Alisa Pavel [1,2,†], Angela Serra [1,2,†], Luca Cattelani [1,2], Antonio Federico [1,2], Dario Greco [1,2,3,4,*]

[1] Faculty of Medicine and Health Technology, Tampere University, Tampere, Finland.

[2] BioMediTech Institute, Tampere University, Tampere, Finland

[3] Institute of Biotechnology, University of Helsinki, Helsinki, Finland

[4] Finnish Center for Alternative Methods (FICAM), Faculty of Medicine and Health Technology, Tampere University, Finland.

* Correspondence: dario.greco@tuni.fi;

† Contributed equally

Running head: Network analysis of microarray data

Network analysis of microarray data

Abstract

DNA microarrays are widely used to investigate gene expression. Even though the classical analysis of microarray data is based on the study of differentially expressed genes, it is well known that genes do not act individually. Network analysis can be applied to study association patterns of the genes in a biological system. Moreover it finds wide application in differential co-expression analysis between different systems. Network based co-expression studies have for example been used in (complex) disease gene prioritization, disease subtyping and patient stratification.

In this chapter we provide an overview of the methods and tools used to create networks from microarray data and describe multiple methods on how to analyze a single network or a group of networks. The described methods range from topological metrics, functional group identification to data integration strategies, topological pathway analysis as well as graphical models.

Keywords (5-10): microarray, co-expression, differential co-expression, multi-layer networks, pathways

Introduction

The ultimate goal of large scale transcriptome analyses, such as DNA microarrays, is the characterization of the molecular alterations underlying a certain biological condition ***(1, 2)***. Although transcriptomics analysis allows the identification of a compendium of up to hundreds of genes which are deregulated under a certain condition, classical univariate analysis of the

individual gene alteration might fail to illustrate the complex interactions in the system under study ***(3)***.

Co-expression network analysis is the method of choice in order to describe gene-gene interactions underlying a certain phenotype. In the particular case of large scale transcriptomics experiments, network-based analyses can support the characterisation of the mechanistic interplay between individual genes based on their expression levels ***(4–7)***.

Starting from gene expression estimates, measured by microarrays, a co-expression network can be constructed (figure 1A-C). In this case, the genes and their associations are represented as a graph where the genes are the nodes of the network and their strength of similarity in their co-expression patterns can be represented as weighted or unweighted edges between the nodes. The advantages of representing microarray data as a network are multiple. For example, it allows the exploitation of a wide range of network topological properties (figure 1E-F) in order to generate new knowledge about the system under analysis ***(8, 9)***. Community detection or module detection allows tightly knit gene areas to be found (figure 1G) and then to functionally characterize them for example through pathway or (gene) ontology enrichment (figure 1H) ***(10)***. Multi-network comparison can provide insights about whether specific functionalities, single genes or gene neighborhoods are affected between multiple conditions.

Co-expression networks, built from DNA microarray data, can be integrated with other prior information (*e.g.* Protein-Protein Interaction (PPI) networks or co-regulation networks) to improve the robustness of the results (figure 1D) ***(11)***. This is based on the assumption that genes (or their proteins) that interact directly with each other or are co-regulated are often part of the same underlying biological function and therefore are likely to be co-expressed ***(12)***. Adding this

information during network generation allows the algorithm to detect noisy correlation patterns. Network analysis can also be applied in the context of multi-omics data analysis. Potential complementary information for the same samples, coming from different (experimental) data layers, are used to build a comprehensive picture of the biological systems in the form of a network.

Since DNA microarray technology became a pivotal instrument to study complex (or multifactorial) diseases, which are the result of complex interactions and perturbations involving large sets of genes, fast progress in the development of gene prioritization methods has been observed ***(13)***. These methods are aimed at uncovering and prioritizing candidate disease-associated gene markers by exploiting large scale omics studies ***(14)***.

Gene prioritization, through network-based methods, have become quite a popular tool. Another level of complexity, in understanding molecular relationships, is due to the fact that most cellular processes are interconnected through key genes (figure 1F). Network analysis aids biomedical researchers in identifying and prioritizing such key genes. Two of the most widespread strategies to exploit networks in order to identify and prioritise disease genes take into account i) the topology of the network and ii) prior information of the genes composing the network ***(15)***.

Another application of network theory in biomedical research is disease subtyping. The huge amount of large scale data made available in the last years for a plethora of human diseases, with a particular regard to multi-factorial ones, allowed the definition of "disease subtype" ***(16)***. Although a certain disease is characterized by a similar phenotype across the affected population, individual patients rarely show the same molecular makeup. This is particularly true for complex

diseases such as cancer as well as metabolic and immunological syndromes ***(17)***. Taking as example the study of cancer biology, the identification of patients subtypes is a central topic of research in order to identify novel drug targets and switch from classical therapeutic approaches (one disease - one therapy) to quasi-personalised pharmacological treatments (one disease - many therapies). Histological subtypes of some cancer types are already well-established ***(17)***. On the contrary, cancer subtyping from a molecular point of view can be less obvious due to the heterogeneity of the molecular alterations in cancer. Tumor stratification in clinically relevant subtypes can be achieved by integrating molecular networks with mutational profiles.

In this chapter, we will define the concept of a graph or network, we will describe algorithms to construct gene co-expression networks along with metrics that can be used to identify relevant nodes and edges based on the network topology. Furthermore, we will touch upon the basic concepts of pathway enrichment analysis, differential co-expression analysis and the use of graphical models on biological networks.

[Figure 1 here]

What is a graph

A graph $G = (V, E)$, consists of a set of nodes (V) and a set of edges (E). The graph in figure 2A has for example a node set of $[w, x, y, z]$ and an edge set of $[yw, yx, yz, xz]$. An edge wy joins two nodes $[w,y]$, which can be used to model relationships between node w and node y. In an undirected network $yw = wy$ holds. This means that there is no direction associated with any edge and it can be travelled in both directions. On the other hand, in a directed network there is a direction associated with each edge and the graph can only be traversed in this direction. For example, in

figure 2A *w* can be reached from *y* through the edge *yw* as well as node *y* can be reached from node *w* through the edge *wy* and *yw = wy* holds. In figure 2B, on the other hand, node *w* can be reached from *y* through the edge *yw* but node *y* cannot be reached from node *w*, since there is no edge *wy*. Further edges can be associated with a weight property, which can for example stand for a distance between node *y* and node *w* or a correlation coefficient between two nodes. In an unweighted network, each edge is considered equal and no weight attributes are assigned to any edge. For example, in figure 2A the "cost" of reaching node *x* from node *y* or node *z* is the same, while in figure 2C less "cost" is associated with the edge *zx* in comparison to edge *yx*. Here it is to be noted that, depending on what the assigned edge attributes stand for, either large values can indicate a larger distance or large values can indicate a higher similarity between the two nodes (e.g. if the edge attribute is a correlation). Further, the weight attributes do not need to be in [0,1] but can be in any range, as defined by the user. This implies that it is the users responsibility to ensure that edge attributes are interpreted in the correct way by applied algorithms. In a binary network representation, edge weights are either 1 or 0, where an edge weight of 1 implies that this edge exists and an edge weight of 0 implies that this edge does not exist.

[Figure 2 here]

Algorithms for genes co-expression networks

The results of a microarray experiment analysis is a normalized expression matrix *D* with *M* rows representing the genes and *N* columns representing the samples. Starting from this matrix, a gene co-expression network can be built, which allows investigation of how the genes jointly behave under the experimental condition. In this case, a gene co-expression network is a graph *G=(V,E)*, where *V* is the set of *M* nodes represented by the genes and *E* is the set of edges representing the

co-expression between all gene pairs. Under the assumption that genes with similar expression patterns are coexpressed, gene co-expression is usually computed by means of information-theoretic methods, such as the pairwise correlation coefficient or mutual information (MI) ***(18, 19)***, to evaluate how similar the expression profiles of two genes (g_i and g_j) across the set of N samples are. For example, correlation based measures assume continuous values between -1 and 1, where positive values mean that two genes have a similar pattern across the samples (e.g both genes have an increase in their expression values), while negative values indicate different patterns (e.g. when one gene increases its expression value, the other decreases). These techniques result in a weighted, undirected, and fully connected graph, where there is an edge connecting each possible pair of genes. However, microarray data are known to be noisy and prone to experimental biases. Thus, an important aspect is to evaluate the amount of non relevant edges inferred from the experimental data and be able to distinguish between real edges (e.g. genes that are actually co-expressed in the system and therefore show a high correlation coefficient) and edges that are inferred due to the noise (e.g. genes that are not co-expressed in the biological system but still achieve a high correlation coefficient) ***(20)***. To this end, simple approaches use a user-defined threshold to cut edges from the networks. In this case, all the edges, whose weight is below the predefined threshold are removed from the network. Here, the assumption is that low correlation or mutual information values can be induced by noise, while strong values might carry relevant information. However, the main issue with these approaches is that the selection of the threshold is arbitrary, and it does not take into account the topology and the structure of the network, and each edge is treated independently.

To overcome these limitations, multiple algorithms, such as RelNet ***(21)***, ARACNE ***(22)*** and CLR ***(23)*** have been developed, which differ mainly in the methods used to compute the co-expression values and how they identify non-relevant edges. The ARACNE algorithm works with both the mutual information and correlation measures. It computes the co-expression values for all gene pairs in a gene expression dataset and subsequently reduces the number of false positive connections, by cutting the less strong associations between every triplet of genes in the network. The CLR algorithm first computes the MI between each pair of genes, then calculates the statistical likelihood of each MI value within its network context ***(23)***. This algorithm compares MI values of gene pairs with the background distribution of MI values. The interactions whose MI scores stand significantly above the background distribution of MI scores are considered as the most probable interactions.

RelNet ***(21)*** works in two steps: it first creates a completely connected gene co-expression matrix where the mutual information between all genes is computed. Subsequently, a threshold is defined, called TMI, that identifies which associations are to be considered significant.

It is important to note that the different algorithms cut the non relevant edges by following different heuristics, thus when executing on the same dataset the resulting networks may not be consistent between them. For this reason, Marwah and collaborators recently proposed a tool, called INfORM (Inference of NetwOrk Response Module) ***(10)***, able to infer a more stable and robust network by applying an ensemble strategy. INfORM derives gene networks from a collection of algorithms, ranks the genes in each network by their relevance and merges them in a final network that ensures

robustness of gene-gene associations. Furthermore, it provides a simple graphical user interface and substantially guides the user in the algorithms setup and execution.

Local and Global connectivity measures

In year 1999, Albert László Barabási and Réka Albert published a study ***(24)*** where they investigated topological properties of real networks. They observed and demonstrated that real networks differ in their connectivity from random networks, which have been the main study subject at the time. The connectivity of real networks, such as the world wide web or molecular networks, follow a scale-free power law distribution, while random networks follow a gaussian distribution. This means that many real networks contain a few nodes, called “hubs”, which have a high number of edges in contrast to the majority of the nodes in the network. Based on this property researchers can explore the role of individual nodes in a network.

There are multiple different local and global network metrics available which have as a main objective to quantify the importance of a node based on the network topology ***(25)***. This can help to identify genes that have a “high” impact on many other genes (e.g. that are key regulators) and therefore may be a good target candidate for treatment ***(26, 27)***. Since all the measures (table 1) have a different approach on how to evaluate a node's importance (for information flow) it can be a good idea to combine multiple measures.

The metrics mentioned in table 1 can be interpreted differently based on the nature of the biological network they are applied to. In a PPI or a co-expression expression network, high degree nodes (hub genes) can for example indicate important regulators (e.g. master regulator of a biological function, such as transcription factors). A PPI network is a representation of how proteins are

known to interact within a biological system. In such a network nodes are proteins and their relationships represent known interactions between them (e.g. created through a Yeast-Two-Hybrid analysis). When comparing multiple networks, for example co-expression networks of different tissues or treatments, the degree distribution can be used to investigate if a significant perturbation of the system has occured or compare gene quantile positions between the networks ***(28, 29)***. In weighted networks (e.g. weighted co-expression networks) strength measures instead of degree measures can be used, which allows for example to add information about the "strength of correlation" between two nodes. Another type of measurement that aims to identify important nodes in the network are centrality metrics, which in contrast to the degree do not only take a node's direct connections into account but its overall position in the network. When comparing multiple networks, these measures can be used to identify nodes (genes) which have changed strongly in their overall connectivity and therefore may have been affected by the condition under investigation ***(28, 30)***. In figure 3 node *w* has the highest degree (centrality) as well as eigenvector centrality, while node c has the highest closeness centrality and node x has the highest betweenness centrality, since all traffic between the two tightly knight groups needs to go through it.

[Figure 3 here]

Next to the above introduced local measures, there are global network measures outlined in table 2, which aim at quantifying a network's overall topology without taking individual nodes into account. This can be a helpful measure to compare multiple networks or when networks without a large amount of common genes are compared.

Structural measures are especially useful to quickly compare multiple networks. For example to quantify if a treatment had a special effect in comparison to the control network(s). A less dense

network could for example suggest that a loss of homeostasis has occurred ***(31)***. Graphlet distribution and cycle distribution can be interesting to be evaluated in certain networks. For example different graphlets have been linked to different biological functions in protein - protein interaction networks or the existence of cycles can indicate existing feedback loops in a biological regulation network ***(32–34)***.

Community detection algorithms

Community detection algorithms aim at grouping the nodes of a graph into sets (communities) based on different properties. This results in sets of nodes which are more tightly connected (based on the grouped by property) with each other than the rest of the network. In figure 4 for example node groups w, z and x are topologically tightly connected with each other but do not have many outgoing edges (edges that go to another node group) and therefore groups w, z and x can be described as three communities of the network. Communities are also sometimes called modules. A community C of a graph G is defined as a node set $C = \{n1, n2,...nn\}$. Depending on the investigated problem or applied algorithm a node can be part of a single community or it can be assigned to multiple ones. Many different algorithms for community detection have been proposed (table 3), that can be classified into four categories: (i) node clustering algorithms aim at assigning each node to a specific community; (ii) overlapping community detection algorithms allow nodes to belong to multiple communities; (iii) probabilistic community detection algorithms estimate the probability of a node belonging to a community; (iv) edge clustering algorithms are similar to node clustering algorithms where instead of grouping the nodes, edges are assigned to distinct communities. A more detailed classification of these algorithms, divides them into weighted or unweighted, depending on whether they take edge weights into account when performing community detection or treat every edge as equal.

[Figure 4 here]

In co-expression networks we assume that nodes which are topologically close together in the network are part of the same process (e.g. genes that take part in the same pathway). Therefore grouping them into their communities, enables you to functionally enrich parts of the network. The most commonly used method is node clustering, but the assumption of a node only belonging to one community does not always hold in biological networks. For example a gene can take part in multiple different processes, such as being part of multiple different pathways.

Selecting a weighted community detection algorithm, allows to take another layer of information into account. For example in a co-expression network, correlation values can be used as edge weights, which tells the algorithm that two genes that are strongly correlated should belong to the same community (e.g. functional group). But depending on which type of network you are working with this information may or may not be available or can become more computationally expensive. There is not one community detection algorithm that fits all problems and most algorithms have not been developed for biological networks. Therefore an algorithm has to be selected based on the problem and network you are investigating.

In table 4 a few metrics are introduced which can evaluate mathematically the “goodness” of the community partitioning. However it needs to be taken into account that different algorithms identify the “best” partitioning based on different parameters and in the same manner different evaluation parameters focus on different metrics to estimate the “goodness” of the partitioning. Therefore, it is advised to select your evaluation metrics based on the community detection

algorithm you have selected. When possible, evaluating your partitioning over a multitude of evaluation parameters is advised. Further it can be useful, but more computationally expensive, to make use of an ensembl community detection method. In ensembl methods multiple (different) paritionings are combined and a consensus partitioning is identified. This allows to combine algorithms with different focus points and to create a more robust community partitioning.

Pathway enrichment analysis

Differential expression analysis at gene level is not able to capture the functional implications of the gene expression deregulation. This has led to the employment of a richer approach where genes that contribute to a single biological function are analyzed together. This kind of procedure is called “pathway analysis”. Pathway analysis is an analytical procedure that can help to clarify the disrupted functional interactions that sustain a certain phenotype. In detail, a pathway is a simplified representation of the functional interactions occurring in a cellular process. Pathways are a collection of several actors that may also be of different nature, spanning from proteins to metabolites, connected by a functional relationship (e.g. protein-protein interaction).

One of the most common solutions in order to relate the molecular findings obtained from omics experiments to a specific phenotype is to leverage the knowledge contained in several databases of biological functional associations (e.g. pathways) ***(35, 36)***. Such databases are for example KEGG (Kyoto Encyclopedia of Genes and Genomes) ***(37, 38)***, Reactome ***(39)***, Biocarta ***(40)*** and PANTHER ***(41)***.

These databases contain collections of genes grouped into pathways or biological functions that can be used to functionally characterize a set of relevant genes (e.g. differentially expressed genes

or the genes in a particular community) instead of studying them individually. Classically, enrichment methods rely on tests applied to evaluate the statistical significance of the overrepresentations of the genes in a pathway or functional group into the set of genes of interest. The most widespread statistics employed in order to verify the enrichment of a certain pathway by differentially expressed genes is the overrepresentation test, for example the Fisher Exact test , chi-square and hypergeometric test ***(42)***.

Alternatively, a slightly more sophisticated method is the Gene Set Enrichment Analysis (GSEA) ***(43)***, where the enrichment of one or more pathways are evaluated against a ranked gene list, by means of the Kolmogorov-Smirnov test ***(44)***. In recent years, pathway analysis methods shifted from a non-topological to a topological approach, where the knowledge about the position of each gene, as well as the type and the direction of a signal, within a biological/cellular pathway is taken into consideration (figures 5-6). The advantage of the topological approach is that a hypothesis testing for pathway expression is often more accurate ***(35, 45)*** .

[Figure 5 here]

Draghici and colleagues ***(46)***, described the first method able to integrate topological information in the classical pathway analysis approach. This method, named impact analysis, takes into consideration two properties: the magnitude of deregulation of the genes (usually represented as the log-fold change) belonging to a certain pathway and the position and the type of gene-gene interactions within the pathway. The first implementation of this method was included in the Pathway-Express package (now included in ROntoTools, https://rdrr.io/bioc/ROntoTools/), which represents a precursor of the following widely used SPIA ***(47)***, graphite ***(48)*** and ROntoTools ***(49)***.

Nguyen et al. ***(35)*** compared 5 topologically-based and 8 non-topologically-based pathway analysis tools, showing that topologically based tools generally perform better than non-topologically based ones, but this is not always true and depends on the specific tools and the specific aspects on which they are compared. The results of their study suggest that when considering only the ranking of pathways on real pathological data, the non-topological PADOG algorithm ***(50)*** shows the best performance. If we consider data from knockout experiments, where the expression of specific genes is artificially silenced, the topological ROntoTools has the best performance, while if we consider the distribution of p-values under the null hypothesis the non-topological GSEA ***(43)*** is the only unbiased one.

[Figure 6 here]

Differential co-expression analysis

Differential co-expression analysis aims to identify significant differences in the structure of two or multiple co-expression networks. The assumption is that genes that are differentially co-expressed between different experimental setups (e.g. diseases *vs.* controls) are more likely to be key regulators, and are therefore likely to explain differences between phenotypes ***(51–54)***.

The simplest approach for differential co-expression analysis consists in ranking the genes in each network according to one or more centrality measures (e.g. degree centrality) and comparing these ranks to identify genes who are ranked at the top only in one co-expression network and not in the others ***(25, 55, 56)***. Other gene-based differential co-expression analysis methods identify genes that show changes in association with other genes across multiple experimental conditions. To this end, different strategies have been developed, and they can be differentiated in global, local or

hybrid methods based on if they compare the expression pattern of one gene with those of all the other genes, with those of a subset of genes, or if they apply a combination of global and local measures ***(29, 57)***. Global gene-based methods include DCglob ***(58)*** and the N-statistic ***(52)***. Local gene-based methods include DCloc ***(58)***, DCp ***(59)***, DCe ***(59)***, DiffK ***(60)***, differential motif centrality ***(25)***, RIF ***(61)***, and metrics based on correlation vectors ***(62)***. DiffRank is a hybrid method where both local and global measures of differential association are computed for each gene ***(63)***.

More complex differential co-expression methods work by identifying communities in each network and comparing them (figure 7). The most simple comparison is the presence (or absence) of a module between the two networks (figure 7A). This might indicate that a particular biological process, associated with the genes in the module, can be (when present) or cannot be (when absent) exerted by a particular experimental condition. Another scenario consists of identifying a set of genes that form a module in both networks and investigating their connection structure (figure 7B). This would allow us to find changes occurring in molecular processes underlying both experimental conditions. For example, by analyzing the strength of interaction between the genes in the same module, one could find out that the module's hub in the two networks are different. This would mean that, even though the two experimental conditions carry out the same biological function, this is driven by a different key gene. More complex patterns can be detected in differential co-expression analysis, such as community division (figure 7C) and gene hopping (figure 7D). In the case of community division, a community of genes that is present in a network is then split in two or more communities in another network. In the case of gene hopping, a set of

genes that is tightly connected with the gene of a community in one group, switch their connections to another community in another group.

[Figure 7 here]

Multiple tools have been developed to perform differential co-expression analysis at module level such as WGCNA ***(64)***, DICER ***(51)***, DiffCoEx ***(65)*** and DINGO ***(66)***, GSCA ***(67)***. WGCNA is a popular tool for module identification which is able to compute the importance of a module in a subpopulation of samples. Similarly, DICER and DiffCoEx identify *de novo* modules and allow for comparisons between multiple conditions. DINGO works by grouping the genes based on how differently they behave in the samples of a particular condition with respect to the baseline co-expression determined from all samples ***(66)***. Another method for *de novo* module identification and differential co-expression analysis is CoXpress, which is only able to compare modules between two experimental conditions ***(68)***. Differently from the previous methods that work by first identifying the gene modules, the GSCA method starts from a known list of genes and ranks them according to a differential co-expression score between multiple conditions ***(67)***. Other methods that only work with binary comparisons and known sets of genes are GSNCA ***(69)***, CoGA ***(70)***, dCoxS ***(71)*** and DiffCorr ***(72)***.

A number of studies have successfully used differential co-expression analyses to identify networks unique to specific tissues ***(73)*** or disease state ***(31)***. For example, in the GTEx project multiple expression data for 35 different human tissues have been collected ***(74)***. Based on the average gene expression of each tissue a hierarchy has been derived and used to generate a single combined co-expression network derived from the tissue specific networks. They showed that in tissue specific networks, transcription factors with functions specific to that tissue are highly

expressed together with tissue-specific genes. However, the tissue specific genes tend to remain at the periphery of the network, while the transcription factors are more central. Thus, transcription factors could be uncovered by identifying modules with increased co-expression strength in tissues-specific networks and by pinpointing the central hubs of these modules. On the other had, genes that are not TFs but are tissue-specific should be detectable by identifying genes that are at the periphery in these modules

Integration strategies for graphs

Data integration strategies can help to increase robustness of your microarray analysis and help in its analysis ***(75–78)***. There are many existing knowledge bases, structured as interaction networks, in the biological domain that contain valuable information about the relationships between genes, such as protein-protein interaction networks or regulation networks ***(37–39, 79–82)***. By combining results from microarray data analysis with these biological networks, hidden relationships and functional implications can be detected. For example differential expressed genes can be combined with a protein-protein interaction network, to investigate what other genes may be involved in the observed response ***(11)***. This can be further expanded by adding knowledge about direct protein interactors with your treatment condition (for chemicals/ drugs such data can for example be retrieved from CTD ***(83)*** or DrugBank ***(84)***). Like the differential expressed genes, the identified interactor gene sets can be mapped onto a protein-protein interaction network, which allows investigation of genes that are very likely propagating the response between these two sets ***(85)***.

Furthermore, such knowledge can directly be used during network creation. For example combining a protein-protein interaction network with a correlation based co-expression network,

can help guide the process of estimating which are the relevant edges to keep or discard during the simplification steps, as is for example implemented in the INfORM tool ***(10)***. However, this kind of approach has to be carefully applied. Evaluating the final results on the basis of similar data used in the integration process should be avoided in order to not introduce bias. For example you should not score edges based on two genes that are known to be in the same pathway and then for example do community detection on the network and perform pathway enrichment on these modules. Such results are likely biased by your data selection method and will not provide meaningful results.

We have now discussed how different external data could be integrated with your microarray data analysis, either by referring to a known biological network or using such data during the analysis process. It is to note that these steps can not only be performed on a single data layer (e.g. where nodes are only of one data type and edges are only of one data type) but can also be performed on multi-layer or multiplex networks. Such networks are heterogeneous networks, which means that nodes and edges can represent different objects and relationships. For example, a Drug - Gene Target network or a Gene - Gene network where there are multiple relationships between genes, such as interactions, co-regulations and take part in the same pathway.

To use such networks together with your microarray data it may be helpful to convert them into homogeneous (e.g. Gene - Gene) networks. This can for example be achieved by estimating relationships (similarities) between genes based on their common neighbors. Multiple of such networks can be merged into a single network by combining their edges or adding their adjacency matrices.

Furthermore, it is to be noted that from microarray technology multiple types of omics data can be produced. These are usually referred to as multi-omics data, which are experimental measures related to the same set of samples on which multiple molecular experimental results have been performed (e.g. gene expression, methylation, copy number variation etc.). These data are particularly useful, since they can show complementary aspects related to the same biological process and can be used to gain a better understanding of the overall phenotype(s) under study. Multiple methods for the integrative analysis of multi-omics data through network analysis have been developed ***(86–88)***. Examples of such methods are SNF ***(76)*** and lemon-tree ***(89)***. The SNF algorithm combines multiple networks with a homogeneous set of nodes into a single one. The SNF methodology can be used for multiple tasks. The authors showcased their integrative methodology for the patient subtyping task. In this case, each co-expression network represents the patient similarity in each omic view. These networks are merged to build a patient similarity network, which accounts for all the different aspects of the multi-omics data and can be used to cluster the patients in multiple subsets.

The lemon-tree method was developed to identify gene co-expression modules starting from gene expression data ***(89)***. It first infers co-expression gene clusters using a model-based Gibbs sampler, then it identifies modules of co-expressed genes by means of a consensus based approach based on the spectral edge clustering. Eventually, another omics data layer, bringing information on candidate regulators such as miRNA expression, CNV and methylation data are combined with the consensus module to infer a regulatory score by using a decision tree structure.

Graphical models

Biological systems are by nature highly complex systems which cannot yet be described accurately (there is no technical instrument to wholly measure them) therefore the possibility of a relationship existence can be described by means of probabilities. Graphical models are a means to compactly define probability distributions on a large number of variables.

Graphical models are graph-based representations of statistical conditional dependence between the variables of a system. Each variable is assigned to a node of the graph, while edges are used to represent dependence. According to the type of graph, each node may represent a categorical, ordinal, or real valued variable, or even a tuple of more than one atomic variable. If an edge from A to B exists in the graph, then P(B|A), the probability of B given A, is a factor in computing the joint distribution on the whole graph. On the opposite, if there is no edge from C to D, knowing P(D|C) is not needed in order to compute the joint distribution on the whole graph. For this reason, graphical models can be considered compact representations of distributions. A node is conditionally independent with respect to the rest of the graph given a set of nearby nodes, this set of nearby nodes depends on the type of the graphical model and is called Markov blanket.
Bayesian networks (figure 8A), also known as belief networks, are directed acyclic graphs, and the Markov blanket of a node is given by its parents, its children and the parents of its children. Markov networks (figure 8B), also known as Markov random fields, are undirected graphs allowing cycles, and the Markov blanket of a node is given by its first neighbors. Dependency networks (figure 8C) are directed possibly cyclic graphs, and the Markov blanket of a node is given by its parents.

[Figure 8 here]

The potentialities of applying graphical models to biological data are well known ***(90–92)***, and many software implementations exist ***(93, 94)***. An advantage of graphical models is that in their general formulation there is no restriction on the functions that model the probability of the values of a node given the values of its parents/neighbors, even non-linear functions can be used. A disadvantage is that if both the graph structure (i.e. the adjacency matrix) and the distributions are to be learned from data, as it is often the case with applications for the analysis of microarray data, the algorithms are often slower that the ones inferring simpler graph representations.

In order to use Bayesian networks to describe a system in time, Dynamic Bayesian networks (DBN) were introduced. A Dynamic Bayesian network can be seen as a Bayesian network that is replicated at each time point, with edges connecting nodes at contiguous time points to represent time-dependent evolution. Grzegorczyk et al. ***(95)*** proposed an extension to DBN and applied it to Affymetrix microarray RNA data from *Arabidopsis thaliana* to study its circadian rhythms.
Directed graphical causal models are directed graphical models where edges represent causality, a stronger concept with respect to just dependency. Learning causality from data, especially if devoid of a sequence of time points, is a particularly difficult problem, but very interesting for its potential to explain biological systems. Glymour et al. ***(96)*** provides a review of causal discovery methods for graphical models including examples of applications to gene expression data.

Conclusions/Summary

In this chapter we described multiple strategies for the analysis of microarray data based on a network approach. Algorithms for the creation of co-expression networks from microarray data, such as ARACNE, CLR and INfORM (an ensemble approach) have been described and different

network types have been defined. Additionally multilayer and data integration approaches, which make use of a wide range of (experimentally) derived data for network creation have been outlined. Including multiple layers of information into your network creation and/ or analysis can yield more robust networks as well as guide analysis (for example functional enrichment).

Different metrics for topological network description and comparison between networks have been introduced of which gene prioritization methods find wide application in disease gene identification or to identify system perturbations caused by treatment conditions. Gene prioritization methods for example include degree centrality and closeness centrality. Next to these local network metrics global metrics can be used to describe a network's overall topology, which can be used to describe a group of networks or if certain treatment methods had significant impact on the gene relationships in a network.

Algorithms to detect functional groups within a network (communities) or to compare their distributions between networks have been discussed as well. Different algorithm types have been introduced together with multiple metrics that can be used to evaluate the created network partitioning. Community detection can be used to describe groups of genes in a network that are more tightly connected with each other than the rest of the network (highly co-expressed) and therefore are likely to take part in similar functionalities. Communities can be enriched by external data, such as pathways to functionally describe them. Topological pathway analysis methods were also discussed, which instead of only looking at (grouped) genes take their connection into account as well. Lastly we introduced graphical models, which try to describe the uncertainty existing in biological networks.

All together established and emerging methods for microarray analysis based on a network approach have been outlined and discussed, which can be used to gain new insight into gene - gene relationships, gene - disease relationships and many more relationship types contained in the data.

Bibliography

1. Kinaret PAS, Serra A, Federico A, et al (2020) Transcriptomics in toxicogenomics, part I: experimental design, technologies, publicly available data, and regulatory aspects. Nanomaterials (Basel) 10
2. Federico A, Serra A, Ha MK, et al (2020) Transcriptomics in toxicogenomics, part II: preprocessing and differential expression analysis for high quality data. Nanomaterials (Basel) 10
3. Serra A, Fratello M, Cattelani L, et al (2020) Transcriptomics in toxicogenomics, part III: data modelling for risk assessment. Nanomaterials (Basel) 10
4. Dam S van, Võsa U, Graaf A van der, et al (2018) Gene co-expression analysis for functional classification and gene-disease predictions. Brief Bioinformatics 19:575–592
5. Zhao W, Langfelder P, Fuller T, et al (2010) Weighted gene coexpression network analysis: state of the art. J Biopharm Stat 20:281–300
6. Liu W, Li L, Ye H, et al (2017) [Weighted gene co-expression network analysis in biomedicine research]. Sheng Wu Gong Cheng Xue Bao 33:1791–1801
7. Kinaret P, Marwah V, Fortino V, et al (2017) Network analysis reveals similar transcriptomic responses to intrinsic properties of carbon nanomaterials in vitro and in vivo. ACS Nano 11:3786–3796
8. Song Z-Y, Chao F, Zhuo Z, et al (2019) Identification of hub genes in prostate cancer using robust rank aggregation and weighted gene co-expression network analysis. Aging (Albany NY) 11:4736–4756
9. Li W, Wang L, Wu Y, et al (2020) Weighted gene co-expression network analysis to identify key modules and hub genes associated with atrial fibrillation. Int J Mol Med 45:401–416
10. Marwah VS, Kinaret PAS, Serra A, et al (2018) Inform: inference of network response modules. Bioinformatics 34:2136–2138
11. Mousavian Z, Nowzari-Dalini A, Rahmatallah Y, et al (2019) Differential network analysis and protein-protein interaction study reveals active protein modules in glucocorticoid resistance for infant acute lymphoblastic leukemia. Mol Med 25:36
12. Yeung KY, Medvedovic M, and Bumgarner RE (2004) From co-expression to co-regulation: how many microarray experiments do we need? Genome Biol 5:R48
13. Valentini G, Paccanaro A, Caniza H, et al (2014) An extensive analysis of disease-gene

associations using network integration and fast kernel-based gene prioritization methods. Artif Intell Med 61:63–78
14. Tiffin N, Andrade-Navarro MA, and Perez-Iratxeta C (2009) Linking genes to diseases: it's all in the data. Genome Med 1:77
15. Köhler S, Bauer S, Horn D, et al (2008) Walking the interactome for prioritization of candidate disease genes. Am J Hum Genet 82:949–958
16. Lötvall J, Akdis CA, Bacharier LB, et al (2011) Asthma endotypes: a new approach to classification of disease entities within the asthma syndrome. J Allergy Clin Immunol 127:355–360
17. Ozturk K, Dow M, Carlin DE, et al (2018) The emerging potential for network analysis to inform precision cancer medicine. J Mol Biol 430:2875–2899
18. Stuart JM, Segal E, Koller D, et al (2003) A gene-coexpression network for global discovery of conserved genetic modules. Science 302:249–255
19. Bansal M, Belcastro V, Ambesi-Impiombato A, et al (2007) How to infer gene networks from expression profiles. Mol Syst Biol 3:78
20. Serra A, Coretto P, Fratello M, et al (2018) Robust and sparse correlation matrix estimation for the analysis of high-dimensional genomics data. Bioinformatics 34:625–634
21. Butte AJ and Kohane IS (2000) Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. Pac Symp Biocomput 418–429
22. Margolin AA, Nemenman I, Basso K, et al (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics 7 Suppl 1:S7
23. Faith JJ, Hayete B, Thaden JT, et al (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS Biol 5:e8
24. Barabási A and Albert R (1999) Emergence of scaling in random networks. Science 286:509–512
25. Koschützki D and Schreiber F (2008) Centrality analysis methods for biological networks and their application to gene regulatory networks. Gene Regul Syst Bio 2:193–201
26. Liu Y, Gu H-Y, Zhu J, et al (2019) Identification of Hub Genes and Key Pathways Associated With Bipolar Disorder Based on Weighted Gene Co-expression Network Analysis. Front Physiol 10:1081
27. Yuan L, Chen L, Qian K, et al (2017) Co-expression network analysis identified six hub genes in association with progression and prognosis in human clear cell renal cell carcinoma (ccRCC). Genom Data 14:132–140
28. Eidsaa M, Stubbs L, and Almaas E (2017) Comparative analysis of weighted gene co-expression networks in human and mouse. PLoS ONE 12:e0187611

29. Lichtblau Y, Zimmermann K, Haldemann B, et al (2017) Comparative assessment of differential network analysis methods. Brief Bioinformatics 18:837–850
30. Jardim VC, Santos S de S, Fujita A, et al (2019) Bionetstat: A tool for biological networks differential analysis. Front Genet 10:594
31. Anglani R, Creanza TM, Liuzzi VC, et al (2014) Loss of connectivity in cancer co-expression networks. PLoS ONE 9:e87075
32. Milenković T and Pržulj N (2008) Uncovering biological network function via graphlet degree signatures. Cancer Inform 6:CIN.S680
33. Hayes W, Sun K, and Pržulj N (2013) Graphlet-based measures are suitable for biological network comparison. Bioinformatics 29:483–491
34. Przulj N (2007) Biological network comparison using graphlet degree distribution. Bioinformatics 23:e177-83
35. Nguyen T-M, Shafi A, Nguyen T, et al (2019) Identifying significantly impacted pathways: a comprehensive review and assessment. Genome Biol 20:203
36. Mubeen S, Hoyt CT, Gemünd A, et al (2019) The impact of pathway database choice on statistical enrichment analysis and predictive modeling. Front Genet 10:1203
37. Kanehisa M, Furumichi M, Tanabe M, et al (2017) KEGG: new perspectives on genomes, pathways, diseases and drugs. Nucleic Acids Res 45:D353–D361
38. Kanehisa M and Goto S (2000) KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res 28:27–30
39. Jassal B, Matthews L, Viteri G, et al (2020) The Reactome Pathway Knowledgebase. Nucleic Acids Res 48:D498–D503
40. Nishimura D (2001) BioCarta. Biotech Software & Internet Report 2:117–120
41. Thomas PD, Campbell MJ, Kejariwal A, et al (2003) PANTHER: a library of protein families and subfamilies indexed by function. Genome Res 13:2129–2141
42. Boyle EI, Weng S, Gollub J, et al (2004) GO::TermFinder--open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. Bioinformatics 20:3710–3715
43. Subramanian A, Tamayo P, Mootha VK, et al (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci USA 102:15545–15550
44. Kolmogorov A (1933) Sulla determinazione empirica di una lgge di distribuzione (On the empirical determination of a distribution law). 83–91
45. Ma J, Shojaie A, and Michailidis G (2019) A comparative study of topology-based pathway enrichment analysis methods. BMC Bioinformatics 20:546
46. Draghici S, Khatri P, Tarca AL, et al (2007) A systems biology approach for pathway level analysis. Genome Res 17:1537–1545
47. Tarca AL, Draghici S, Khatri P, et al (2009) A novel signaling pathway impact analysis. Bioinformatics 25:75–82
48. Sales G, Calura E, Cavalieri D, et al (2012) graphite - a Bioconductor package to convert

pathway topology to gene network. BMC Bioinformatics 13:20
49. Voichita C, Ansari S, and Draghici S (2019) ROntoTools: The R Onto-Tools suite.
50. Tarca AL, Draghici S, Bhatti G, et al (2012) Down-weighting overlapping genes improves gene set analysis. BMC Bioinformatics 13:136
51. Amar D, Safer H, and Shamir R (2013) Dissection of regulatory networks that are altered in disease via differential co-expression. PLoS Comput Biol 9:e1002955
52. Hu R, Qiu X, Glazko G, et al (2009) Detecting intergene correlation changes in microarray analysis: a new approach to gene selection. BMC Bioinformatics 10:20
53. Kostka D and Spang R (2004) Finding disease specific alterations in the co-expression of genes. Bioinformatics 20 Suppl 1:i194-9
54. Hudson NJ, Reverter A, and Dalrymple BP (2009) A differential wiring analysis of expression data correctly identifies the gene containing the causal mutation. PLoS Comput Biol 5:e1000382
55. Wang H, Li M, Wang J, et al (2011) A new method for identifying essential proteins based on edge clustering coefficient, In: Chen, J., Wang, J., and Zelikovsky, A. (eds.) Bioinformatics research and applications, pp. 87–98 Springer Berlin Heidelberg, Berlin, Heidelberg
56. Odibat O and Reddy CK (2011) Ranking differential genes in co-expression networks, In: Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine - BCB '11, pp. 350–354 ACM Press, New York, New York, USA
57. Bhuva DD, Cursons J, Smyth GK, et al (2019) Differential co-expression-based detection of conditional relationships in transcriptional data: comparative analysis and application to breast cancer. Genome Biol 20:236
58. Bockmayr M, Klauschen F, Györffy B, et al (2013) New network topology approaches reveal differential correlation patterns in breast cancer. BMC Syst Biol 7:78
59. Yu H, Liu B-H, Ye Z-Q, et al (2011) Link-based quantitative methods to identify differentially coexpressed genes and gene pairs. BMC Bioinformatics 12:315
60. Fuller TF, Ghazalpour A, Aten JE, et al (2007) Weighted gene coexpression network analysis strategies applied to mouse weight. Mamm Genome 18:463–472
61. Reverter A, Hudson NJ, Nagaraj SH, et al (2010) Regulatory impact factors: unraveling the transcriptional regulation of complex traits from expression data. Bioinformatics 26:896–904
62. Gonzalez-Valbuena E-E and Treviño V (2017) Metrics to estimate differential co-expression networks. BioData Min 10:32
63. Odibat O and Reddy CK (2012) Ranking differential hubs in gene co-expression networks. J Bioinform Comput Biol 10:1240002
64. Langfelder P and Horvath S (2008) WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics 9:559
65. Tesson BM, Breitling R, and Jansen RC (2010) DiffCoEx: a simple and sensitive method to find differentially coexpressed gene modules. BMC Bioinformatics 11:497

66. Ha MJ, Baladandayuthapani V, and Do K-A (2015) DINGO: differential network analysis in genomics. Bioinformatics 31:3413–3420
67. Choi Y and Kendziorski C (2009) Statistical methods for gene set co-expression analysis. Bioinformatics 25:2780–2786
68. Watson M (2006) CoXpress: differential co-expression in gene expression data. BMC Bioinformatics 7:509
69. Rahmatallah Y, Emmert-Streib F, and Glazko G (2014) Gene Sets Net Correlations Analysis (GSNCA): a multivariate differential coexpression test for gene sets. Bioinformatics 30:360–368
70. Santos S de S, Galatro TF de A, Watanabe RA, et al (2015) CoGA: An R Package to Identify Differentially Co-Expressed Gene Sets by Analyzing the Graph Spectra. PLoS ONE 10:e0135831
71. Cho SB, Kim J, and Kim JH (2009) Identifying set-wise differential co-expression in gene expression microarray data. BMC Bioinformatics 10:109
72. Fukushima A (2013) DiffCorr: an R package to analyze and visualize differential correlations in biological networks. Gene 518:209–214
73. Pierson E, GTEx Consortium, Koller D, et al (2015) Sharing and Specificity of Co-expression Networks across 35 Human Tissues. PLoS Comput Biol 11:e1004220
74. Melé M, Ferreira PG, Reverter F, et al (2015) The human transcriptome across tissues and individuals. Science 348:660–665
75. Serra A, Letunic I, Fortino V, et al (2019) INSIdE NANO: a systems biology framework to contextualize the mechanism-of-action of engineered nanomaterials. Sci Rep 9:179
76. Wang B, Mezlini AM, Demir F, et al (2014) Similarity network fusion for aggregating data types on a genomic scale. Nat Methods 11:333–337
77. Chierici M, Bussola N, Marcolini A, et al (2020) Integrative Network Fusion: A Multi-Omics Approach in Molecular Profiling. Front Oncol 10:1065
78. Ma T and Zhang A (2017) Integrate multi-omic data using affinity network fusion (ANF) for cancer patient clustering, In: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 398–403 IEEE
79. Han H, Shim H, Shin D, et al (2015) TRRUST: a reference database of human transcriptional regulatory interactions. Sci Rep 5:11432
80. Han H, Cho J-W, Lee S, et al (2018) TRRUST v2: an expanded reference database of human and mouse transcriptional regulatory interactions. Nucleic Acids Res 46:D380–D386
81. Fornes O, Castro-Mondragon JA, Khan A, et al (2020) JASPAR 2020: update of the open-access database of transcription factor binding profiles. Nucleic Acids Res 48:D87–D92
82. Sandelin A, Alkema W, Engström P, et al (2004) JASPAR: an open-access database for eukaryotic transcription factor binding profiles. Nucleic Acids Res 32:D91-4
83. Davis AP, Grondin CJ, Johnson RJ, et al (2021) Comparative Toxicogenomics Database

(CTD): update 2021. Nucleic Acids Res 49:D1138–D1143
84. Wishart DS, Knox C, Guo AC, et al (2006) DrugBank: a comprehensive resource for in silico drug discovery and exploration. Nucleic Acids Res 34:D668-72
85. Pavel A, Giudice G del, Federico A, et al (2021) Integrated network analysis reveals new genes suggesting COVID-19 chronic effects and treatment. Brief Bioinformatics
86. Hawe JS, Theis FJ, and Heinig M (2019) Inferring Interaction Networks From Multi-Omics Data. Front Genet 10:535
87. Huang S, Chaudhary K, and Garmire LX (2017) More Is Better: Recent Progress in Multi-Omics Data Integration Methods. Front Genet 8:84
88. Dimitrakopoulos C, Hindupur SK, Häfliger L, et al (2018) Network-based integration of multi-omics data for prioritizing cancer genes. Bioinformatics 34:2441–2448
89. Bonnet E, Calzone L, and Michoel T (2015) Integrative multi-omics module network inference with Lemon-Tree. PLoS Comput Biol 11:e1003983
90. Friedman N, Linial M, Nachman I, et al (2000) Using Bayesian networks to analyze expression data. J Comput Biol 7:601–620
91. Opgen-Rhein R and Strimmer K (2006) Inferring gene dependency networks from genomic longitudinal data: a functional data approach. Revstat Stat J 4:53–65
92. Markowetz F and Spang R (2007) Inferring cellular networks--a review. BMC Bioinformatics 8 Suppl 6:S5
93. Murphy K (1999) The bayes net toolbox for matlab.
94. Scutari M (2009) Learning Bayesian Networks with the bnlearn R Package . arXiv:0908.3817
95. Grzegorczyk M (2016) A non-homogeneous dynamic Bayesian network with a hidden Markov model dependency structure among the temporal data points. Mach Learn 102:155–207
96. Glymour C, Zhang K, and Spirtes P (2019) Review of causal discovery methods based on graphical models. Front Genet 10:524
97. Freeman LC (1978) Centrality in social networks conceptual clarification. Soc Networks 1:215–239
98. Brandes U (2001) A faster algorithm for betweenness centrality*. J Math Sociol 25:163–177
99. Bonacich P (1987) Power and Centrality: A Family of Measures. Am J Sociol 92:1170
100. Katz L (1953) A new status index derived from sociometric analysis. Psychometrika 18:39–43
101. Junker BH, Koschützki D, and Schreiber F (2006) Exploration of biological network centralities with CentiBiN. BMC Bioinformatics 7:219
102. Piraveenan M, Prokopenko M, and Hossain L (2013) Percolation centrality: quantifying graph-theoretic impact of nodes during percolation in networks. PLoS ONE 8:e53095
103. Saramäki J, Kivelä M, Onnela J-P, et al (2007) Generalizations of the clustering coefficient to weighted complex networks. Phys Rev E Stat Nonlin Soft Matter Phys

75:027105
104. Kaiser M (2008) Mean clustering coefficients: the role of isolated nodes and leafs on clustering measures for small-world networks. New J Phys 10:083042
105. Zhan FB and Noon CE (1998) Shortest path algorithms: an evaluation using real road networks. Transportation Science 32:65–73
106. Dijkstra EW (1959) A note on two problems in connexion with graphs. Numer Math 1:269–271
107. Aingworth D, Chekuri C, Indyk P, et al (1999) Fast estimation of diameter and shortest paths (without matrix multiplication). SIAM J Comput 28:1167–1181
108. Milenković T, Ng WL, Hayes W, et al (2010) Optimal network alignment with graphlet degree vectors. Cancer Inform 9:121–137
109. Paton K (1969) An algorithm for finding a fundamental set of cycles of a graph. Commun ACM 12:514–518
110. Blondel VD, Guillaume J-L, Lambiotte R, et al (2008) Fast unfolding of communities in large networks. J Stat Mech 2008:P10008
111. Clauset A, Newman MEJ, and Moore C (2004) Finding community structure in very large networks. Phys Rev E 70
112. Traag VA, Waltman L, and Eck NJ van (2019) From Louvain to Leiden: guaranteeing well-connected communities. Sci Rep 9:5233
113. Pons P and Latapy M (2005) Computing communities in large networks using random walks, In: Yolum, pInar, Güngör, T., Gürgen, F., et al (eds.) Computer and Information Sciences - ISCIS 2005, pp. 284–293 Springer Berlin Heidelberg, Berlin, Heidelberg
114. Girvan M and Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci USA 99:7821–7826
115. Raghavan UN, Albert R, and Kumara S (2007) Near linear time algorithm to detect community structures in large-scale networks. Phys Rev E 76
116. Enright AJ, Van Dongen S, and Ouzounis CA (2002) An efficient algorithm for large-scale detection of protein families. Nucleic Acids Res 30:1575–1584
117. Rossetti G (2020) Exorcising the Demon: Angel, Efficient Node-Centric Community Discovery, In: Cherifi, H., Gaito, S., Mendes, J.F., et al (eds.) Complex networks and their applications VIII, pp. 152–163 Springer International Publishing, Cham
118. Whang JJ, Gleich DF, and Dhillon IS (2013) Overlapping community detection using seed set expansion, In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13, pp. 2099–2108 ACM Press, New York, New York, USA
119. Gregory S (2008) A fast algorithm to find overlapping communities in networks, In: Daelemans, W., Goethals, B., and Morik, K. (eds.) Machine learning and knowledge discovery in databases, pp. 408–423 Springer Berlin Heidelberg, Berlin, Heidelberg
120. Yang J and Leskovec J (2013) Overlapping community detection at scale, In: Proceedings of the sixth ACM international conference on Web search and data mining -

WSDM ’13, p. 587 ACM Press, New York, New York, USA
121. Kundu S and Pal SK (2015) Fuzzy-rough community in social networks. Pattern Recognit Lett 67:145–152
122. Ahn Y-Y, Bagrow JP, and Lehmann S (2010) Link communities reveal multiscale complexity in networks. Nature 466:761–764
123. Radicchi F, Castellano C, Cecconi F, et al (2004) Defining and identifying communities in networks. Proc Natl Acad Sci USA 101:2658–2663
124. Shi J and Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22:888–905
125. Flake GW, Lawrence S, and Giles CL (2000) Efficient identification of Web communities, In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’00, pp. 150–160 ACM Press, New York, New York, USA
126. Newman MEJ and Girvan M (2004) Finding and evaluating community structure in networks. Phys Rev E Stat Nonlin Soft Matter Phys 69:026113
127. Fortunato S (2010) Community detection in graphs. Physics Reports 486:75–174

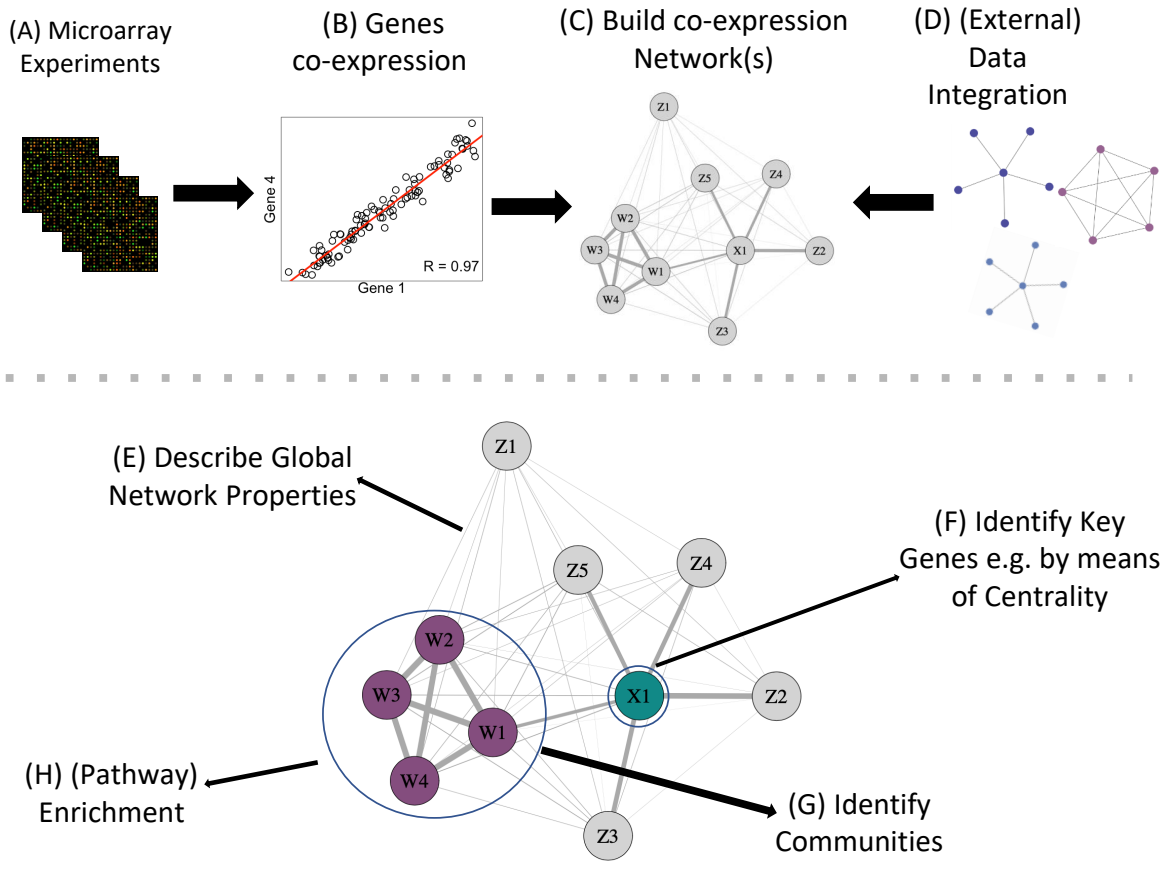

Figure 1: Example of co-expression network analysis. Starting from microarray experiments (A) the gene co-expression values (B) can be computed by means of correlation or mutual information metrics. From gene co-expression values, the co-expression network(s) can be computed by means of multiple algorithms (C). External data can be integrated during the co-expression network creation in order to obtain more robust and reliable results (D). Once the final network is obtained, global measures can be computed to evaluate network properties (E). Moreover local centrality measures can be used to identify key genes in the network(s) such as for example hub genes (F).Community detection algorithms can be used to identify groups of genes with strong correlation patterns (G) and pathways enrichment analysis can be used to functionally characterize the genes in the communities (H).

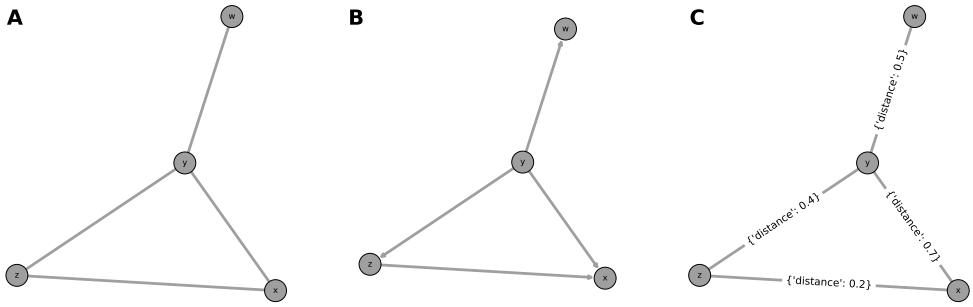

Figure 2: An undirected unweighted graph (A), directed unweighted graph (B) and un undirected weighted graph (C).

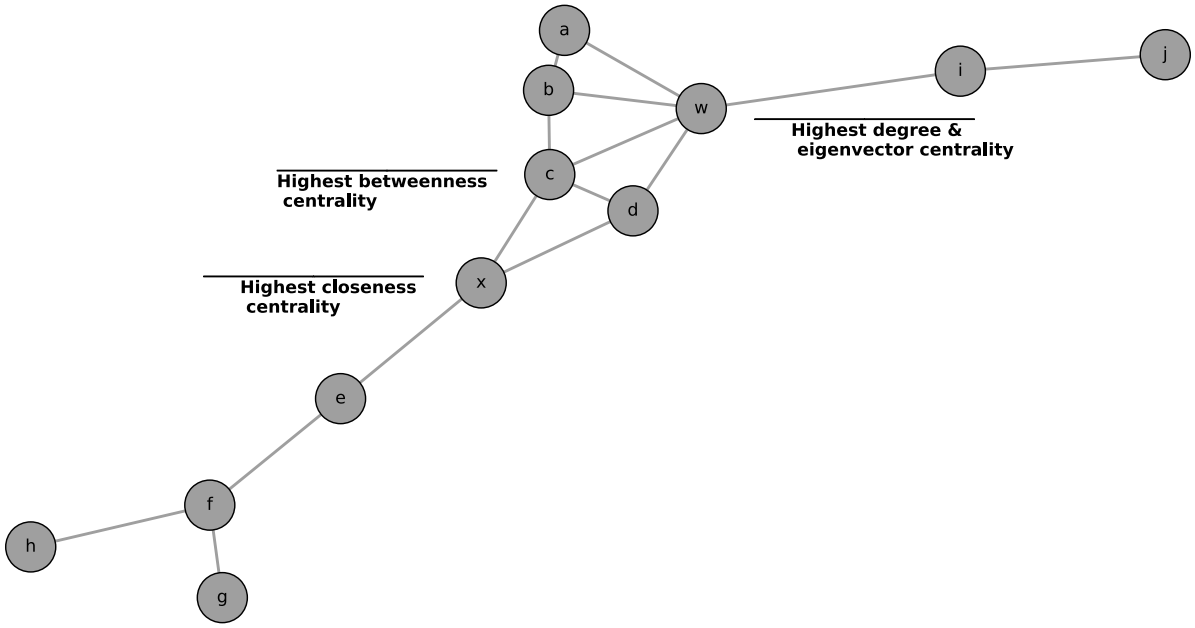

Figure 3: Showing how different centrality measures identify different “high importance” nodes in a network. Node c has the highest betweenness centrality, node x the highest closeness centrality and node w has the highest degree and eigenvector centrality.

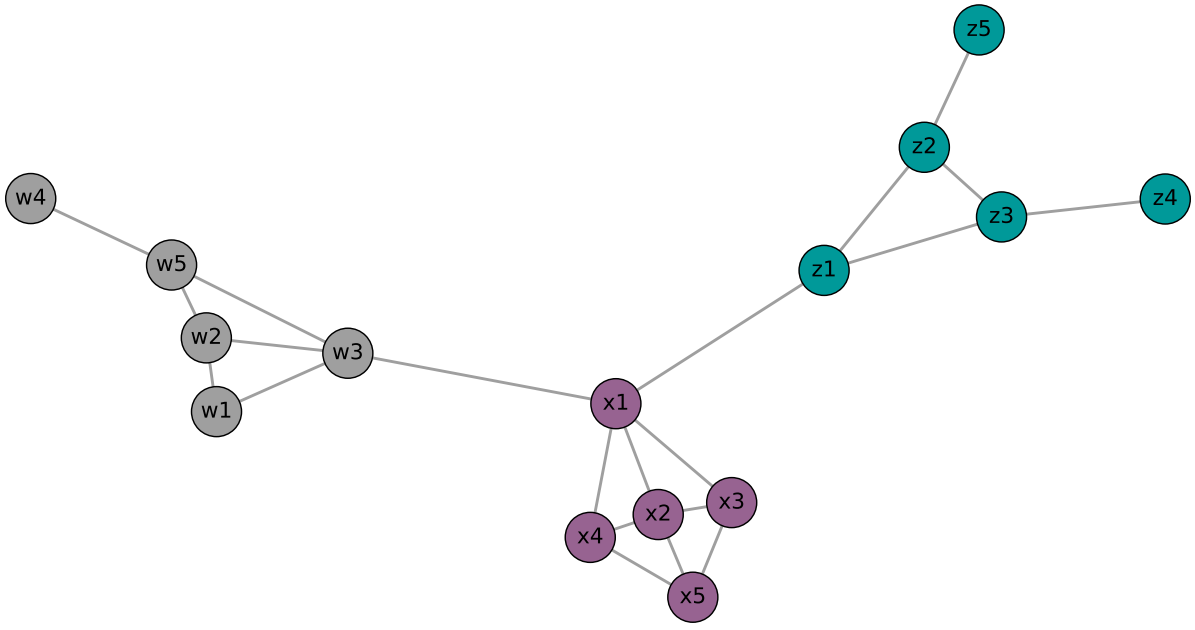

Figure 4: A graph with three tightly connected structures (communities): w, z and x.

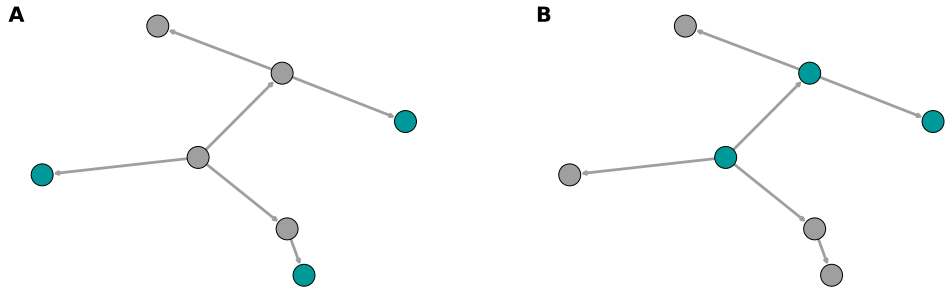

Figure 5: In topological pathway analysis the position of the differentially expressed (DE) genes in a pathway graph is taken into account. DE genes placed on different paths (on the left) have less impact than DE genes on the same path (on the right).

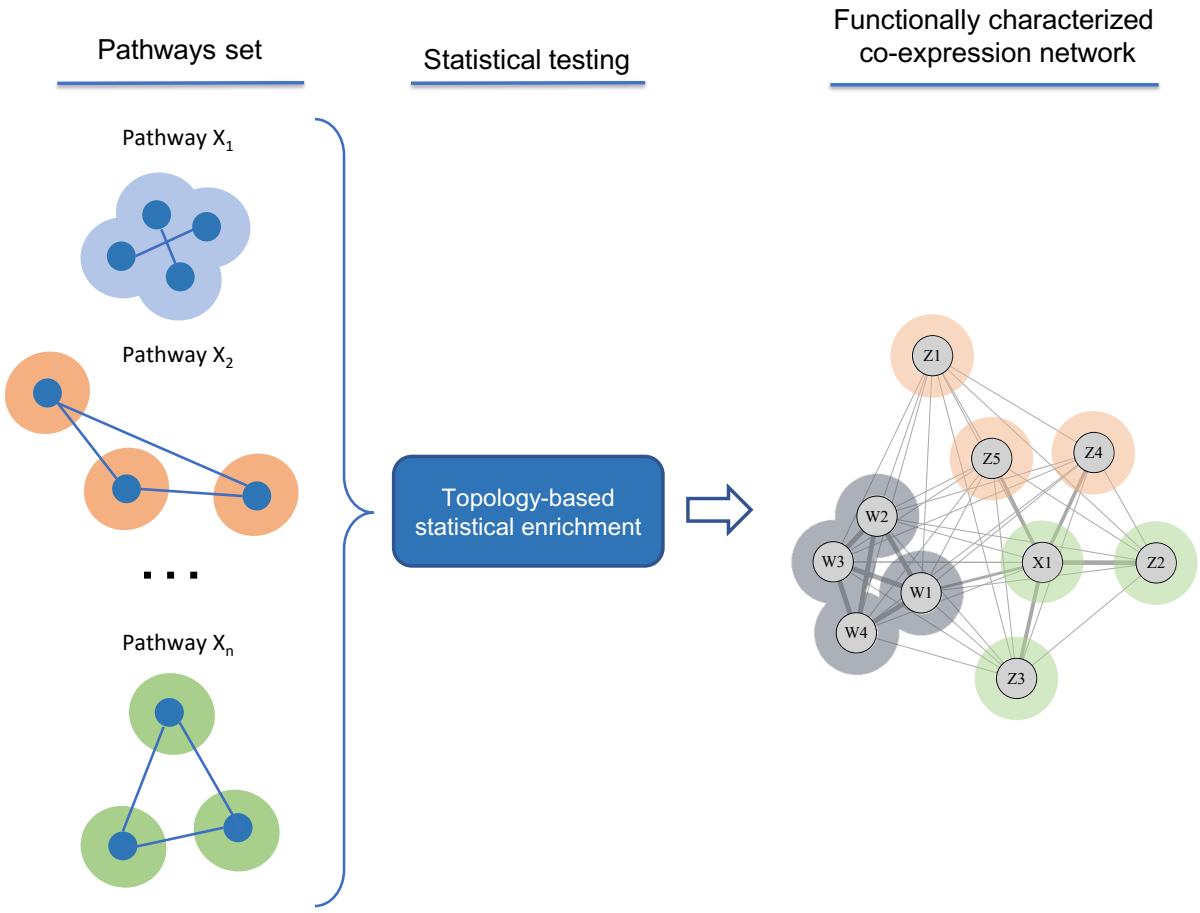

Figure 6: A generic outline of a topological pathway analysis. A co-expression network is functionally characterized by assessing the statistical enrichment of its connections over a set of pathways.

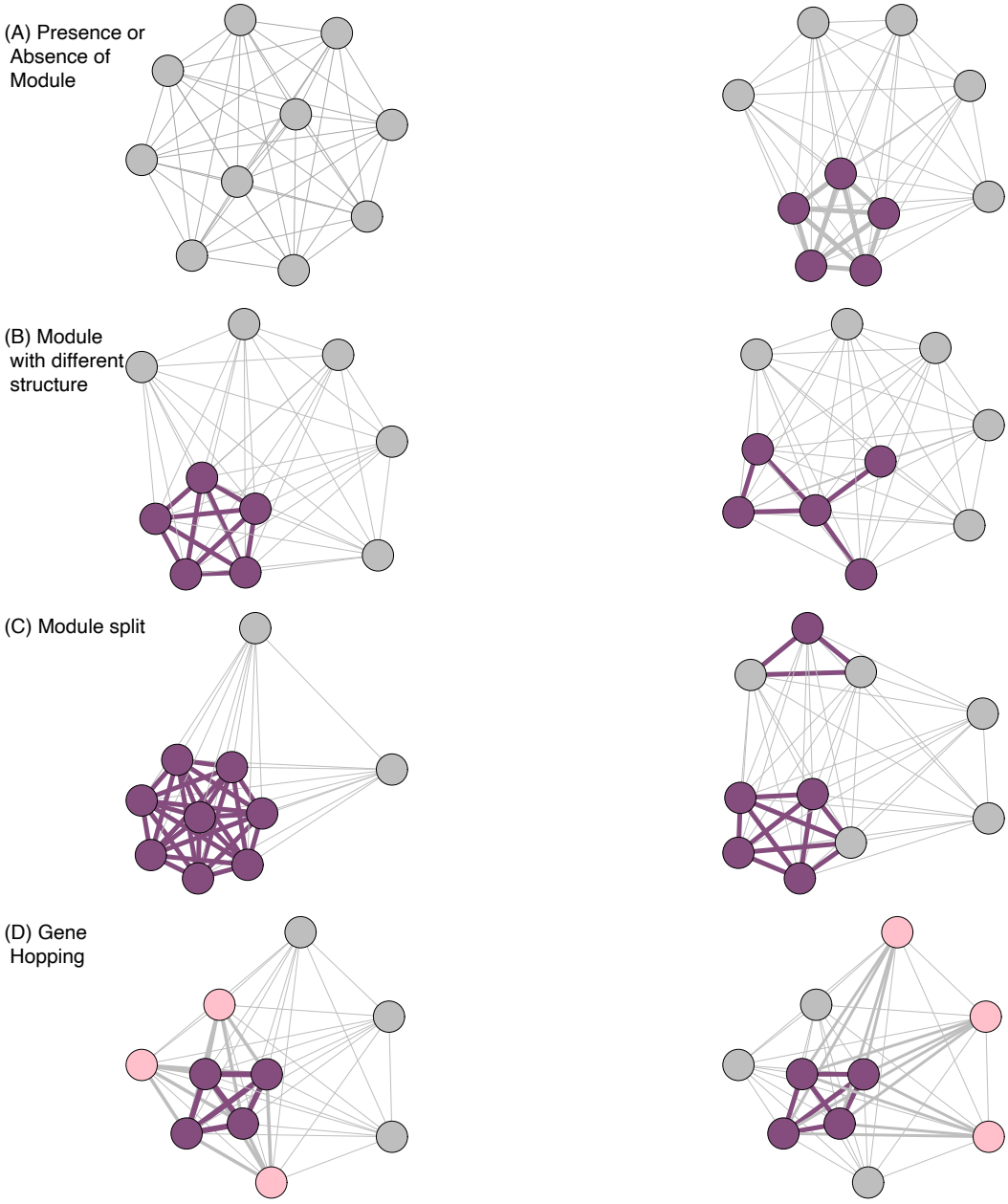

Figure 7. Module based differential co-expression analysis. A different co-expression pattern can happen because a module is present or not in two co-expression networks (A); A different co-expression pattern can happen because a module of genes is present in both networks but with

different structures (B); A different co-expression pattern can happen because a module of genes in a network is splitted in two modules in another network (C); A different co-expression pattern can happen because a module of genes is tightly connected with a set of genes in a network, while it changes genes connection in another network (D).

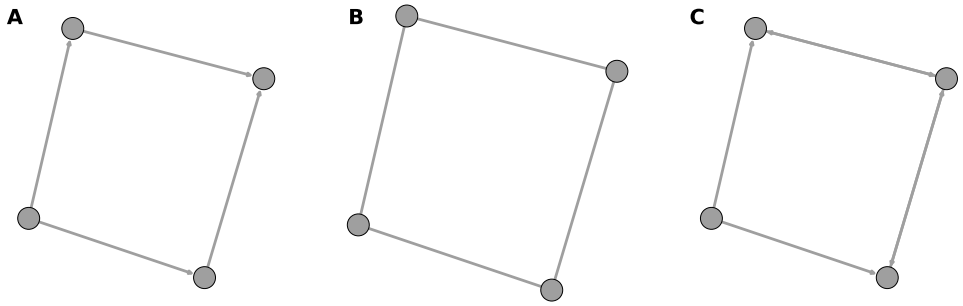

Figure 8: Showing a Bayesian network (A), Markov network (B) and dependency network (C).

Table 1: Different local network metrics.

Type	Measures	Meaning
Connectivity of a node/ all nodes	● Degree ● Degree Distribution ● Strength	● Number of a nodes incident edges ● Distribution of all nodes in a graph's degree ● Sum of the weights of a nodes incident edges
Centrality - a nodes position in the network w.r.t. all other nodes	● Closeness Centrality ● Betweenness Centrality ● Eigenvector Centrality	● How close (how many steps/ weighted paths) is a node to any other node in the network (that can be reached) - average length of the shortest paths between a node and all other nodes ***(97)*** ● How important is a node for information flow in the network - quantifies how often a node lies on a shortest path between two other nodes ***(97, 98)*** ● How influential is a node on the network - quantifies to how many "important" nodes a node is connected to ***(99)***

	● Katz Centrality ● Cross Clique Centrality ● Percolation Centrality	● How influential is a node on the network - generalization of eigenvector centrality, takes into account immediate neighbors and all other nodes that can be reached from a node ***(100, 101)*** ● How important is a node for information propagation - Estimates to how many cliques (a subgraph of the network where all nodes are connected to each other; the subgraph is a complete graph) a node belongs ***(101)*** ● How important is a node for "information flow" over time - quantifies how many percolated paths go through a node, can be used to model infection spreading in a network; when all prelocation values are the same then percolation centrality = betweenness centrality holds ***(102)***

Table 2: Global network metrics.

Type	Measure	Meaning
Connectivity	● Clustering coefficient ● Density ● Shortest Path Length Distribution	● Measures the degree to which nodes tend to cluster together. ***(103, 104)*** ● Quantifies how many of all possible edges exist in a network, it is 1 for a complete network and 0 for a network without any edges ● Distribution of all shortest paths in a network (how close any node is to any other node) ***(105, 106)***
Size	● Diameter ● Radius	● A networks diameter is the largest path among all longest shortest path between any two nodes that exists in a network (maximum eccentricity) ***(107)*** ● A networks radius is the smallest among all the longest shortest paths between any pair of nodes that exists in a network (minimum eccentricity)

Connectivity patterns	• Graphlet Distribution • Cycle Distribution	• Graphlets are small (often up to 5 nodes) subgraphs ***(32, 34, 108)*** • Quantifies the size of cycles that exist in a network ***(109)***

Table 3: Groups of community detection algorithms and some example algorithms.

Type	Algorithms
Node Clustering	
weighted	• Louvain ***(110)*** • Greedy modularity ***(111)*** • Leiden ***(112)***
unweighted	• Walktrap (non probabilistic random walks) ***(113)*** • Girvan Newman ***(114)*** • Label propagation ***(115)*** • Markov clustering ***(116)***
Overlapping	• Angel ***(117)*** • Seed set expansion ***(118)*** • CONGO/CONGA ***(119)*** • Big Clam ***(120)***
Fuzzy/ Probabilistic	• Fuzzy Rough Community Detection ***(121)***
Edge Clustering	• Hierarchical Link Clustering ***(122)***

Table 4: A collection of graph partitioning evaluation metrics.

Evaluation Metric	
Community Size Distribution	Are there strong differences in community size? Which distribution is to be prefered depends on the use case and type of network, but often the aim is an equal community size distribution.
Average Internal Degree	How tightly are nodes within a community connected. A high value indicates a tight knit community ***(123)***.
Internal Edge Density/ Density w.r.t.	How tightly are nodes within a community connected -

Graph density	this can also be estimated in comparison to the graph density. A high within community score indicates a tightly knit community structure ***(123)***.
Conductance	Fraction of edges leaving a community. A small value indicates that there are not many connections to other communities ***(124)***.
Fraction of weak members	How many nodes in a community have more outgoing than in-going edges? A small value indicates that members of a community are tightly knit with each other w.r.t. to the outside ***(125)***.
Modularity	Fraction of edges existing in a community w.r.t. to the expected number of edges. A high value indicates a tightly knit community ***(126)***.
Cut Ratio	Fraction of edges (of all possible edges) that leave a community. A small value indicates a more condensed community structure ***(127)***.
Average shortest path within a community/ w.r.t. Whole graph	How close are nodes within a community. A small value indicates a more tightly knit community structure.
Average Edge Weight (weighted graph)/ w.r.t. the whole graph	Are “stronger” connected nodes clustered together? This measurement is for weighted graphs. Either a small or high value may be preferred (indicating that strongly connected nodes are clustered together) but this depends on the type of edge weight being used.
Average Clustering Coefficient in a Community/ w.r.t. the whole graph	How tight are nodes connected? A high value indicates tightly connected community structures w.r.t. to the whole graph structure.
Hub Dominance	Is a community based around a hub node? A high value indicates this.
Node Embeddedness	Node Degree within a community w.r.t. To its overall degree. A high value indicates a strongly connected community structure.