. Data Structures for Statistical Computing in Python, *Proceedings of the 9th Python in Science Conference*, 51-56 (2010)

[4] Hagberg A, Schult D, Swart P. Exploring network structure, dynamics, and function using NetworkX. *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Pasadena, CA. p. 11-15, Aug 2008.

[5] Hunter J. Matplotlib: A 2D Graphics Environment. *Computing in Science and Engineering*, 9, 90-95 (2007), DOI:10.1109/MCSE.2007.55

[6] Tretyakov K. Matplotlib-Venn. Web. 06 June 2017. `JohnD.Hunter.Matplotlib:A2DGraphicsEnvironment,ComputinginScience&Engineering,9,90-95(2007),DOI:10.1109/MCSE.2007.55`

[7] Van der Walt S, Colbert S, Varoquaux G. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science and Engineering*, 13, 22-30 (2011), DOI:10.1109/MCSE.2011.37

[8] Gautier L. RPy2. Web. 06 June 2017. `https://pypi.python.org/pypi/rpy2/2.8.6`

[9] Xin J, Mark A, Afrasiabi C, et al. High-performance web services for querying gene and variant annotation. *Genome Biol.* 2016;17(1):91.

[10] R Core Team (2017). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, Vienna, Austria. URL https://www.R-project.org/.

[11] Adamusiak T, Burdett T, Kurbatova N, Van der Velde K, Abeygunawardena N, Antonakaki D, Kapushesky M, Parkinson H, Swertz M. OntoCAT–simple ontology search and integration in Java, R and REST/JavaScript. *BMC Bioinformatics.* 2011;12:218. doi:10.1186/1471-2105-12-218.

[12] Kerrien S, Orchard S, Montecchi-Palazzi L, et al. Broadening the horizon: level 2.5 of the HUPO-PSI format for molecular interactions. *BMC Biology.* 2007;5:44. DOI:10.1186/1741-7007-5-44.

[13] Chatr-aryamontri A, Oughtred R, Boucher L, et al. The BioGRID interaction database: 2017 update. *Nucleic Acids Res.* 2017;45(D1):D369-D379.

[14] Orchard S, Ammari M, Aranda B, et al. The MIntAct project–IntAct as a common curation platform for 11 molecular interaction databases. *Nucleic Acids Res.* 2014;42(Database issue):D358-63.

[15] Salwinski L, Miller CS, Smith AJ, Pettit FK, Bowie JU, Eisenberg D. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.* 2004;32(Database issue):D449-51.

[16] Federhen S. The NCBI Taxonomy database. *Nucleic Acids Research.* 2012;40(Database issue):D136-D143. doi:10.1093/nar/gkr1178.

[17] Maglott D, Ostell J, Pruitt KD, Tatusova T. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Res.* 2005;33(Database issue):D54-8.

[18] Boutet E, Lieberherr D, Tognolli M, et al. UniProtKB/Swiss-Prot, the Manually Annotated Section of the UniProt KnowledgeBase: How to Use the Entry View. *Methods Mol Biol.* 2016;1374:23-54.

[19] Cline MS, Smoot M, Cerami E, et al. Integration of biological networks and gene expression data using Cytoscape. *Nat Protoc.* 2007;2(10):2366-82.

[20] Bastian M., Heymann S., Jacomy M. (2009). Gephi: an open source software for exploring and manipulating networks. *International AAAI Conference on Weblogs and Social Media.*

[21] Huttlin EL, Bruckner RJ, Paulo JA, et al. Architecture of the human interactome defines protein communities and disease networks. *Nature.* 2017;545(7655):505-509.

[22] Menche J, Sharma A, Kitsak M, et al. Disease networks. Uncovering disease-disease relationships through the incomplete interactome. *Science.* 2015;347(6224):1257601.

[23] Willer CJ, Schmidt EM, Sengupta S, et al. Discovery and refinement of loci associated with lipid levels. *Nat Genet.* 2013;45(11):1274-83.