Grundlagen der Bioinformatik

Exercises – Assignment 4

Ulf Leser

Overview

- Download a PPI network and represent as graph
- Compute graph properties
- Perform a gene set enrichment analysis

Task 1.1

- In moodle, you find a file from STRING
  - Source: https://string-db.org
  - File: 9606.protein.physical.links.v11.5.txt.gz
  - Contains ~2 Million edges between human proteins
  - Remove first line
- Write a program with one parameter
  - Name of a file with edges of a PPI
  - Format: ID_PROT1 ID_PROT2 value
  - Reads the file and represent internally as graph (ignore value)
- Output the degree distribution of the graph
  - How many nodes with degree 1, 2, 3, … n, in reverse order
- Submission: program called "**graph_degree**"

Comments

- Interpret edges as undirected
- No need to store the IDs of the proteins
  - You may use internal (more compact) IDs
- Ignore the value (third column) of the edges
- Graph libraries (jGragh, iGraph, NetworKit …) are forbidden
- Store the graph as adjacency lists
  - Graph is sparse
  - Adjacency matrix would be VERY big
  - Good idea: Keep adjacency lists sorted
- Important: Fast access to all neighbors of an arbitrary node

Task 1.2

- Plot the distribution in a PDF
  - Here, you may use a library, e.g., JFreeChart, charts4J
- Answer: Is this a (a) random graph or a (b) scale free graph?
  - If your answer is (a): Compute edge probability p
  - If your answer is (b): Estimate parameter γ
    - Recall: $P(k) = k^{-\gamma}$
    - Try & error is OK, no programmatic optimal fitting required
- Submission: As PDF

Task 1.3

- Write a program with one parameter
  - Name of a file with edges of a PPI
- Program must compute the maximal clique(s) in this graph
  - Test with *small* graphs – first 1000, 2000, … lines of the PPI network from Task 1.1
- Output
  - Size of the maximal clique(s)
  - For every maximal clique: All proteins involved
- Submission: program called "**graph_clique**"

Comment

- First time in your life you implement an algorithm for an NP-complete problem?
  - See why this algorithm is NP?
- No worries: In sparse graphs, the size of S_k decreases very fast with k
- If adjacency lists are sorted, the overlap computation is linear
- When computing the overlap, also remember the "extra" nodes and test their neighbors only when testing for clique

```
build set S_2 of all cliques of size 2
i:= 2;
repeat
  i := i+1;
  S_i := ∅;
  for j := 1 to |S_i-1|
    for k := j+1 to |S_i-1|
      T := S_i-1[j] ∩ S_i-1[k];
      if |T|=i-2 then
        N := S_i-1[j] ∪ S_i-1[k];
        if N is a clique then
          S_i := S_i ∪ N;
        end if;
      end if;
    end for;
  end for;
until |S_i| = 0:
```

Competition

- Compute the size of the maximal clique as fast as possible
  - Use whatever tricks you find
  - Implementation may be different from solution to Task 1.3
  - Submission (voluntarily): program called "**graph_clique_competition**" with two parameters
- Only Java and Python are allowed
- We will measure wall clock time (unix time) for a small fraction of the PPI network

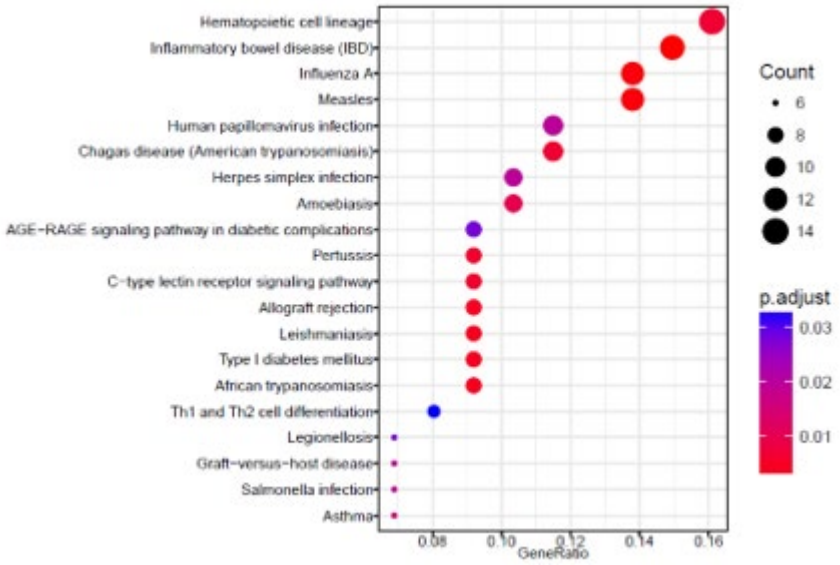
Things you may consider for speeding-up alignment

- Immediately remove all nodes with degree 1
- When you found a clique of size k – remove all nodes with degree smaller than k+1
  - Creates increasingly small filtered copies of the graph
  - Adjacency lists are reduced a lot
  - Filtering very effective especially for smaller k
- Many more tricks in the literature – feel free to explore

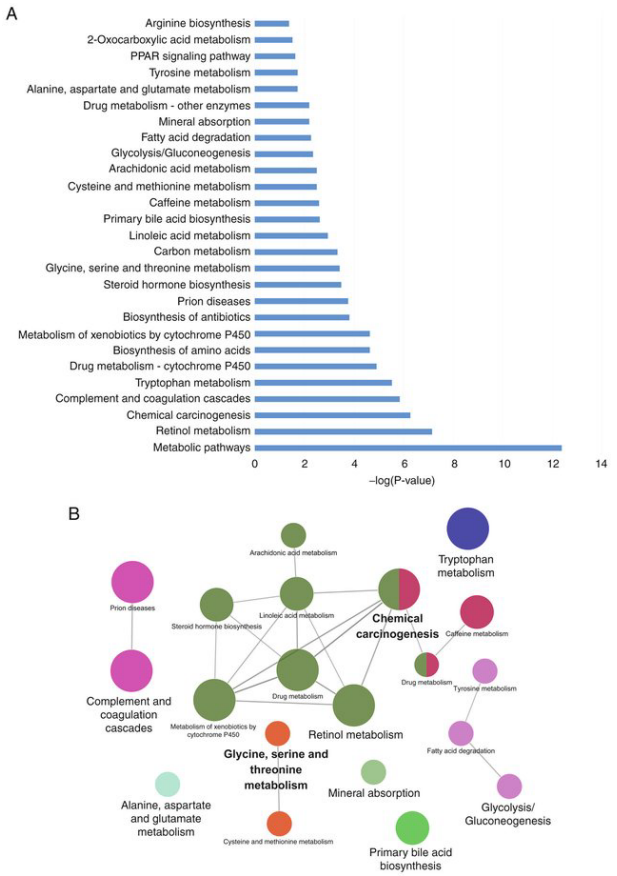
Task 2.1: Compute a pathway enrichment

- Very often, biomedical research identifies sets of "interesting" genes (signatures)
  - Overexpressed in a disease
  - Overexpressed in a cell type or developmental stage
  - Regulated by the same transcription factor
  - Sharing a motif/domain in their protein
  - …
- But: What functions do these genes have?
  - Biological "function" is always carried out by groups of proteins
    - Pathways
  - Most gene participate in several pathways
  - Whatever gene set we have – some pathways will be affected
  - Which pathways are affected more than expected by chance?

Gene Set Enrichtment Analysis (GSEA): Examples



https://www.biotrend.com



Zhang et al. "Prediction and analysis of weighted genes in hepatocellular carcinoma using bioinformatics analysis", Molecular Medicine Reports , 2019

Task 2.1. Obtain gene signatures

- Go to MSig-DB
  - One of several databases of interesting gene sets / signatures
  - Sorry: Will require a registration
  - Note URL of database in PDF (1)
- Find the signature "KRAS.LUNG.BREAST_UP.V1_DN"
  - Note URL of gene set and size of gene set
  - Download gene set and note in PDF (2)
- Go to DAVID - Functional Annotation Tool
  - Note the URL in PDF (3)
  - Compute a gene set enrichment
    - Paste and submit gene list (official symbols, human, "gene list")
    - Compute a "functional annotation chart"
  - Note screenshot of top-~40 pathways in PDF (4)

General requirements

- Remember to name all programs as requested
- All programs must run without further installations on GRUENAU2
  - *ssh username@gruenau2.informatik.hu-berlin.de*
- For all programs, source code must be submitted as well
  - Document your code
  - For Java/C etc.: Submit the source code and the compiled binary
- All responses must be submitted as PDF, where the task /assignment of every answer is clearly recognizable
- Zip everything into one file per task and upload via Moodle
  - **AssignmentX_groupY_taskZ.zip**
- Deadline for submissions: 7.7.2022, 0 o'clock

Questions?