Comparative Genomics Practical 8—Interaction Networks

Group 6 Tianlin He Xueqing Wang

SUMMARY

Interaction network is an important subject in system biology; while gene specific functional database can provide exceptional depth and coverage of available data for given genes, interaction networks can effectively explore the biological relationship associated with hundreds or thousands of genes in parallel. [1] In this practical we extracted the interactomes for given species, explored the network topology of them by looking into distributions, compared networks of differentially expressed genesets in Saccharomyces cerevisiae chromosome by the help of advanced online tools, as well as performed pathway enrichment analysis for experimentally defined gene sets. We further explored a little into the different performance of several online tools in terms of their principles.

ACTIVITIES

Comparative network analysis using STRING

1. Extract the networks for all your bacterial and eukaryotic genomes from the STRING (protein.links.v10.txt.gz).

Tip1: The file is very large don't decompress it but use python gzip.open() to extract the links for your species. Proteins in the links file are prefixed with the NCBI taxonomy ID of the species.

Tip2: Python ETE3 package contains NCBITaxa class which can be used to translate NCBI taxa identifiers to species names.

We first used the module gzip.open() to write all the interactions out in one file. After that we extracted the NCBI taxonomy ID of each genome from STRING and searched against the overall interaction file to extract all the interactions within these five species by looking for the taxa ID. The interactome of each species is written into five files individually. The code for this part is included in AnnexI.

2. Write a Python script to calculate the average connectivity (nr of links/nr of proteins) for each interactome.

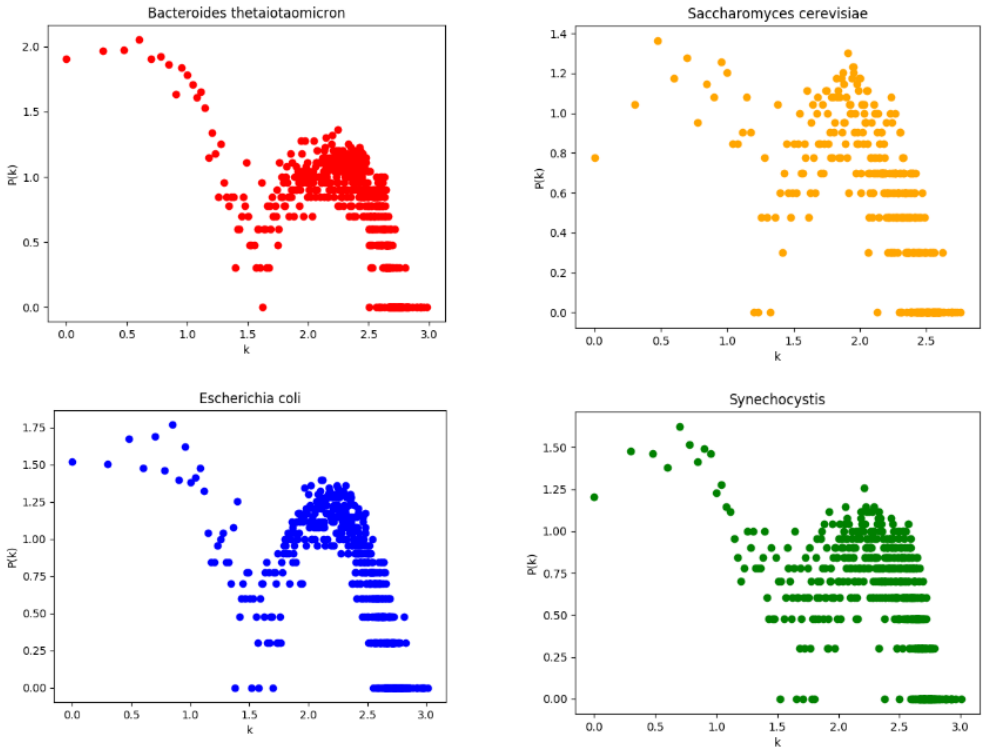
Each row in our interactome file corresponds to an interaction between two proteins in that species; however, as a protein is both recorded in the first and second colomn, every interaction is counted twice. Thus, the number of links should be half of the number of rows. We also counted the number of unique proteins showing up in each interactome. The code is attached in AnnexII. The results are as follows:

Average connectivity in Bacteroides thetaiotaomicron (species 03)= 88.948701709943
Average connectivity in Dictyoglomus turgidum (species 08)= 58.960986804360296
Average connectivity in Escherichia coli (species 09)= 96.46691336525903
Average connectivity in Synechocystis (species 18)= 108.0041022404544
Average connectivity in Saccharomyces cerevisiae (species 30)= 146.46276098473044

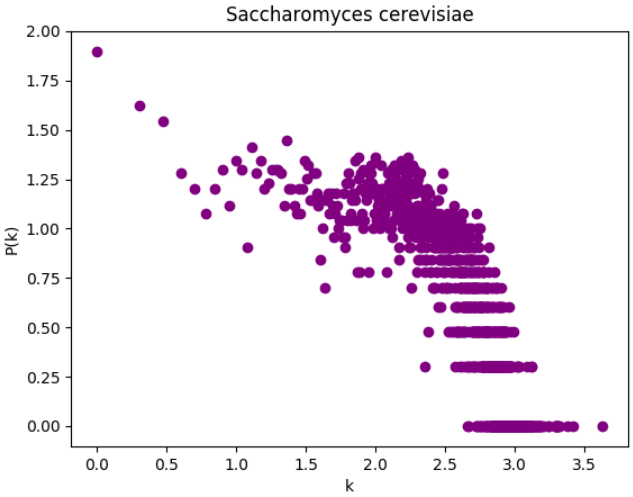
3. Plot the degree distribution as a log-log scale scatter plot for each interactome. On one axis of the plot should be the node degree the other axis the frequency. Do you observe a power-law distribution?

The code for this part is also included in AnnexII(together with the answer for the upper question). Here the log of frequency (probability)that a randomly selected node has exactly k edges is plotted against the node degree k of different nodes.

The log-log scatter for all the prokaryotes in our group are shown as follows:



The log-log scatter for yeast is shown as follows:



From the scatters we can roughly tell that, the distribution of node degrees for the four prokaryotes somehow follows the power-law (the overwhelming majority of nodes each hold low degrees; a few hubs possess an extraordinary number of neighbours), with the exception in the area where k has a medium value: the frequency of nodes drops dramatically at these parts. A heavy tail can be inspected on each of these plots. As the power-law refers to that the network is scale-free or hierarchical type, this phenomenon might be interpreted as that the networks within these genomes show somewhat scale-free characteristics, but not totally.

The scatter of yeast also demonstrates a heavy tail, with the drop in the middle part still existing but not as obvious as those in the prokaryotes. Although not perfectly fit with power-law, it is more close to it,

which might be interpreted as that the interactome of yeast is more similar to scale-free style than those in the prokaryotes.

Experimental gene sets

Imagine an experiment has been performed and e.g. differentially expressed genes have been detected. Obviously, these would be a subset of your yeast chromosome.

1. Download the S.cerevisiae S228C proteome from Uniprot (http://www.uniprot.org/proteomes/) and use BLAST to match your predicted genes against UniProt proteome.

1) We first downloaded the chromosome X from S.cerevisiae S228C proteomes from Uniprot.
2) In the same directory, make a protein blast database using the downloaded proteomes by typing the below command:

```
makeblastdb -in YeastChrX.fa -dbtype prot
```

3) Blast the query multi-fasta file generated from GENESCAN in practical 2 against the database in 2)

```
blastp -outfmt 5 -query  protein30.fa -db YeastChrX.fa -out out.yeast.blastp.txt
```

2. Parse blast results to extract gene names (GN=) for genes present on chromosome.

The python blastResultParser.py was modified such that it displays the best-hit in YeastChrX.fa. An output file “out.yeast.blastp.txt” was obtained.

```
python2 blastResultParser.py out.yeast.blastp.txt
```

The modified script is in AnnexIII.

3. Find two experimental gene sets (experiments.txt) that overlap most with genes on your chromosome. The goal is to find out if any of these are present on the chromosome.

The python script readID.py(included as AnnexIV) that extracts the overlap between experiments.txt and out.yeast.bastp.txt is in attachment no.2. It reads the two files and outputs a dictionary containing line as key and number of overlapping proteins as value, such as :

```
Line40: 0
```

Moreover, it sorts the dictionary so that keys are ordered according to ascending number of overlapping proteins.

4. Report how many genes overlap and save two experimental gene sets for further analysis.

The output from readID.py is as below:

```
[('line40', 0), ('line34', 0), ('line17', 0), ('line39', 0), ('line51', 0), ('line5', 0), ('line4', 0), ('line19', 0),
('line55', 0), ('line16', 0), ('line30', 0), ('line1', 1), ('line22', 1), ('line53', 1), ('line43', 1), ('line8', 1),
('line42', 1), ('line31', 1), ('line49', 1), ('line20', 1), ('line28', 1), ('line7', 1), ('line6', 1), ('line50', 1),
('line33', 1), ('line52', 1), ('line25', 1), ('line2', 1), ('line27', 1), ('line38', 1), ('line12', 1), ('line23', 1),
('line37', 2), ('line24', 2), ('line48', 2), ('line47', 2), ('line45', 2), ('line44', 2), ('line11', 2), ('line21', 2),
('line3', 2), ('line10', 2), ('line14', 2), ('line41', 2), ('line36', 2), ('line46', 2), ('line18', 2), ('line35', 2),
('line15', 2), ('line29', 2), ('line13', 2), ('line9', 2), ('line32', 2), ('line26', 3), ('line54', 4)]
```

Therefore, line 54 and line 26 in experiments.txt should be selected for further use, as they have the most overlaps (4 proteins and 3 proteins respectively) with the out.yeast.blastp.txt.

Comparative network analysis using FunCoup and STRING

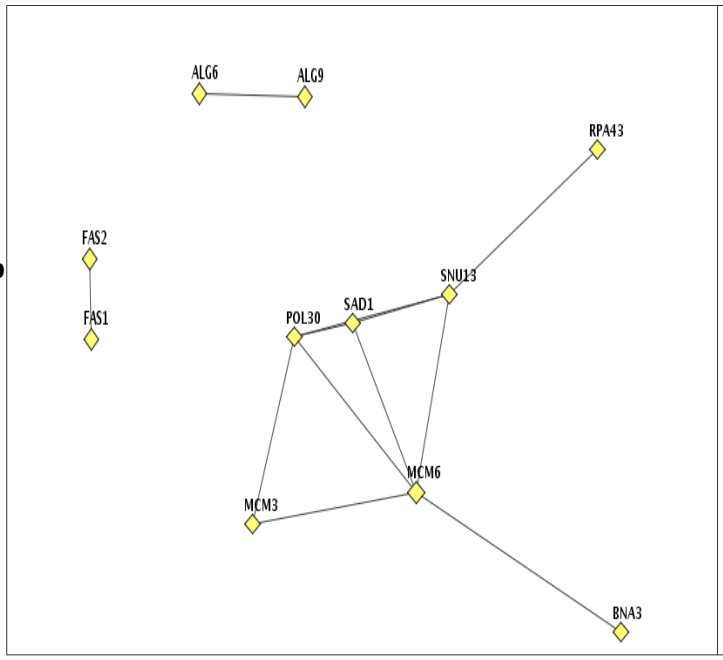
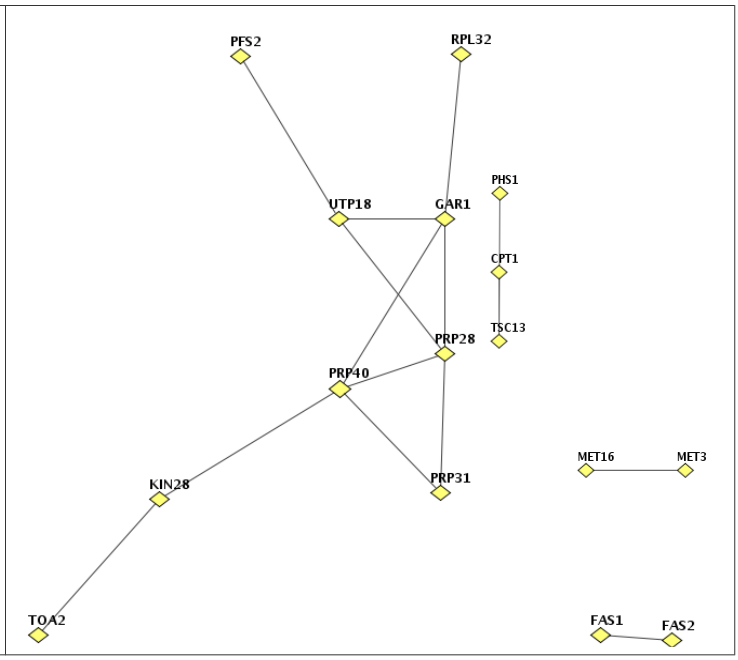
1. Using your two experimental gene sets query FunCoup and STRING for sub-networks containing these genes.
a. FunCoup works with space delimited list of gene names, STRING expects each gene name in new line for a query.

b. Use the same expansion depth or max number of interactors for searching both databases, so the results are comparable.
In FunCoup, we set the expansion depth as 0, while in STRING, we set the max number of interactions as “none/query proteins only”, so that this parameter is comparable in the two methods.

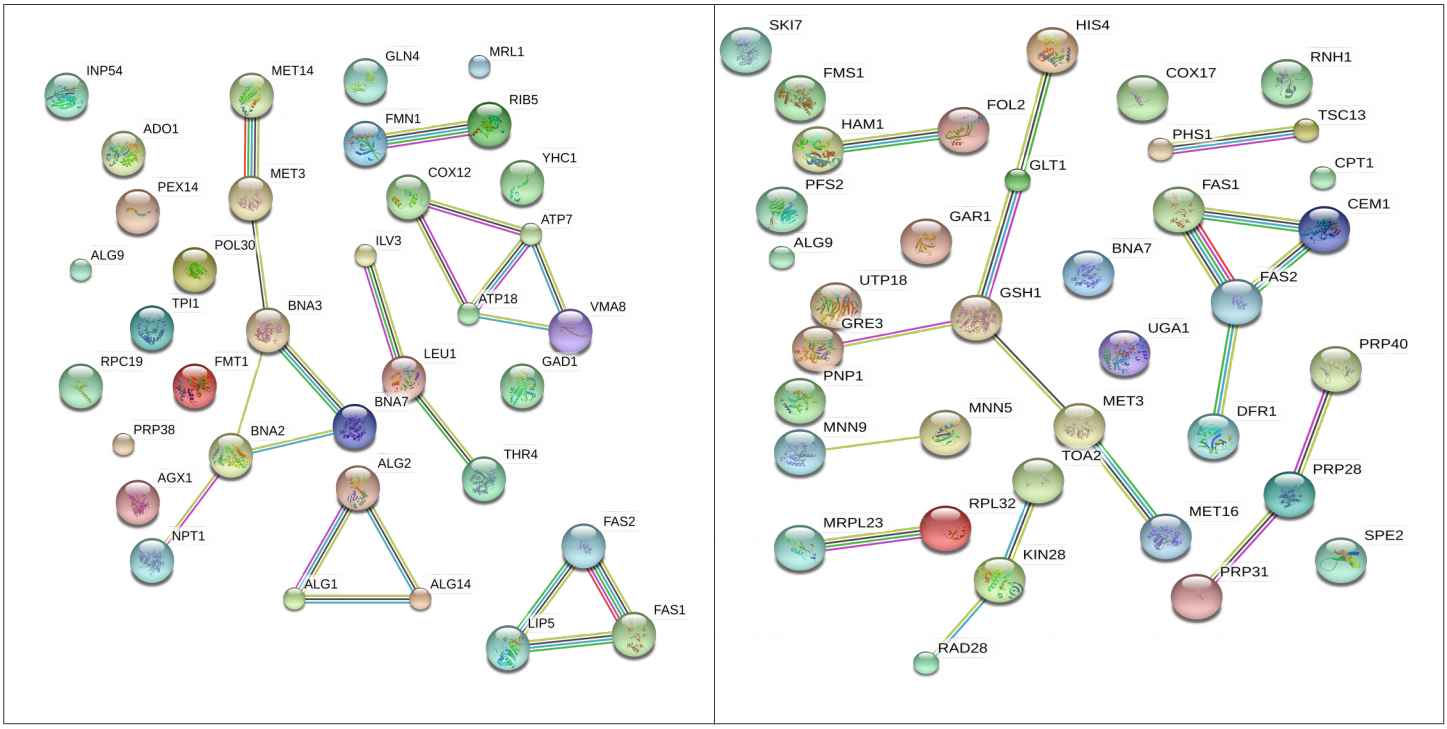
2. Compare results

a. How do these networks differ in terms of nodes, links, and hubs (the three nodes with highest degree)?

- For set no. 26, the agreement between Funcoup and STRING is not as obvious as one would expect. Their only shared feature is the interaction between FAS1 and FAS2.
- For set no. 54, they have the below shared features:
  - Nodes: MET3, MET16, PRP28, PRP31, PRP40, KIN28, TOA2, PHS1, TSC13
  - Links: MET3-MET16, PRP40-PRP31-PRP28, KIN28-TOA2
  - Hubs: by Funcoup (PRP28, PRP40, GAR1), by STRING (GSH1, FAS1, PRP28)

	Set no. 26	**Set no. 54**
Funcoup		

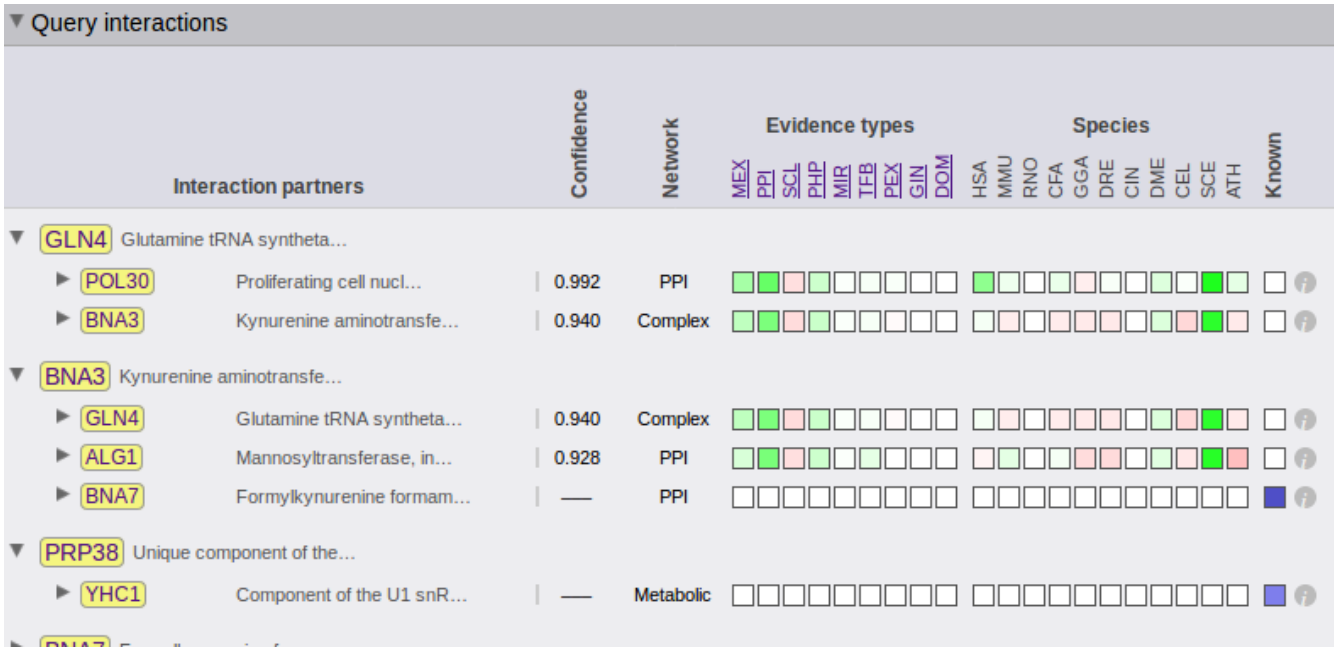
STRING



b. What is the most common evidence type with high confidence (>0.9)?

	Funcoup	STRING
set26	Protein-protein interaction	textmining
set54	mRNA co-expression	textmining

Take set 16 by Funcoup as an example, It is shown that PPI (protein-protein interaction) is the most common evidence type, as it is bright green in color, which symbolizes the highest supporting score.



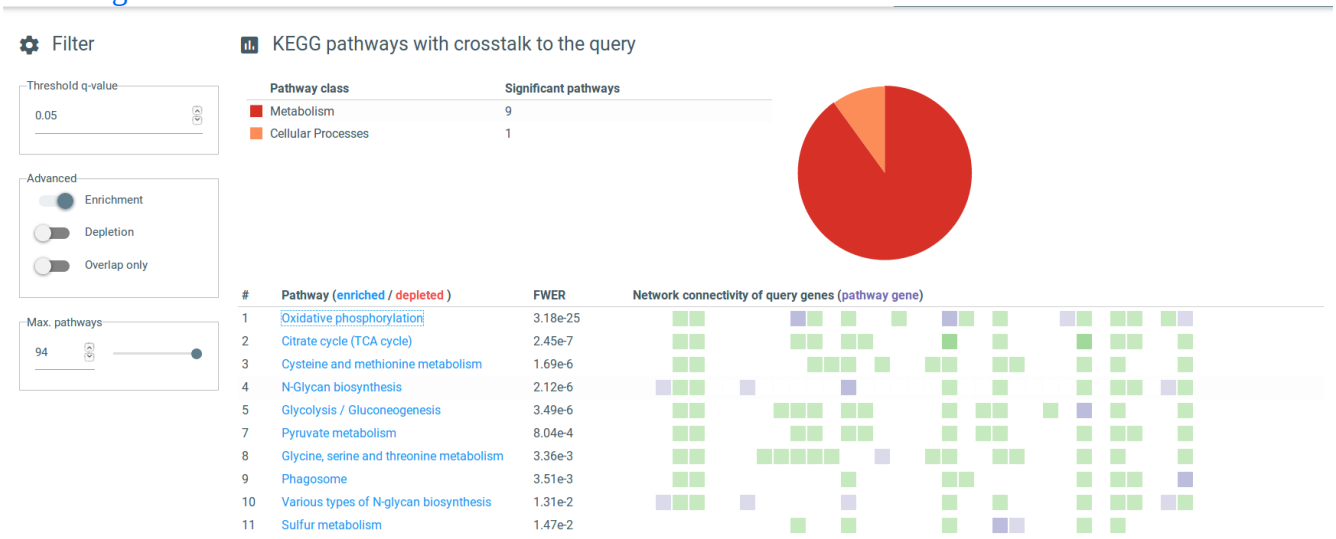
c. Can you explain the differences in terms of underlying data sources in the databases?

- Funcoup:
  - Its data sources are mainly experimental, e.g. physical protein-protein interactions, mRNA/protein co-expression, co-regulation, which should be the most reliable
  - Or can come from literature, e.g. genetic profile, subcellular localization, shared transcription factor
  - Bioinformatics: domain prediction or phylogenic relationship
- STRING:
  - Experimental: co-expression or high-throughput experiment
  - Bioinformatics: textmining, genomic context prediction
  - From other primary databases [2]

As a consequence, Funcoup should be more accurate, as it is heavily based on multiple types of experimental results, such as mRNA co-expression, protein co-expression, miRNA co-regulation....which are more reliable. It should be more sensitive than STRING as well, because it also takes into account the evolutionary relationship between proteins (i.e. using InParanoid as reference).[3]
This can be reflected by the fact that in question 2b, the most common evidence type found by STRING is textmining, which is by no means comparable to Funcoup.

Enrichment analysis using PathwAX and DAVID

1. Analyze the same two experimental sets for enrichment of pathways via PathwAX and DAVID, using KEGG.
■ Which pathways are enriched?
■ Disparity between results from PathwAX and DAVID?
■ Does the number of (input) genes matter for results?
■ If so explain why.
We selected all the gene names from set 26 and set 52, respectively, and searched on PathwAX and DAVID websites to enrich the pathways containing as many of these gene IDs. The gene IDs provided here is official gene symbol. The result interfaces of these two online databases are exemplified in the following screen shots:



The Pathways Enriched for Set 26 by PathwAX

Sublist	Category	Term	RT	Genes	Count	%	P-Value	Benjamini
☐	KEGG_PATHWAY	Fatty acid biosynthesis	RT		2	5.9	7.6E-2	9.2E-1

32 gene(s) from your list are not in the output.

The Pathways Enriched for Set 26 by DAVID

The pathways collected for each sets and a fraction of them is shown as follows:

	Set 26 (total)	Set 26 (Partial)	Set 52 (Total)	Set 52 (Partial)
PathwAX	1. Oxidative phosphorylation 2. Citrate cycle(TCA cycle) 3. Cysteine and methionine metabolism 4. N-Glycan biosynthesis 5.Glycolysis/Gluconeogenesis 6. Pyruvate metabolism 7. Glysine, serine and threonine metabolism 8. Phagosome 9. Various types of N-glycan biosynthesis 10. Sulfur metabolism	1. Oxidative phosphorylation 2. N-Glycan biosynthesis 3. Cysteine and methionine metabolism 4. Citrate cycle(TCA cycle) 5. Various types of N-glycan biosynthesis 6. Sulfur metabolism 7. Peroxisome 8. Glysine, serine and threonine metabolism 9.Glycosylphosphatidylinositol(GPI)-anchor biosynthesis	1. Basal transcription factors 2. mRNA surveillance pathway	1. mRNA surveillance pathway 2. Basal transcription factors 3. Various types of N-glycan biosynthesis
DAVID	Fatty acid biosynthesis	1. N-Glycan biosynthesis 2. Fatty acid biosynthesis	Fatty acid biosynthesis	Fatty acid biosynthesis

The results given by PathwAX and DAVID are distinct: For each set, PathwAX gives out much more enriched pathways than DAVID. This could be explained by the principles of the two plats. PathwAX is built based on the crosstalk with FunCoup; a pathway is statistically enriched if this crosstalk (the number of links between the pathway and submitted gene set) is more than the number of that in a random network. In this way, PathwAX make it possible to visualize the gene sets in a bigger context, allowing more evidence to be used to calculate enriched pathway terms. On the other hand, DAVID utilizes Fisher's Exact method as well as Global annotation relationship to extract the best overlapping pathways within the selected gene sets; it has higher threshold and does not include as much as PathwAX does. [4,5]

When we deduced the number of input genes the enriched pathways and their total numbers also change. This might be interpreted as that, the websites try to collect the pathways by every single gene and integrate them with a certain overlapping threshold. When the number of input genes is changed,

some pathways will not be selected any more while some new ones may emerge, thus the overlapping parts will also vary.

DISCUSSION

We explored the topology of established interactomes as well as tried to build networks and enrich related pathways with experimentally defined gene sets. For the topology researching part, we learned how to summarize the overall network type of a huge interactome by simply looking into the distribution of node degree, which can make an abstract notion to be specific.

In the network establishing and pathway enrichment parts we probed into the usage of different online databases and got dramatically distinct outcomes with the same input into different websites. We attempted to explain these disparities from the difference of algorithms, calculation thresholds, etc in various databases.

Furthermore, as each online database contains huge information and substantial functions, it is of great importance to understand the input data thoroughly, since that tiny changes in parameter setting might cause great distinction to the content of outcome, which would influence the analysis. It is also notable to accumulate as much background principles as possible for each method we utilize.

References:

[1] Glynn DennisJr, Brad T Sherman, etc. DAVID: Database for Annotation, Visualization, and Integrated Discovery. Genome Biology, 2003 **4**:R60.

[2] Szklarczyk D, Morris JH, Cook H, Kuhn M, Wyder S, Simonovic M, Santos A, Doncheva NT, Roth A, Bork P, Jensen LJ, von Mering C. 2017. The STRING database in 2017: quality-controlled protein-protein association networks, made broadly accessible. *Nucleic Acids Research* 45: D362-368.

[3] Schmitt T, Ogris C, Sonnhammer EL. 2017.FunCoup 3.0: database of genome-wide functional coupling networks. *Nucleic Acids Research* 42 (Database issue): D380-388.

[4] Huang DW, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID Bioinformatics Resources. *Nature Protoc. 2009;4(1):44-57.*

[5] Huang DW, Sherman BT, Lempicki RA. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res. 2009;37(1):1-13.*