Exploring the Utility of Ratio-based Co-expression Networks using a GPU Implementation of Semantic Similarity

By

Michael J Greer

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Biomedical Informatics

December 16, 2017

Nashville, Tennessee

Approved:

Bing Zhang, Ph.D.
Qi Liu, Ph.D.
Alissa Weaver, Ph.D.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 – INTRODUCTION

Cells are dynamic biological systems that require the coordinated activity of multiple gene products for growth and survival. With the advent of next-generation sequencing, bioinformatics methods have been developed to unravel gene function relationships, however most methods utilize steady state sequence data. The decreased cost of sequencing in recent years has made it feasible to obtain multi-tissue sequence data from the same patient [2-3]. This means that samples from diseased and adjacent normal tissue can both be sequenced and the patient specific alterations that caused a transition to the disease state can be analyzed. It is currently unknown how best to integrate these multi-tissue data sets and what unique information can be extracted.

Co-expression network analysis is a common technique used for gene function prediction. Previous studies have suggested that networks constructed using the ratio of values can offer performance improvement [4-5]. Using ratios to construct a network is one way to integrate multi-tissue information; however, the interpretation of these networks is slightly different than traditional co-expression networks. Ratio networks capture coordinated change and I hypothesize that these networks will be more functionally relevant since the edges represent alterations that influenced deviations from the normal state. A systematic study of the utility of ratio-based co-expression networks applied to cancer data has not been performed. A common method to evaluate the utility of a co-expression network is to test for the enrichment of functionally similar edges, which can be accomplished using semantic similarity scores[6].

Semantic similarity is a knowledge driven approach used to quantify the relationship between terms of an ontology[7]. Once term-wise similarity score have been computed, they can be used to evaluate the performance of co-expression networks [6,8]. Several methods use these scores to derive meaning from large high-dimensional NGS data sets with applications that include: gene function prediction, validation of gene product interactions, and gene product localization prediction[7-9]. However, calculating similarity scores is computational intensive [9-10]. Since every score can be computed independently, it is a fine-grained parallel problem and I hypothesize that implementing the scoring algorithms on a GPU will be orders of magnitude faster than equivalent CPU approaches.

Here, I develop a GPU implementation of a semantic similarity measure then use the scores to evaluate the performance of cancer type specific ratio-based co-expression networks compared to tumor and consensus tumor networks.

CHAPTER 2 – BACKGROUND

2.1 Semantic Similarity

The gene ontology (GO) project provides a vocabulary of terms that describe the properties of gene products[11]. It is organized as a directed acyclic graph, which means that links connecting terms are directionally distinct and the GO structure does not contain a set of links that form a loop [11-12]. In other words, if A and B are terms then (A → B) is different from (A ← B), and if (A → B) and (B → C) both exist then (C → A) cannot exist. GO is organized as three separate sub-ontologies that describe specific features within a living system. The biological process (BP) sub-ontology terms describe the biological objectives to which gene products contribute. The molecular function (MF) sub-ontology terms describe the biochemical activity of gene products. The cellular function (CC) sub-ontology terms describe locations within the cell where gene products are active[11-13]. Semantic similarity measures are used to quantify how similar one GO term is to one another[8].

There are three major approaches used to define GO term-wise semantic similarity: path- (edge-) based, information content (IC) based, and a hybrid of both[7-8]. The earliest approaches were path-based; these measures defined similarity based on the path from the root term of the ontology to the term of interest. A major drawdown of path-based measures is a lack of specificity since terms at the same level within the ontology are given identical scores[14]. Information content-based methods were introduced next; these approaches borrow concepts from information theory to define similarity. More specifically, these methods define the information stored within a term as the negative log of the frequency with which the term is annotated. The basic idea is that a sparsely annotated term will have higher information content because that term is more specific[15]. For example, if one GO term only has five gene annotations then it is more specific, and has more information, than the root term which might have several thousand gene annotations. If two terms have a sparsely annotated term as a common ancestor then the two terms are likely very similar to one another.

Since a single gene can be annotated to multiple terms within GO, different combination methods are required to produce gene-wise similarity scores. Just as there are different term-wise semantic similarity scoring methods there are different methods of combining term-wise scores with the optimal method being condition specific[14].

Once term- and gene-wise similarity scores are generated they can be used in many applications. For example, they can be used to validate proposed protein-protein interactions with the idea being that gene products that interact are likely to be involved in similar biological processes and active at similar parts within the cell, that is, they should have high semantic similarity using the biological process and cellular component sub-ontologies[7]. These scores can also be used to assess the quality of co-expression networks where higher quality networks will be enriched with edges of higher gene-wise semantic similarity[6].

2.2 GPU Computing

Computer programs are typically written to execute on a central processing unit (CPU) although programs can also be written to execute on a graphics processing unit (GPU)[16-17]. Sometimes implementing a solution on a GPU can result in dramatic performance improvements[18]. The main difference between a CPU and a GPU is the underlying computing architecture. A greater portion of a graphic processing unit is composed of arithmetic logic units (ALUs) and less space is allocated for control logic and caching; both of which are emphasized on a CPU[16]. More ALUs mean that many arithmetic calculations can occur in parallel and result in a potential performance boost.

Previous studies have shown that GPUs can be used effectively in bioinformatics because of the nature of the problems being solved [18-20]. A key attribute of the ideal problem is that the computational workload can be broken up into independent parts, that is, they display fine-grained parallelism. Since computing semantic similarity score can be done in parallel I hypothesize that a GPU based approach will perform better than CPU based approach.

2.3 Co-expression Network Analysis

A network consists of a collection of nodes with links connecting one node to another[21]. The network is a flexible data structure since the nodes and links of a network can represent any general entity and relationship[22-25]. Networks are used extensively in bioinformatics research and different types of networks are constructed depending on the data type used[6]. Some commonly studied biological networks include: protein-protein interaction networks, metabolic processing networks, and gene co-expression networks[27-29].

Two key applications of co-expression network analysis include: 1) to suggest the biological process a gene may be involved in using functional enrichment and 2) to find novel genes that may be part of a specific biological process[26]. These applications rely on a key assumption of co-expression network analysis known as the guilt by association (GBA) principle. The GBA principle states that genes that are co-expressed likely have a functional relationship[28].

There are many ways to construct a co-expression networks, however, all construction methods follow a similar pattern. First, a similarity measure is chosen to quantify the relationship between expression patterns. Common co-expression similarity measures include: spearman correlation, pearson correlation, and mutual information. Once pairwise similarity scores are computed you must determine whether two genes should be considered co-expressed. The simplest approach is to use a numerical cutoff where all gene pairs with a score higher than the threshold are considered co-expressed. Other approaches include clustering, using random permutations of the expression matrix to determine a significance threshold, and Bayesian approaches. Different approaches perform better under different conditions, but the best method is often determined empirically. Once a network is constructed, the performance of the network can be evaluated based on how the quality of the edges it identifies as being co-expressed. One method of doing this is to use semantic similarity to assess whether the edges identified are similar based on their gene-wise score.

CHAPTER 3 – METHODS

GPU Semantic Similarity Tool Implementation GPUGOSim is an open-source command line utility I developed using C++11 and the CUDA 7.0 software platform. The tool was developed using the Maxwell compute architecture. The Resnik scoring method (see below) was implemented and used in this study.

mRNA Expression Data Log base 2 transformed gene-level mRNA expression data was downloaded from The Cancer Genome Atlas (TCGA) using the Firehose web portal. All data sets analyze were generated using Illumina sequencing technology and normalized with the RSEM algorithm[29].

Computing Functional Similarity Between Gene Pairs The Gene Ontology (GO) structure and annotation data was downloaded July 2017. To ensure high quality GO annotations IEA and ND annotations were excluded. The Resnik semantic similarity method was used to compute term wise scores[30]. Given two GO terms, x and y, the Resnik scoring method is defined as follows:

$$sim(x,y) = max_{z \in P(x,y)} \left[log \frac{1}{p(z)} \right]$$

where P (x, y) represents the set of common ancestor terms including the root term of the given sub ontology. After defining term wise similarity scores, functional similarity was generated using the average combination method.

$$sim_{avg}(g_1, g_2) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} sim(go_{1i}, go_{2j})}{m \times n}$$

where go_1 and go_2 represent the set of GO terms that have genes g_1 and g_2 annotated, and m and n represent the total number of go terms with g_1 and g_2 annotations. After computing functional similarity scores for all gene pairs, the top 25% scoring gene pairs were classified as similar and the bottom 25% scoring gene pairs were classified as dissimilar.

Co-expression Network Construction The NetSAM R package[31] was used to construct all co-expression networks. The rank method, a K-means clustering approach, was used to construct all networks. The optimal K for each network was determined by varying K from 0.1%×D to 1%×D, where D is the number of genes in the data set. The K that produced the highest functional relevance (see below) while producing less than 15% of nodes with degree 1 was selected for downstream analysis.

Computing Functional Relevance Of Co-expression Network Given pairwise gene similarity scores, the functional relevance of a co-expression network was defined as follows:

$$LLR = \log\left(\frac{P(S \mid N)/P(D \mid N)}{P(S)/P(D)}\right)$$

where P (S | N) represent the frequency of similar gene pairs in the network, P (D | N) represents the frequency of dissimilar gene pairs in the network, P(S) represent the frequency of similar genes pairs, and P(D) represent the frequency of dissimilar gene pairs.

Gene Function Prediction Using Random Walk With Restart Gene function prediction was performed using the random walk with restart algorithm[32]. Briefly, this procedure takes a network, n, and a set of m seed genes (m > 0) that exists within the network as input and outputs a priority score for every other gene in the network using the following iterative procedure:

$$p^{t+1} = (1-r)Wp^{t} + rp^{0}$$

where p^0 represents the initial priority vector that contains scores of all genes in the network, p^t and p^{t+1} represent the priority vectors at time *t* and *t+1* respectively, W represents the column normalized adjacency matrix of the network, and *r* represents the restart frequency. Initially, all seed genes are given uniform probability and every other gene is given a probability of 0. This procedure is repeated until $\Sigma \mid p^{t+1} - p^t \mid < 1 \times 10^{-6}$. The final vector will contain priority scores with higher scores implying a closer relationship to the seed genes based on the topology of the network[32].

The predictive ability of the scores was assessed using GO BP terms. To compute the AUC associated with a single GO term a gold-standard positive gene set was defined as those genes annotated to the term and appearing within the network and a gold-standard negative gene set was defined as the other genes in the network. Prediction was done using five-fold cross validation where four of the five equally sized subgroups were combined as the training set to predict the remaining subgroup. AUC scores were computed using the scikit-learn toolkit.

Consensus co-expression network construction I explored the following two consensus construction techniques: *Method 1)* Pearson's correlation coefficients were first computed for all gene pairs of each cancer type. Gene pairs were then binned based on their correlation coefficient, ranging from -1.0 to 1.0 in 0.1 increments. The log likelihood of each bin for each cancer type was then computed as follows:

$$LLR_{ij} = \log\left(\frac{P\ (S \mid N_{ij})/P\ (D \mid N_{ij})}{P\ (S)/P\ (D)}\right)$$

where P (S | N_{ij}) represents the frequency of similar gene pairs in the network of cancer type *i* and correlation bin *j*, P (D | N_{ij}) represents the frequency of dissimilar gene pairs in the network of cancer type *i* and correlation bin *j*, P(S) represent the frequency of similar genes pairs, and P(D) represent the frequency of dissimilar gene pairs.

Each gene pair was then given a score based on the LLR of the bin it fell into across all the cancer types. For example, the score of gene pair k would be computed according to the following formula:

$$LLR_k = \sum_{i=1}^{n} LLR_{ij}$$

where i represents the cancer data set index and j represents whichever bin gene pair k appears. *Method 2)* Compute co-expression networks using clustering approach described above for all cancer data sets available. Next, select those edges that appear in *x* number of cancer types. The optimal *x* was chosen such that the number of nodes with a single edge was kept under 15%. In this study the optimal *x* was 4.

CHAPTER 4 – RESULTS

4.1 Performance comparison of GPU and CPU semantic similarity tools

Table 1 summarizes the term-wise running times of several semantic similarity tools. GOSemSim had the worst performance with the 10M term calculation and 100M term calculation failing to complete within 24 hours. The tools A-DaGO-Fun and SML had similar term-wise performance until the 10M term calculation. At this point, the SML tool began to display superior performance. The GPU based tool I've developed, called GPUGOSim, had the best overall term-wise performance with the difference becoming more exaggerated as the number of terms increased. For the 100M term calculation, GPUGOSim performed approximately 3x faster than the nearest competitor.

Table 2 summarizes the gene-wise running times of the same semantic similarity tools. Once again, the GOSemSim tool had the worst performance compared to the other tools while the GPU based tool performed the best. As the number of calculations increased the difference became more exaggerated. The 100M term running time of the GPU approach completed more than 6x faster than the nearest competitor.

4.2 Summary of mRNA profiling data sets

Table 3 summarizes the number of tumor, normal, and matched tumor normal samples for each cancer type considered in this study. Only cancer types with more than 20 samples were used when comparing tumor- and ratio-based co-expression networks. However, all cancer types were considered during construction of the tumor consensus network.

4.3 Comparing of tumor- and ratio-based co-expression network structure

The networks produced displayed good coverage with all networks containing greater than 10,000 nodes (Figure 1). There was also a high level of overlap between the nodes that appeared in the tumor network compared to the nodes that appear in the ratio-based networks. The percentage of overlapping nodes was always above 70%. The number of edges appearing within the network is shown in Figure 2. There was consistent small overlap between network edges suggesting that each type of network is capturing different functional relationships.

4.4 Functional relevance of ratio-, tumor-, and consensus networks

The functional relevance of all co-expression networks is shown in Figure 3. Although most of the top performing networks were ratio-based (9 / 12), the scoring differences were not significantly different based on the Wilcoxon sign rank test (p ≈ 0.1099). The functional relevance of both consensus networks (see methods) were computed, however, the method that selected frequently appearing edges produced a higher score (log(LLR) =1.63) compared to the other approach (log(LLR) = 1.42). The consensus network producing the highest score was chosen for all downstream analysis.

4.5 Comparing gene function prediction ability of ratio- versus tumor-networks and consensus- versus tumor networks

The AUCs of ratio- versus tumor- networks are shown in Figure 5. The ratio-based networks performed slightly better than the tumor networks with 7/12 of the networks having a majority of term AUCs above the 50% line. Furthermore, the difference between the number of top performing (AUC > 0.7) ratio- and tumor- terms was significant based on the Wilcoxon sign rank test (p ≈ 0.0111).

The AUCs of the same consensus network versus tumor- networks is shown in Figure 6. The consensus network performed consistently worse than the tumor networks with 10/12 of the networks having most AUCs below the 50% line, additionally, the difference between the top performing (AUC > 0.80) consensus and tumor-terms was not significant (Wilcoxon sign rank test p ≈ 0.2298)

4.6 Examining the top performing ratio- and consensus- network terms

The top performing terms (AUC > 0.80) of each network type were examined to determine if any processes were consistently better predicted using a particular type of network. The top performing terms identified using the ratio networks were related mainly to DNA damage repair, with the highest frequency term related to mitotic spindle organization.

The top performing terms identified using the consensus network were related to different forms of nucleus activity and different forms of cell signaling. The highest frequency term identified was related to protein import into the nucleus.

CHAPTER 5 – DISCUSSION

Using the CUDA parallel computing platform, we have developed the first tool capable of computing common semantic similarity measures on a GPU. Previous studies have focused on developing high performance multi-threaded solutions[10], but none to date have attempted to implement the algorithms on a GPU.

The superior performance of our tool provides another example of the usefulness of GPU computing in bioinformatics tool development, and I believe future work may result in increased performance. Since only a single graphics card was used in this study, the data had to be split into chunks before execution. Scaling to multiple GPU cards will eliminate this problem and result in improved performance.

Ratio based networks showed a modest improvement over tumor networks and the difference between the number of top performing terms was significant. However, whether ratio networks had better overall performance often depended on the cancer type. In certain cancer types the ratio-based approach performed better than tumor networks by a wide margin (e.g. LUAD and KICH), while in others the difference was marginal. This suggests that the ratio approach could be beneficial in some cases, but not all. Many of the top performing terms identified using ratio networks were related to DNA damage repair, meaning that these networks could be useful in helping researcher predict gene relationships involved in these processes.

A potential reason for the low performance of the ratio networks is that the genes were not filtered for variance in the normal expression vector. Low variance in the normal expression vector would cause the ratio and tumor results to be biased since you'd be dividing by a constant. Future work will explore whether selecting for highly variable normal expression leads to any performance improvement.

In conclusion, the GPU tool developed represents a performance improvement over existing methods. However, ratio-based co-expression networks showed only a modest improvement in predictive capability compared to tumor networks.

APPENDIX

Tables 1 – 3

Table 1: Performance comparison of term wise semantic similarity score generation

Tool	Lang.	1K	10K	100K	1M	10M	100M
GOSemSim	R	1m57s	12m31s	2h14m11s	23h18m51s	X	X
A-DaGO-Fun	Python	0m02s	0m03s	0m12s	1m49s	13m52s	X
SML	Java	0m09s	0m10s	0m11s	0m11s	1m22s	13m03s
GPUGOSim	**C++**	**0m09s**	**0m09s**	**0m09s**	**0m13**	**0m23s**	**4m06s**

Table 2: Performance comparison of gene wise semantic similarity score generation

Tool	Lang.	1K	10K	100K	1M	10M	100M
GOSemSim	R	0m44s	6m26s	1h03m36s	16h33m13s	X	X
A-DaGO-Fun	Python	0m05s	0m26s	4m33s	31m56s	4h39m58s	X
SML	Java	0m09s	0m10s	0m11s	0m53s	6m54s	1h08m30s
GPUGOSim	**C++**	**0m09s**	**0m09s**	**0m09s**	**0m14s**	**0m58s**	**10m23s**

Table 3: Summary of TCGA mRNA profiling data sets

	Tumor mRNA	Normal mRNA	Matched data
Cancer Type	Sample #	Sample#	Sample #
ACC	79	0	0
BLCA	408	19	19
BRCA	1093	112	112
CESC	304	3	3
CHOL	36	9	9
COAD	285	41	26
DLBC	48	0	0
ESCA	184	11	11
GBM	153	5	0
HNSC	520	44	43
KICH	66	25	25
KIRC	533	72	72
KIRP	290	32	32
LGG	516	0	0
LIHC	371	50	50
LUAD	515	59	58
LUSC	501	51	51
MESO	87	0	0
OV	303	0	0
PAAD	178	4	4
PCPG	179	3	3
PRAD	497	52	52
READ	94	10	6
SARC	259	2	2
SKCM	103	1	0
STAD	415	35	32
TGCT	150	0	0
THCA	501	59	59
THYM	120	2	2
UCEC	176	24	7
UCS	57	0	0
UVM	80	0	0

Figures 1 – 7

Figure 1 - Cancer type specific number of network nodes

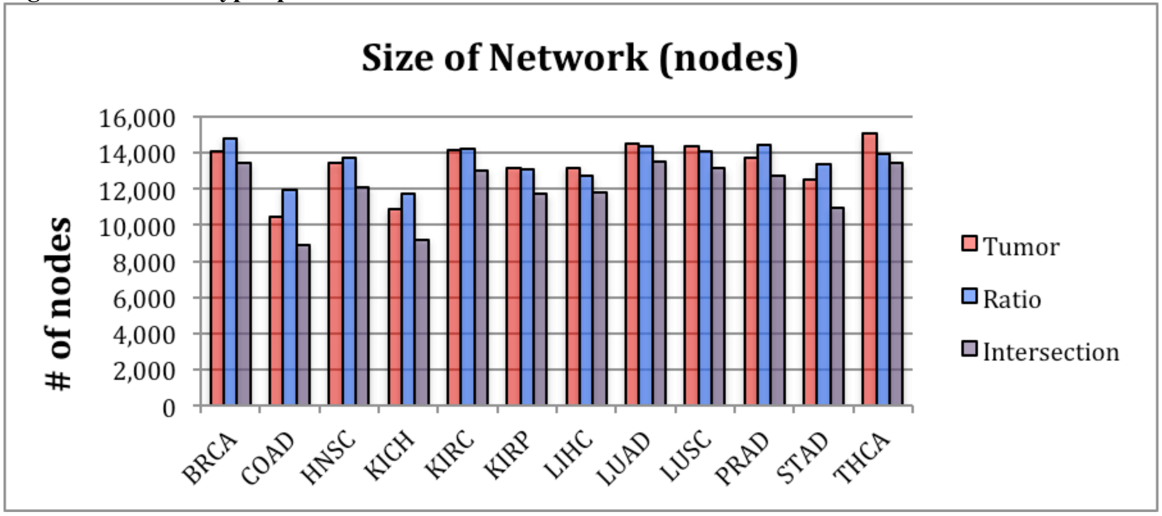


Figure 2 - Cancer type specific number of network edges

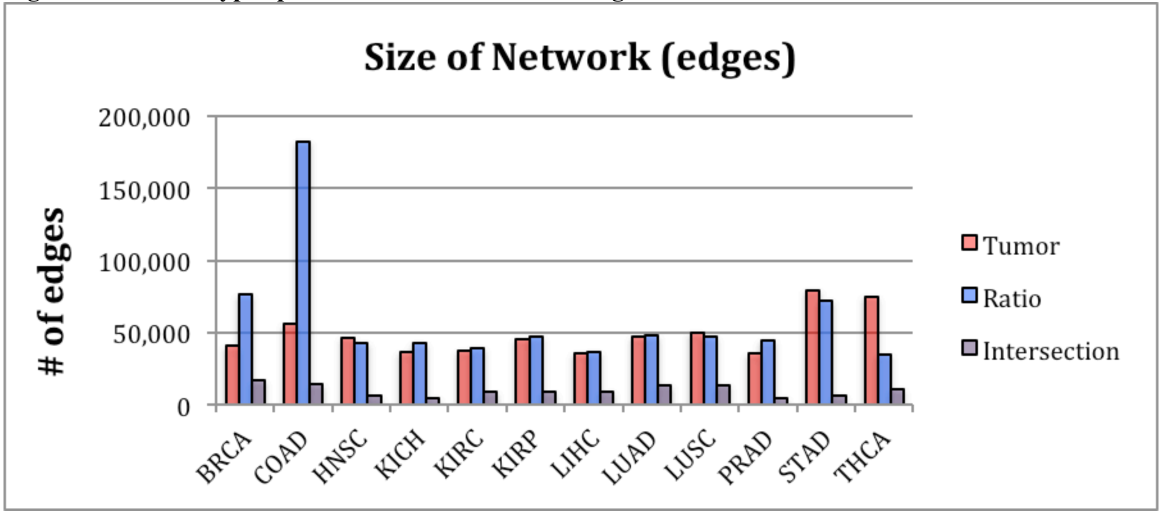


Figure 3 - Functional relevance of co-expression networks

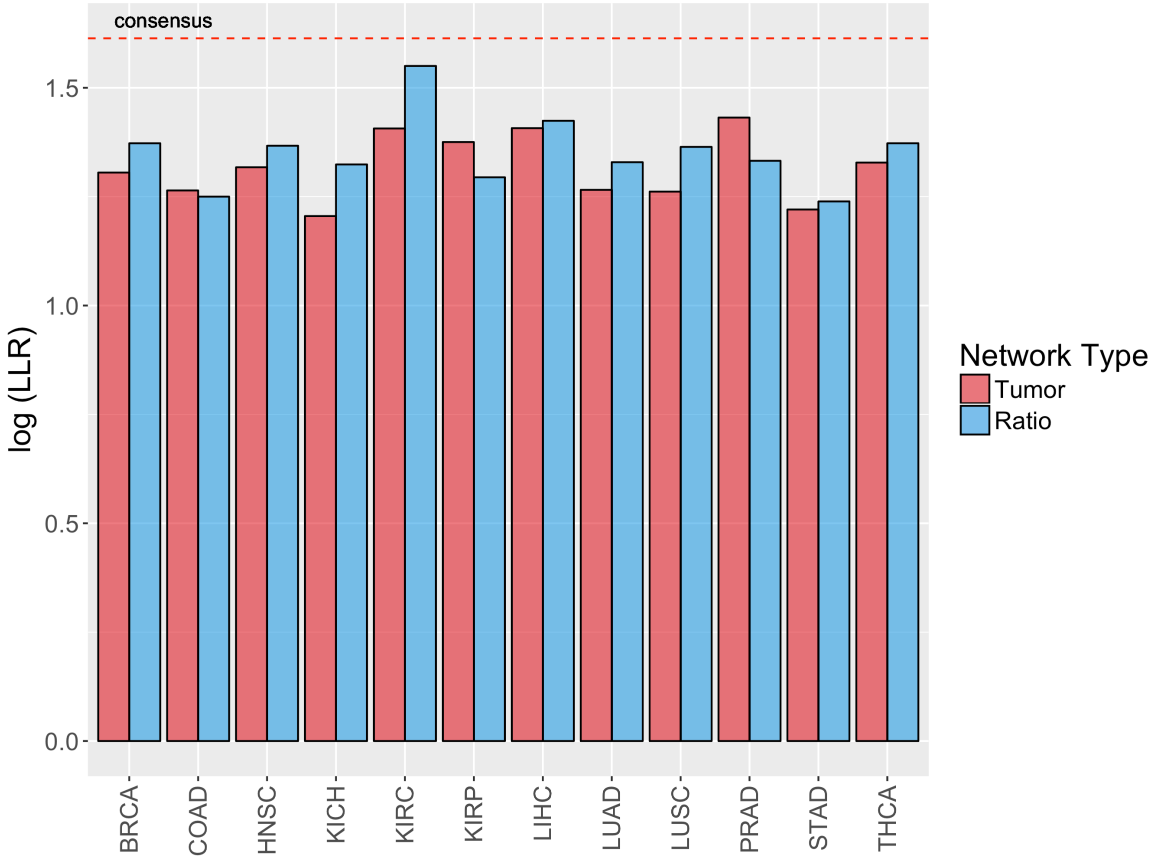


Figure 4 - Gene function prediction using tumor- and ratio-based networks

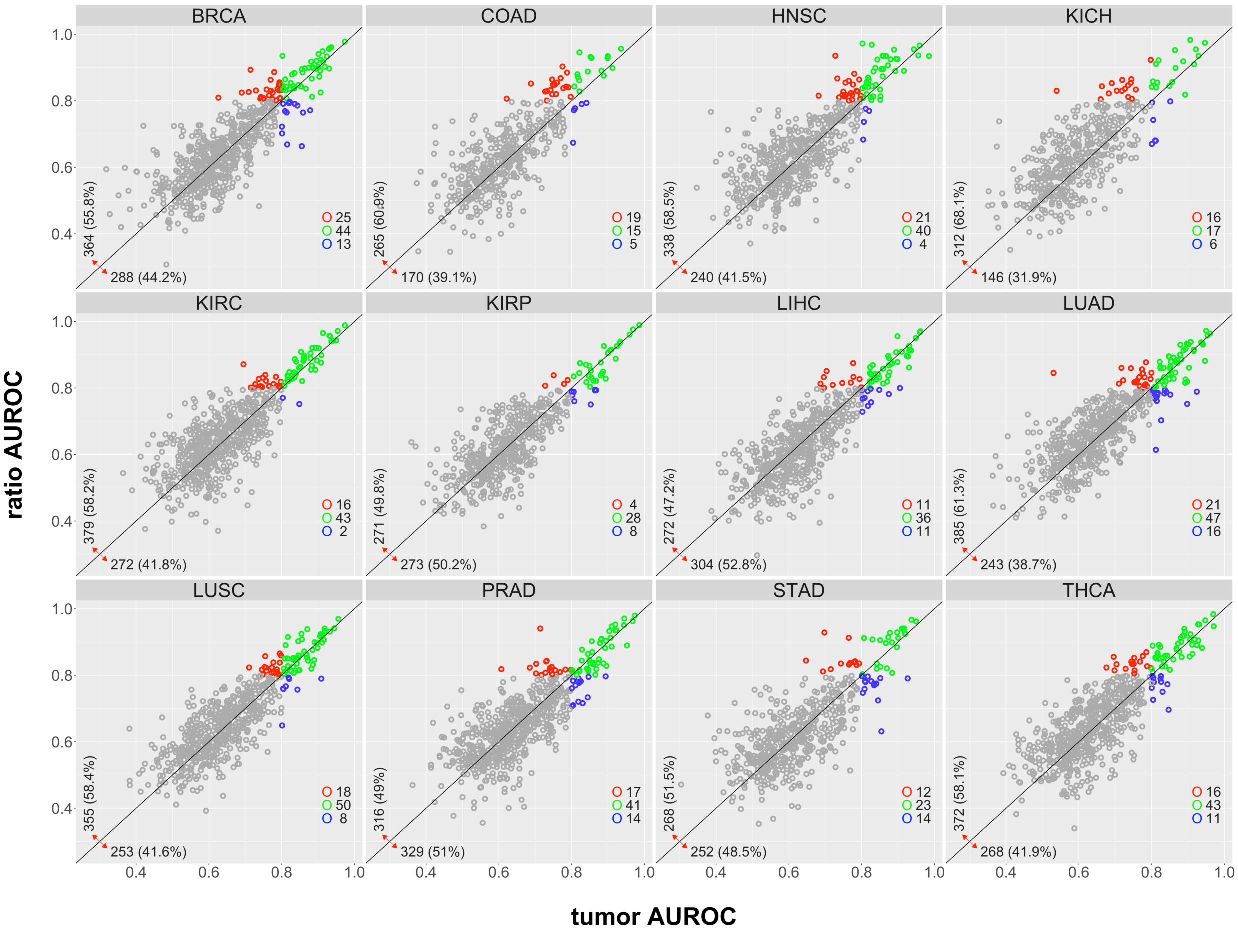


Figure 5 - Gene function prediction using tumor- and conserved network

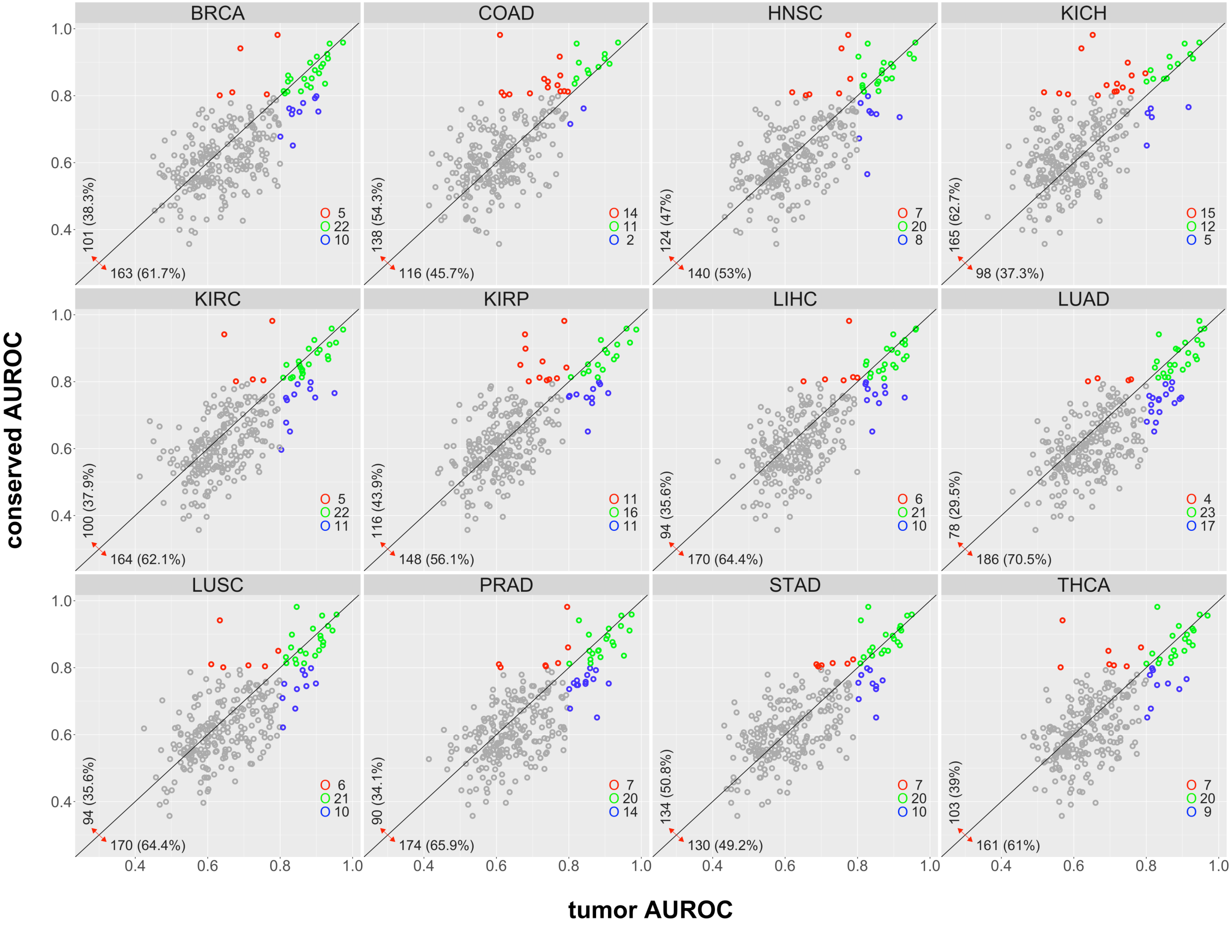


Figure 6 - Ratio-based network top performing terms

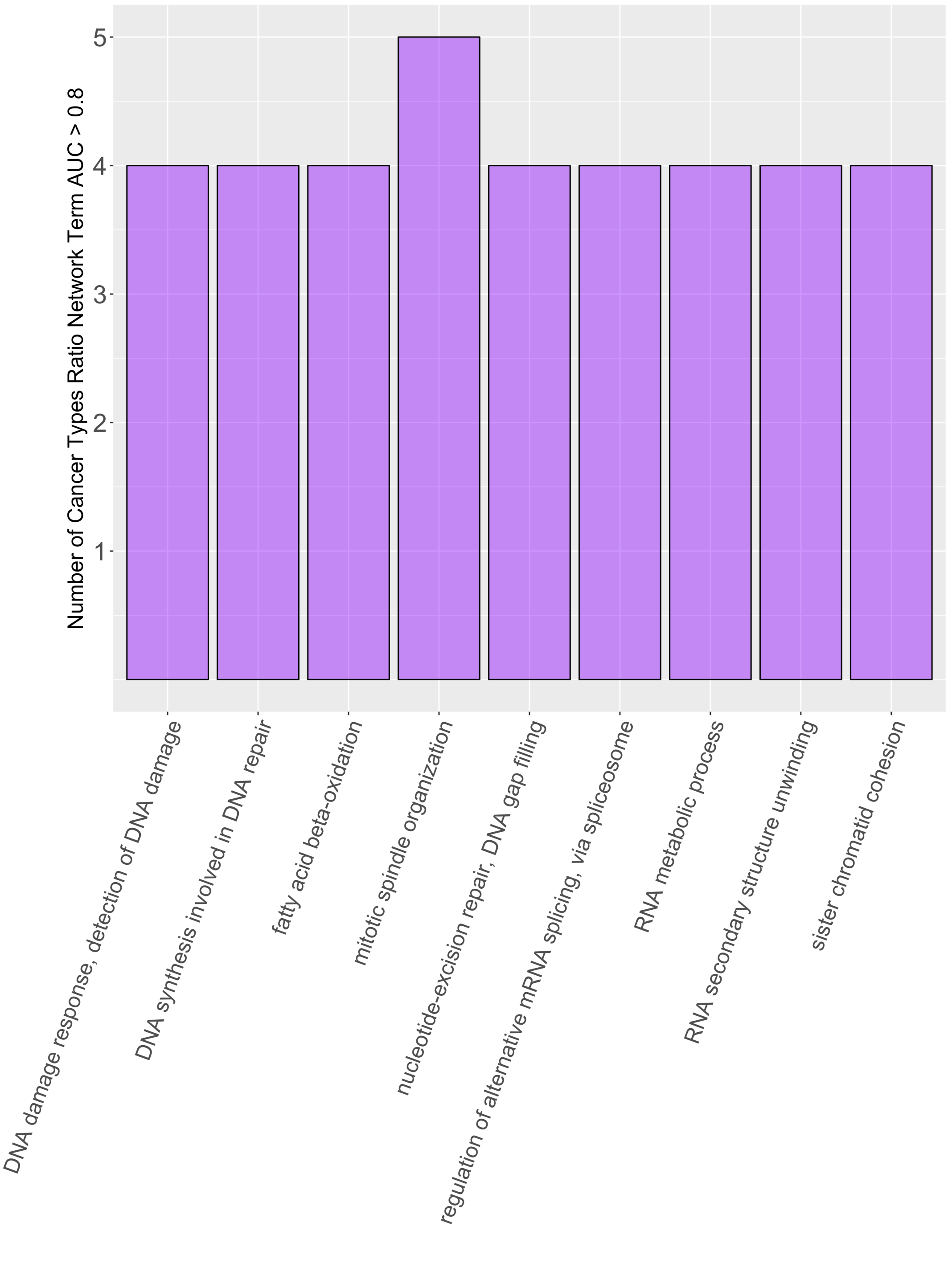
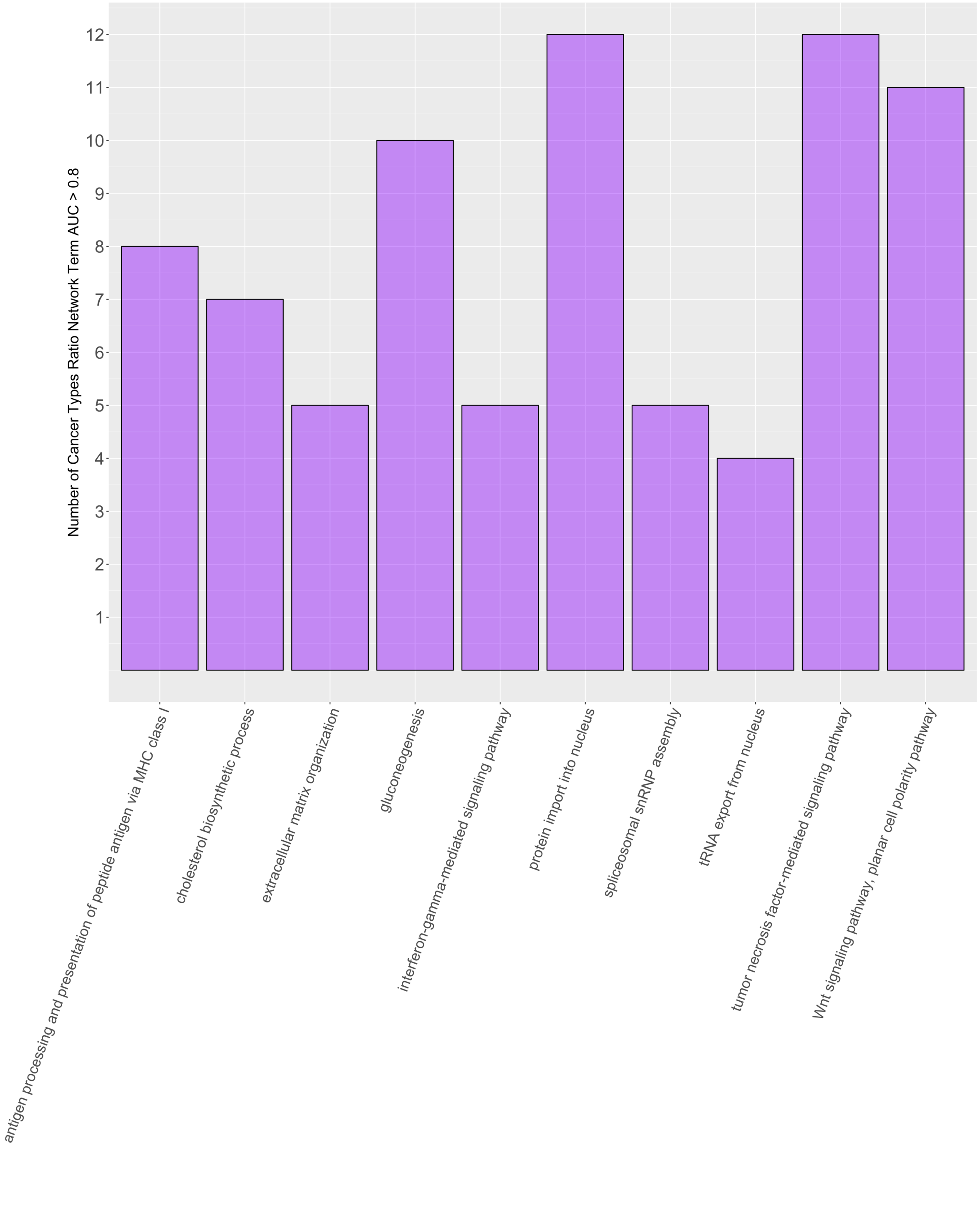


Figure 7 - Conserved network top performing terms



BIBLIOGRAPHY

1. Murali, T. M., Wu, C. J., & Kasif, S. (2006). The art of gene function prediction. *Nature biotechnology*, *24*(12), 1474-1475.

2. Goodwin, S., McPherson, J. D., & McCombie, W. R. (2016). Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, *17*(6), 333-351.

3. Reuter, J. A., Spacek, D. V., & Snyder, M. P. (2015). High-throughput sequencing technologies. *Molecular cell*, *58*(4), 586-597.

4. Netzer, M., Kugler, K. G., Müller, L. A., Weinberger, K. M., Graber, A., Baumgartner, C., & Dehmer, M. (2012). A network-based feature selection approach to identify metabolic signatures in disease. *Journal of theoretical biology*, *310*, 216-222.

5. Huang, X., Zeng, J., Zhou, L., Hu, C., Yin, P., & Lin, X. (2016). A New Strategy for Analyzing Time-Series Data Using Dynamic Networks: Identifying Prospective Biomarkers of Hepatocellular Carcinoma. *Scientific reports*, *6*.
6. Wang, J., Ma, Z., Carr, S. A., Mertins, P., Zhang, H., Zhang, Z., ... & McDermott, J. E. (2017). Proteome profiling outperforms transcriptome profiling for coexpression based gene function prediction. *Molecular & Cellular Proteomics*, *16*(1), 121-134.

7. Pesquita, C., Faria, D., Falcao, A. O., Lord, P., & Couto, F. M. (2009). Semantic similarity in biomedical ontologies. *PLoS computational biology*, *5*(7), e1000443.

8. Mazandu, G. K., Chimusa, E. R., & Mulder, N. J. (2016). Gene Ontology semantic similarity tools: survey on features and challenges for biological knowledge discovery. *Briefings in bioinformatics*, bbw067.

9. Guzzi, P. H., Mina, M., Guerra, C., & Cannataro, M. (2011). Semantic similarity analysis of protein data: assessment with biological features and issues. *Briefings in bioinformatics*, *13*(5), 569-585.

10. Harispe, S., Ranwez, S., Janaqi, S., & Montmain, J. (2013). The semantic measures library and toolkit: fast computation of semantic similarity and relatedness using biomedical ontologies. *Bioinformatics*, *30*(5), 740-742.

11. Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., ... & Harris, M. A. (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, *25*(1), 25.

12. Barabasi, A. L., & Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature reviews. Genetics*, *5*(2), 101.

13. Rhee, S. Y., Wood, V., Dolinski, K., & Draghici, S. (2008). Use and misuse of the gene ontology annotations. *Nature reviews. Genetics*, *9*(7), 509.

14. Mazandu, G. K., & Mulder, N. J. (2014). Information content-based gene ontology functional similarity measures: which one to use for a given biological data type?. *PloS one*, *9*(12), e113859.

15. Stone, J. V. (2015). *Information theory: a tutorial introduction*. Sebtel Press.

16. Cheng, J., Grossman, M., & McKercher, T. (2014). *Professional Cuda C Programming*. John Wiley & Sons.

17. Sanders, J., & Kandrot, E. (2010). *CUDA by Example: An Introduction to General-Purpose GPU Programming, Portable Documents*. Addison-Wesley Professional.

18. Nobile, M. S., Cazzaniga, P., Tangherloni, A., & Besozzi, D. (2016). Graphics processing units in bioinformatics, computational biology and systems biology. *Briefings in bioinformatics*, bbw058.

19. Payne, J. L., Sinnott-Armstrong, N. A., & Moore, J. H. (2010). Exploiting graphics processing units for computational biology and bioinformatics. *Interdisciplinary Sciences: Computational Life Sciences*, *2*(3), 213-220.

20. Kobus, R., Hundt, C., Müller, A., & Schmidt, B. (2017). Accelerating metagenomic read classification on CUDA-enabled GPUs. *BMC bioinformatics*, *18*(1), 11.

21. Barabasi, A. L., & Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature reviews. Genetics*, *5*(2), 101.

22. Jackson, M. (2015). The past and future of network analysis in economics. In *The Oxford Handbook of the Economics of Networks*.

23. Luke, D. A., & Harris, J. K. (2007). Network analysis in public health: history, methods, and applications. *Annu. Rev. Public Health*, *28*, 69-93.

24. Burt, R. S., Kilduff, M., & Tasselli, S. (2013). Social network analysis: Foundations and frontiers on advantage. *Annual review of psychology*, *64*, 527-547.

25. Barabási, A. L., Gulbahce, N., & Loscalzo, J. (2011). Network medicine: a network-based approach to human disease. *Nature reviews. Genetics*, *12*(1), 56.

26. Serin, E. A., Nijveen, H., Hilhorst, H. W., & Ligterink, W. (2016). Learning from co-expression networks: possibilities and challenges. *Frontiers in plant science*, *7*.

27. Raman, K. (2010). Construction and analysis of protein–protein interaction networks. *Automated experimentation*, *2*(1), 2.

28. Oliver, S. (2000). Guilt-by-association goes global. *Nature*, *403*(6770), 601-603.

29. Li, B., & Dewey, C. N. (2011). RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, *12*(1), 323.

30. Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.(JAIR)*, *11*, 95-130.

31. Shi, Z., Wang, J., & Zhang, B. (2013). NetGestalt: integrating multidimensional omics data over biological networks. *Nature methods*, *10*(7), 597-598.

32. Köhler, S., Bauer, S., Horn, D., & Robinson, P. N. (2008). Walking the interactome for prioritization of candidate disease genes. The American Journal of Human Genetics, 82(4), 949-958