Analysis and Visualization of Biological Networks with Cytoscape

John "Scooter" Morris, Ph.D., UCSF (scooter@cgl.ucsf.edu)
Allan Kuchinsky, Agilent (allan_kuchinsky@agilent.com)
Alex Pico, Ph.D., Gladstone Institutes (apico@gladstone.ucsf.edu)

Table of Contents

Revised 09/17/2014

Overview

Networks have long been used to represent important biological processes. Many of us remember memorizing the Krebs (TCA) cycle, which is usually shown as a directed graph, itself a type of network (Figure 1). Recently, however, the use of networks in biology has changed from purely illustrative and didactic to more analytic, even including hypothesis formulation. This shift has resulted, in part, from the confluence of advances in computation, informatics, and high-throughput techniques in systems biology. Today the analysis and visualization of biologically relevant networks has become commonplace, whether the networks represent metabolic, regulatory, or signaling pathways; protein-protein or genetic interactions; or more abstract connections between similar proteins or similar ligands. Networks are now routinely used to show relationships between biologically relevant molecules, and analysis of those networks is proving valuable for helping us understand those relationships and formulate hypotheses about biological function.

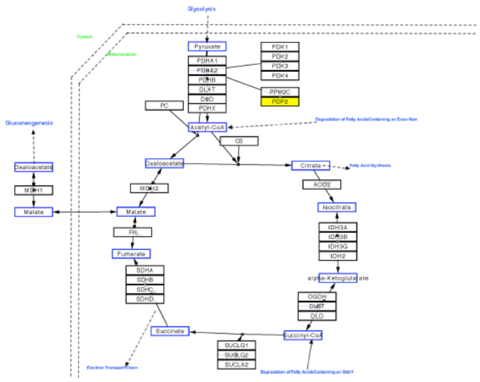
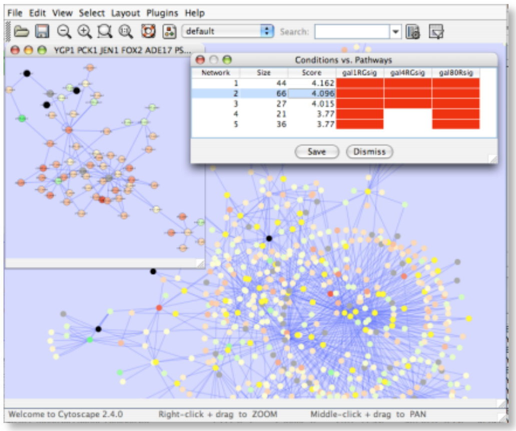


Figure 1. The TCA cycle from WikiPathways

With the advent of high-throughput methods that generate vast amounts of data from diverse measurement sources – for example gene expression data from microarrays, protein or metabolite abundance from mass spectrometry – biological networks have become increasingly important as an integrating context for data. As a commonly understood diagrammatic representation for concepts and relationships, networks provide structure that helps reduce underlying complexity of the data. Network tools give us functionality for studying complex processes. We can analyze global characteristics of the data, via metrics such as degree, clustering coefficient, shortest paths, centrality, density. We can identify key elements (hubs) and 'interesting' subnets, which can help us to elucidate mechanisms of interaction. Also, visualization of data superimposed upon the network can help us understand how a process is modulated or attenuated by a stimulus.

Network tools have proven to be extremely useful in analyzing and visualizing important biological processes. Some general applications of networks in biology include:

- **Gene Function Prediction** – Examining genes (proteins) in a network context shows connections to sets of genes/proteins involved in same biological process that are likely to function in that process [1-4].



Conditions vs. Pathways

Network	Size	Score	gal1RGsig	gal4RGsig	gal80Rsig
1	44	4.162	🟥	🟥	🟥
2	66	4.096	🟥	🟥	🟥
3	27	4.015	🟥	🟥	🟥
4	21	3.77	🟥		🟥
5	36	3.77	🟥		🟥

Welcome to Cytoscape 2.4.0 Right-click + drag to ZOOM Middle-click + drag to PAN

Figure 2. Gene Function Prediction using jActiveModules

- **Detection of protein complexes/other modular structures** – although interaction networks are based on pair-wise interactions, there is clear evidence for modularity & higher order organization (motifs, feedback loops) [5-9]

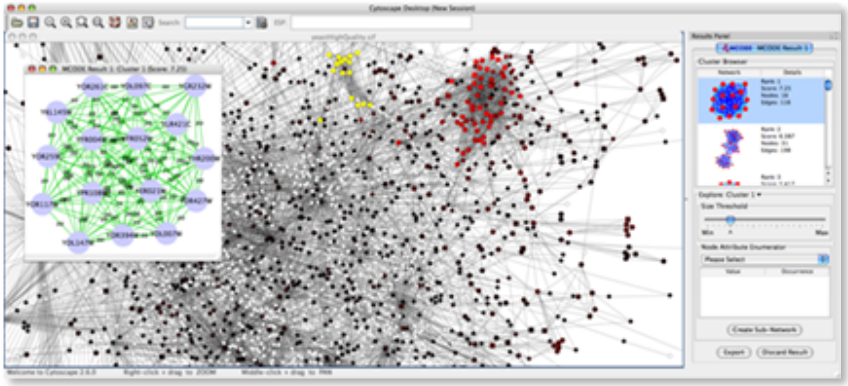

Cytoscape Desktop (New Session)
MCODE Result 1: Cluster 1 (Score: 7.25)
Results Panel
MCODE Result 1
Cluster Browser
Network
Details
Explore: Cluster 1
Size Threshold
Node Attribute Enumerator
Please Select
Value
Occurrence
Create Sub-Network
Export
Discard Result
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN

Figure 3. Identifying molecular complexes in large protein interaction networks using MCODE

- **Prediction of new interactions and functional associations** – There are several methods for predicting interactions and functional associations, based upon network structure and correlations amongst data. For example, orthology-based methods have been used to predict interactions for a species based upon orthology to interacting pairs of proteins in evolutionarily similar organisms[10]. Other researchers have used Bayesian network approaches to inferring gene regulatory networks from time course gene expression data[11]. In another approach, shown on the example below, statistically significant domain-domain correlations in protein interaction network suggest that certain domain (and domain pairs) mediate protein binding. Machine learning extends this to predict protein-protein or genetic interaction through integration of diverse types of evidence for interaction [12-14].

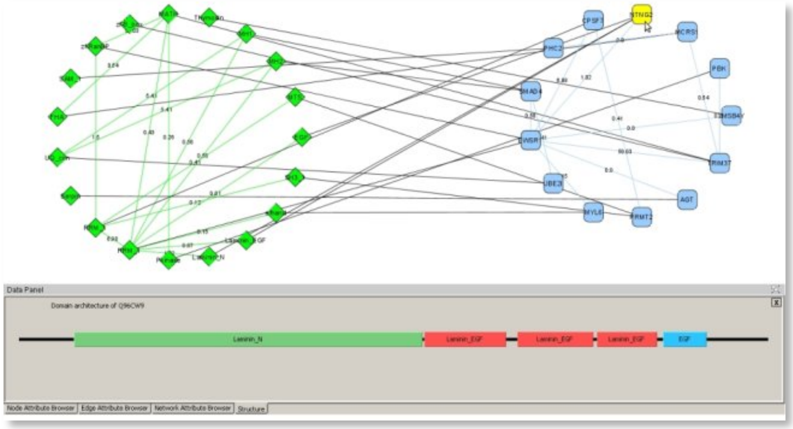


Figure 4. Visualizing domain interactions and alternative splicing using DomainGraph

Moreover, these same tools and their associated analysis and visualization methods can provide key insights in the study of disease and in drug development. These include:

- **Identification of disease subnetworks** – identification of disease network subnetworks that are transcriptionally active in disease. These suggest key pathway components in disease progression and provide leads for further study and potential therapeutic targets [15-20].

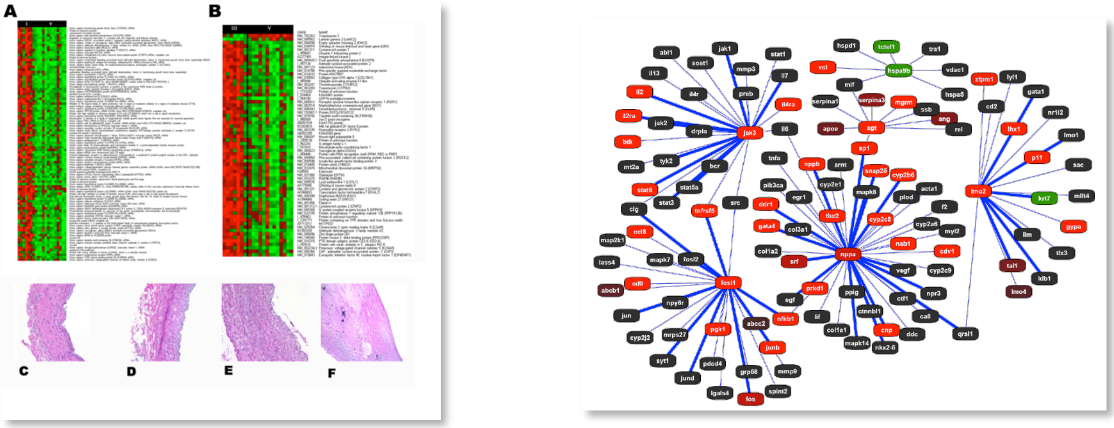


Figure 5. Gene expression profiles and American Heart Association (AHA) histological classification of atherosclerotic lesions (left panel).   Differentiation scores were calculated for all genes across pairwise conditions (e.g. diabetic vs. non-diabetic patients).   A large literature network was built for atherosclerosis.   Connectivity analysis was used to extract a transcriptionally-active subnetwork for diabetic vs. non-diabetic conditions (right panel).

- **Subnetwork-based diagnosis** – subnetworks also provide a rich source of biomarkers for disease classification, based on mRNA profiling integrated with protein networks to identify subnetwork biomarkers (interconnected genes whose aggregate expression levels are predictive of disease state[21, 22]).

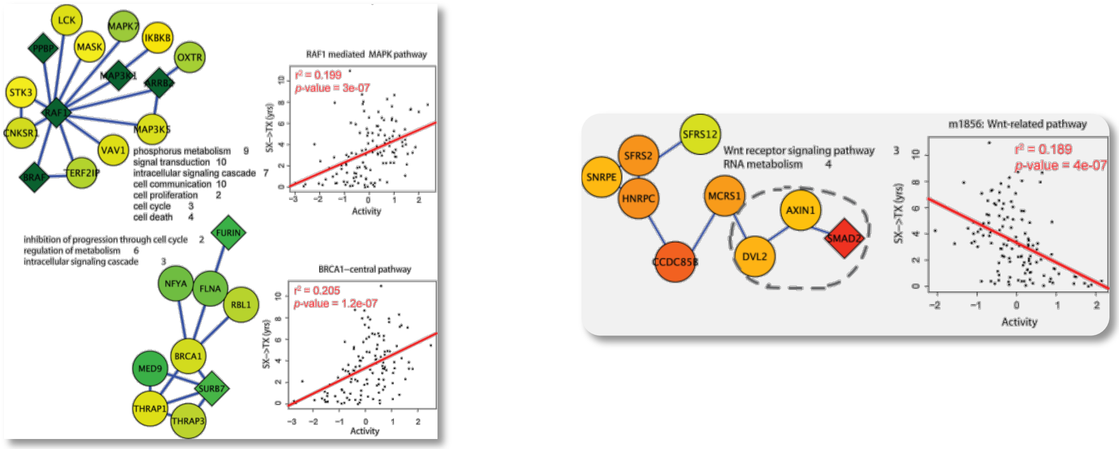
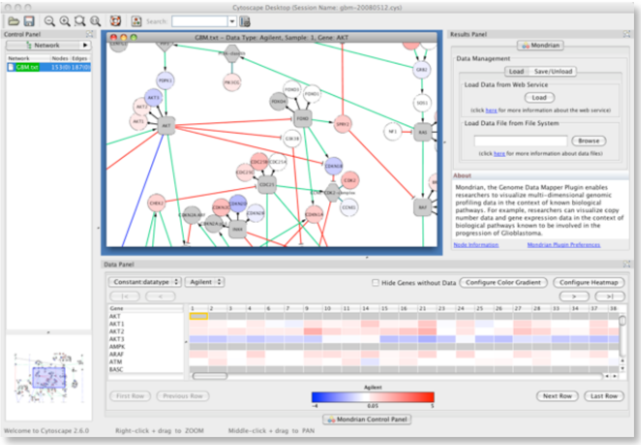


Figure 6. A network-based approach identified prognostic markers not as individual genes but as subnetworks extracted from molecular interaction databases.   Gene expression profiles from Chronic Lymphocytoic Leukemia patients were mapped to a large human molecular interaction network.   A search over this network was performed to identify prognostic subnetworks that could be used to predict treatment-free survival.

- **Subnetwork-based gene association** – molecular networks will provide a powerful framework for mapping common pathway mechanisms affected by collection of genotypes[23, 24].


Cytoscape Desktop (Session Name: gbm-20080512.cys)

GBM.txt – Data Type: Agilent, Sample: 1, Gene: AKT

Mondrian, the Genome Data Mapper Plugin enables researchers to visualize multi-dimensional genomic profiling data in the context of known biological pathways. For example, researchers can visualize copy number data and gene expression data in the context of biological pathways known to be involved in the progression of Glioblastoma.

Welcome to Cytoscape 2.6.0 Right-click + drag to ZOOM Middle-click + drag to PAN

Figure 7. Cytoscape Mondrian plugin with a dataset derived from the TCGA Glioblastoma Pilot Project. This dataset contains mutations, copy-number alterations, and expression data for 91 samples.

For the purposes of this tutorial, we will classify biological networks into three major categories: pathways, similarity networks, and interaction networks. Pathways include metabolic, regulatory, and signaling networks. Figure 2 shows a pathway containing genes involved in glioblastoma multiforme, a major form of brain cancer [25]. These genes were identified by a large-scale genetic analysis of copy number variation and genetic changes in 206 glioblastoma multiforme patients. The study was conducted as part of The Cancer Genome Atlas (TCGA) project. Notably, the study demonstrated that there was no single genetic defect responsible for glioblastoma multiforme, but that all of the cases showed significant pathway changes – strongly suggesting that this form of cancer is a "pathway disease." From a visualization standpoint, the real power is the ability to map expression, mutation, or copy

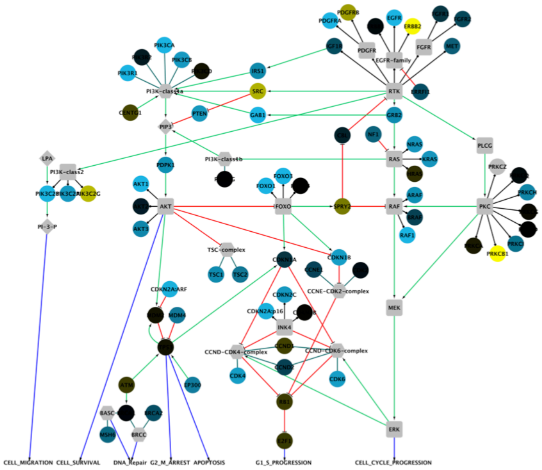


Figure 8. Partial pathway showing genes implicated in glioblastoma multiforme colored by the changes in copy number

number variation data onto pathways to reveal (or suggest) how the pathway and its components function under different sets of conditions, including disease states. Thus, the ability to analyze a variety of data sources and types and to map that data onto pathways is crucial. There are also techniques for deriving putative pathways from expression data[1] and for modeling the kinetics of biological processes [26] that are beyond the scope of this talk.

Interaction networks comprise the second category. In these networks, nodes represent biological entities and edges represent some form of interaction or relationship. A common example of this type is a protein-protein interaction (PPI) network. Figure 3 shows a yeast protein-protein interaction network generated by tandem affinity purification followed by mass spectrometry (TAP/MS) [27]. Analogous networks have been generated based on ligand similarities [28], protein similarities [29], and drug-target networks [30]. Generally, this class of biological networks can present as a “hair ball”, where there is so much information that the meaningful relationships are difficult to discern. There is good evidence that analysis of a PPI network to find highly connected “hubs” can be used to predict protein complexes [8], and clustering of protein similarity networks can provide clues to protein family (and hence functional) assignments (Figure 4).

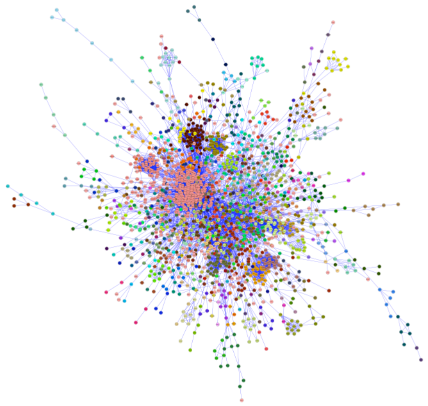


Figure 9. Partial protein-protein interaction network for *Saccharomyses cerevisiea* colored by predicted complexes.

A variety of analytical techniques can help to elucidate interaction networks. Clustering methods such as MCL [31] have proven valuable, although several algorithms more specific to various types of interaction networks have also been developed (c.f.[5]. In addition to clustering, a variety of metrics can be applied to an interaction network or nodes within the network. The average density (node degree) of the network, average shortest-path distance, number of connected components, measures of centrality, and the extent to which the network fits a scale-free model are all useful descriptors for the analysis of an interaction network. Altering the layout and visual attributes of the network can also be helpful.

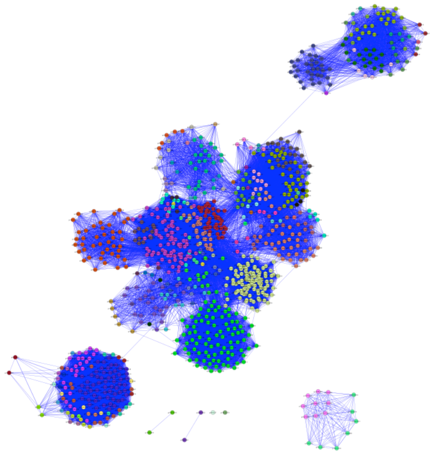


Figure 10. Protein similarity network of the amidohydrolase enzyme superfamily colored by subgroup.

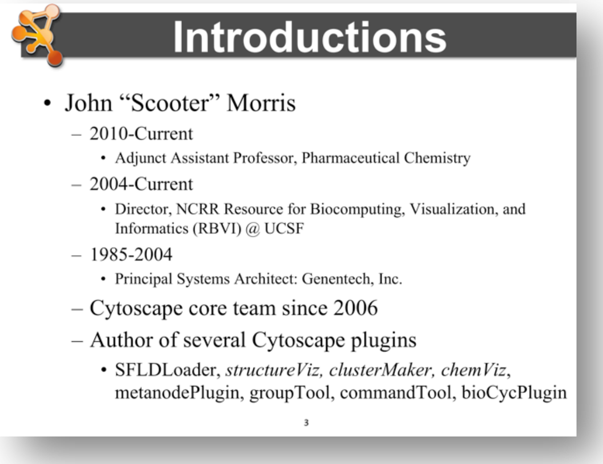
Cytoscape is an open-source application for the visualization and analysis of (biological) networks. During my talk, I will use Cytoscape to demonstrate some of the techniques for visualizing and analyzing biological networks. In addition, I will demonstrate some ways that biological networks can be combined with other data to help elucidate function or the possible implications of changes in biological function due to perturbation, mutation, or infection.

[1] c.f. the ExpressionCorrelation plugin from Gary Bader's lab: http://baderlab.org/Software/ExpressionCorrelation

Introductions and setup

Introductions

The three instructors that initiated this tutorial are experienced Cytoscape developers, with a cumulative of 15 years of participation in the Cytoscape core team. All three have a strong background in Cytoscape development, both from the perspective of core development, but also from the perspective of developing plugins that extend Cytoscape functionality. They also have a long history of working in the biomedical field, both from the perspective of tool developers and the underlying science.

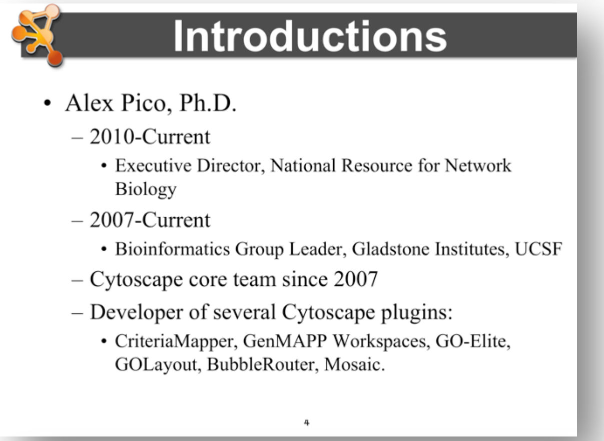


Introductions

- John “Scooter” Morris
  - 2010-Current
    - Adjunct Assistant Professor, Pharmaceutical Chemistry
  - 2004-Current
    - Director, NCRR Resource for Biocomputing, Visualization, and Informatics (RBVI) @ UCSF
  - 1985-2004
    - Principal Systems Architect: Genentech, Inc.
  - Cytoscape core team since 2006
  - Author of several Cytoscape plugins
    - SFLDLoader, *structureViz, clusterMaker, chemViz*, metanodePlugin, groupTool, commandTool, bioCycPlugin

3

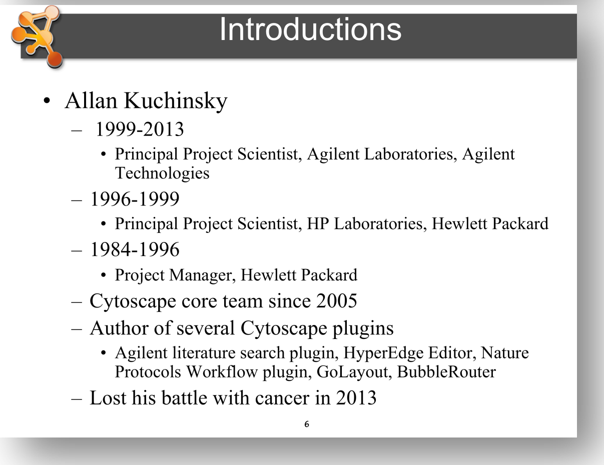
Notes



Introductions

- Alex Pico, Ph.D.
  - 2010-Current
    - Executive Director, National Resource for Network Biology
  - 2007-Current
    - Bioinformatics Group Leader, Gladstone Institutes, UCSF
  - Cytoscape core team since 2007
  - Developer of several Cytoscape plugins:
    - CriteriaMapper, GenMAPP Workspaces, GO-Elite, GOLayout, BubbleRouter, Mosaic.

4



Introductions

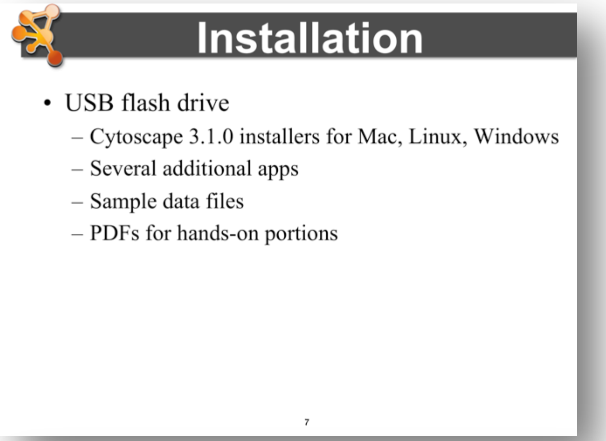
- Allan Kuchinsky
  - 1999-2013
    - Principal Project Scientist, Agilent Laboratories, Agilent Technologies
  - 1996-1999
    - Principal Project Scientist, HP Laboratories, Hewlett Packard
  - 1984-1996
    - Project Manager, Hewlett Packard
  - Cytoscape core team since 2005
  - Author of several Cytoscape plugins
    - Agilent literature search plugin, HyperEdge Editor, Nature Protocols Workflow plugin, GoLayout, BubbleRouter
  - Lost his battle with cancer in 2013

6

Setup

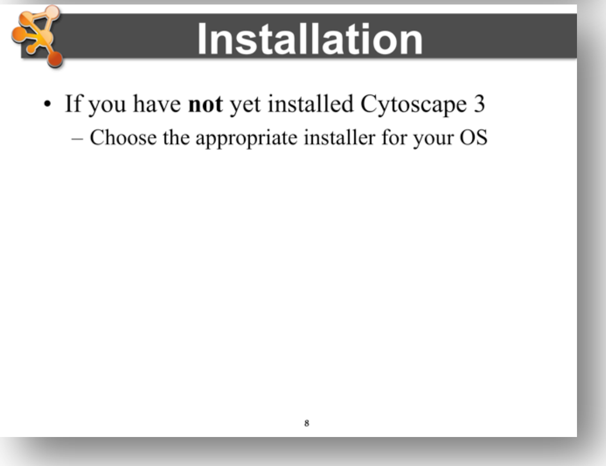
For the purposes of this tutorial, we will be using Cytoscape 3.1. 3.x is the whole new version of Cytoscape with modular architecture. It is designed for long-term maintainability and eventually it replaces 2.x series. New major features, including new user interfaces, headless (command-line) distribution, and multiple rendering engine support, will be released for this version. Cytoscape 3.1 is available as installers for Mac, Windows, and Linux, which include the core and sample files. Apps are generally available for download with Cytoscape's App Manager or from the App Store at http://apps.cytoscape.org/.

To avoid potential network problems or contention, we have provided all of the plugins that we will use for today on the CD that we've distributed.



Installation

- USB flash drive
  - Cytoscape 3.1.0 installers for Mac, Linux, Windows
  - Several additional apps
  - Sample data files
  - PDFs for hands-on portions



Installation

- If you have **not** yet installed Cytoscape 3
  - Choose the appropriate installer for your OS

Notes

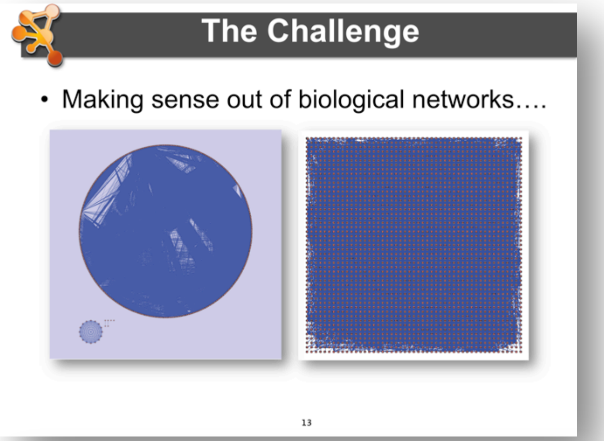
Biological Networks

In this section, we will begin to explore the use of networks in biology. We begin by posing a challenge: how do we make sense of biological networks? We pose that challenge by providing a series of pictorial examples of networks in biology.

The Challenge

The images on this slide are all representations of biological networks. The challenge we are faced with is to extract the "meaning" behind these representations, which may be a purely visual challenge, but it might also involve analytical approaches.

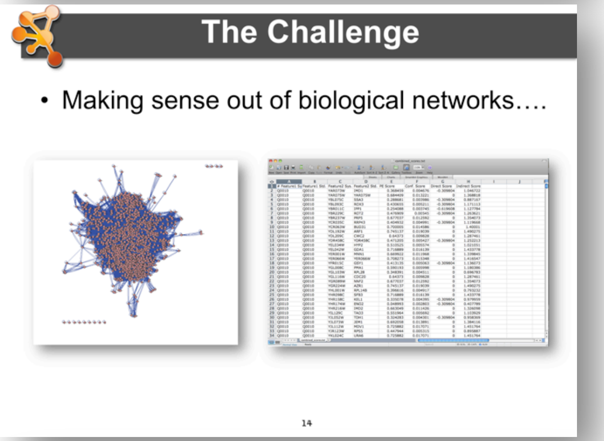
All of the images at the right represent biological networks, including the Excel spreadsheet. Without more information about the content, these images don't tell us much. How can we extract this meaning? What are the analytical techniques? What are the common visualization approaches?



The Challenge

- Making sense out of biological networks….

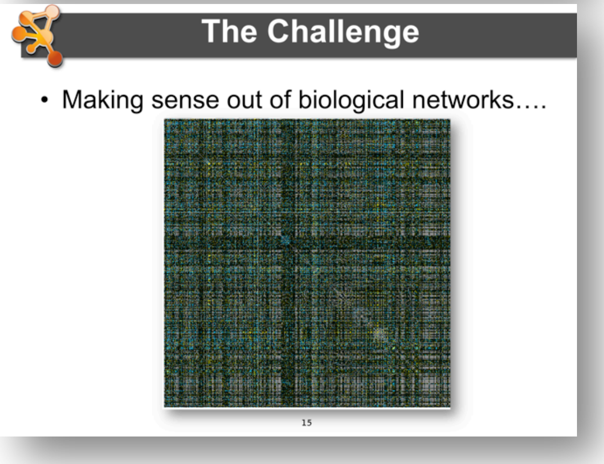
13



The Challenge

- Making sense out of biological networks….

14



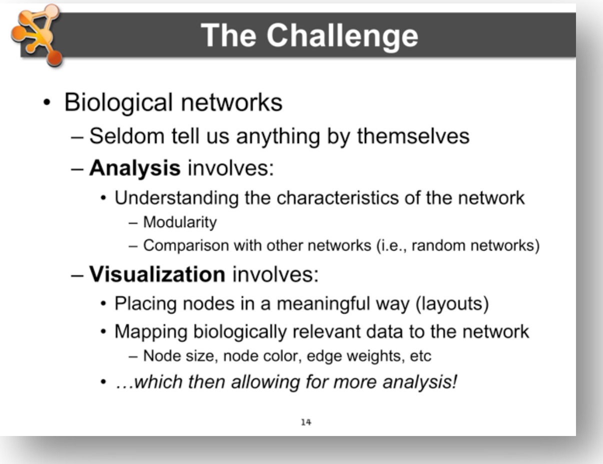
The Challenge

- Making sense out of biological networks….

15

Notes

If we simply think of a biological network as a list of nodes and the edges that connect them, we're not going to be able to gain much information. However, if we add information to those nodes and edges to that we can analyze the interactions (or similarities) in more depth, or we use that additional information to visualize the nodes in some meaningful manner, we will find it easier to gain (or communicate) insight about aspects of the network. There are a number of analytical and visualization approaches that can help us, which are described below.

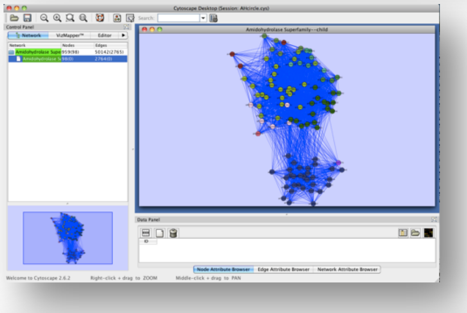


The Challenge

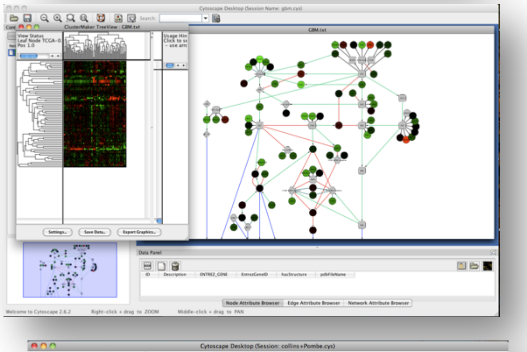
- Biological networks
  - Seldom tell us anything by themselves
  - **Analysis** involves:
    - Understanding the characteristics of the network
      - Modularity
      - Comparison with other networks (i.e., random networks)
  - **Visualization** involves:
    - Placing nodes in a meaningful way (layouts)
    - Mapping biologically relevant data to the network
      - Node size, node color, edge weights, etc
    - *…which then allowing for more analysis!*

14

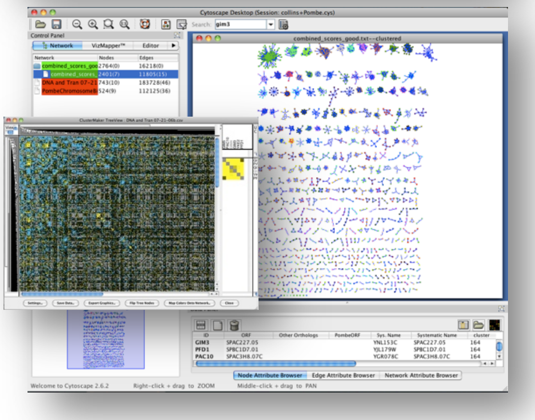
Taking the networks that we showed before, we can begin to analyze or visualize additional data. In the image at the right, we've colored the nodes in the network by protein family membership (members of protein families share functional characteristics), and then performed an edge-weighted layout where the edge weights represent the BLAST similarity between the proteins. As you can see pretty quickly that similar proteins tend to group together.



In this example, we've combined a network representation with an analysis of some of the associated data. The image at the left is a hierarchical clustering of all of the genes in the TCGA glioblastoma study vs. all of the patients in the study. This allows us to look for patterns in the heat map and associate those patterns with specific genes or groups of genes in the pathway.



In the final example on the right, we have combined two different visualizations with two different analyses. The heat map on the left represents a hierarchical cluster of genetic interactions and the network shows the results of an MCL cluster of a set of physical interactions. These views are linked, allowing users to select groups in one view and determine if the same groups exist in the other view. This allows researchers to explore areas where there are tight protein-protein physical interactions as well as genetic interactions, providing pretty strong evidence for the existence of a complex.

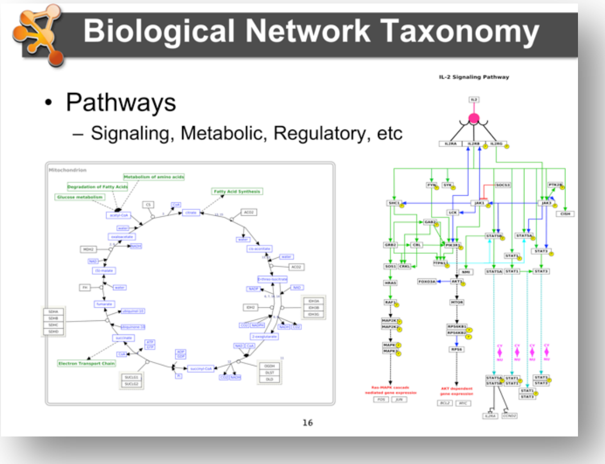

Cytoscape Desktop (Session: collins+Pombe.cys)
Control Panel
Network
VizMapper™
Editor
combined_scores_good.txt--clustered
Node Attribute Browser
Edge Attribute Browser
Network Attribute Browser

But, how do we know what kinds of analyses make sense, and what kinds of visualizations are appropriate?

Biological Network Taxonomy

Before tackling this question, we need to understand that not all "biological networks" are the same. In particular, there is a sort of taxonomy of networks – each visualization or analytical technique can be either more or less appropriate for the different network types. For our purposes, we can divide these biological networks into 3 main groups: pathways, interaction networks, and similarity networks.

Pathways



Biological Network Taxonomy

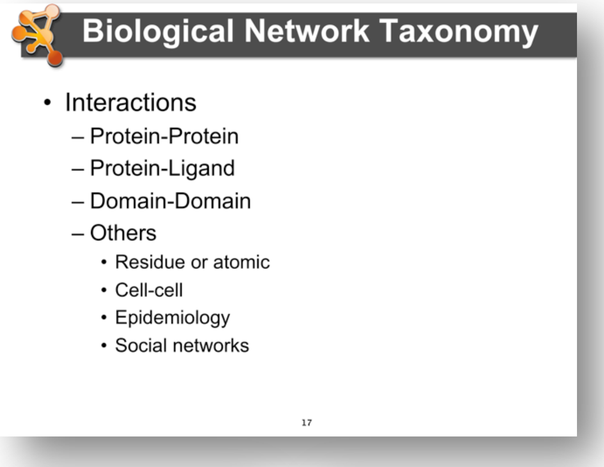
- Pathways
  - – Signaling, Metabolic, Regulatory, etc

16

The first type of network in our taxonomy is probably the most familiar. We've all seen pathway diagrams, whether those pathways represent signaling pathways, metabolic pathways, or regulatory pathways. These networks are often hand-curated diagrams that have been constructed for didactic purposes. However, even though the positions and graphical annotations associated with these networks does not lend them to the normal types of network analysis techniques, they can be extremely useful as templates on which we can paint expression profile information, or any other kinds of annotation that we want to use to show information associated with the curated pathway. Phylogenetic trees can also be thought of in a similar fashion of those trees have been hand curated like the kinase phylogenetic tree [32] shown in the slide.

Notes

Interactions

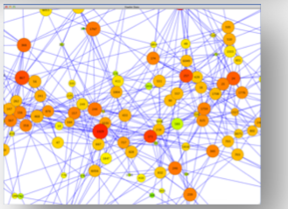


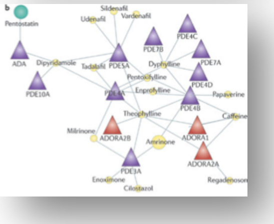
Biological Network Taxonomy

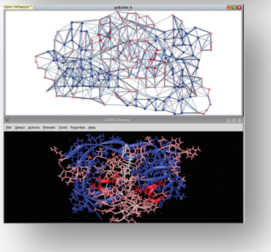
- Interactions
  - – Protein-Protein
  - – Protein-Ligand
  - – Domain-Domain
  - – Others
    - Residue or atomic
    - Cell-cell
    - Epidemiology
    - Social networks

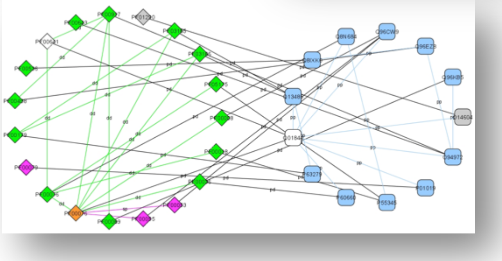
17

The second type of networks in our taxonomy are interaction networks. While pathways are probably familiar to most because of their use for educational purposes, interaction networks are what most people of when we think of "network biology". Basically, these networks reflect the interactions between biological entities. The entities might all be proteins, giving us the canonical protein-protein interaction network shown to the right in the first frame. The interacting entities also be genes, in which case, the network could be a genetic interaction network. The middle panel at the right shows a particular representation of an epistatic miniarray profile (EMAP). These networks are formed by recording the differential results of double-delete mutants when compared to the expected combination of single-delete mutants. The last network shows a protein-ligand interaction network. Interaction networks don't necessarily need to have only one interacting entity, and as we are rediscovering the importance of metabolic pathways, the "metabolome", which combines metabolites with the enzymes and regulatory proteins which control metabolism. There are also efforts underway to understand how the interactomes of pathogens interact with the interactomes of their hosts – yet another kind of "mixed" interaction network.









Of course, there are many kinds of biological interactions we might be interested in, up to and including how people interact with each other. Such social networks are beyond our scope, but social network analysis is very similar to biological networks analysis and provide a fruitful source of algorithms and visualization techniques.

Notes

____

____

____

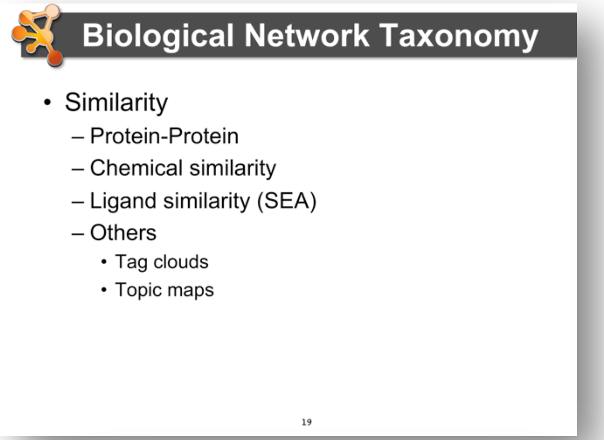
____

____

____

____

Similarity



Biological Network Taxonomy

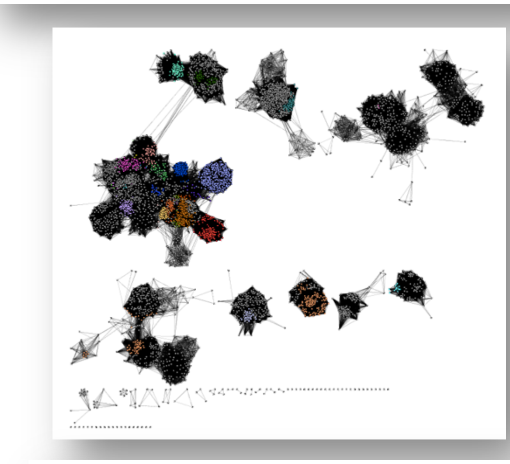
- Similarity
  - Protein-Protein
  - Chemical similarity
  - Ligand similarity (SEA)
  - Others
    - Tag clouds
    - Topic maps

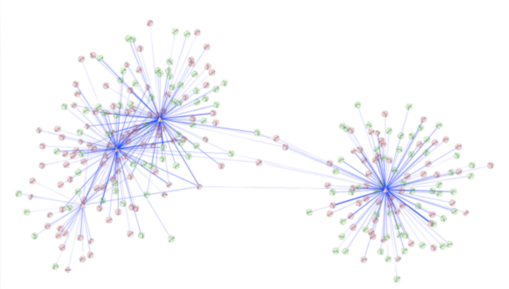
19

The final type of networks we want to discuss are similarity networks. In similarity networks, the nodes represent biological entities and the edges represent some measure of the similarity between them. There are several types of similarity networks that are commonly used in biology today. One common similarity metric is the Tanimoto coefficient[33-35], which represents the similarity between two small molecules based on the chemical fingerprints of each of them[36]. Other similarity metrics include sequence similarity as measured by BLAST[29, 37], PSI-BLAST[38], or Smith-Waterman[39], structural similarity as measured by RMSD or other structural similarity measures[40-45], or the ligand similarity as measure by the similarity ensemble approach (SEA) method[28].

There are other types of non-biological networks that use various kinds of similarity measures. Tag clouds[46] and topic maps[47], which is one of the semantic web technologies.

The images at the right show two examples of similarity networks. The network on top is a protein-protein similarity network showing the Amidohydrolase enzyme superfamily from the Structure-Function Linkage Database (SFLD)[48]. The colors on the network represent proteins of similar function. Note that these proteins tend to group together based on their BLAST similarity[29].





The network on the bottom shows a network of small-molecules where the edges represent the Tanimoto similarity between them. These networks can be useful to find molecules with similar structural characteristics

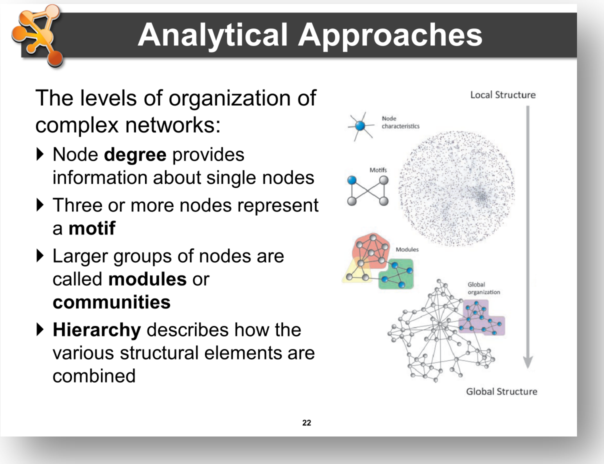
Notes

Analytical Approaches

The analysis of networks is a large and complex topic that we can't do justice in a single tutorial (even less a tutorial handout). In general, network analysis is part of the mathematics known as graph theory, and there are entire conferences (and many textbooks) devoted to the area. A good starting point might the Wikipedia article[49] or the online book "Graph Theory with Applications"[50]. Our goal here is to provide a brief introduction and touch on some of the main approaches used with biological networks.

Organization of complex networks

Complex networks have different levels of organization. The illustrations on the right show how to breakdown the *hairball* that arises when we usually plot a large complex network. First, we can look at single nodes and their local properties as the *node degree*. These nodes are then linked to form *motifs*, small subnetworks of three or more nodes. Motifs are combined to form *communities* or *modules* and communities are joined into the entire network. The *hierarchy* of the network describes how the various structural elements are combined.



Analytical Approaches

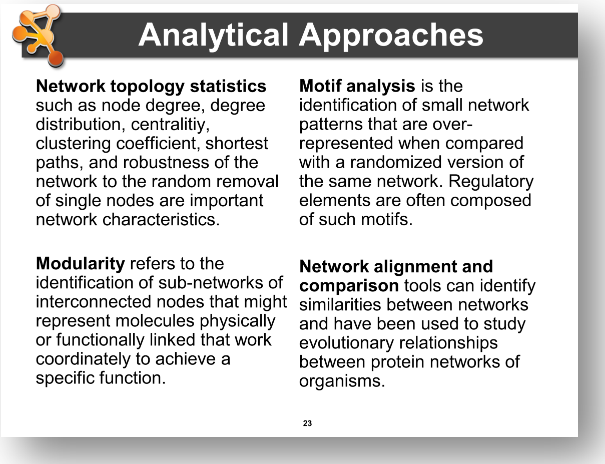
The levels of organization of complex networks:

- Node **degree** provides information about single nodes
- Three or more nodes represent a **motif**
- Larger groups of nodes are called **modules** or **communities**
- **Hierarchy** describes how the various structural elements are combined

22

There are some typical analysis tasks that are often performed with biological networks.

- Network topology statistics are easy to compute and to compare among different networks. Among others they include node centrality, betweenness, degree distribution of nodes, clustering coefficient, shortest path between nodes and robustness of the network to the random removal of single nodes.
- Modularity refers to the identification of sub-networks of interconnected nodes that might represent molecules physically or functionally linked that work coordinately to achieve a specific function.
- Motif analysis is the identification of small network patterns (or subgraphs) that are over-represented when compared with a randomized version of the same network. Discrete biological processes such as regulatory elements are often composed of such motifs.
- Network alignment and comparison tools can identify similarities between networks (such as common subgraphs) and have been used to study evolutionary relationships between protein networks of organisms.



Analytical Approaches

Network topology statistics such as node degree, degree distribution, centralitiy, clustering coefficient, shortest paths, and robustness of the network to the random removal of single nodes are important network characteristics.

Modularity refers to the identification of sub-networks of interconnected nodes that might represent molecules physically or functionally linked that work coordinately to achieve a specific function.

Motif analysis is the identification of small network patterns that are over-represented when compared with a randomized version of the same network. Regulatory elements are often composed of such motifs.

Network alignment and comparison tools can identify similarities between networks and have been used to study evolutionary relationships between protein networks of organisms.

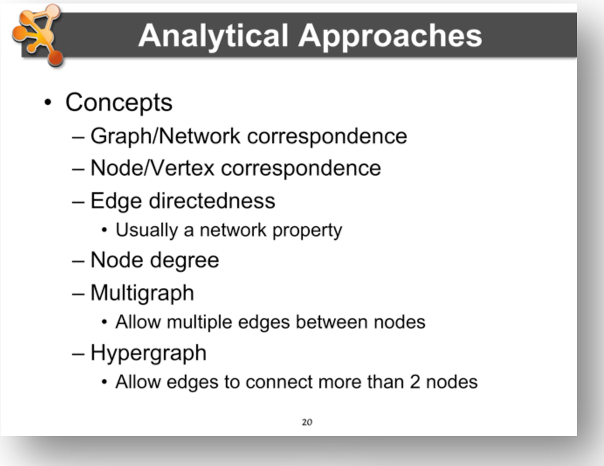
23

Concepts

In mathematical terms, a biological network (any network for that matter) is a graph, often written:

$$G = (V(G), E(G), \psi_G)$$

where *V(G)* are the set of vertices (nodes) in the graph and *E(G)* are the set of edges. In this particular notation, ψ_G is the set of incidence functions that define which edge goes with which vertices.

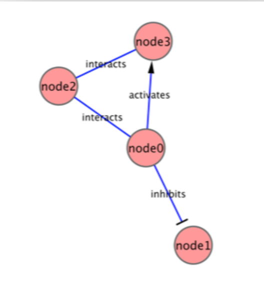


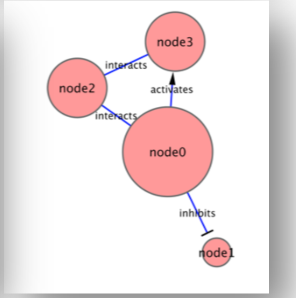
Analytical Approaches

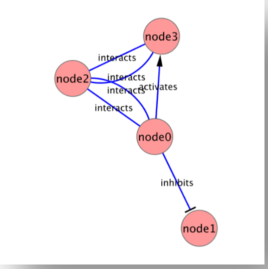
- Concepts
  - Graph/Network correspondence
  - Node/Vertex correspondence
  - Edge directedness
    - Usually a network property
  - Node degree
  - Multigraph
    - Allow multiple edges between nodes
  - Hypergraph
    - Allow edges to connect more than 2 nodes

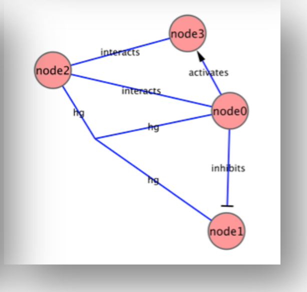
20

The edges between nodes can either be directed or undirected. This is easiest to understand when considering the *degree* of a node. In an undirected network, the degree of a node is simply the number of edges connected to it. In the first simple network at the right, the node (**node0**) has three edges connected to it, so it has a degree of 3. In a network with directed edges, we need to expand our concept of degree to include *in-degree*, the number of edges that connect *to* this node, and *out-degree*, the number of edges that originate *from* this node. In the second network at the right, the size of the nodes reflects the node degree.







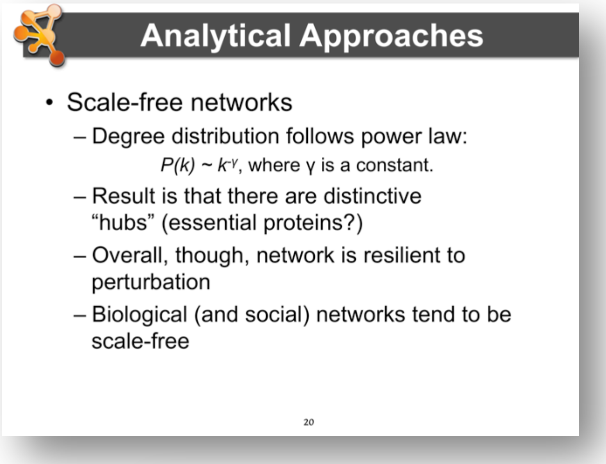


There are also differences between the types of networks. The first network at the right is a *multigraph*. In a multigraph, there can be multiple edges between nodes. The network at the far right on the other hand, is a *hypergraph*. In a hypergraph, an edge can be connected to more than two edges.

Notes

Scale-free networks

One property of network topology that is of interest is the *degree distribution* – that is, the distribution of how many edges each node has (also referred to as the *connectivity distribution*)[51]. A network is said to be *scale-free* if the degree distribution fits a power law. It has been reported that many types of biological networks are scale free[52-62]. The characteristics of scale-free networks are that there is a short path from any node to another node (*small world property*), there are many nodes with few connections and a few nodes with many connections (*hubs*), and the hubs are enriched with essential/legal nodes (*centrality and lethality principal*)[52, 63].

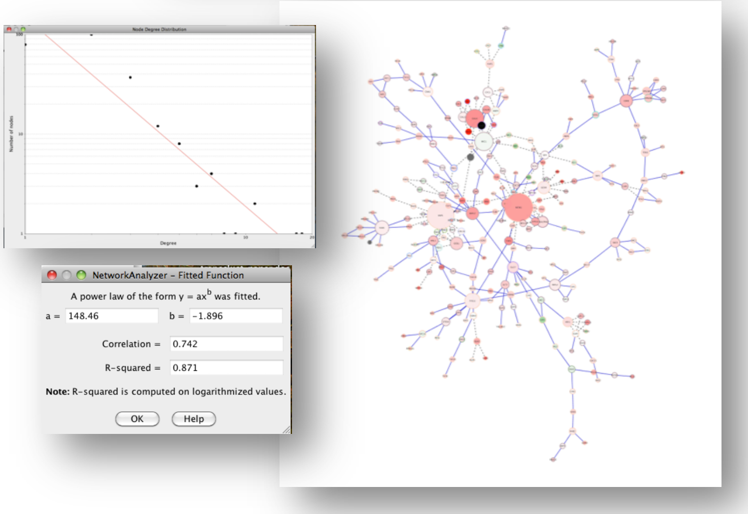


Analytical Approaches

- Scale-free networks
  - Degree distribution follows power law:
    $P(k) \sim k^{-\gamma}$, where γ is a constant.
  - Result is that there are distinctive "hubs" (essential proteins?)
  - Overall, though, network is resilient to perturbation
  - Biological (and social) networks tend to be scale-free

20

Scale-free networks have interesting properties for biological systems – in particular, they are robust to random breakdowns[64]. They are also (as the name implies) invariant to changes in scale. On the other hand, recent analysis of several data sources have begun to throw into question exactly how well many biological networks fit the scale-free power law distribution[63, 65-67]. So, while none of the authors have suggested that biological networks don't exhibit some scale-free characteristics, they don't fit the power-law degree distribution well enough to be considered scale-free.



NetworkAnalyzer – Fitted Function

A power law of the form $y = ax^b$ was fitted.

a = 148.46 b = –1.896

Correlation = 0.742

R–squared = 0.871

Note: R–squared is computed on logarithmized values.

It should also be noted that biological networks aren't the only network type that tends to be scale-free. For example, both social networks and the Internet tend to be scale-free[68, 69]. In both cases the overall topology tends to be one with a few hubs of high degree and lots of lower-degree nodes.

Notes

____

____

____

____

____

____

Small-world networks

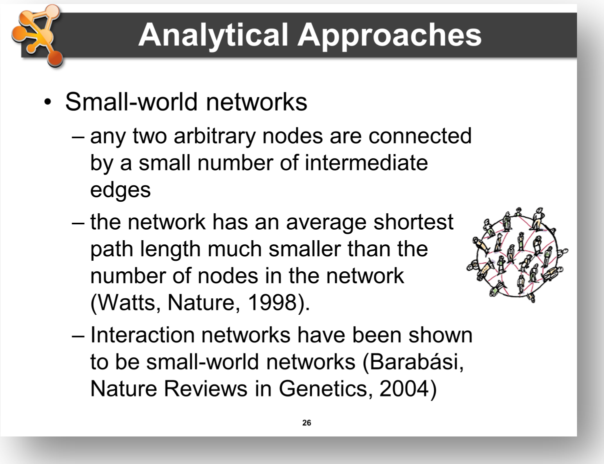
One of the striking properties of many empirical networks is that despite their huge size, the average path length is usually surprisingly small [74]. Such networks are called small-world networks.

Formally, a small-world network is defined to be a network where the typical distance L between two randomly chosen nodes grows proportionally to the logarithm of the number of nodes N in the network:

$$L \propto \log N$$

Many real-world systems have small-world properties when represented as networks. They include the WWW, food chains, electric power grids, metabolite processing networks, networks of brain neurons, telephone call graphs, and social influence networks.

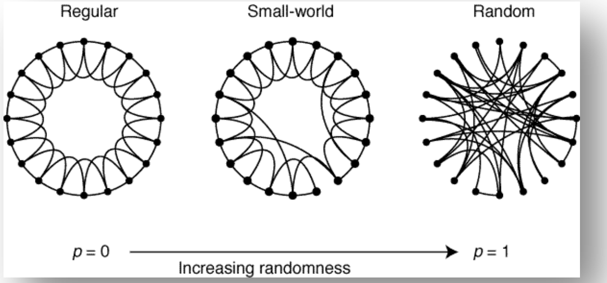
Interaction networks are also shown to be small-world networks [51]. This could indicate that local pertrubations such as regulation of the biological activity of a given protein could reach the whole network very quickly.



Analytical Approaches

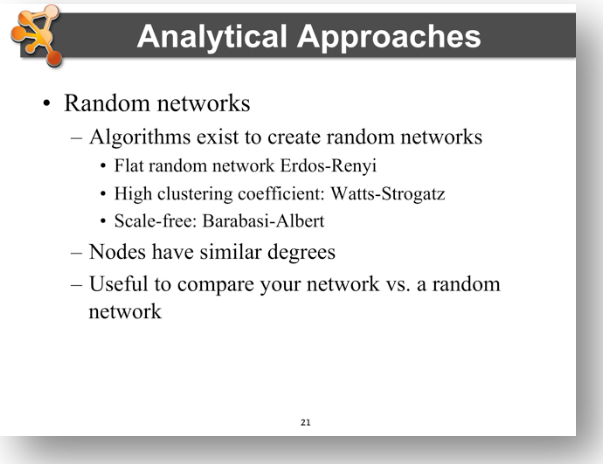
- Small-world networks
  - any two arbitrary nodes are connected by a small number of intermediate edges
  - the network has an average shortest path length much smaller than the number of nodes in the network (Watts, Nature, 1998).
  - Interaction networks have been shown to be small-world networks (Barabási, Nature Reviews in Genetics, 2004)

26



Notes

Random networks



Analytical Approaches

- Random networks
  - Algorithms exist to create random networks
    - Flat random network Erdos-Renyi
    - High clustering coefficient: Watts-Strogatz
    - Scale-free: Barabasi-Albert
  - Nodes have similar degrees
  - Useful to compare your network vs. a random network

21

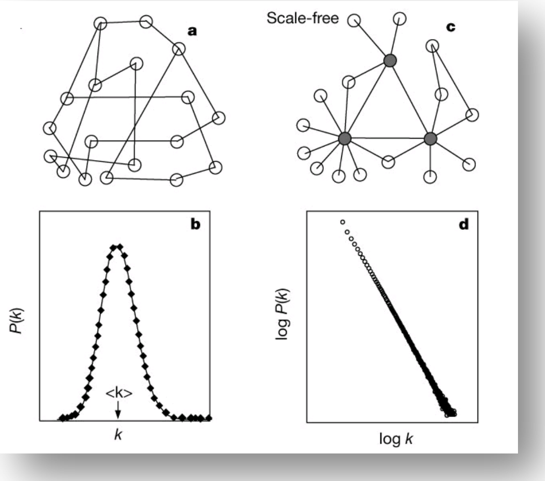
Random networks (random graphs) are important tools for determining the extent to which a computationally derived network differs from a similar “random network”. This is, in principal, the same idea behind the BLAST expectation value or the *p* value that you might get from a statistical test.

Networks, however, are complicated, and developing an appropriate probability model is non-trivial. There are several algorithms commonly used to generate random networks. In the simplest case, you can just generate a graph, *G(n,p)*, where for any two nodes N_1 and N_2, there is a probability *p* that there is an edge between them[70]. This is similar to the Erdős-Rényi model([71, 72] as cited in [73]), but in the Erdős-Rényi model, the number of edges is restricted to a fixed number, *M*. Thus, the graph, *G(n,M)*, is a graph where all of the *M* edges appear with equal probability.

The problem with both of these “flat” models is that neither of the models are likely to result in graphs that exhibit the characteristics of biological networks (small world, scale-free) discussed above. One approach to this is to explicitly model the random graph such that it exhibits small-world properties (short average path lengths and high clustering). The is the approach proposed by Watts and Strogatz[74]. In the Watts and Strogatz model, there are three key parameters: the number of nodes, *N*, the mean degree of the nodes, *K*, and a tuning parameter *β*, which is between 0 and 1. The algorithms begins by generating a network with *N* nodes, each connected to *K* neighbors, *K*/2 on each side. Then for every edge (n_i,n_j) rewire that edge with probability *β* such that there are no loops and there is no duplicate edges. The result depends on the value of *β*. If *β* is near zero, the result is a regular lattice. If *β* is one, this approaches the random graph similar to the Erdős-Rényi model with $p = \frac{NK}{2\binom{N}{2}}$

Another approach is to implement a random graph that is scale-free. The Barabási-Albert model is an approach to generating random scale-free graphs[68]. This approach starts with a small network *G(n,m)*, where *n* is the number of nodes (≥2) and *m* is the number of edges. The requirement is that all nodes have degree of at least 1. Then new nodes are added according to a probability p_i:

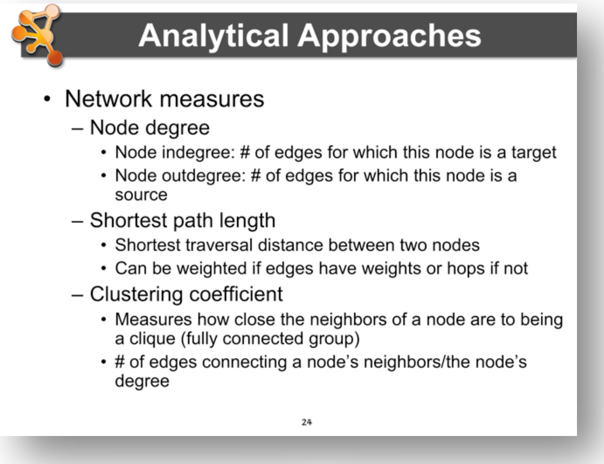
$$p_i = \frac{k_i}{\sum_j k_j}$$



where k_i is the degree of the node *i*. This results in hubs (nodes with more edges) continuing to get more edges and nodes with fewer edges being less likely to get new edges. This results in a degree distribution that fits the scale-free model quite well, but is still random in nature.

Network measures

We've mentioned the three most common network measures already: node degree, path length, and clustering coefficient. The first two of these are intuitively understandable. The third is a little more difficult to conceptualize since it doesn't fit our concept of clusters (i.e. groupings of nodes or modularity) very well.

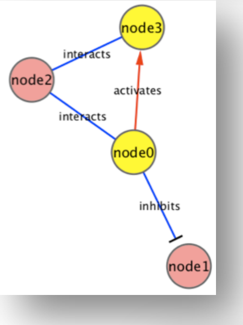


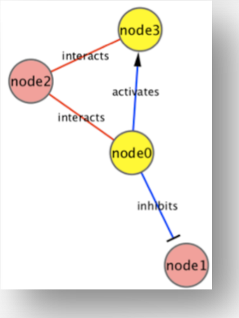
Analytical Approaches

- Network measures
  - Node degree
    - Node indegree: # of edges for which this node is a target
    - Node outdegree: # of edges for which this node is a source
  - Shortest path length
    - Shortest traversal distance between two nodes
    - Can be weighted if edges have weights or hops if not
  - Clustering coefficient
    - Measures how close the neighbors of a node are to being a clique (fully connected group)
    - # of edges connecting a node's neighbors/the node's degree

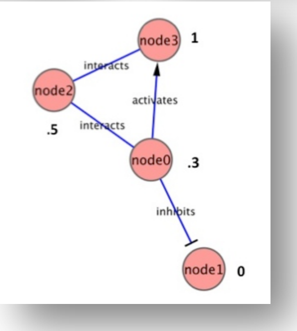
24

Node degree is, as we've already mentioned, the number of edges connected to this node. In a directed network, the node *indegree* is the number of edges directed towards this node, and the node *outdegree* is the number of edges directed away from this node. In the network at the right, for example, **node3** has an *indegree* of 2 and an *outdegree* of 1 (assuming we count the undirected edge as both in and out).



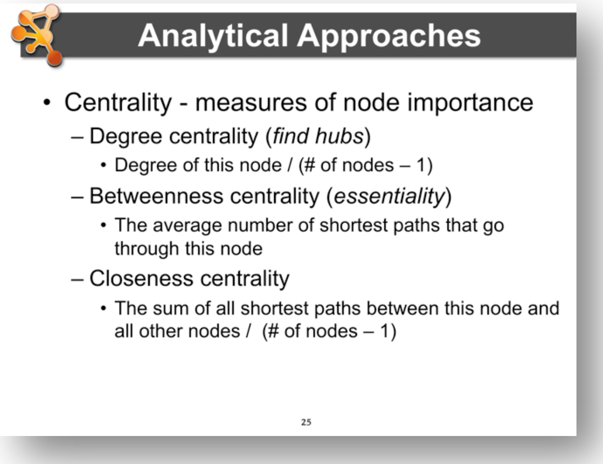


Path length is also relatively easy to imagine. If we look for the shortest path from **node0** to **node3** (the first network at the right) it's the edge between them. On the other hand, the shortest path from **node3** to **node0** goes through **node2** (because the edge between **node0** and **node3** is directed). The length of the path is often just a hop count (1 in the first example, 2 in the second), but can also be weighted, which might mean the shortest path is not the path that traverses the fewest nodes.



The clustering coefficient is a measure of the degree to which nodes form a complete graph. It was originally defined to measure the degree to which a network exhibits small-world properties[74]. For undirected graphs, the local clustering coefficient is given as the number of edges between neighbors of node i divided by the maximum possible number of edges: $C_i = \frac{2\left|\{e_{jk}\}\right|}{k_i(k_i - 1)}$

In the network example at the right (assuming it's undirected), **node3** has two neighbors (degree 2), **node2** and **node0** share an edge, so we have (2*1)/2(2-1) = 1. On the other hand, **node0** is degree 3, but only **node2** and **node3** are connected, so we have (2*1)/3(3-1) = .3. The network average clustering coefficient can be used to express the degree to which a graph exhibits small-world properties. The average is simply: $\overline{C} = \frac{1}{n}\sum_{i=1}^{n} Ci$



Analytical Approaches

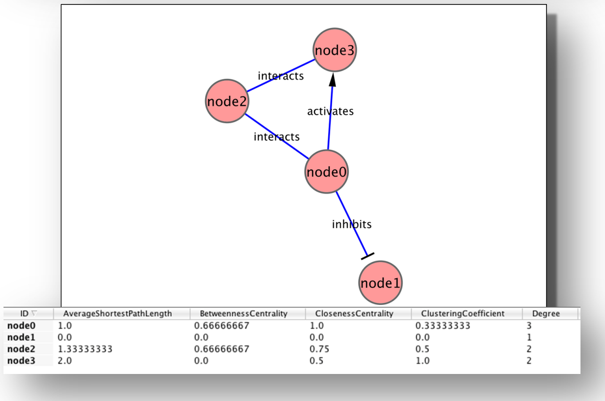
- Centrality - measures of node importance
  - Degree centrality (*find hubs*)
    - Degree of this node / (# of nodes – 1)
  - Betweenness centrality (*essentiality*)
    - The average number of shortest paths that go through this node
  - Closeness centrality
    - The sum of all shortest paths between this node and all other nodes / (# of nodes – 1)

25

Another important set of networks measures that has important properties are the various centrality measures. These approaches (in general) attempt to provide a measure of the importance of a given node. There are many centrality measures, but we'll just discuss three of them here.

The first centrality measure we'll discuss is *degree centrality*. Nodes with high degree centrality are the hubs in scale-free networks, for example. This is an easy measure to compute the degree centrality (C_D) of node v: $C_D(v) = \frac{\deg(v)}{n-1}$, where n is the number of nodes in the network.

Betweenness centrality is another centrality measure than tends to reflect the essentiality of a node in the network. Essentially it measures the extent to which "all roads lead through" this node. The betweenness centrality for a node v is calculated as: $C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$, where σ_{st} is the number of shortest paths from s to t and σ_{st} (v) is the number of shortest paths from s to t that go through v. Betweenness is usually normalized by dividing with the number of node pairs. Intuitively, this makes sense – if a large percentage of the shortest paths between two nodes go through a given node, removal of that node will have a significant effect on the network topology (from the perspective of those two nodes).



ID	AverageShortestPathLength	BetweennessCentrality	ClosenessCentrality	ClusteringCoefficient	Degree
node0	1.0	0.66666667	1.0	0.33333333	3
node1	0.0	0.0	0.0	0.0	1
node2	1.33333333	0.66666667	0.75	0.5	2
node3	2.0	0.0	0.5	1.0	2

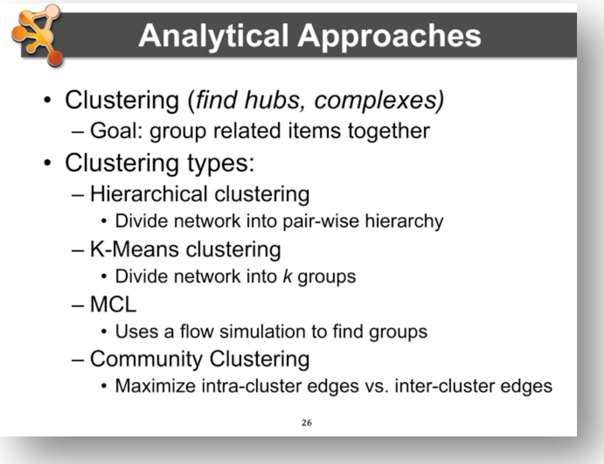
Closeness centrality is the degree to which this node is close to all other nodes. It is again calculated based on shortest paths: $C_C(v) = \frac{\sum_{t \neq v \in V} S(v,t)}{n-1}$, where *S(v,t)* is the shortest path between v and t. So, in a star topology, where all nodes are connected to a single hub, the closeness centrality measure for the hub is 1 and ~2 for all other nodes[2]. This value is usually reported as the inverse, 1/Cc(v).

Notes

[2] It's approximately 2 because the shortest path between a non-hub node and all of the other nodes is 2 except for the hub node, in which case the shortest path is 1.

Clustering

Clustering is a heavily used technique for analyzing networks, both biological and otherwise. The overall goal of clustering is to group items together that are related based on some measure. Clustering is an active area of research and there are many clustering algorithms that have long been used for biological applications, and even more algorithms that are being developed for specialized purposes.



Analytical Approaches

- Clustering (*find hubs, complexes)*
  - Goal: group related items together
- Clustering types:
  - Hierarchical clustering
    - Divide network into pair-wise hierarchy
  - K-Means clustering
    - Divide network into *k* groups
  - MCL
    - Uses a flow simulation to find groups
  - Community Clustering
    - Maximize intra-cluster edges vs. inter-cluster edges

26

Before we talk about specific clustering approaches, it is important to understand that all of the clustering approaches depend on some metric for determining the similarity of the items being clustered. This similarity metric is termed a *distance* metric in clustering terms, and there are a number of ways to calculate the distance in feature space (that is, the terms or values you are using to determine the similarity between objects). A common measure is the *Euclidean* distance, which is simply the distance between two points in n-dimensional space:

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \ldots + (q_n - p_n)^2}$$

Other common techniques are based on the *Pearson correlation*, *r*, between any two series of numbers $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$, which is defined as:

$$r = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - \overline{x}}{\sigma_x}\right)\left(\frac{y_i - \overline{y}}{\sigma_y}\right)$$

where σ_x is the standard deviation of the *x* series, and σ_y is the standard deviation of the *y* series.

This term can be either *centered* (as above), or *uncentered*, which assumes a mean of zero (even if it's not). There are many other approaches to calculating the distance, from taking the negative log of the BLAST e-value to much more complicated approaches designed to account for specific characteristics of the data.

Hierarchical Clustering

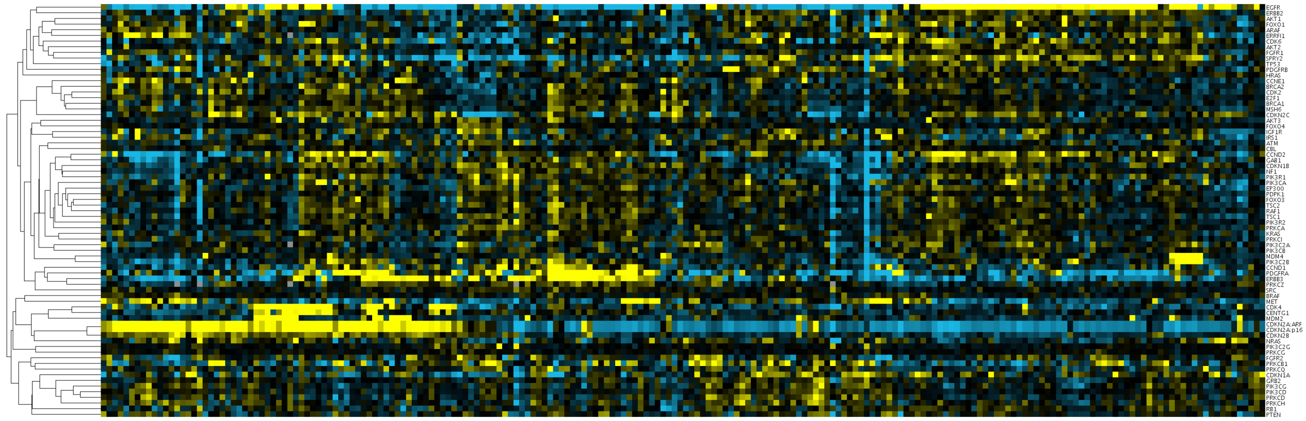
A very common clustering approach is *hierarchical* clustering[75]. As the name implies, this approach divides the objects into a pairwise hierarchy. Hierarchical clustering has been used for many years as one of the major approaches to analyzing and visualizing microarray data[76]. An important first step in performing hierarchical clustering is to determine the distance metric (above). The second step is to determine how to link the pairwise distances[3]:

- *Single linkage* clustering takes the minimum pairwise distance,
- *Complete linkage* clustering takes the maximum pairwise distance,

[3] This list is taken from the clustering approaches used in the original Cluster program from Eisen and colleagues, which has been inherited by clusterMaker and other Cluster-clones.

- *Average linkage* clustering (UPGMA) takes the average of all of the pairwise distances,
- *Centroid linkage* clustering takes the distance between the centroids of all pairs of elements.

Once the metrics and linkages have been selected, clustering may be accomplished by either an *agglomerative* (bottom-up) or *divisive* (top-down) method. In either case, the result is tree (hierarchy) where the nodes closer together in the tree are more similar. For microarray data, this is often shown as a dendrogram associated with the heatmap that reflects the fold changes in the expression data (see the example below).



k-Means Clustering

Another common clustering technique is *k-means[77, 78]*. In k-means clustering the algorithm divides the data set up into *k* groups in such a way that the value of the item gets assigned to the cluster with the nearest mean. The approach is relatively simple: given a set of **n** data items the idea is to partition the **n** items into *k* sets so as to minimize the within-cluster sum of squares (WCSS):

$$\underset{S}{\arg\min} \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2$$

where **S** = (S_1, S_2, …, S_n) are the clusters and $\boldsymbol{u}_i$ are the mean of the points in each cluster S_i. k-means has been used in a number of applications, and has been incorporated in to a number of other algorithms.

There are many other clustering algorithms and combinations of algorithms used in network applications – far too many for us to cover here. Often these algorithms are general algorithms (e.g. Community clustering[79], MCL[31, 80, 81], Spectral Clustering[82-86], and Affinity Propagation[87, 88]) and often they designed for special purposes (e.g. SCPS[89], MCODE[5], FORCE[90], TransClust[91]). Some algorithms are actually combinations of algorithms (e.g. AutoSOME[92]). We're going to cover only three of these algorithms (MCL, Spectral, and Affinity Propagation), but the interested reader is encouraged to explore the references below.

MCL Clustering

MCL clustering (MCL is short for Markov Clustering) is a clustering approach that simulates a weighted random walk through a network. The idea behind the algorithm is that because edges within the natural groupings will most likely stay within the group, the vast majority of the steps in a random walk will be within the natural group. The other way to think about it is by imagining edges as flows – most of the flow through a network with natural clusters will stay within the clusters – very little will flow between the clusters. The simulation of the random walk is by alternate application of two operations: *expansion* and *inflation*. First, the distance matrix is converted to a stochastic matrix (a non-negative matrix where each of the columns sums to 1). In the expansion step, the stochastic matrix is squared using the normal matrix product. In the inflation step, the Hadamard product of the matrix (entry-wise multiplication by an inflation parameter, *I*) is taken. After the inflation step, a scaling step is added which returns the matrix to a stochastic matrix. Repeated expansion and inflation will have the result of removing cells in the distance matrix (i.e. edges) that represent inter-cluster edges.

MCL clustering has been used for a large number of biological applications, including the finding of protein complexes in protein-protein interaction networks and the grouping of proteins in protein similarity networks. MCL has proven to be very fast and robust with then number of edges is reasonably low, but can have problems resolving dense networks necessitating some form of algorithm or user-chosen cut-off value to reduce the edge density[93]. MCL has the nice characteristic that it does not necessitate the user to select the number of clusters in advance, although the inflation parameter *I* does have to be specified.

Spectral Clustering

Spectral clustering takes in name from the use of spectral properties of the similarity (or distance) matrix constructed from the network. Given a set of data points A, the similarity matrix may be defined as a matrix *S* where S_{ij} represents a measure of the similarity between points *i* and *j* which are members of the set A. Spectral clustering techniques make use of the spectrum of this matrix of the data to perform dimensionality reduction for clustering in fewer dimensions.

One such technique is the Normalized Cuts algorithm[94, 95], commonly used for image segmentation. It partitions points into two sets (S_1,S_2) based on the eigenvector *v* corresponding to the second-smallest eigenvalue of the Laplacian matrix

$$L = I - D^{-\frac{1}{2}} S D^{-\frac{1}{2}}$$

of S, where D is the diagonal matrix

$$D_{ij} = \sum_j S_{ij}$$

This partitioning may be done in various ways, such as by taking the median m of the components in v, and placing all points whose component in v is greater than m in S_1, and the rest in S_2. The algorithm can be used for hierarchical clustering by repeatedly partitioning the subsets in this fashion.

A related algorithm is the Meila-Shi algorithm[82], which takes the eigenvectors corresponding to the k largest eigenvalues of the matrix $P = SD^{-1}$ for some k, and then invokes another (e.g. k-means) to cluster points by their respective k components in these eigenvectors.

Spectral clustering techniques are very useful in biology, but they have the disadvantage that since they essentially divide the data into two sets, you must either combine them with something like k-means or use a hierarchical decomposition to arrive at a more refined clustering.

Affinity Propagation

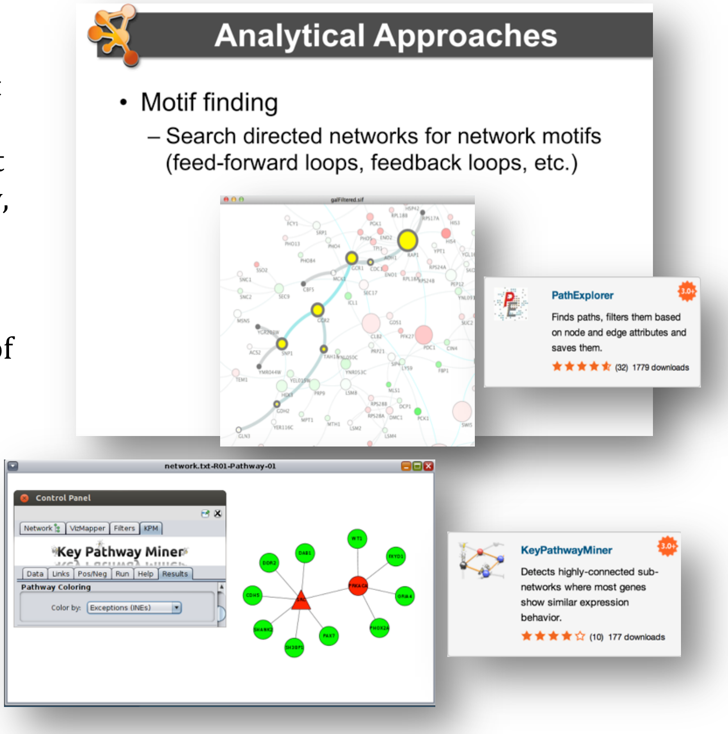
Affinity propagation[87] is a newer algorithm that takes a message passing approach rather than a mathematical approach to clustering. Basically, as with the other approaches, affinity propagation takes a similarity matrix $s(i,j)$, which represents the starting point of the algorithm. In addition, each point is given a preference value $s(k,k)$ which is used to seed the likelihood of this point being an exemplar for the formation of a cluster (this is often just set to a flat value to allow the algorithm to learn the number of clusters). Then, the points exchange messages of two types: responsibilities ($r(i,k)$) are sent from point i to point k and reflects the degree to which k is a good exemplar for point i; and availability ($a(i,k)$) is sent from point k to point i to reflect the evidence for i to choose k for its exemplar. The algorithm runs until some stopping point usually based on the degree to which $r(i,k)$ and $a(i,k)$ change during each pass. See their web site (http://www.psi.toronto.edu/index.php?q=affinity propagation) for more information about the algorithm and its application.

Affinity propagation has numerous applications in biology and seems to perform well in the datasets provided by the authors. Some comparative analysis by others[96] suggests that other algorithms might be less susceptible to noise and more robust for some applications.

Notes

Network motifs

A network motif is a pattern of connectivity that occurs more frequently than might be expected by a random connection of nodes[97]. As might be expected by the reuse we often see in biology, biological networks tend to have a small set of network motifs that act like components in a larger circuit[98, 99]. Network motifs have been identified in the gene regulation network of *E. coli*[100] as well as a larger set of networks[101]. There are a number of network motifs that have been identified in biology, including feed forward loops[102-106] (like the one shown at the right), feedback loops[107-109], positive and negative auto-regulation loops[110]. These biological circuits are critical to regulatory processes in the cell, so identifying them in protein-protein interaction networks can provide important clues to the pathway which the protein participate in[111-113].



Analytical Approaches

- Motif finding
  - Search directed networks for network motifs (feed-forward loops, feedback loops, etc.)

PathExplorer

Finds paths, filters them based on node and edge attributes and saves them.

★★★★★ (32) 1779 downloads

KeyPathwayMiner

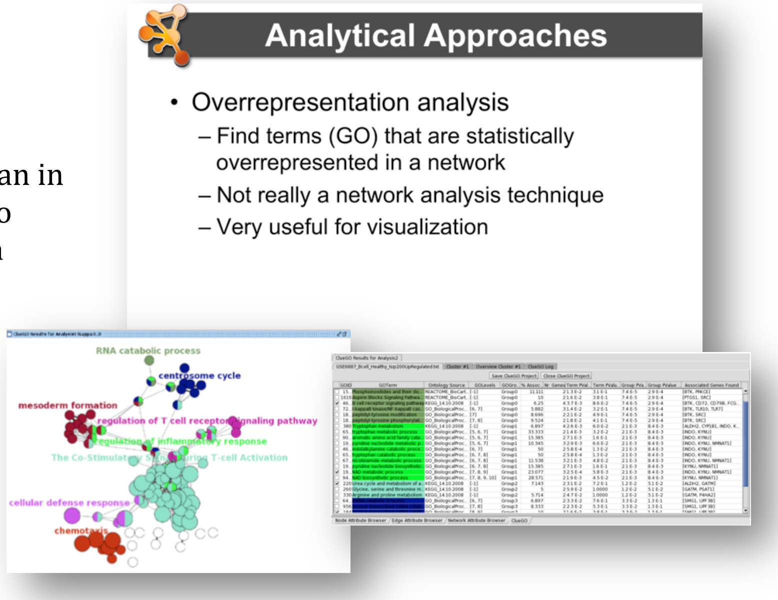
Detects highly-connected sub-networks where most genes show similar expression behavior.

★★★★☆ (10) 177 downloads

Notes

Overrepresentation analysis

Overrepresentation analysis (ORA) is an important tool used to identify aspects or attributes of a subset of nodes that are statistically more common in those nodes than in the full set. The most common approach is to cluster a group of genes based on expression data and look for overrepresentation of various gene ontology (GO)[114] terms in the groups to determine if a particular expression pattern suggests a particular biological process[115-118].



Analytical Approaches

- Overrepresentation analysis
  - Find terms (GO) that are statistically overrepresented in a network
  - Not really a network analysis technique
  - Very useful for visualization

RNA catabolic process
centrosome cycle
mesoderm formation
regulation of T cell receptor signaling pathway
regulation of inflammatory response
The Co-Stimulatory Signal During T-cell Activation
cellular defense response
chemotaxis

ClueGO Results for Analysis2

GSE6887_Bcell_Healthy_top200UpRegulated.txt | Cluster #1 | Overview Cluster #1 | ClueGO Log

Save ClueGO Project | Close ClueGO Project

	GOID	GOTerm	Ontology Source	GOLevels	GOGro...	% Assoc...	Nr. Genes	Term PVal...	Term PValu...	Group PVa...	Group PValue ...	Associated Genes Found
□	15...	Phosphoinositides and their do...	REACTOME_BioCart...	[-1]	Group0	11.111	2	1.3 E-2	3.1 E-1	7.4 E-5	2.9 E-4	[BTK, PRKCE]
□	1619	Aspirin Blocks Signaling Pathwa...	REACTOME_BioCart...	[-1]	Group0	10	2	1.6 E-2	3.8 E-1	7.4 E-5	2.9 E-4	[PTGS1, SRC]
☑	46...	B cell receptor signaling pathway	KEGG_14.10.2008	[-1]	Group0	6.25	4	3.7 E-3	8.6 E-2	7.4 E-5	2.9 E-4	[BTK, CD72, CD79B, FCG...
□	72...	I-kappaB kinase/NF-kappaB cas...	GO_BiologicalProc...	[6, 7]	Group0	5.882	3	1.4 E-2	3.2 E-1	7.4 E-5	2.9 E-4	[BTK, TLR10, TLR7]
□	18...	peptidyl-tyrosine modification	GO_BiologicalProc...	[7]	Group0	8.696	2	2.1 E-2	4.9 E-1	7.4 E-5	2.9 E-4	[BTK, SRC]
□	18...	peptidyl-tyrosine phosphorylati...	GO_BiologicalProc...	[7, 8]	Group0	9.524	2	1.8 E-2	4.1 E-1	7.4 E-5	2.9 E-4	[BTK, SRC]
□	380	Tryptophan metabolism	KEGG_14.10.2008	[-1]	Group1	6.897	4	2.6 E-3	6.0 E-2	2.1 E-3	8.4 E-3	[ALDH2, CYP1B1, INDO, K...
□	65...	tryptophan metabolic process	GO_BiologicalProc...	[5, 6, 7]	Group1	33.333	2	1.4 E-3	3.2 E-2	2.1 E-3	8.4 E-3	[INDO, KYNU]
□	90...	aromatic amino acid family cata...	GO_BiologicalProc...	[5, 6, 7]	Group1	15.385	2	7.1 E-3	1.6 E-1	2.1 E-3	8.4 E-3	[INDO, KYNU]
□	19...	pyridine nucleotide metabolic p...	GO_BiologicalProc...	[5, 6, 7]	Group1	10.345	3	2.9 E-3	6.6 E-2	2.1 E-3	8.4 E-3	[INDO, KYNU, NMNAT1]
□	46...	indolalkylamine catabolic proce...	GO_BiologicalProc...	[6, 7]	Group1	50	2	5.8 E-4	1.3 E-2	2.1 E-3	8.4 E-3	[INDO, KYNU]
□	65...	tryptophan catabolic process	GO_BiologicalProc...	[6, 7, 8]	Group1	50	2	5.8 E-4	1.3 E-2	2.1 E-3	8.4 E-3	[INDO, KYNU]
□	67...	nicotinamide metabolic process	GO_BiologicalProc...	[6, 7, 8]	Group1	11.538	3	2.1 E-3	4.8 E-2	2.1 E-3	8.4 E-3	[INDO, KYNU, NMNAT1]
□	19...	pyridine nucleotide biosynthetic...	GO_BiologicalProc...	[6, 7, 8]	Group1	15.385	2	7.1 E-3	1.6 E-1	2.1 E-3	8.4 E-3	[KYNU, NMNAT1]
☑	19...	NAD metabolic process	GO_BiologicalProc...	[7, 8, 9]	Group1	23.077	3	2.5 E-4	5.8 E-3	2.1 E-3	8.4 E-3	[INDO, KYNU, NMNAT1]
□	94...	NAD biosynthetic process	GO_BiologicalProc...	[7, 8, 9, 10]	Group1	28.571	2	1.9 E-3	4.5 E-2	2.1 E-3	8.4 E-3	[KYNU, NMNAT1]
☑	220	Urea cycle and metabolism of a...	KEGG_14.10.2008	[-1]	Group2	7.143	2	3.1 E-2	7.2 E-1	1.2 E-2	5.1 E-2	[ALDH2, GATM]
□	260	Glycine, serine and threonine m...	KEGG_14.10.2008	[-1]	Group2	5	2	5.9 E-2	1.0000	1.2 E-2	5.1 E-2	[GATM, PSAT1]
□	330	Arginine and proline metabolism	KEGG_14.10.2008	[-1]	Group2	5.714	2	4.7 E-2	1.0000	1.2 E-2	5.1 E-2	[GATM, P4HA2]
□	64...	mRNA catabolic process	GO_BiologicalProc...	[6, 7]	Group3	6.897	2	3.3 E-2	7.6 E-1	3.3 E-2	1.3 E-1	[SMG1, UPF3B]
□	956	nuclear-transcribed mRNA catab...	GO_BiologicalProc...	[7, 8]	Group3	8.333	2	2.3 E-2	5.3 E-1	3.3 E-2	1.3 E-1	[SMG1, UPF3B]
☑	184	[illegible]	GO_BiologicalProc...	[8, 9]	Group3	10	2	1.6 E-2	3.8 E-1	3.3 E-2	1.3 E-1	[SMG1, UPF3B]

Node Attribute Browser | Edge Attribute Browser | Network Attribute Browser | ClueGO

One of the things to keep in mind when doing ORA is that the resultant *p*-values may need to be adjusted since multiple tests are conducted. This makes sense – if we're performing multiple tests we increase the possibility that we'll get a false positive based on random chance. Two methods for correcting for multiple tests are the Dunn-Bonferroni Familywise Error Rate (FWER)[119] correction and the Benjamini & Hochberg False Discovery Rate (FDR)[120] correction.

The image at the right shows the results of an overrepresentation analysis of a yeast expression data set using the Cytoscape BiNGO plugin[121].

Notes

____

____

____

____

____

____

____

____

____

____

____

____

____

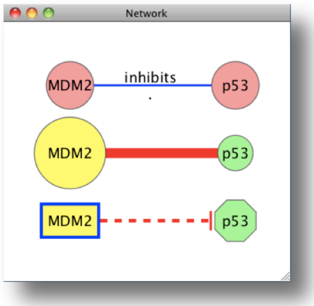
____

Visualization

In the previous sections, we made use of a number of visualization techniques that are easily taken for granted. In this section, we will detail the techniques and key decisions involved in producing network and pathway visualizations. Using biological networks to visualize data is a critical aspect of exploratory analysis: facilitating interpretation, new insights and new hypotheses (PMID: 20824171). We are visual creatures, after all.

Depiction

The basic visual motif of networks in Cytoscape is that of nodes and edges. In biological networks, the nodes often represent genes, proteins or small molecules, while the edges (or lines) represent interactions and relationships between connected nodes (see figure at the right). Beyond this core motif, all other visual features (e.g., shape, size, color, thickness, label, transparency, etc) are flexible and can be used to represent practically any data value, annotation or attribute.

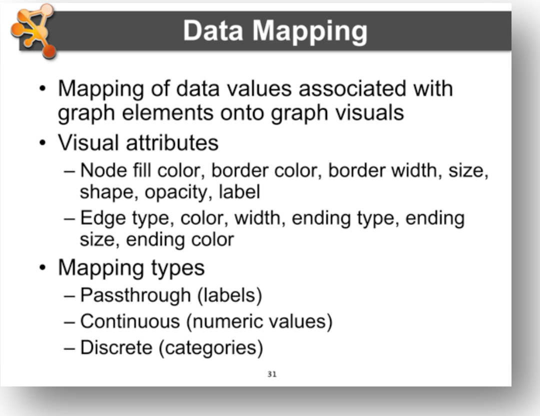


Node and edge motif in various visual styles

Data Mapping

The first thing most users want to do in Cytoscape is to map their data onto networks for visualization. The variety of data and network types has already been explored in previous sections. Here, we will focus on the mechanics of data mapping using the VizMapper interface.

VizMapper provides a user interface for controlling the mapping of data attributes to visual attributes. There is a long list of available visual attributes that can be mapped to, including node properties such as fill color, border color, shape, width, height, opacity and label, and edge properties such as type, color, thickness, as well as arrow type, size and color.



Data Mapping

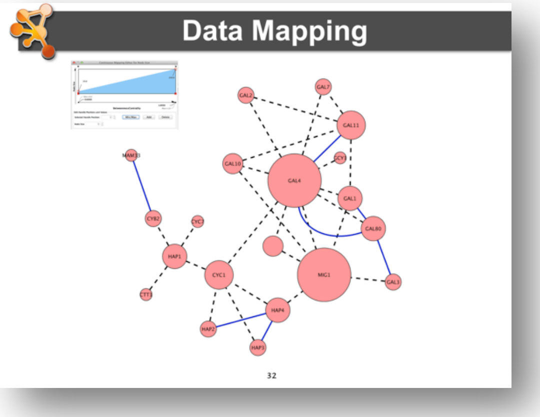
- Mapping of data values associated with graph elements onto graph visuals
- Visual attributes
  - Node fill color, border color, border width, size, shape, opacity, label
  - Edge type, color, width, ending type, ending size, ending color
- Mapping types
  - Passthrough (labels)
  - Continuous (numeric values)
  - Discrete (categories)

31

Notes

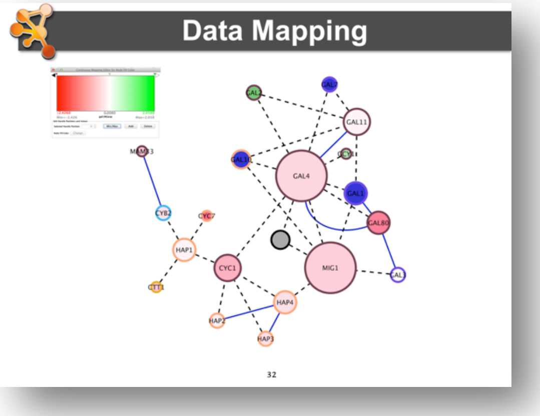
Data attributes can be mapped in three main ways: **Passthrough** – directly passing the data value to the visual attribute, e.g., labels. **Continuous** – mapping a continuous range of numerical values to a range of visual attributes, e.g., expression values to a color gradient. **Discrete** – mapping discrete data values (string or numeric) to specified visual attributes, e.g., five different categories to five different colors.

The next two examples focus on two types of Continuous data mapping, since these are the most useful and most challenging. First, we map degree-of-connectivity (the data attribute) to node size (the visual attribute), so that more connected nodes appear proportionally larger, thus highlighting potential "hubs." In the VizMapper interface, you would begin by double-clicking on 'Node Size' and selecting the data attribute containing degree information. Then select "Continuous Mapping" type and click the graphic to edit the mapping parameters. The min and max of the data attribute is given as the x-axis and the visual attribute is the y-axis.


Data Mapping

32

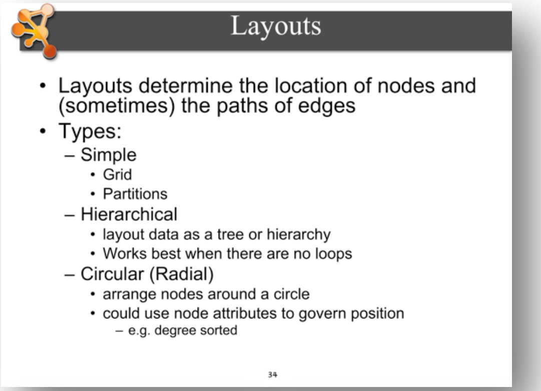
The second example maps continuous expression values to node color. Once again in the VizMapper interface, you would double-click the visual attribute, pull-down the data attribute and then choose "Continuous Mapping." When you click on the graphic, you will notice a different parameterization. Once again, the data attribute is given as the x-axis, but now instead of a y-axis, you will find thresholds that control the ends and mid-point of a color gradient as well as step-function thresholds to set discrete colors for values exceeding the gradient range. This is a handle tool for focusing the continuous mapping of color to a critical range of data.


Data Mapping

32

Notes

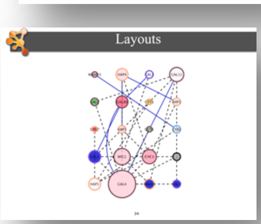
Layouts

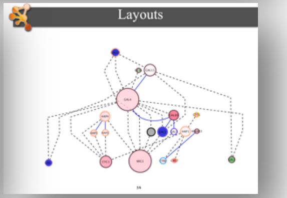


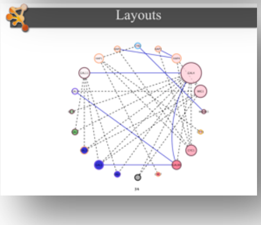
Layouts

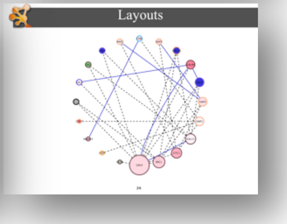
- Layouts determine the location of nodes and (sometimes) the paths of edges
- Types:
  - Simple
    - Grid
    - Partitions
  - Hierarchical
    - layout data as a tree or hierarchy
    - Works best when there are no loops
  - Circular (Radial)
    - arrange nodes around a circle
    - could use node attributes to govern position
      - e.g. degree sorted

34









The majority of network information does not come with fixed coordinates. With the exception of manually curated pathway diagrams, networks typically rely on automated layout algorithms to position nodes and edges. Cytoscape comes with a wide variety of built-in layout algorithms that can be applied to any pathway or network. In addition, a number of plugin extensions have been developed to support additional layouts.

Here, we will describe the main layout types natively supported by Cytoscape. You can find these in the menu *Layout > Cytoscape Layouts*.

Grid Layout – a simple layout of nodes in arbitrary order arranged in a grid pattern. This layout does *not* take in account edge crossings, weights or degree of connectivity.

Group Attributes Layout – performs a grid layout but orders nodes according to a user-selected attribute, e.g., ascending order based on a numerical attribute.

Hierarchical – based on connectivity, this layout defines ordered layers of nodes in a tree structure, e.g., phylogenetic trees.

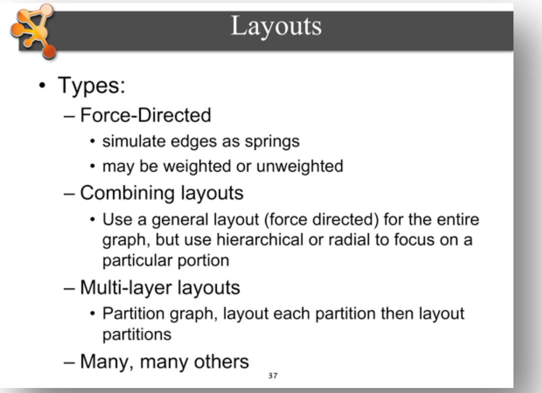
Circular Layout – arranges nodes around the circumference of a circle. The order of the nodes is arbitrary in the basic version. There two other versions: **Attribute Circle Layout**, which orders nodes based on a user-selected attribute, and **Degree Sorted Circle Layout**, which orders nodes based on their number of connections. *Pro-tip: The Degree Sorted Circle Layout calculates the degree for each node and creates a new attribute that can be used for other purposes as well, e.g., data mapping.*

Notes

Force-Directed Layout – simulates edges as springs, resulting in clusters of highly connected nodes with minimally connected nodes spaced and in the periphery. You can also choose to influence the layout based on an edge attribute, if available. A related layout is **Spring Embedded**, which also simulates edges as springs. Both of these layouts also have explicit **Edge-Weighted-** versions that provide more control.

Furthermore, you can apply layouts to selected subsets of nodes. If you make a node selection prior to browsing the Layout menu, you will see an additional submenu option to apply the layout to “All nodes” or “Selected Nodes Only.” By using this feature, you can effectively combine different layouts for a single network. For example, after applying a Force-Directed Layout, you could then select a connected subset and apply a Hierarchical Layout just to that set.

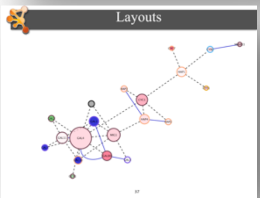
To achieve just the right visual layout for your network, you may need to “tune” a layout algorithm. You can do this by going to *Layout > Settings…* and then select the layout algorithm you want to tune. The settings expose the parameters of each algorithm so that you can explore different layout behaviours.

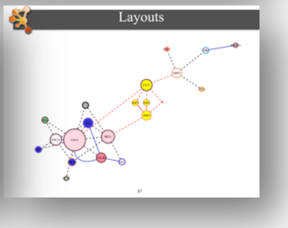


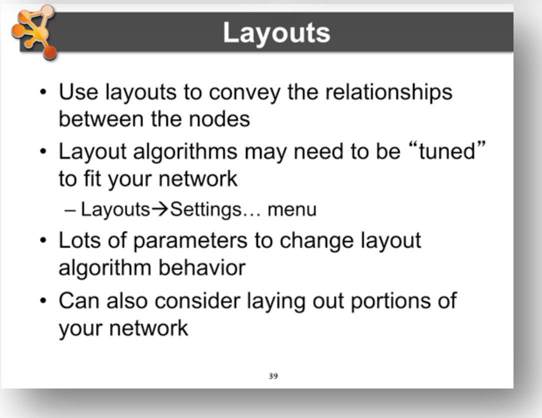
Layouts

- Types:
  - Force-Directed
    - simulate edges as springs
    - may be weighted or unweighted
  - Combining layouts
    - Use a general layout (force directed) for the entire graph, but use hierarchical or radial to focus on a particular portion
  - Multi-layer layouts
    - Partition graph, layout each partition then layout partitions
  - Many, many others

37







Layouts

- Use layouts to convey the relationships between the nodes
- Layout algorithms may need to be “tuned” to fit your network
  - Layouts→Settings… menu
- Lots of parameters to change layout algorithm behavior
- Can also consider laying out portions of your network

39

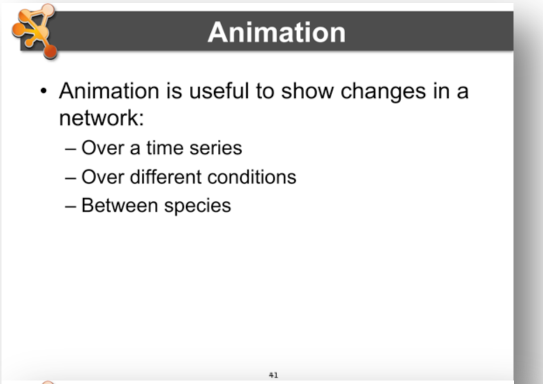
Notes

Animation

There are cases where a static image doesn’t tell the whole story. Perhaps you have collected data in a time series or have more than one condition you want to compare. Animation is a key technique for visualizing change.

There are a handful of plugins for Cytoscape that support automatic, step-wise animation through defined VizMapper styles (e.g., *clusterMaker* and VistaClara). But there is also a dedicated plugin called **CyAnimator** that supports flexible animation creation.

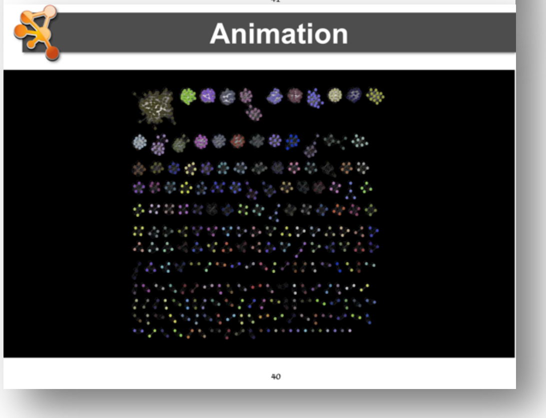
Using CyAnimator, you choose the frames of the animation as you work. Interpolation fills-in the transitions between frames for a wide range of visual features, including position, size, color, and opacity. A final set of images are generated and ready to be imported into a number of free or commercial movie file generators.



Animation

- Animation is useful to show changes in a network:
  - – Over a time series
  - – Over different conditions
  - – Between species

41



Animation

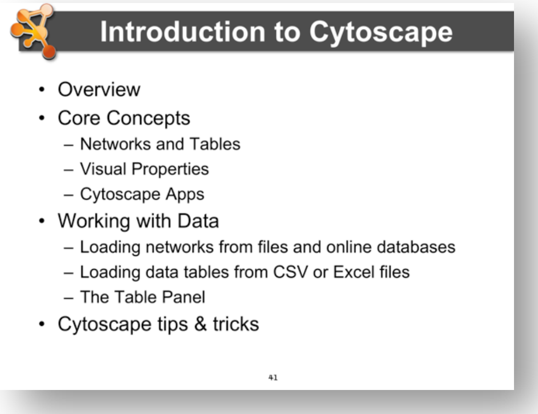
40

Notes

Introduction to Cytoscape

Cytoscape is an open source bioinformatics software platform for ***visualizing*** molecular interaction networks and biological pathways and ***integrating*** these networks with annotations, gene expression profiles and other state data. Although Cytoscape was originally designed for biological research, now it is a general platform for complex network analysis and visualization. Cytoscape *core* distribution provides a basic set of features for data integration and visualization. Additional features are available as ***plugins*** (now called ***Apps***). Plugins are available for network and molecular profiling analyses, new layouts, additional file format support, scripting, and connection with databases. Plugins may be developed by anyone using the Cytoscape open API based on Java™ technology and plugin community development is encouraged. Most of the plugins are freely available.

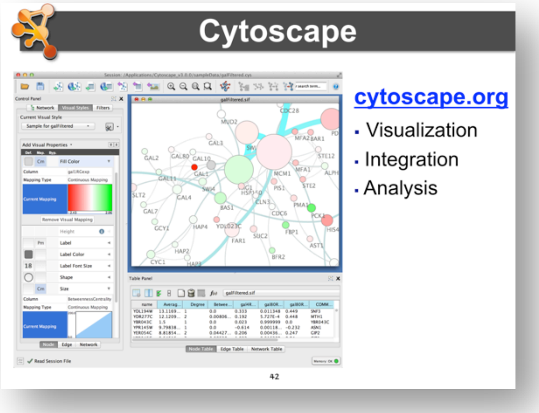
Cytoscape is a collaborative project between the Institute for Systems Biology (Leroy Hood lab), the University of California San Diego (Trey Ideker lab), Memorial Sloan-Kettering Cancer Center (Chris Sander lab), the Institut Pasteur (Benno Schwikowski lab), Agilent Technologies (Annette Adler team), the University of Toronto (Gary Bader Lab), Gladstone Institutes (Bruce Conklin and Alex Pico), and the University of California, San Francisco (Tom Ferrin lab).



Introduction to Cytoscape

- Overview
- Core Concepts
  - Networks and Tables
  - Visual Properties
  - Cytoscape Apps
- Working with Data
  - Loading networks from files and online databases
  - Loading data tables from CSV or Excel files
  - The Table Panel
- Cytoscape tips & tricks

41



Cytoscape

cytoscape.org

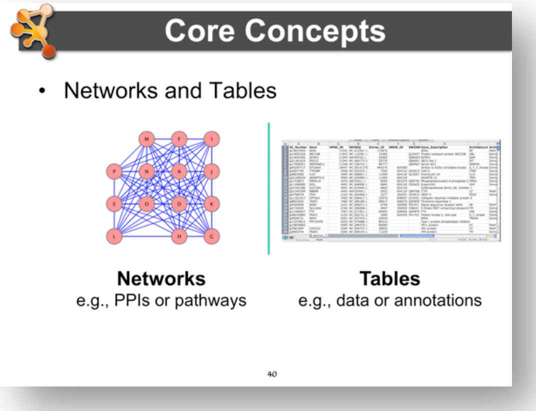
- Visualization
- Integration
- Analysis

42

Notes

Core Concepts

Cytoscape creates networks, where nodes of the network represent objects (such as proteins) and connecting edges represent relationships between them (such as physical interactions). Each Edge connects two Nodes. Edges can be directed or undirected. In the case of a directed edge, there is a Source and a Target Node. Once this basic network is created, various attributes of the nodes and edges (such as protein expression levels or strength of interaction) can be added to the network and incorporated as visual cues like shape or color.



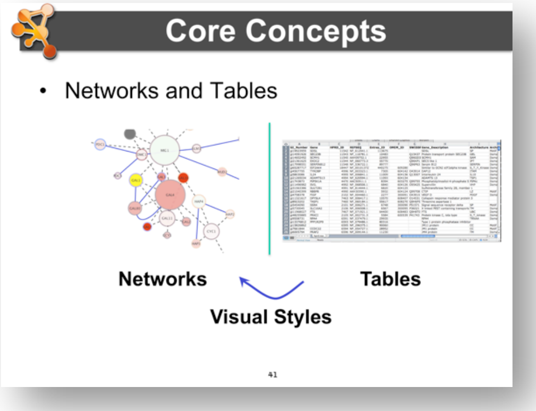
Core Concepts

- Networks and Tables

Networks
e.g., PPIs or pathways

Tables
e.g., data or annotations

40



Core Concepts

- Networks and Tables

Networks

Tables

Visual Styles

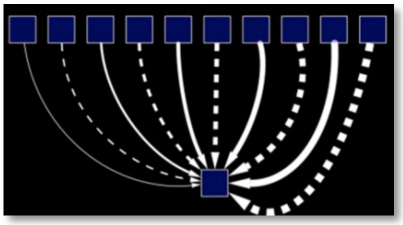
41

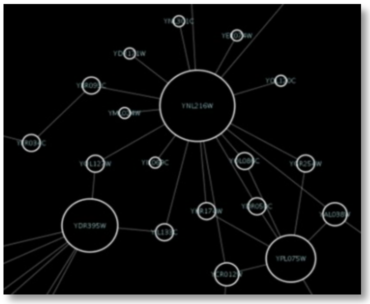
Notes

Visual Styles

One of Cytoscape's strengths in network visualization is the ability to allow users to encode any attribute of their data (name, type, degree, weight, expression data, etc.) as a visual property (such as color, size, transparency, or font type). A set of these encoded or mapped attributes is called a ***Visual Style*** and can be created or edited using the Cytoscape ***VizMapper***. With the VizMapper, the visual appearance of your network is easily customized. For example, you can:

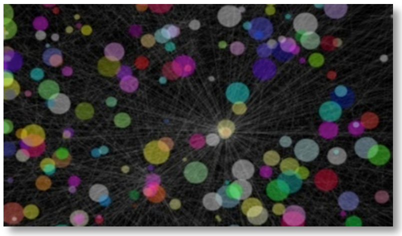
Use specific line types to indicate different types of interactions.

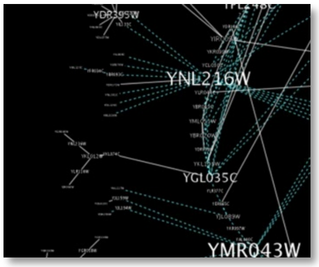




Set node sizes based on the degree of connectivity of the nodes.

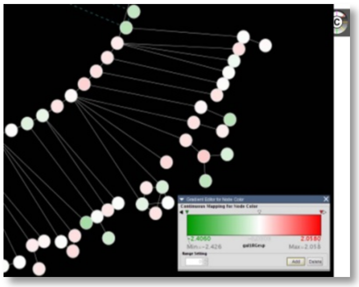
Browse extremely dense networks by controlling for the opacity of nodes.





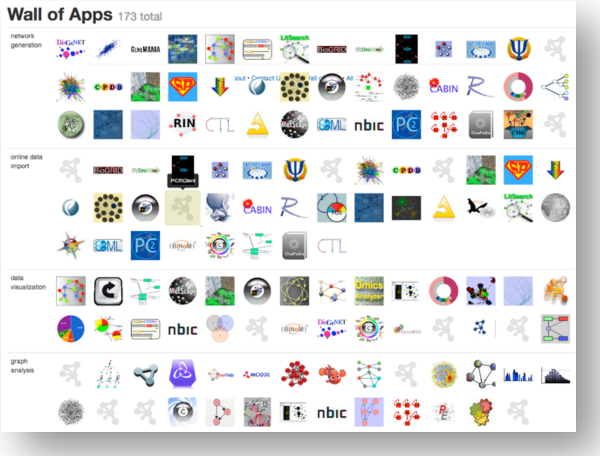
Set node font sizes based on the degree of connectivity of the nodes.

Visualize Gene Expression data its biological context by superimposing colors onto the nodes based upon their Gene Expression data values.



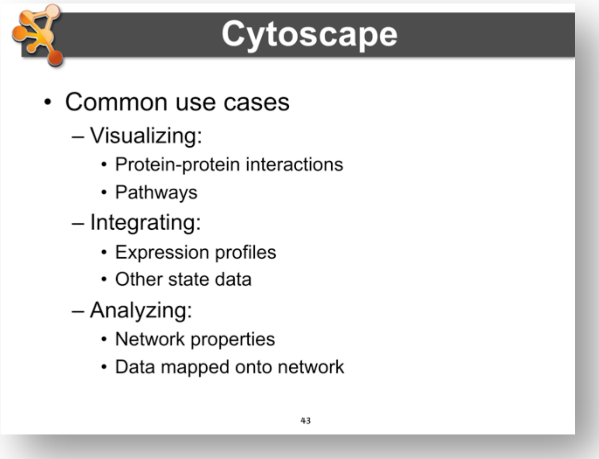
Apps

Cytoscape allows users to extend its functionality by creating or downloading additional software modules known as “Apps”. These extensions provide additional functionality in areas such as network data query and download services; network data integration and filtering; attribute-directed network layout; GO enrichment analysis[7]; as well as identification of network motifs, functional modules, protein complexes, or domain interactions.



Links to these apps can be found at http://apps.cytoscape.org

Altogether, Cytoscape and its Apps provide a powerful tool kit designed to help researchers answer specific biological questions using large amounts of cellular network and molecular profiling information.



Cytoscape

- Common use cases
  - Visualizing:
    - Protein-protein interactions
    - Pathways
  - Integrating:
    - Expression profiles
    - Other state data
  - Analyzing:
    - Network properties
    - Data mapped onto network

43

Notes

____

____

____

____

____

____

____

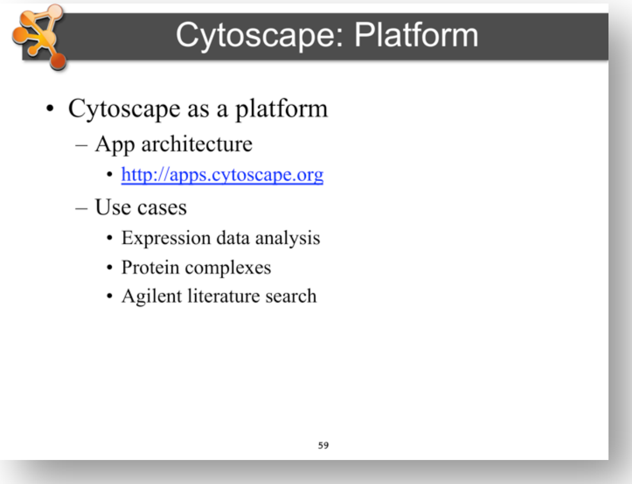
____

____

____

BiNGO

BiNGO is a Java-based tool to determine which Gene Ontology (GO) categories are statistically overrepresented in a set of genes or a subgraph of a biological network. BiNGO is implemented as a plugin for Cytoscape.



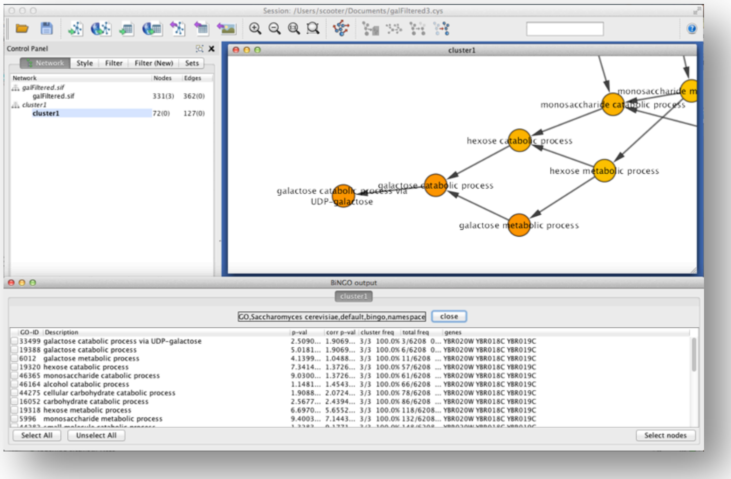
Cytoscape: Platform

- Cytoscape as a platform
  - App architecture
    - http://apps.cytoscape.org
  - Use cases
    - Expression data analysis
    - Protein complexes
    - Agilent literature search

59

BiNGO maps the predominant functional themes of a given gene set on the GO hierarchy, and outputs this mapping as a Cytoscape graph. Gene sets can either be selected or computed from a Cytoscape network (as subgraphs) or compiled from sources other than Cytoscape (e.g. a list of genes that are significantly upregulated in a microarray experiment).

The main advantage of BiNGO over other GO tools is the fact that it can be used directly and interactively on molecular interaction graphs. Another plus is that BiNGO takes full advantage of Cytoscape's versatile visualization environment. This allows you to produce customized high-quality figures.



Session: /Users/scooter/Documents/galFiltered3.cys
Control Panel
Network
Style
Filter
Filter (New)
Sets
cluster1
monosaccharide m
monosaccharide catabolic process
hexose catabolic process
hexose metabolic process
galactose catabolic process
galactose catabolic process via UDP-galactose
galactose metabolic process
BiNGO output
cluster1
close
Select All
Unselect All
Select nodes

Network	Nodes	Edges
galFiltered.sif		
galFiltered.sif	331(3)	362(0)
cluster1		
cluster1	72(0)	127(0)

GO,Saccharomyces cerevisiae,default,bingo,namespace

GO-ID	Description	p-val	corr p-val	cluster freq	total freq	genes
33499	galactose catabolic process via UDP-galactose	2.5090...	1.9069...	3/3 100.0%	3/6208 0...	YBR020W YBR018C YBR019C
19388	galactose catabolic process	5.0181...	1.9069...	3/3 100.0%	6/6208 0...	YBR020W YBR018C YBR019C
6012	galactose metabolic process	4.1399...	1.0488...	3/3 100.0%	11/6208 ...	YBR020W YBR018C YBR019C
19320	hexose catabolic process	7.3414...	1.3726...	3/3 100.0%	57/6208 ...	YBR020W YBR018C YBR019C
46365	monosaccharide catabolic process	9.0300...	1.3726...	3/3 100.0%	61/6208 ...	YBR020W YBR018C YBR019C
46164	alcohol catabolic process	1.1481...	1.4543...	3/3 100.0%	66/6208 ...	YBR020W YBR018C YBR019C
44275	cellular carbohydrate catabolic process	1.9088...	2.0724...	3/3 100.0%	78/6208 ...	YBR020W YBR018C YBR019C
16052	carbohydrate catabolic process	2.5677...	2.4394...	3/3 100.0%	86/6208 ...	YBR020W YBR018C YBR019C
19318	hexose metabolic process	6.6970...	5.6552...	3/3 100.0%	118/6208...	YBR020W YBR018C YBR019C
5996	monosaccharide metabolic process	9.4003...	7.1443...	3/3 100.0%	132/6208...	YBR020W YBR018C YBR019C

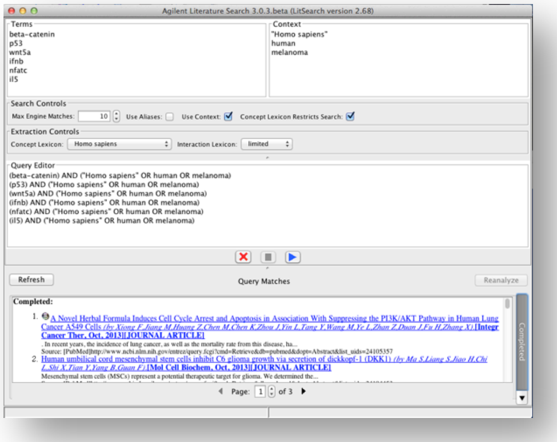
BiNGO features include:

- Assessing overrepresentation or underrepresentation of GO categories
- Graph or gene list input
- Batch mode: analyze several clusters simultaneously using same settings
- Different GO and GOSlim ontologies
- Wide range of organisms
- Evidence code filtering
- Hypergeometric or binomial test for overrepresentation
- Multiple testing correction using Bonferroni (FWER) or Benjamini&Hochberg (FDR) correction
- Interactive visualization of results mapped on the GO hierarchy.
- Extensive results in tab-delimited text file format
- Ability to make and use custom annotations, ontologies and reference sets

Agilent Literature Search

Agilent Literature Search Software is a meta-search tool for automatically querying multiple text-based search engines (both public and proprietary) in order to aid biologists faced with the daunting task of manually searching and extracting associations among genes/proteins of interest.

Agilent Literature Search Software can be used in conjunction with Cytoscape, which provides a means of generating an overview network view of gene/protein associations.


Agilent Literature Search 3.0.3.beta (LitSearch version 2.68)
Refresh
Query Matches
Reanalyze
Completed

Terms
beta-catenin
p53
wnt5a
ifnb
nfatc
il5

Context
"Homo sapiens"
human
melanoma

Search Controls
Max Engine Matches: 10 Use Aliases: ☐ Use Context: ☑ Concept Lexicon Restricts Search: ☑

Extraction Controls
Concept Lexicon: Homo sapiens Interaction Lexicon: limited

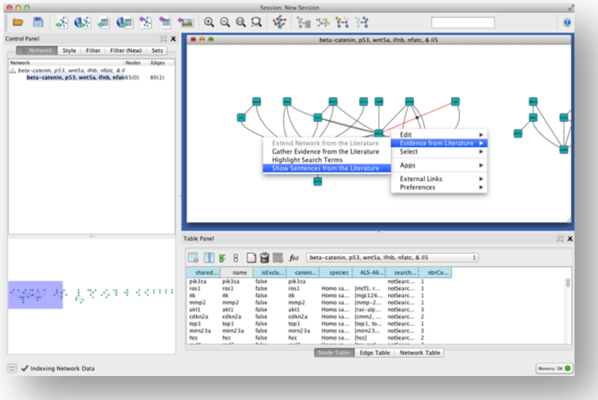
Query Editor
```
(beta-catenin) AND ("Homo sapiens" OR human OR melanoma)
(p53) AND ("Homo sapiens" OR human OR melanoma)
(wnt5a) AND ("Homo sapiens" OR human OR melanoma)
(ifnb) AND ("Homo sapiens" OR human OR melanoma)
(nfatc) AND ("Homo sapiens" OR human OR melanoma)
(il5) AND ("Homo sapiens" OR human OR melanoma)
```

Completed:

1. A Novel Herbal Formula Induces Cell Cycle Arrest and Apoptosis in Association With Suppressing the PI3K/AKT Pathway in Human Lung Cancer A549 Cells *(by Xiong F,Jiang M,Huang Z,Chen M,Chen K,Zhou J,Yin L,Tang Y,Wang M,Ye L,Zhan Z,Duan J,Fu H,Zhang X)*.**[Integr Cancer Ther, Oct, 2013][JOURNAL ARTICLE]**
. In recent years, the incidence of lung cancer, as well as the mortality rate from this disease, ha...
Source: [PubMed]http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=pubmed&dopt=Abstract&list_uids=24105357
2. Human umbilical cord mesenchymal stem cells inhibit C6 glioma growth via secretion of dickkopf-1 (DKK1) *(by Ma S,Liang S,Jiao H,Chi L,Shi X,Tian Y,Yang B,Guan F)*.**[Mol Cell Biochem, Oct, 2013][JOURNAL ARTICLE]**
Mesenchymal stem cells (MSCs) represent a potential therapeutic target for glioma. We determined the...

Page: 1 of 3

Agilent Literature Search software provides an easy-to-use interface to its powerful querying capabilities. When a query is entered, it is submitted to multiple user-selected search engines, and the retrieved results (documents) are fetched from their respective sources. Each document is then parsed into sentences and analyzed for protein-protein associations. Agilent Literature Search Software uses a set of "context" files (lexicons) for defining protein names (and aliases) and association terms (verbs) of interest. Associations extracted from these documents are collected into a Cytoscape network. The sentences and source hyperlinks for each association are further stored as attributes of the corresponding Cytoscape edges.



Network	Nodes	Edges
beta-catenin, p53, wnt5a, ifnb, nfatc, & il		
beta-catenin, p53, wnt5a, ifnb, nfat	65(0)	80(1)

beta-catenin, p53, wnt5a, ifnb, nfatc, & il5

shared...	name	isExclu...	canoni...	species	ALS-Ali...	search...	nbrCo...
pik3ca	pik3ca	false	pik3ca			notSearc...	1
ros1	ros1	false	ros1	Homo sa...	[mcf3, r...	notSearc...	1
itk	itk	false	itk	Homo sa...	[mgc126...	notSearc...	1
mmp2	mmp2	false	mmp2	Homo sa...	[mmp-2...	notSearc...	1
akt1	akt1	false	akt1	Homo sa...	[rac-alp...	notSearc...	3
cdkn2a	cdkn2a	false	cdkn2a	Homo sa...	[cmm2, ...	notSearc...	2
top1	top1	false	top1	Homo sa...	[top1, to...	notSearc...	1
mirn23a	mirn23a	false	mirn23a	Homo sa...	[mirn23...	notSearc...	3
hcc	hcc	false	hcc	Homo sa...	[hcc]	notSearc...	2

✓ Indexing Network Data

Agilent Literature Search Plugin Features:

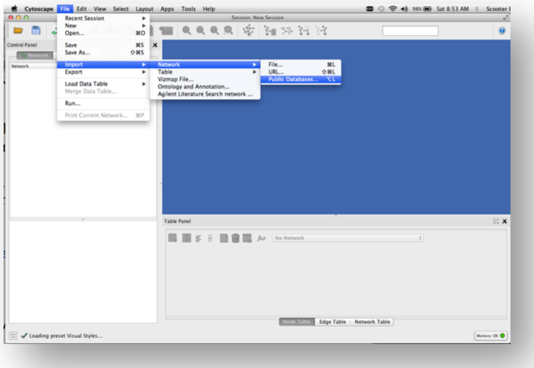
- Meta-search engine combining Information Retrieval & Knowledge Extraction
- PubMed, OMIM, USPTO
- Load/Save/Reanalyze search results
- Paged Search results view
- User context-based aliasing
- File-based lexicon management
- Symbol identification, interaction extraction
- Cytoscape session load/save compatible
- Putative network generation from literature
- Literature-based evidence gathering for Cytoscape Edges
- Extend a Cytoscape network with associations extracted from the literature

Revised 09/17/2014

Loading Networks

There are 4 different ways of creating networks in Cytoscape:

1. Importing networks from Public Databases
2. Importing pre-existing, unformatted text or Excel files.
3. Importing pre-existing, formatted network files..
4. Creating an empty network and manually adding nodes and edges.

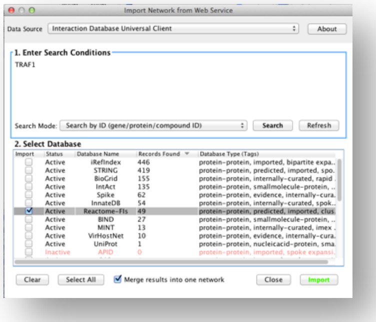


Loading Networks from a Web Service

In this section we will look at how to import networks using publicly accessible databases.

First, select the **File→Import→Network** from Public Databases menu item. By default, Cytoscape only provides one public database item (Interaction Database Universal Client), but there are several others available as apps.

Step 1: Search. Type in a search term or set of search terms separated by newlines. In this example we use the Interaction Database Universal Client, and enter TRAF1 as our search term.



Data Source: Interaction Database Universal Client

1. Enter Search Conditions

TRAF1

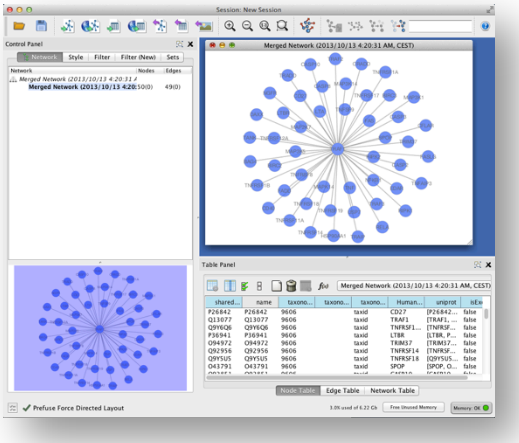
Search Mode: Search by ID (gene/protein/compound ID)

2. Select Database

Import	Status	Database Name	Records Found	Database Type (Tags)
□	Active	iRefIndex	446	protein-protein, imported, bipartite expa..
□	Active	STRING	419	protein-protein, predicted, imported, spo.
□	Active	BioGrid	155	protein-protein, internally-curated, rapid .
□	Active	IntAct	135	protein-protein, smallmolecule-protein, ..
□	Active	Spike	62	protein-protein, evidence, internally-cura.
□	Active	InnateDB	54	protein-protein, internally-curated, spok..
☑	Active	Reactome-FIs	49	protein-protein, predicted, imported, clus.
□	Active	BIND	27	protein-protein, smallmolecule-protein, ..
□	Active	MINT	13	protein-protein, internally-curated, imex .
□	Active	VirHostNet	10	protein-protein, evidence, internally-cura.
□	Active	UniProt	1	protein-protein, nucleicacid-protein, sma.
□	Inactive	APID	0	protein-protein, imported, spoke expansi.

☑ Merge results into one network

Step 2: Select. Select Reactome-FIs as our database source and select Import. This will import the interactions provided by Reactome into a Cytoscape network. When the network has successfully loaded, you will see it displayed in the top center panel (Network View). There will also be a 'birds eye' overview in bottom-left panel that shows the entire network.



Network	Nodes	Edges
Merged Network (2013/10/13 4:20:31 A		
Merged Network (2013/10/13 4:20:	50(0)	49(0)

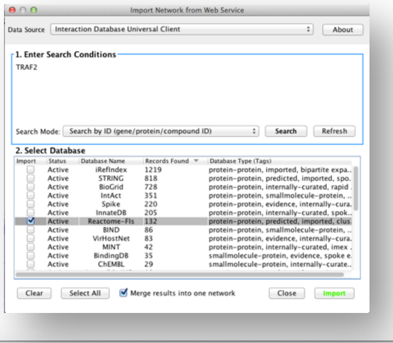
Merged Network (2013/10/13 4:20:31 AM, CEST)

shared...	name	taxono...	taxono...	taxono...	Human...	uniprot	isEx
P26842	P26842	9606		taxid	CD27	[P26842...	false
Q13077	Q13077	9606		taxid	TRAF1	[TRAF1, ...	false
Q9Y6Q6	Q9Y6Q6	9606		taxid	TNFRSF1...	[TNFRSF...	false
P36941	P36941	9606		taxid	LTBR	[LTBR, P...	false
O94972	O94972	9606		taxid	TRIM37	[TRIM37...	false
Q92956	Q92956	9606		taxid	TNFRSF14	[TNFRSF...	false
Q9Y5U5	Q9Y5U5	9606		taxid	TNFRSF18	[Q9Y5U5...	false
O43791	O43791	9606		taxid	SPOP	[SPOP, O...	false

✓ Prefuse Force Directed Layout

3.0% used of 6.22 Gb

Now let’s extend our network by merging in the known protein-protein interactions for TRAF2.   Follow the same procedure as above but this time select TRAF2 in Step 1 to search and import the interactions from Reactome.



Import Network from Web Service

Data Source: Interaction Database Universal Client

1. Enter Search Conditions

TRAF2

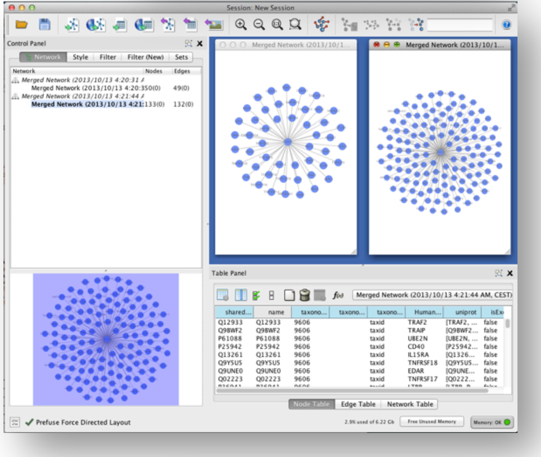
Search Mode: Search by ID (gene/protein/compound ID)

2. Select Database

Import	Status	Database Name	Records Found	Database Type (Tags)
□	Active	iRefIndex	1219	protein-protein, imported, bipartite expa..
□	Active	STRING	818	protein-protein, predicted, imported, spo.
□	Active	BioGrid	728	protein-protein, internally-curated, rapid .
□	Active	IntAct	351	protein-protein, smallmolecule-protein, ..
□	Active	Spike	220	protein-protein, evidence, internally-cura.
□	Active	InnateDB	205	protein-protein, internally-curated, spok..
☑	Active	Reactome-FIs	132	protein-protein, predicted, imported, clus.
□	Active	BIND	86	protein-protein, smallmolecule-protein, ..
□	Active	VirHostNet	83	protein-protein, evidence, internally-cura.
□	Active	MINT	42	protein-protein, internally-curated, imex .
□	Active	BindingDB	35	smallmolecule-protein, evidence, spoke e.
□	Active	ChEMBL	29	smallmolecule-protein, internally-curate..

☑ Merge results into one network

This will bring up the protein-protein interaction network for TRAF2. The image to the right shows the two star networks: one for TRAF1 and one for TRAF2. We can use Cytoscape’s internal tools to merge them.



Session: New Session

Control Panel

Network	Nodes	Edges
Merged Network (2013/10/13 4:20:31 A		
Merged Network (2013/10/13 4:20:35	0(0)	49(0)
Merged Network (2013/10/13 4:21:44 A		
Merged Network (2013/10/13 4:21:	133(0)	132(0)

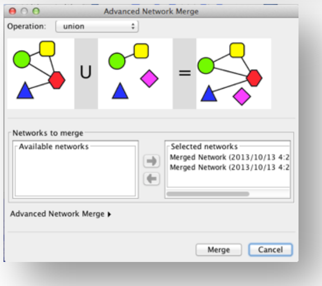
Table Panel: Merged Network (2013/10/13 4:21:44 AM, CEST)

shared...	name	taxono...	taxono...	taxono...	Human...	uniprot	isEx
Q12933	Q12933	9606		taxid	TRAF2	[TRAF2, ...	false
Q9BWF2	Q9BWF2	9606		taxid	TRAIP	[Q9BWF2...	false
P61088	P61088	9606		taxid	UBE2N	[UBE2N, ...	false
P25942	P25942	9606		taxid	CD40	[P25942...	false
Q13261	Q13261	9606		taxid	IL15RA	[Q1326...	false
Q9Y5U5	Q9Y5U5	9606		taxid	TNFRSF18	[Q9Y5U5...	false
Q9UNE0	Q9UNE0	9606		taxid	EDAR	[Q9UNE...	false
Q02223	Q02223	9606		taxid	TNFRSF17	[Q0222...	false

✓ Prefuse Force Directed Layout

2.9% used of 6.22 Gb

Under the Cytoscape **Tools** menu, select **Merge Networks**. Select both networks (TRAF1 and TRAF2) that you want to merge and click “Merge”.



Advanced Network Merge

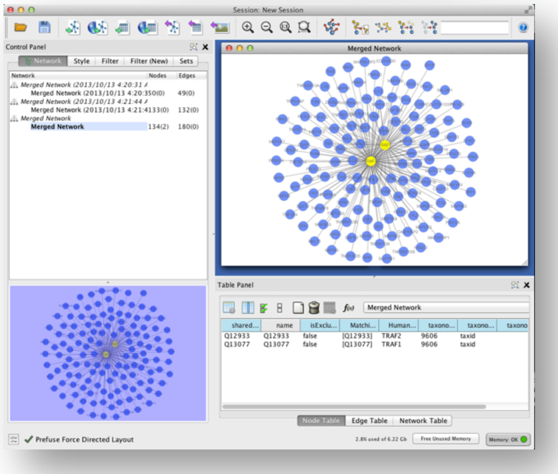
Operation: union

Networks to merge

Available networks

Selected networks
Merged Network (2013/10/13 4:2
Merged Network (2013/10/13 4:2

The combined TRAF1/TRAF2 protein-protein interaction network will be displayed (the image at right shows TRAF1 and TRAF2 selected for clarity).



Session: New Session

Control Panel

Network	Nodes	Edges
Merged Network (2013/10/13 4:20:31 A		
Merged Network (2013/10/13 4:20:35	0(0)	49(0)
Merged Network (2013/10/13 4:21:44 A		
Merged Network (2013/10/13 4:21:41	33(0)	132(0)
Merged Network		
Merged Network	134(2)	180(0)

Table Panel: Merged Network

shared...	name	isExclu...	Matchi...	Human...	taxono...	taxono...	taxono
Q12933	Q12933	false	[Q12933]	TRAF2	9606	taxid	
Q13077	Q13077	false	[Q13077]	TRAF1	9606	taxid	

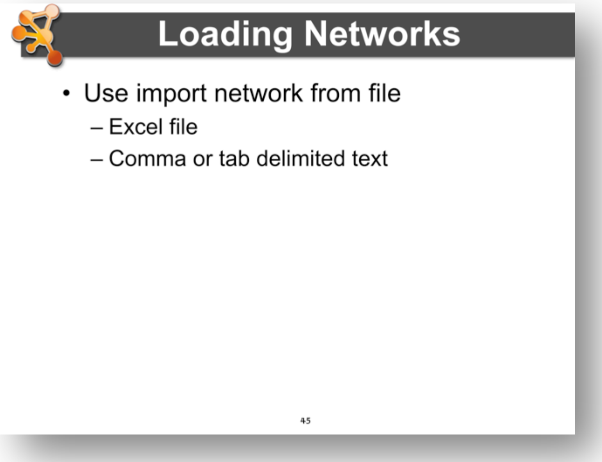
✓ Prefuse Force Directed Layout

2.8% used of 6.22 Gb

Load a Network from a Table

In this section we will explore how to create Cytoscape network by importing a pre-existing text or Excel file. The figure at right shows one such example network, consisting of four nodes and four edges.

Let's begin creating the network by selecting the **File→Import→Network → File…** menu item.

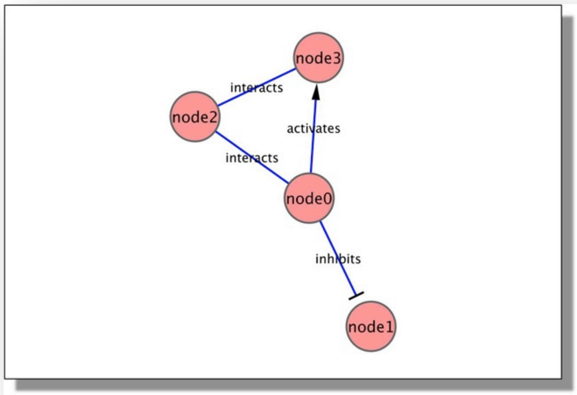


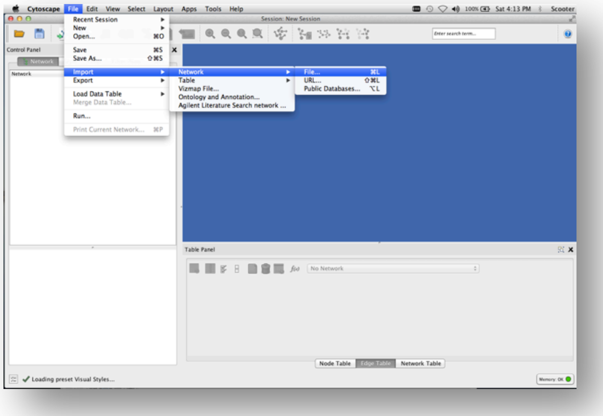
Loading Networks

- Use import network from file
  - Excel file
  - Comma or tab delimited text

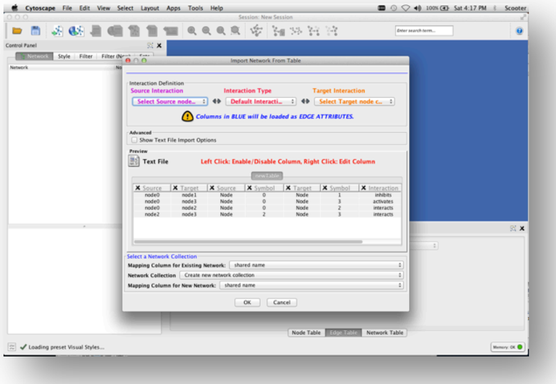
45

Notes





An interactive graphical user interface allows you to specify parsing options for specified files. The screen provides a preview that shows how the file will be parsed given the current configuration. As the configuration changes, the preview updates automatically. In addition to specifying how the file will be parsed, you also choose the columns that represent the Source nodes, the Target nodes, and an optional edge interaction type.



Import Network From Table

Interaction Definition

Source Interaction: Select Source node... | Interaction Type: Default Interacti... | Target Interaction: Select Target node c...

Columns in BLUE will be loaded as EDGE ATTRIBUTES.

Advanced

□ Show Text File Import Options

Preview

Text File — Left Click: Enable/Disable Column, Right Click: Edit Column

newTable

✕ Source	✕ Target	✕ Source	✕ Symbol	✕ Target	✕ Symbol	✕ Interaction
node0	node1	Node	0	Node	1	inhibits
node0	node3	Node	0	Node	3	activates
node0	node2	Node	0	Node	2	interacts
node2	node3	Node	2	Node	3	interacts

Select a Network Collection

Mapping Column for Existing Network: shared name

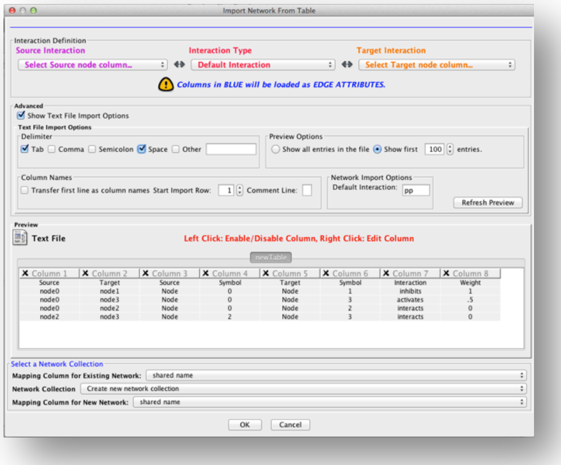
Network Collection: Create new network collection

Mapping Column for New Network: shared name

OK | Cancel

✓ Loading preset Visual Styles...

Under the **Advanced** section, check the checkbox labeled **Show Text File Import Options.**



Import Network From Table

Interaction Definition

Source Interaction: Select Source node column... | Interaction Type: Default Interaction | Target Interaction: Select Target node column...

Columns in BLUE will be loaded as EDGE ATTRIBUTES.

Advanced

☑ Show Text File Import Options

Text File Import Options

Delimiter: ☑ Tab □ Comma □ Semicolon ☑ Space □ Other

Preview Options: ○ Show all entries in the file ● Show first 100 entries.

Column Names: □ Transfer first line as column names Start Import Row: 1 Comment Line:

Network Import Options: Default Interaction: pp

Refresh Preview

Preview

Text File — Left Click: Enable/Disable Column, Right Click: Edit Column

newTable

✕ Column 1	✕ Column 2	✕ Column 3	✕ Column 4	✕ Column 5	✕ Column 6	✕ Column 7	✕ Column 8
Source	Target	Source	Symbol	Target	Symbol	Interaction	Weight
node0	node1	Node	0	Node	1	inhibits	1
node0	node3	Node	0	Node	3	activates	.5
node0	node2	Node	0	Node	2	interacts	0
node2	node3	Node	2	Node	3	interacts	0

Select a Network Collection

Mapping Column for Existing Network: shared name

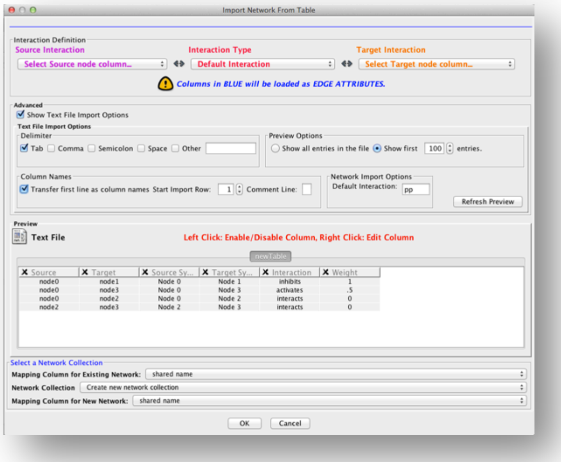
Network Collection: Create new network collection

Mapping Column for New Network: shared name

OK | Cancel

You will see a set of checkboxes appear. These allow you to choose the:

- **Delimiter**. The delimiter character that separates columns (fields) in the import file. This can be a tab, comma, semicolon, space, or any arbitrary delimiter character that you define.
- **Preview options**. This is a control for how many preview lines you see in the bottom Preview pane of the dialog. You can set this to preview all entries in the file or a subset of the entries (typically the first 100 entries).
- **Attribute Names**. You can choose whether to use the first line of the file to supply attribute names, one name per delimited column in the file.
- **Start import row**. You can set the import line number so that you can skip over any initial header or **comment** lines in the file.
- **Comment Line**. You can indicate a character, e.g. '#', to distinguish comment lines in the import files, so that they are not treated as network data.
- **Default Interaction**: You can set the name of the Default Interaction type, which is used to name an edge. The example in our figure uses 'pp' (for protein-protein interaction) as its default interaction.



Import Network From Table

Interaction Definition

Source Interaction: Select Source node column... | Interaction Type: Default Interaction | Target Interaction: Select Target node column...

Columns in BLUE will be loaded as EDGE ATTRIBUTES.

Advanced

☑ Show Text File Import Options

Text File Import Options

Delimiter: ☑ Tab □ Comma □ Semicolon □ Space □ Other

Preview Options: ○ Show all entries in the file ● Show first 100 entries.

Column Names: ☑ Transfer first line as column names Start Import Row: 1 Comment Line:

Network Import Options: Default Interaction: pp

Refresh Preview

Preview

Text File — Left Click: Enable/Disable Column, Right Click: Edit Column

newTable

✕ Source	✕ Target	✕ Source Sy...	✕ Target Sy...	✕ Interaction	✕ Weight
node0	node1	Node 0	Node 1	inhibits	1
node0	node3	Node 0	Node 3	activates	.5
node0	node2	Node 0	Node 2	interacts	0
node2	node3	Node 2	Node 3	interacts	0

Select a Network Collection

Mapping Column for Existing Network: shared name

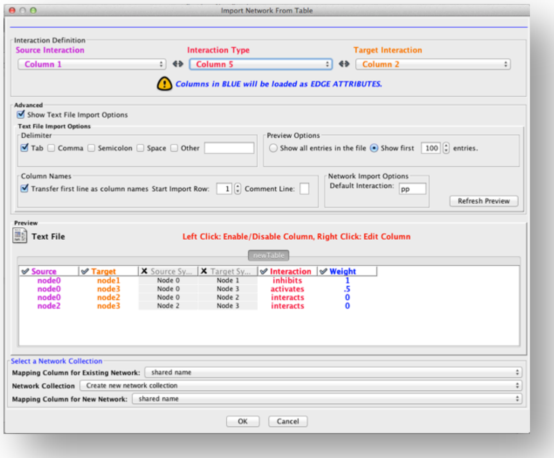
Network Collection: Create new network collection

Mapping Column for New Network: shared name

OK | Cancel

Now use the **Source Interaction**, **Interaction Type**, and **Target Interaction** combo boxes in the **Interaction Definition** to chose the columns for edge source, edge interaction type, and edge target, respectively. The figure above shows **Column 1** is being used for **Source Interaction, Column 5** for **Interaction Type**, and **Column 2** for **Target Interaction.**

You can also use additional columns of data as edge attribute values by checking their column header. In this case the column labeled **Weight** has been checked and marked blue to indicate that its values will be loaded as edge attributes.



Source Interaction: Column 1 ⇔ Interaction Type: Column 5 ⇔ Target Interaction: Column 2

Columns in BLUE will be loaded as EDGE ATTRIBUTES.

☑ Show Text File Import Options

Delimiter: ☑ Tab ☐ Comma ☐ Semicolon ☐ Space ☐ Other

Preview Options: ○ Show all entries in the file ● Show first 100 entries.

Column Names: ☑ Transfer first line as column names Start Import Row: 1 Comment Line:

Network Import Options: Default Interaction: pp

Left Click: Enable/Disable Column, Right Click: Edit Column

✓ Source	✓ Target	✗ Source Sy...	✗ Target Sy...	✓ Interaction	✓ Weight
node0	node1	Node 0	Node 1	inhibits	1
node0	node3	Node 0	Node 3	activates	.5
node0	node2	Node 0	Node 2	interacts	0
node2	node3	Node 2	Node 3	interacts	0

Select a Network Collection

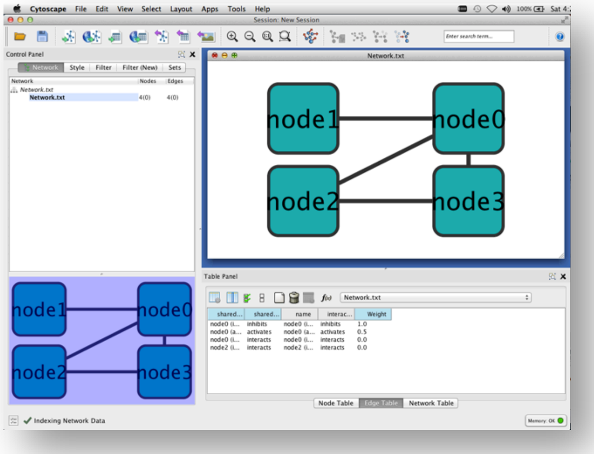
Mapping Column for Existing Network: shared name

Network Collection: Create new network collection

Mapping Column for New Network: shared name

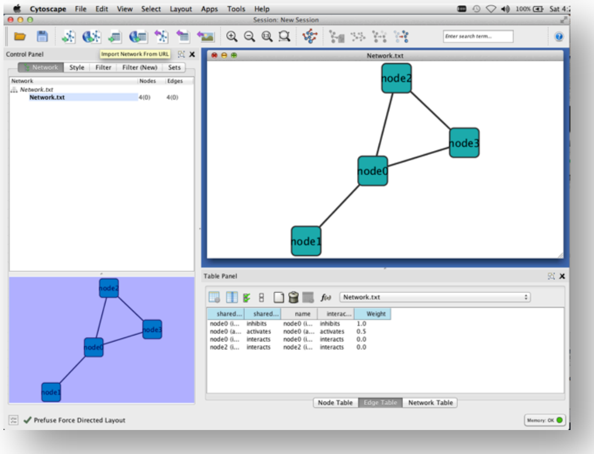
When you are satisfied with the settings, press the import button and the network will be imported. You will see a figure that looks like the figure on right.

The values of edge attributes can be used as arguments to



shared...	shared...	name	interac...	Weight
node0 (i...	inhibits	node0 (i...	inhibits	1.0
node0 (a...	activates	node0 (a...	activates	0.5
node0 (i...	interacts	node0 (i...	interacts	0.0
node2 (i...	interacts	node2 (i...	interacts	0.0

graph layout and other computational operations. In the bottom example on right, the attribute for **Weight** is used in the calculation of coordinate positions using Cytoscape's. **Force-Directed Layout.**



shared...	shared...	name	interac...	Weight
node0 (i...	inhibits	node0 (i...	inhibits	1.0
node0 (a...	activates	node0 (a...	activates	0.5
node0 (i...	interacts	node0 (i...	interacts	0.0
node2 (i...	interacts	node2 (i...	interacts	0.0

Notes

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

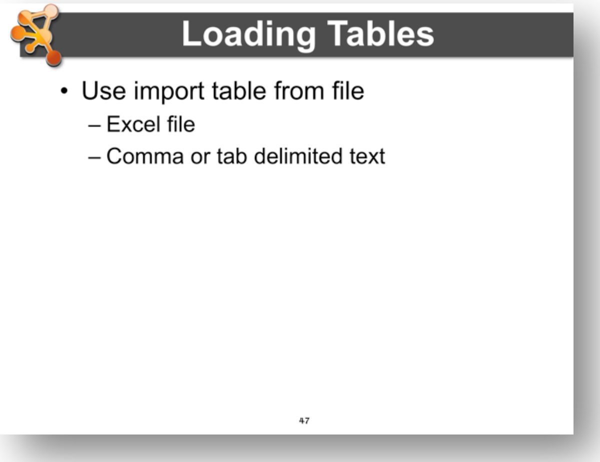
____

Load Tables

In this section we will explore how to create Cytoscape attributes and values by importing a pre-existing text or Excel file.

Let's begin creating the network by selecting the **File -> Import -> Table -> File…** menu item.

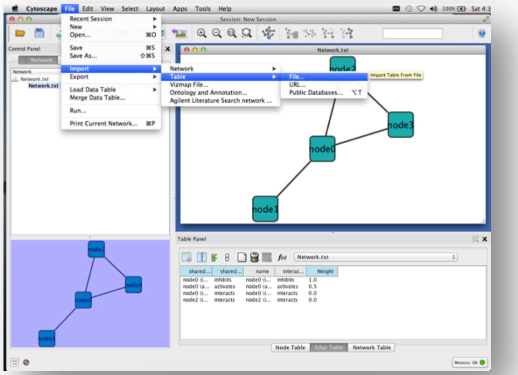
Select under **Advanced** section the **Show Mapping Options** and **Show Text File Import Options** checkboxes, which set up dialogs for controlling Text Import options (as in example with networks) and **Show Mapping Options,** which give you a number of options for associating network nodes with elements in the dataset. This enables us to encode the data as visual properties, such as color, shape, and overlay the network nodes with the values of those properties.



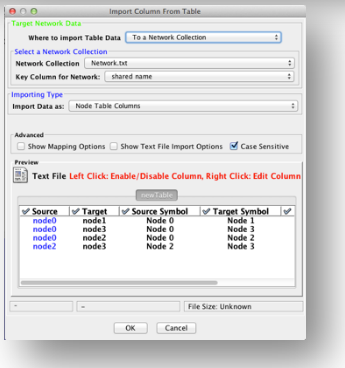
Loading Tables

- Use import table from file
  - Excel file
  - Comma or tab delimited text

47



shared…	shared…	name	interac…	Weight
node0 (i…	inhibits	node0 (i…	inhibits	1.0
node0 (a…	activates	node0 (a…	activates	0.5
node0 (i…	interacts	node0 (i…	interacts	0.0
node2 (i…	interacts	node2 (i…	interacts	0.0



Where to import Table Data: To a Network Collection

Network Collection: Network.txt

Key Column for Network: shared name

Import Data as: Node Table Columns

Advanced

□ Show Mapping Options □ Show Text File Import Options ☑ Case Sensitive

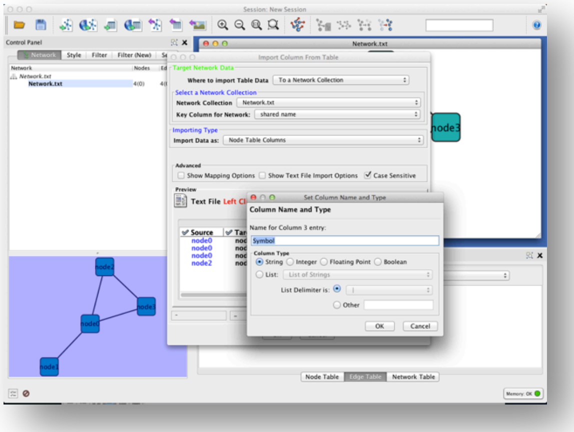
Text File Left Click: Enable/Disable Column, Right Click: Edit Column

✓ Source	✓ Target	✓ Source Symbol	✓ Target Symbol	✓
node0	node1	Node 0	Node 1	
node0	node3	Node 0	Node 3	
node0	node2	Node 0	Node 2	
node2	node3	Node 2	Node 3	

File Size: Unknown

Notes

Now we are ready to map the nodes of the network to the data you have.



Import Column From Table

Target Network Data

Where to import Table Data: To a Network Collection

Select a Network Collection

Network Collection: Network.txt

Key Column for Network: shared name

Importing Type

Import Data as: Node Table Columns

Advanced

□ Show Mapping Options □ Show Text File Import Options ☑ Case Sensitive

Preview

Text File Left Cl

Set Column Name and Type

Column Name and Type

Name for Column 3 entry: Symbol

Column Type

● String ○ Integer ○ Floating Point ○ Boolean

○ List: List of Strings

List Delimiter is: ● |

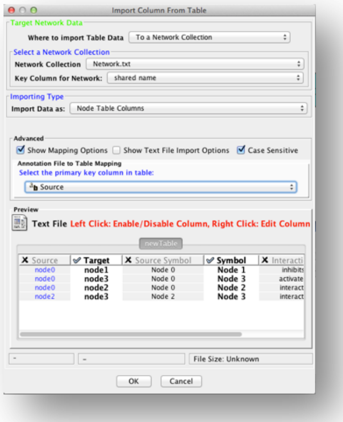
○ Other

OK Cancel

If you right-click on a column header, a dialog box will be displayed. You can fill in a name for the attribute. You can also set the type of the elements in the data column, to one of: the primitive data types that Cytoscape supports are: String , Integer, Floating Point, and Boolean. You can also set the datatype of the column to be a list of primitive elements of one datatype.

Now you need to map unique identifiers between the entries in the data and the nodes in the network. The key point of this is to identify which nodes in the network are equivalent to which entries in the table. This enables mapping of data values into visual properties like Color and Shape. This kind of mapping is typically done by comparing the unique Identifier attribute value for each node with the unique Identifier value for each data value. As a default, Cytoscape looks for an attribute value of 'ID' in the network and a user-supplied *Primary Key* in the dataset. The user can change these values via combo boxes in the **Mapping** section:

- **Primary Key:** combo box that allows you to choose the column that is to be used as key for mapping values in the dataset. You can also set an arbitrary number of columns as **aliases** via checkbox, in which case those supplied alias will be used in addition to the Primary Key in the attempt to map identifiers.
- **Key Attribute for Network:** combo box that allow you to set the node attribute that is to be used as used as key to map to.



Import Column From Table

Target Network Data

Where to import Table Data: To a Network Collection

Select a Network Collection

Network Collection: Network.txt

Key Column for Network: shared name

Importing Type

Import Data as: Node Table Columns

Advanced

☑ Show Mapping Options □ Show Text File Import Options ☑ Case Sensitive

Annotation File to Table Mapping

Select the primary key column in table: Source

Preview

Text File Left Click: Enable/Disable Column, Right Click: Edit Column

newTable

✗ Source	✓ Target	✗ Source Symbol	✓ Symbol	✗ Interacti
node0	node1	Node 0	Node 1	inhibits
node0	node3	Node 0	Node 3	activate
node0	node2	Node 0	Node 2	interact
node2	node3	Node 2	Node 3	interact

File Size: Unknown

OK Cancel

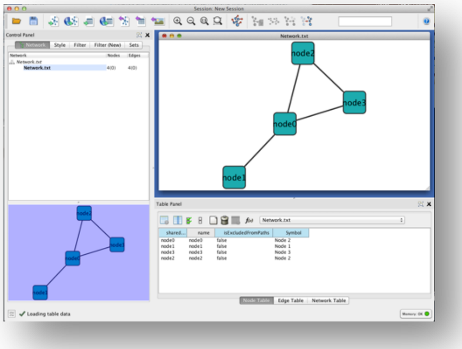
If there is a match between the value of a Primary Key in the dataset and and the value the Key Attribute For Network field in the network, then all attribute-value pairs associated with the element in the dataset are assigned as well to the matching node in the network.

You can control some of the options for ID Mapping by using the controls in the **Advanced** section. Select under **Advanced** section the **Show Mapping Options. Show Mapping Options,** which give you a number of options for associating network nodes with elements in the dataset. This enables us to encode the data as visual properties, such as color, shape, and overlay the network nodes with the values of those properties.

There is a checkbox entitled "Import everything (Key is always ID)". If this is checked, Cytoscape will create an attribute value pair for every entry in the dataset, regardless of whether it maps to an equivalent attribute name. Now check **Show Mapping Options** checkbox under **Advanced** section. You can also check the **Case Sensitive** checkbox if the string comparison is to be done in a case insensitive way (e.g. if stat1 gene is the same as STAT2 gene).

Once attributes are imported, we can inspect them in the Data Panel. Right clicking on the leftmost icon in the toolbar under DataPanel title bar will bring up a menu of checkboxes. If you check a checkbox for any attribute in that attribute and its values will be displayed as a column in of values on the DataPanel.

If you a select a node in the network, its attribute values will be displayed in the control panel, but by default, if no nodes are selected, all attribute rows are shown.



Network	Nodes	Edges
Network.txt		
Network.txt	4(0)	4(0)

Network.txt

shared...	name	isExcludedFromPaths	Symbol
node0	node0	false	Node 2
node1	node1	false	Node 1
node3	node3	false	Node 3
node2	node2	false	Node 2

Notes

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

____

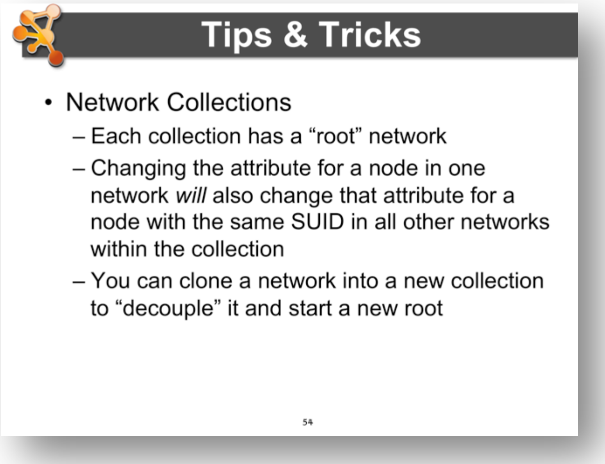
____

____

____

Tips and Tricks

Cytoscape is a large, complex, and dynamic software system. A little knowledge of the internals organization and operational model of the software will enable more efficient use of the software. Here are some useful Tips & Tricks to help you get the most out of your Cytoscape usage.



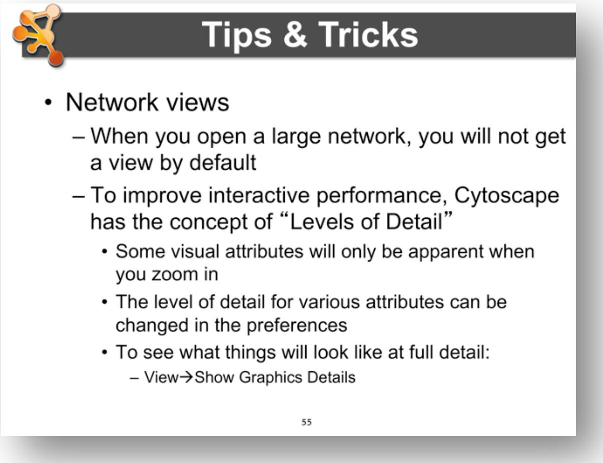
Tips & Tricks

- Network Collections
  - Each collection has a “root” network
  - Changing the attribute for a node in one network *will* also change that attribute for a node with the same SUID in all other networks within the collection
  - You can clone a network into a new collection to “decouple” it and start a new root

54

The “’Root Graph”

There is one central *root graph* that contains all nodes and edges. Thus all networks are ‘views’ on that single graph, and nodes and edges are unique across all networks. Modifying a node in one network will modify that node in all other networks that it appears in. There is no way to have two or more copies of a node with the same ID. The only workaround would be to make a copy of a Cytoscape session.



Tips & Tricks

- Network views
  - When you open a large network, you will not get a view by default
  - To improve interactive performance, Cytoscape has the concept of “Levels of Detail”
    - Some visual attributes will only be apparent when you zoom in
    - The level of detail for various attributes can be changed in the preferences
    - To see what things will look like at full detail:
      - View→Show Graphics Details

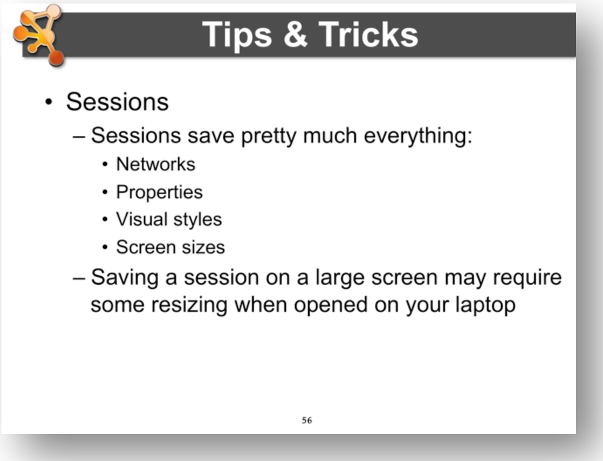
55

Network Views

For efficiency in dealing with large networks, a view is not automatically generated when the size of the network is over a user-definable threshold. You can manually generate a Network View by right-clicking on its entry in the Network Navigator Panel (upper left o Cytoscape desktop)., then selecting ‘Create View’. You can also use that right-menu item to ‘Destroy View’, ‘Destroy Network”, and edit the Network’s title.

To improve interactive performance, Cytoscape has the concept of *Levels of Detail*? This is basically a mechanism for *semantic zooming*, where different levels of detail com into play at different levels of detail (think of the Google Maps interface where a City is represented by a yellow patch at high level then shows more of the structure of streets and avenues as you zoom in.

Some Cytoscape attributes will only be apparent when you zoom in. The level of detail for various attributes can be changed in the preferences. To see what things look like in full detail, select the **View→Show Graphics Details** menu item..



Tips & Tricks

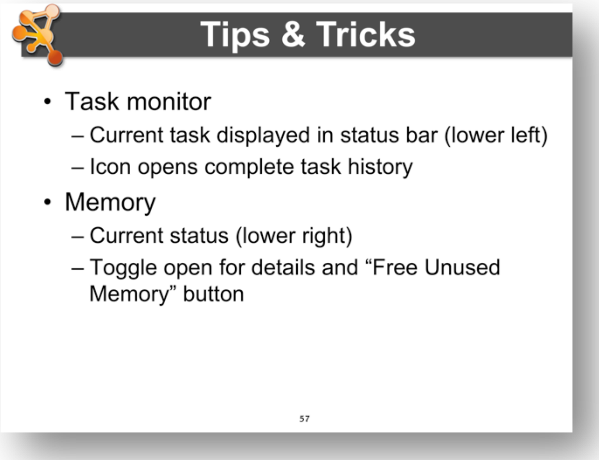
- Sessions
  - Sessions save pretty much everything:
    - Networks
    - Properties
    - Visual styles
    - Screen sizes
  - Saving a session on a large screen may require some resizing when opened on your laptop

56

Sessions

Sessions save pretty much everything: Networks, Properties. Visual styles, Screen sizes, and many other types of information. When working on a complex study of workflow, it is often prudent to save one's intermediate results as a session, so that the current state of an activity is persisted and can be resumed without having to repeat earlier low-level operations. Not all state is the same, however. For example, saving a session on a large screen may require some resizing when re-opened.

Task Monitor



Tips & Tricks

- Task monitor
  - Current task displayed in status bar (lower left)
  - Icon opens complete task history
- Memory
  - Current status (lower right)
  - Toggle open for details and "Free Unused Memory" button

57

The Cytoscape task monitor will show the messages for the current task and all of the completed tasks. You can show the task monitor by clicking its icon in the lower left-hand corner.

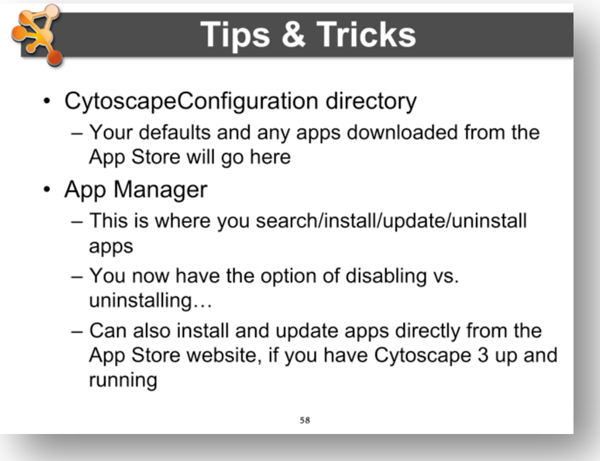
Cytoscape will also log more detailed messages to the framework-cytoscape.log file in CytoscapeConfiguration/3 directory.

Memory

Cytoscape uses a lot of memory and, as a Java system, doesn't like to let go of it. When working with large networks, an occasional save session and restart will help clear out memory. Another efficiency measure is to destroy large network views when not needed.

Cytoscape now provides a memory indicator in the lower right-hand corner of the window. This will tell you if you have enough memory (green) or you are running out of memory (yellow, red). Details are available by clicking the button and you will be given the option to "Free Unused Memory" to lower memory as much as possible.

Final points on Tips and Tricks



Tips & Tricks

- CytoscapeConfiguration directory
  - Your defaults and any apps downloaded from the App Store will go here
- App Manager
  - This is where you search/install/update/uninstall apps
  - You now have the option of disabling vs. uninstalling…
  - Can also install and update apps directly from the App Store website, if you have Cytoscape 3 up and running

58

- CytoscapeConfiguration directory
  - This directory is typically located under your home directory, for example on a Macintosh system it will be under

    /Users/<username>
  - Your defaults and any plugins downloaded from the app manager will go in this directory. Also, apps may use this directory to store configuration
  - Sometimes,, if things get really messed up, deleting (or renaming) this directory can give you a “clean slate”
- App manager
  - This is where you search/install/update/uninstall apps
  - You now have the option of disabling vs. uninstalling…
  - Can also install and update apps directly from the App Store website, if you have Cytoscape 3 up and running

Notes

Demo/Sample use cases

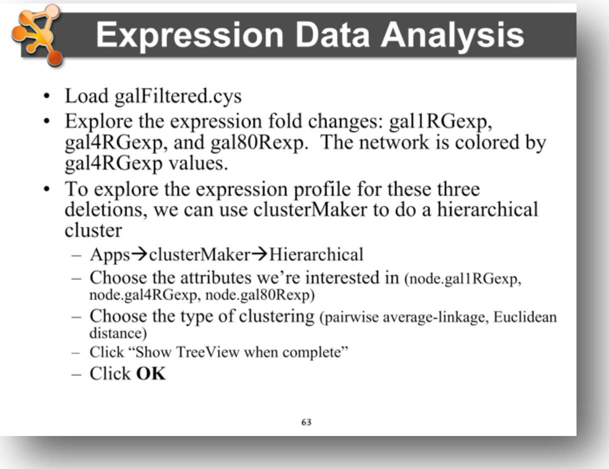
Use case 1: Expression data analysis

This use case highlights the visual display of expression data, integrated clustering features, and basic Gene Ontology overrepresentation analysis. *Note: we are starting with an expression dataset that has already been normalized, statistically analyzed, formatted, imported and associated with an interaction network.*

The dataset

Differential gene expression of GAL deletion mutants grown in the presence and absence of galactose[122]. Fold values were mapped onto a protein-protein interaction network focusing on galactose utilization in yeast, *Saccharomyces cerevisiae*. Additional annotations (e.g., gene and protein identifiers, GO terms, pathway associations, etc) have also been added as node attributes. This dataset is included with the download of Cytoscape in the sampleData folder and is called **galFiltered.cys**.

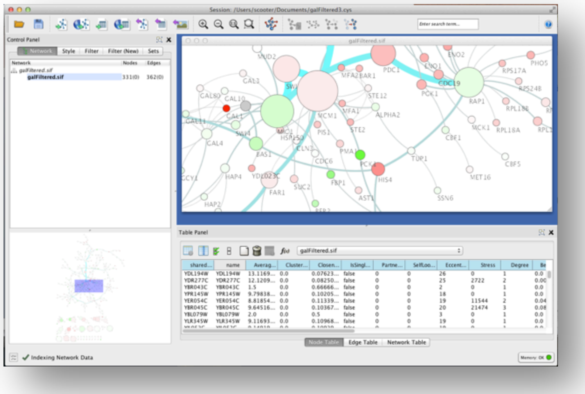
Locate and open the galFiltered.cys session file. Check out the VizMapper settings for node color. By default, the nodes are colored by the fold value **gal4RGexp.** Explore the visualization of other fold values in the dataset: **gal1RGexp** and **gal80Rexp**.



Expression Data Analysis

- Load galFiltered.cys
- Explore the expression fold changes: gal1RGexp, gal4RGexp, and gal80Rexp. The network is colored by gal4RGexp values.
- To explore the expression profile for these three deletions, we can use clusterMaker to do a hierarchical cluster
  - Apps→clusterMaker→Hierarchical
  - Choose the attributes we're interested in (node.gal1RGexp, node.gal4RGexp, node.gal80Rexp)
  - Choose the type of clustering (pairwise average-linkage, Euclidean distance)
  - Click "Show TreeView when complete"
  - Click **OK**

63



shared...	name	Averag...	Cluster...	Closen...	IsSingl...	Partne...	SelfLoo...	Eccent...	Stress	Degree	Be
YDL194W	YDL194W	13.1169...	0.0	0.07623...	false	0	0	26	0	1	0.0
YDR277C	YDR277C	12.1209...	0.0	0.08250...	false	0	0	25	2722	2	0.00
YBR043C	YBR043C	1.5	0.0	0.66666...	false	0	0	2	0	1	0.0
YPR145W	YPR145W	9.79838...	0.0	0.10205...	false	0	0	18	0	1	0.0
YER054C	YER054C	8.81854...	0.0	0.11339...	false	0	0	19	11544	2	0.04
YBR045C	YBR045C	9.64516...	0.0	0.10367...	false	0	0	20	21474	3	0.08
YBL079W	YBL079W	2.0	0.0	0.5	false	0	0	3	0	1	0.0
YLR345W	YLR345W	9.11693...	0.0	0.10968...	false	0	0	19	0	1	0.0

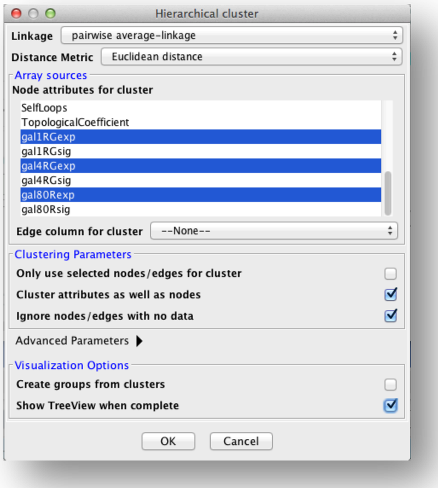
Notes

Cluster analysis

To explore the expression profiles for the three deletions, we can perform clustering within Cytoscape using the **clusterMaker** app.

In the Apps menu, select clusterMaker > Hierarchical.

- Choose the type of clustering:
  - pairwise average-linkage
- Choose the attributes of array data:
  - node.gal1RGexp
  - node.gal4RGexp
  - node.gal80Rexp
- Check **Show Treeview when complete**
- Click: **OK**



Hierarchical cluster

Linkage: pairwise average-linkage

Distance Metric: Euclidean distance

Array sources

Node attributes for cluster

- SelfLoops
- TopologicalCoefficient
- gal1RGexp (selected)
- gal1RGsig
- gal4RGexp (selected)
- gal4RGsig
- gal80Rexp (selected)
- gal80Rsig

Edge column for cluster: --None--

Clustering Parameters

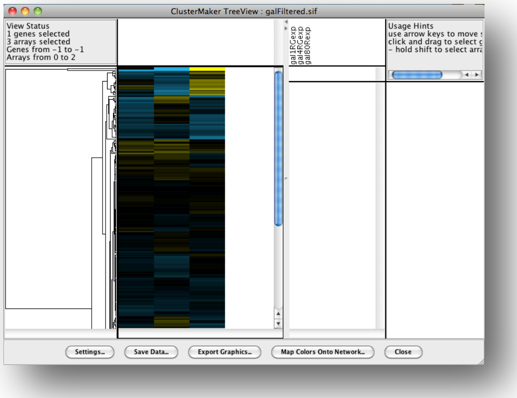
- ☐ Only use selected nodes/edges for cluster
- ☑ Cluster attributes as well as nodes
- ☑ Ignore nodes/edges with no data

Advanced Parameters ▶

Visualization Options

- ☐ Create groups from clusters
- ☑ Show TreeView when complete

This will bring up the TreeView of your cluster results. Each row is a gene and the three columns correspond to the three data attributes. A dendrogram to the left expresses the relationship between clusters, and the region to the right shows a close-up and labeled view of selected rows.



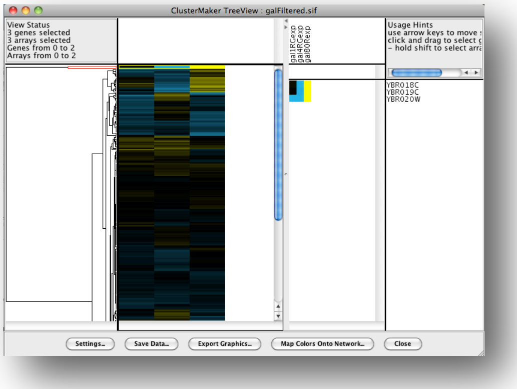
ClusterMaker TreeView : galFiltered.sif

View Status
1 genes selected
3 arrays selected
Genes from -1 to -1
Arrays from 0 to 2

Usage Hints
use arrow keys to move s
click and drag to select g
- hold shift to select arra

If the colors are too dark, or if you prefer other colors altogether, you can open Settings... and adjust a number of preferences.

Now, select the top most branch of the dendrogram, as shown on the right. *Notice that selections in TreeView correspond to selections in the network!*



ClusterMaker TreeView : galFiltered.sif

View Status
3 genes selected
3 arrays selected
Genes from 0 to 2
Arrays from 0 to 2

Usage Hints
use arrow keys to move s
click and drag to select g
- hold shift to select arra

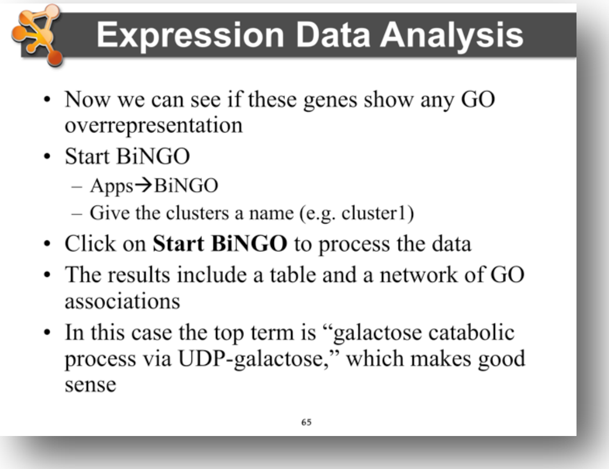
YBR018C
YBR019C
YBR020W

Notes

GO term overrepresentation analysis

Now we can see if any of the selected genes from that first cluster show any GO term overrepresentation. In other words, are there particular GO terms that are enriched (or overrepresented) in this subset of genes? We can do this using the **BiNGO** app.

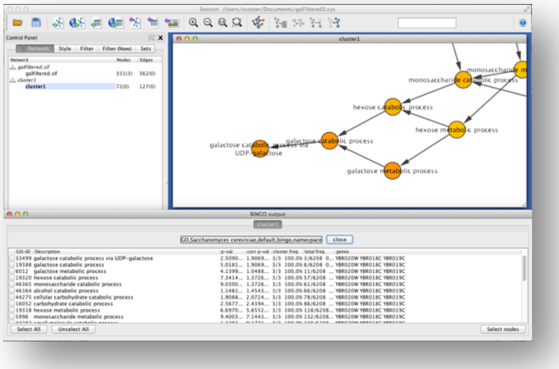
- Apps > BiNGO
- Give the cluster a name
- Click: Start BiNGO
- *Note: there are many parameters you can play with. The defaults are usually sufficient for a first pass as major trends.*



Expression Data Analysis

- Now we can see if these genes show any GO overrepresentation
- Start BiNGO
  - Apps→BiNGO
  - Give the clusters a name (e.g. cluster1)
- Click on **Start BiNGO** to process the data
- The results include a table and a network of GO associations
- In this case the top term is “galactose catabolic process via UDP-galactose,” which makes good sense

The results are displayed as a table and as a network of GO associations. The GO terms are connected based on their inherent hierarchical relationship and they are colored based on the significance of their overrepresentation in your cluster.


Session: /Users/scooter/Documents/galFiltered3.cys
Control Panel
Network
Style
Filter
Filter (New)
Sets
cluster1
monosaccharide m
monosaccharide catabolic process
hexose catabolic process
hexose metabolic process
galactose catabolic process
galactose catabolic process via UDP-galactose
galactose metabolic process
BiNGO output
cluster1
close
Select All
Unselect All
Select nodes

Network	Nodes	Edges
galFiltered.sif		
galFiltered.sif	331(3)	362(0)
cluster1		
cluster1	72(0)	127(0)

GO,Saccharomyces cerevisiae,default,bingo,namespace

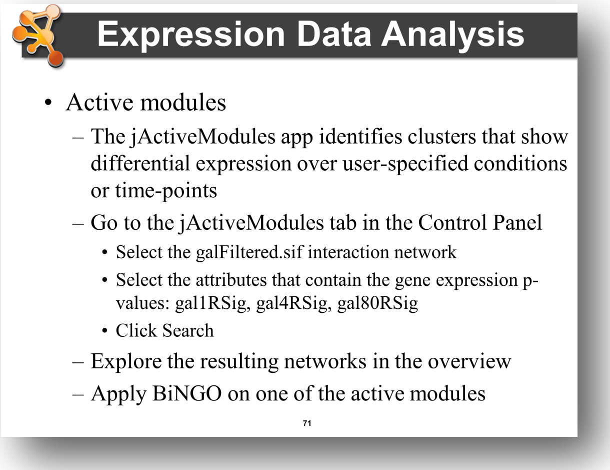
GO-ID	Description	p-val	corr p-val	cluster freq	total freq	genes
33499	galactose catabolic process via UDP-galactose	2.5090...	1.9069...	3/3 100.0%	3/6208 0...	YBR020W YBR018C YBR019C
19388	galactose catabolic process	5.0181...	1.9069...	3/3 100.0%	6/6208 0...	YBR020W YBR018C YBR019C
6012	galactose metabolic process	4.1399...	1.0488...	3/3 100.0%	11/6208 ...	YBR020W YBR018C YBR019C
19320	hexose catabolic process	7.3414...	1.3726...	3/3 100.0%	57/6208 ...	YBR020W YBR018C YBR019C
46365	monosaccharide catabolic process	9.0300...	1.3726...	3/3 100.0%	61/6208 ...	YBR020W YBR018C YBR019C
46164	alcohol catabolic process	1.1481...	1.4543...	3/3 100.0%	66/6208 ...	YBR020W YBR018C YBR019C
44275	cellular carbohydrate catabolic process	1.9088...	2.0724...	3/3 100.0%	78/6208 ...	YBR020W YBR018C YBR019C
16052	carbohydrate catabolic process	2.5677...	2.4394...	3/3 100.0%	86/6208 ...	YBR020W YBR018C YBR019C
19318	hexose metabolic process	6.6970...	5.6552...	3/3 100.0%	118/6208...	YBR020W YBR018C YBR019C
5996	monosaccharide metabolic process	9.4003...	7.1443...	3/3 100.0%	132/6208...	YBR020W YBR018C YBR019C

Notes

Active modules

Using the jActiveModules app, we can also identify clusters that show differential expression over user-specified conditions or time-points. Here, we will use the p-values for the differential expression of the GAL deletion mutants.

- Select the jActiveModules tab in the Control Panel
- Choose the galFiltered.sif interaction network
- Choose the attributes that contain the differential expression p-values:
    - gal1RSig,
    - gal4RSig
    - gal80RSig
- Click Search

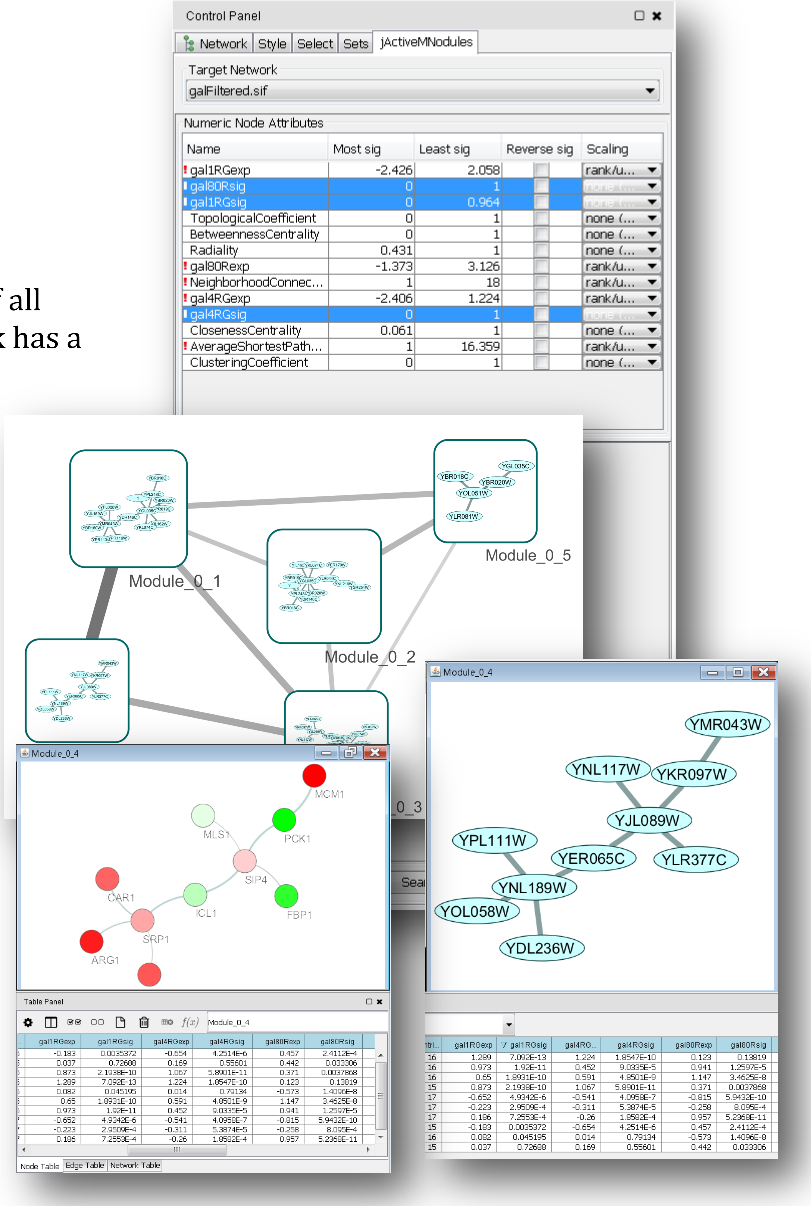


Expression Data Analysis

- Active modules
  - The jActiveModules app identifies clusters that show differential expression over user-specified conditions or time-points
  - Go to the jActiveModules tab in the Control Panel
    - Select the galFiltered.sif interaction network
    - Select the attributes that contain the gene expression p-values: gal1RSig, gal4RSig, gal80RSig
    - Click Search
  - Explore the resulting networks in the overview
  - Apply BiNGO on one of the active modules

71

This will invoke the creation of several new networks: one for each cluster identified by jActiveModules and one overview network of all modules. Each node in the overview network has a node score that shows how significant the identified module is and the edges are weighted by the number of nodes they share. Each module network has the same attributes as the original network and we can look at the significance values in the Node Table panel.



Target Network: galFiltered.sif

Numeric Node Attributes

Name	Most sig	Least sig	Reverse sig	Scaling
gal1RGexp	-2.426	2.058	☐	rank/u...
gal80Rsig	0	1	☐	none (...
gal1RGsig	0	0.964	☐	none (...
TopologicalCoefficient	0	1	☐	none (...
BetweennessCentrality	0	1	☐	none (...
Radiality	0.431	1	☐	none (...
gal80Rexp	-1.373	3.126	☐	rank/u...
NeighborhoodConnec...	1	18	☐	rank/u...
gal4RGexp	-2.406	1.224	☐	rank/u...
gal4RGsig	0	1	☐	none (...
ClosenessCentrality	0.061	1	☐	none (...
AverageShortestPath...	1	16.359	☐	rank/u...
ClusteringCoefficient	0	1	☐	none (...

Module_0_4

gal1RGexp	gal1RGsig	gal4RGexp	gal4RGsig	gal80Rexp	gal80Rsig
-0.183	0.0035372	-0.654	4.2514E-6	0.457	2.4112E-4
0.037	0.72688	0.169	0.55601	0.442	0.033306
0.873	2.1938E-10	1.067	5.8901E-11	0.371	0.0037868
1.289	7.092E-13	1.224	1.8547E-10	0.123	0.13819
0.082	0.045195	0.014	0.79134	-0.573	1.4096E-8
0.65	1.8931E-10	0.591	4.8501E-9	1.147	3.4625E-8
0.973	1.92E-11	0.452	9.0335E-5	0.941	1.2597E-5
-0.652	4.9342E-6	-0.541	4.0958E-7	-0.815	5.9432E-10
-0.223	2.9509E-4	-0.311	5.3874E-5	-0.258	8.095E-4
0.186	7.2553E-4	-0.26	1.8582E-4	0.957	5.2368E-11

ntri...	gal1RGexp	▽ gal1RGsig	gal4RG...	gal4RGsig	gal80Rexp	gal80Rsig
16	1.289	7.092E-13	1.224	1.8547E-10	0.123	0.13819
16	0.973	1.92E-11	0.452	9.0335E-5	0.941	1.2597E-5
16	0.65	1.8931E-10	0.591	4.8501E-9	1.147	3.4625E-8
15	0.873	2.1938E-10	1.067	5.8901E-11	0.371	0.0037868
17	-0.652	4.9342E-6	-0.541	4.0958E-7	-0.815	5.9432E-10
17	-0.223	2.9509E-4	-0.311	5.3874E-5	-0.258	8.095E-4
17	0.186	7.2553E-4	-0.26	1.8582E-4	0.957	5.2368E-11
15	-0.183	0.0035372	-0.654	4.2514E-6	0.457	2.4112E-4
16	0.082	0.045195	0.014	0.79134	-0.573	1.4096E-8
15	0.037	0.72688	0.169	0.55601	0.442	0.033306

Notes

____

____

____

____

____

____

____

____

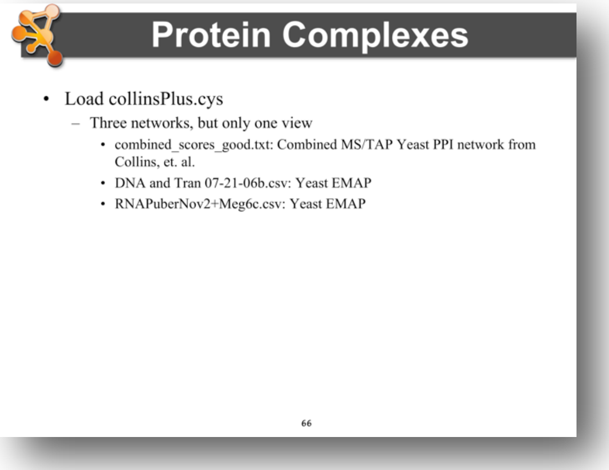
____

____

____

Use case 2: Protein complexes in protein-protein interaction networks

This use case highlights the combined use of MCL clustering of protein-protein interaction (PPI) networks and hierarchical clustering of epistatic mini-array profile (EMAP) data to explore potential biological protein complexes.



Protein Complexes

- Load collinsPlus.cys
  - Three networks, but only one view
    - combined_scores_good.txt: Combined MS/TAP Yeast PPI network from Collins, et. al.
    - DNA and Tran 07-21-06b.csv: Yeast EMAP
    - RNAPuberNov2+Meg6c.csv: Yeast EMAP

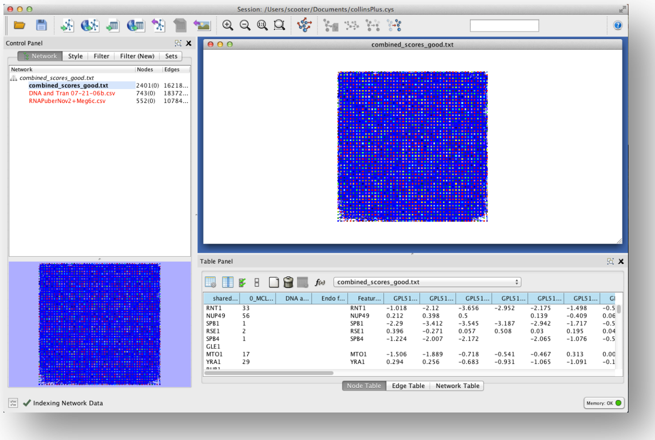
66

The dataset

We will be working with a Cytoscape session file containing three networks: one is a yeast PPI[27] and the other two are yeast EMAP datasets [7, 123]. *Note: we will not bother viewing the EMAP datasets as networks, but rather treat them as sets of nodes and attributes. You can perform clustering on sets of nodes without creating a network view!* The key to making this analysis work is having the same node identifiers in both the PPI and EMAPs.

The dataset is provided with this tutorial and is called **collinsPlus.cys**:

- combined_scores_good.txt (PPI)
- DNA and Tran 07-21-06b.csv (EMAP)
- RNAPuberNov2+Meg6c.csv (EMAP)



Session: /Users/scooter/Documents/collinsPlus.cys

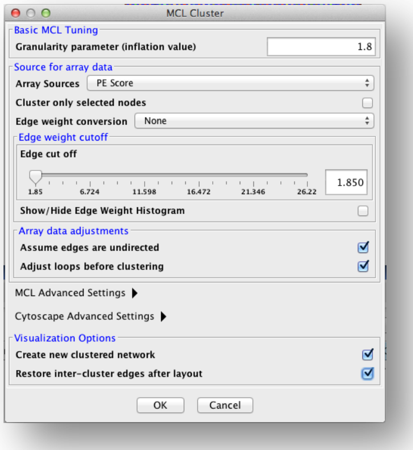
Network	Nodes	Edges
combined_scores_good.txt		
combined_scores_good.txt	2401(0)	16218...
DNA and Tran 07-21-06b.csv	743(0)	18372...
RNAPuberNov2+Meg6c.csv	552(0)	10784...

shared...	0_MCL...	DNA a...	Endo f...	Featur...	GPL51...	GPL51...	GPL51...	GPL51...	GPL51...	GPL51...	Gl
RNT1	33			RNT1	-1.018	-2.12	-3.656	-2.952	-2.175	-1.498	-0.5
NUP49	56			NUP49	0.212	0.398	0.5		0.139	-0.409	0.06
SPB1	1			SPB1	-2.29	-3.412	-3.545	-3.187	-2.942	-1.717	-0.5
RSE1	2			RSE1	0.396	-0.271	0.057	0.508	0.03	0.195	0.04
SPB4	1			SPB4	-1.224	-2.007	-2.172		-2.065	-1.076	-0.5
GLE1											
MTO1	17			MTO1	-1.506	-1.889	-0.718	-0.541	-0.467	0.313	0.00
YRA1	29			YRA1	0.294	0.256	-0.683	-0.931	-1.065	-1.091	-0.1

Notes

MCL clusters in the PPI network

First, we will identify the MCL clusters in the protein-protein interaction network. Under the Plugins menu, choose Cluster and then MCL cluster:



MCL Cluster

Basic MCL Tuning

Granularity parameter (inflation value): 1.8

Source for array data

Array Sources: PE Score

Cluster only selected nodes ☐

Edge weight conversion: None

Edge weight cutoff

Edge cut off: 1.850

Show/Hide Edge Weight Histogram ☐

Array data adjustments

Assume edges are undirected ☑

Adjust loops before clustering ☑

Visualization Options

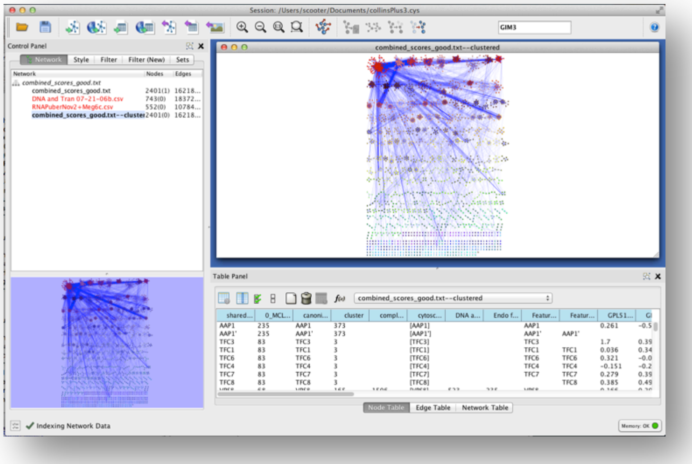
Create new clustered network ☑

Restore inter-cluster edges after layout ☑

- Density Parameter: 1.8
- Array source: PE Score
- Check **Create new clustered network** and **Restore inter-cluster edges after layout**
- Click: OK

There are your MCL clusters. Beautiful, aren't they! These are our first approximation of potential protein complexes based solely on tightly interacting protein clusters.

Next, we consider the clusters generated from EMAP data as an orthogonal form of evidence based on genetic interactions. Combining both cluster results provides a more complete picture.



Session: /Users/scooter/Documents/collinsPlus3.cys

Network	Nodes	Edges
combined_scores_good.txt		
combined_scores_good.txt	2401(1)	16218...
DNA and Tran 07-21-06b.csv	743(0)	18372...
RNAPuberNov2+Meg6c.csv	552(0)	10784...
combined_scores_good.txt--cluster	2401(0)	16218...

combined_scores_good.txt--clustered

Table Panel

combined_scores_good.txt--clustered

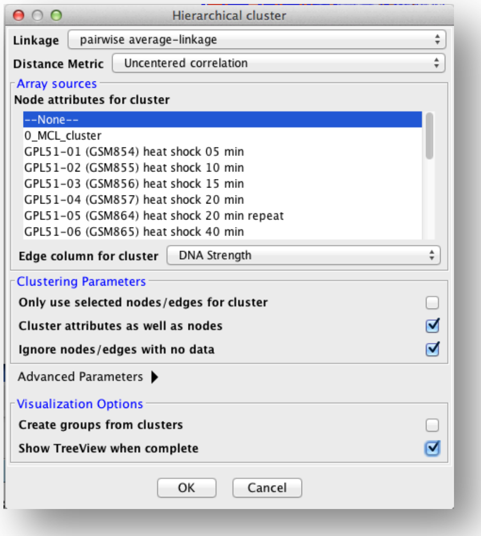
shared...	0_MCL...	canoni...	cluster	compl...	cytosc...	DNA a...	Endo f...	Featur...	Featur...	GPLS1...	GI
AAP1	235	AAP1	373		[AAP1]			AAP1		0.261	-0.5
AAP1'	235	AAP1'	373		[AAP1']			AAP1'	AAP1'		
TFC3	83	TFC3	3		[TFC3]			TFC3		1.7	0.39
TFC1	83	TFC1	3		[TFC1]			TFC1	TFC1	0.036	0.34
TFC6	83	TFC6	3		[TFC6]			TFC6	TFC6	0.321	-0.0
TFC4	83	TFC4	3		[TFC4]			TFC4	TFC4	-0.151	-0.2
TFC7	83	TFC7	3		[TFC7]			TFC7	TFC7	0.279	0.39
TFC8	83	TFC8	3		[TFC8]				TFC8	0.385	0.49

Notes

Hierarchical clustering of EMAP data

Select the “DNA and Tran…” dataset in the Network panel on the left. *Note: the red highlight simply indicates that no network view has been created. No problem.* Once again, go to Apps > clusterMaker > Hierarchical Cluster:

- Linkage: pairwise average-linkage
- Distance Metric: Uncentered correlation
- Edge column for cluster: DNA Strenth
- Check: **Show TreeView when complete**
- Click **OK**



Hierarchical cluster

Linkage: pairwise average-linkage

Distance Metric: Uncentered correlation

Array sources

Node attributes for cluster:

--None--
0_MCL_cluster
GPL51-01 (GSM854) heat shock 05 min
GPL51-02 (GSM855) heat shock 10 min
GPL51-03 (GSM856) heat shock 15 min
GPL51-04 (GSM857) heat shock 20 min
GPL51-05 (GSM864) heat shock 20 min repeat
GPL51-06 (GSM865) heat shock 40 min

Edge column for cluster: DNA Strength

Clustering Parameters

Only use selected nodes/edges for cluster □

Cluster attributes as well as nodes ☑

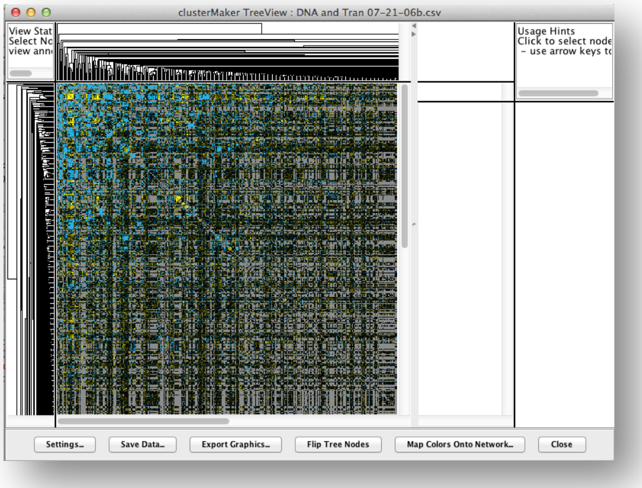
Ignore nodes/edges with no data ☑

Advanced Parameters ▶

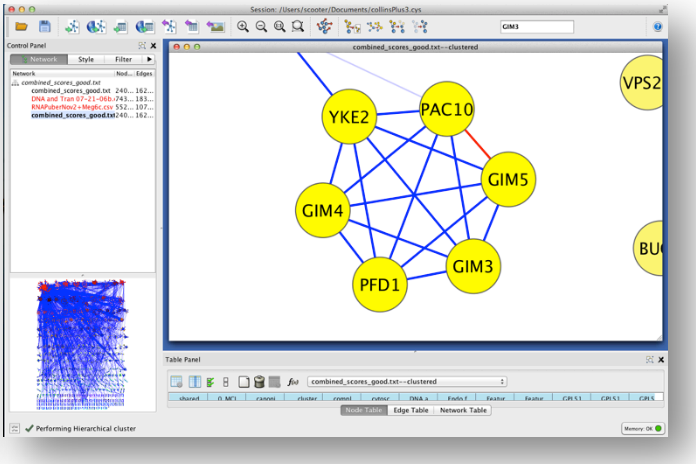
Visualization Options

Create groups from clusters □

Show TreeView when complete ☑



clusterMaker TreeView : DNA and Tran 07-21-06b.csv



Session: /Users/scooter/Documents/collinsPlus3.cys

Network	Nod...	Edges
combined_scores_good.txt		
combined_scores_good.txt	240...	162...
DNA and Tran 07-21-06b.	743...	183...
RNAPuberNov2+Meg6c.csv	552...	107...
combined_scores_good.txt	240...	162...

✓ Performing Hierarchical cluster

The EMAP clusters identify potential complexes based on genetic (functional) interactions. Now, we can explore the correspondence of evidence from these two methods. For example, search for GIM5 and select the entire cluster. Notice how the corresponding interactions are dynamically highlighted in the TreeView. Notice how both EMAP and PPI data do not provide strong support for the inclusion of BUD27 in this potential complex.

Notes

Hands-on tutorial: Introduction to Cytoscape

This tutorial will cover:

1. Navigating Cytoscape
2. Visualizing Data on Networks
3. Network and Pathway Resources
4. Plugin Manager
5. Plugin Demos

By the end of the tutorial, you should be able to use Cytoscape to import networks and attributes and visualize those attributes on the network.

Notes

Revised 09/17/2014

Hands-on tutorial: Working with data

This tutorial will introduce you to:

1. Searching Internet interaction databases with query terms.
2. Mapping Identifiers of different types to networks.
3. Finding your query terms in the downloaded network.

The second half of the tutorial will introduce you to some advanced basics in Cytoscape:

1. Apply filters to filter out low-confidence edges.
2. Perform basic edits using the Cytoscape graph editor.

Notes

Hands-on tutorial: Analysis of microarray data

This tutorial will introduce you to:

1. Combining data from two different sources: experimental data in the form of microarrray expression data and network data in the form of interaction data.
2. Visualizing networks using expression data.
3. Filtering networks based on expression data.

NOTE: The expression data used in this example has been pre-processed to work with the interaction network used.

Notes

Bibliography

1. Ideker T, Ozier O, Schwikowski B, Siegel AF: **Discovering regulatory and signalling circuits in molecular interaction networks**. *Bioinformatics* 2002, **18 Suppl 1**:S233-240.
2. Breitling R, Amtmann A, Herzyk P: **Graph-based iterative Group Analysis enhances microarray interpretation**. *BMC Bioinformatics* 2004, **5**:100.
3. Bandyopadhyay S, Kelley R, Ideker T: **Discovering regulated networks during HIV-1 latency and reactivation**. *Pac Symp Biocomput* 2006:354-366.
4. Qiu YQ, Zhang S, Zhang XS, Chen L: **Detecting disease associated modules and prioritizing active genes based on high throughput data**. *BMC Bioinformatics* 2010, **11**:26.
5. Bader GD, Hogue CW: **An automated method for finding molecular complexes in large protein interaction networks**. *BMC Bioinformatics* 2003, **4**:2.
6. Bandyopadhyay S, Kelley R, Krogan NJ, Ideker T: **Functional maps of protein complexes from quantitative genetic interaction data**. *PLoS computational biology* 2008, **4**(4):e1000065.
7. Collins SR, Miller KM, Maas NL, Roguev A, Fillingham J, Chu CS, Schuldiner M, Gebbia M, Recht J, Shales M *et al*: **Functional dissection of protein complexes involved in yeast chromosome biology using a genetic interaction map**. *Nature* 2007, **446**(7137):806-810.
8. Krogan NJ, Cagney G, Yu H, Zhong G, Guo X, Ignatchenko A, Li J, Pu S, Datta N, Tikuisis AP *et al*: **Global landscape of protein complexes in the yeast Saccharomyces cerevisiae**. *Nature* 2006, **440**(7084):637-643.
9. Vlasblom J, Wu S, Pu S, Superina M, Liu G, Orsi C, Wodak SJ: **GenePro: a Cytoscape plug-in for advanced visualization and analysis of interaction networks**. *Bioinformatics* 2006, **22**(17):2178-2179.
10. Florez AF, Park D, Bhak J, Kim BC, Kuchinsky A, Morris JH, Espinosa J, Muskus C: **Protein network prediction and topological analysis in Leishmania major as a tool for drug target selection**. *BMC Bioinformatics* 2010, **11**:484.
11. Bansal M, Della Gatta G, di Bernardo D: **Inference of gene regulatory networks and compound mode of action from time course gene expression profiles**. *Bioinformatics* 2006, **22**(7):815-822.
12. Emig D, Salomonis N, Baumbach J, Lengauer T, Conklin BR, Albrecht M: **AltAnalyze and DomainGraph: analyzing and visualizing exon expression data**. *Nucleic acids research* 2010, **38**(Web Server issue):W755-762.
13. Liu S, Zhang C, Zhou Y: **Domain graph of Arabidopsis proteome by comparative analysis**. *J Proteome Res* 2005, **4**(2):435-444.
14. Kann MG, Jothi R, Cherukuri PF, Przytycka TM: **Predicting protein domain interactions from coevolution of conserved regions**. *Proteins* 2007, **67**(4):811-820.
15. Vailaya A, Bluvas P, Kincaid R, Kuchinsky A, Creech M, Adler A: **An architecture for biological information extraction and representation**. *Bioinformatics* 2005, **21**(4):430-438.
16. Kohler S, Bauer S, Horn D, Robinson PN: **Walking the interactome for prioritization of candidate disease genes**. *Am J Hum Genet* 2008, **82**(4):949-958.
17. Kann MG: **Protein interactions and disease: computational approaches to uncover the etiology of diseases**. *Brief Bioinform* 2007, **8**(5):333-346.

18. Mohammad Shafkat Amin AB, Russel L. Finley, Jr., Hasan Jamin: **A stochastic approach to candidate disease gene subnetwork extraction**. In: *2010 ACM Symposium on Applied Computing*. 2010.

19. Dobrin R, Zhu J, Molony C, Argman C, Parrish ML, Carlson S, Allan MF, Pomp D, Schadt EE: **Multi-tissue coexpression networks reveal unexpected subnetworks associated with disease**. *Genome biology* 2009, **10**(5):R55.

20. King JY, Ferrara R, Tabibiazar R, Spin JM, Chen MM, Kuchinsky A, Vailaya A, Kincaid R, Tsalenko A, Deng DX *et al*: **Pathway analysis of coronary atherosclerosis**. *Physiol Genomics* 2005, **23**(1):103-118.

21. Chuang H, Ressenti, L., Ideker, T., Kipps, T.: **Interactome-based modeling and diagnosis of Chrinic Lymphocytic Leukemia**. In: *50th Annual Meeting of the American Society of Hematology*. 2008.

22. Chuang HY, Lee E, Liu YT, Lee D, Ideker T: **Network-based classification of breast cancer metastasis**. *Mol Syst Biol* 2007, **3**:140.

23. Mileyko Y, Joh RI, Weitz JS: **Small-scale copy number variation and large-scale changes in gene expression**. *Proceedings of the National Academy of Sciences of the United States of America* 2008, **105**(43):16659-16664.

24. Chen L, Xuan J, Wang Y, Hoffman EP, Riggins RB, Clarke R: **Identification of condition-specific regulatory modules through multi-level motif and mRNA expression analysis.** *Int J Comput Biol Drug Des* 2009, **2**(1):1-20.

25. McLendon R, et. al.: **Comprehensive genomic characterization defines human glioblastoma genes and core pathways**. *Nature* 2008, **455**(7216):1061-1068.

26. Moraru, II, Schaff JC, Slepchenko BM, Blinov ML, Morgan F, Lakshminarayana A, Gao F, Li Y, Loew LM: **Virtual Cell modelling and simulation software environment**. *IET Syst Biol* 2008, **2**(5):352-362.

27. Collins SR, Kemmeren P, Zhao XC, Greenblatt JF, Spencer F, Holstege FC, Weissman JS, Krogan NJ: **Toward a comprehensive atlas of the physical interactome of Saccharomyces cerevisiae**. *Mol Cell Proteomics* 2007, **6**(3):439-450.

28. Keiser MJ, Roth BL, Armbruster BN, Ernsberger P, Irwin JJ, Shoichet BK: **Relating protein pharmacology by ligand chemistry**. *Nat Biotechnol* 2007, **25**(2):197-206.

29. Atkinson HJ, Morris JH, Ferrin TE, Babbitt PC: **Using sequence similarity networks for visualization of relationships across diverse protein superfamilies**. *PLoS One* 2009, **4**(2):e4345.

30. Yildirim MA, Goh KI, Cusick ME, Barabasi AL, Vidal M: **Drug-target network**. *Nat Biotechnol* 2007, **25**(10):1119-1126.

31. Enright AJ, Van Dongen S, Ouzounis CA: **An efficient algorithm for large-scale detection of protein families**. *Nucleic Acids Res* 2002, **30**(7):1575-1584.

32. Scheeff ED, Bourne PE: **Structural evolution of the protein kinase-like superfamily**. *PLoS computational biology* 2005, **1**(5):e49.

33. Jaccard P: **Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines.** *Bulletin del la Société Vaudoise des Sciences Naturelles* 1901, **37**:241-272.

34. Tanimoto TT. In: *IBM Internal Report.* 1957.

35. Tversky A: **Features of similarity**. *Psychological Review* 1977, **84**(4):327-352.

36. **Daylight Theory: Fingerprints** [http://www.daylight.com/dayhtml/doc/theory/theory.finger.html]

37. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool**. *Journal of molecular biology* 1990, **215**(3):403-410.
38. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ: **Gapped BLAST and PSI-BLAST: a new generation of protein database search programs**. *Nucleic acids research* 1997, **25**(17):3389-3402.
39. Smith TF, Waterman MS: **Identification of common molecular subsequences.** *Journal of molecular biology* 1981, **147**(1):195-197.
40. Shindyalov IN, Bourne PE: **Protein structure alignment by incremental combinatorial extension (CE) of the optimal path**. *Protein Eng* 1998, **11**(9):739-747.
41. Holm L, Sander C: **Mapping the protein universe**. *Science* 1996, **273**(5275):595-603.
42. Taylor WR, Flores TP, Orengo CA: **Multiple protein structure alignment**. *Protein Sci* 1994, **3**(10):1858-1870.
43. Ilyin VA, Abyzov A, Leslin CM: **Structural alignment of proteins by a novel TOPOFIT method, as a superimposition of common volumes at a topomax point**. *Protein Sci* 2004, **13**(7):1865-1874.
44. Kolbeck B, May P, Schmidt-Goenner T, Steinke T, Knapp EW: **Connectivity independent protein-structure alignment: a hierarchical approach**. *BMC Bioinformatics* 2006, **7**:510.
45. Guerler A, Knapp EW: **Novel protein folds and their nonsequential structural analogs**. *Protein Sci* 2008, **17**(8):1374-1382.
46. Bausch P, Bumgardner, J.: **Make a Flickr-Style Tag Cloud**. In: *Flickr Hacks.* O'Reilly Press.; 2006.
47. **ISO/IEC JTC1/SC34/WG3** [http://www.isotopicmaps.org/]
48. Pegg SC, Brown SD, Ojha S, Seffernick J, Meng EC, Morris JH, Chang PJ, Huang CC, Ferrin TE, Babbitt PC: **Leveraging enzyme structure-function relationships for functional inference and experimental design: the structure-function linkage database**. *Biochemistry* 2006, **45**(8):2545-2555.
49. **Graph theory** [http://en.wikpedia.org/wiki/Graph_theory]
50. Bondy JA, Murty, U.S.R.: **Graph Theory with Applications**. In.
51. Barabasi AL, Oltvai ZN: **Network biology: understanding the cell's functional organization**. *Nat Rev Genet* 2004, **5**(2):101-113.
52. Jeong H, Mason SP, Barabasi AL, Oltvai ZN: **Lethality and centrality in protein networks**. *Nature* 2001, **411**(6833):41-42.
53. Jeong H, Tombor B, Albert R, Oltvai ZN, Barabasi AL: **The large-scale organization of metabolic networks**. *Nature* 2000, **407**(6804):651-654.
54. Fell DA, Wagner A: **The small world of metabolism**. *Nat Biotechnol* 2000, **18**(11):1121-1122.
55. Ma HW, Zeng AP: **The connectivity structure, giant strong component and centrality of metabolic networks**. *Bioinformatics* 2003, **19**(11):1423-1430.
56. Maslov S, Sneppen K: **Specificity and stability in topology of protein networks**. *Science* 2002, **296**(5569):910-913.
57. Rzhetsky A, Gomez SM: **Birth of scale-free molecular networks and the number of distinct DNA and protein domains per genome**. *Bioinformatics* 2001, **17**(10):988-996.
58. Wuchty S: **Scale-free behavior in protein domain networks**. *Mol Biol Evol* 2001, **18**(9):1694-1702.

59. Tong AH, Lesage G, Bader GD, Ding H, Xu H, Xin X, Young J, Berriz GF, Brost RL, Chang M *et al*: **Global mapping of the yeast genetic interaction network**. *Science* 2004, **303**(5659):808-813.
60. van Noort V, Snel B, Huynen MA: **The yeast coexpression network has a small-world, scale-free architecture and can be explained by a simple model**. *EMBO Rep* 2004, **5**(3):280-284.
61. Featherstone DE, Broadie K: **Wrestling with pleiotropy: genomic and topological analysis of the yeast gene expression network**. *Bioessays* 2002, **24**(3):267-274.
62. Agrawal H: **Extreme self-organization in networks constructed from gene expression data**. *Phys Rev Lett* 2002, **89**(26):268702.
63. Khanin R, Wit, E.: **How Scale-Free Are Biological Networks**. *Journal of Computational Biology* 2006, **13**(3):810-818.
64. Albert R, Jeong H, Barabasi AL: **Error and attack tolerance of complex networks**. *Nature* 2000, **406**(6794):378-382.
65. Stumpf MP, Wiuf C, May RM: **Subnets of scale-free networks are not scale-free: sampling properties of networks**. *Proceedings of the National Academy of Sciences of the United States of America* 2005, **102**(12):4221-4224.
66. Stumpf MP, Ingram, P.J.: **Probablity models for degree distribution of protein interaction networks.** *Europhys Lett* 2005, **71**(1):152-158.
67. Stumpf MP, INgram, P.J., Nouvel, I., Wiuf, C.: **Statistical model selection methods applied to biological network data.** *Trans Comp Syst Biol* 2005, **3**:65-77.
68. Barabasi AL, Albert R: **Emergence of scaling in random networks**. *Science* 1999, **286**(5439):509-512.
69. Barabasi AL: **Scale-free networks: a decade and beyond**. *Science* 2009, **325**(5939):412-413.
70. Gilbert EN: **Random Graphs**. *Ann Math Stat* 1959, **30**(4):1141-1144.
71. Erdős P, Rényi, A.: **On Random Graphs, I.** *Publicationes Mathematicae* 1959, **6**:290-297.
72. Erdős P, Rényi, A.: **The Evolution of Random Graphs**. *Magyar Tud Akad Mat Kutató Int Közl* 1960, **5**:17-61.
73. **Erdős–Rényi model** [http://en.wikipedia.org/wiki/Erd%C5%91s%E2%80%93R%C3%A9nyi_model]
74. Watts DJ, Strogatz SH: **Collective dynamics of 'small-world' networks**. *Nature* 1998, **393**(6684):440-442.
75. Ward JH: **Hierarchical Grouping to Optimize an Objective Function**. *Journal of the Americal Statistical Association* 1963, **58**(301):236-244.
76. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns**. *Proc Natl Acad Sci U S A* 1998, **95**(25):14863-14868.
77. MacQueen JB: **Some Methods for classification and Analysis of Multivariate Observations**. In: *5th Berkeley Symposium on Mathematical Statistics and Probability: 1967*. University of California Press: 281-297.
78. Steinhaus H: **Sur la division des corps matériels et parties**. *Bull Acad Polon Sci* 1956, **4**(12):801-804.
79. Newman ME, Girvan M: **Finding and evaluating community structure in networks**. *Phys Rev E Stat Nonlin Soft Matter Phys* 2004, **69**(2 Pt 2):026113.
80. van Dongen S: **A cluster algorithm for graphs.** In. Amsterdam: National Research Institue in the Netherlands; 2000.

81. van Dongen S: **Graph Clustering by Flow SImulation**. University of Utrecht; 2000.
82. Meila M, Shi, J.: **A random walks view of spectral segmentation**. In: *International Workshop on AI and Statistics*. 2001.
83. Fiedler M: **Algebraic connectivity of graphs**. *Czech Math J* 1973, **23**:298-305.
84. Fiedler M: **A property of eigenvectors of non-negative symmetric matrices and its application to graph theory.** *Czech Math J* 1975, **25**:619-633.
85. Donath WE, Hoffman, A.J.: **Lower bounds for paritioning of graph.s**. *IBM J Res Develop* 1973, **17**:420-425.
86. Paccanaro A, Casbon JA, Saqi MA: **Spectral clustering of protein sequences.** *Nucleic acids research* 2006, **34**(5):1571-1580.
87. Frey BJ, Dueck D: **Clustering by passing messages between data points**. *Science* 2007, **315**(5814):972-976.
88. Givoni IE, Frey BJ: **A binary variable model for affinity propagation**. *Neural Comput* 2009, **21**(6):1589-1600.
89. Nepusz T, Sasidharan R, Paccanaro A: **SCPS: a fast implementation of a spectral method for detecting protein families on a genome-wide scale**. *BMC Bioinformatics* 2010, **11**:120.
90. Wittkop T, Baumbach J, Lobo FP, Rahmann S: **Large scale clustering of protein sequences with FORCE -A layout based heuristic for weighted cluster editing**. *BMC Bioinformatics* 2007, **8**:396.
91. Wittkop T, Emig D, Lange S, Rahmann S, Albrecht M, Morris JH, Bocker S, Stoye J, Baumbach J: **Partitioning biological data with transitivity clustering.** *Nat Methods* 2010, **7**(6):419-420.
92. Newman AM, Cooper JB: **AutoSOME: a clustering method for identifying gene expression modules without prior knowledge of cluster number**. *BMC Bioinformatics* 2010, **11**:117.
93. Apeltsin L, Morris JH, Babbitt PC, Ferrin TE: **Improving the quality of protein similarity network clustering algorithms using the network edge weight distribution**. *Bioinformatics* 2011, **27**(3):326-333.
94. Shi J, Malik, J.: **Normalized Cuts and IMage Segmentation**. In: *International Conference on Computer Vision and Pattern Recognition: June 1997; San Juan, Puerto Rico*. 1997.
95. Shi J, Malik, J.: **Normalized Cuts and Image Segmentation**. In.: UC Berkeley; 1997.
96. Vlasblom J, Wodak SJ: **Markov clustering versus affinity propagation for the partitioning of protein interaction graphs**. *BMC Bioinformatics* 2009, **10**:99.
97. **Network Motif** [http://en.wikipedia.org/wiki/Network_motif]
98. Alon U: **Network motifs: theory and experimental approaches.** *Nat Rev Genet* 2007, **8**(6):450-461.
99. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U: **Network motifs: simple building blocks of complex networks**. *Science* 2002, **298**(5594):824-827.
100. Shen-Orr SS, Milo R, Mangan S, Alon U: **Network motifs in the transcriptional regulation network of Escherichia coli**. *Nature genetics* 2002, **31**(1):64-68.
101. Milo R, Itzkovitz S, Kashtan N, Levitt R, Shen-Orr S, Ayzenshtat I, Sheffer M, Alon U: **Superfamilies of evolved and designed networks**. *Science* 2004, **303**(5663):1538-1542.
102. Mangan S, Alon U: **Structure and function of the feed-forward loop network motif**. *Proceedings of the National Academy of Sciences of the United States of America* 2003, **100**(21):11980-11985.

103. Mangan S, Zaslaver A, Alon U: **The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks**. *Journal of molecular biology* 2003, **334**(2):197-204.

104. Kalir S, Mangan S, Alon U: **A coherent feed-forward loop with a SUM input function prolongs flagella expression in Escherichia coli**. *Mol Syst Biol* 2005, **1**:2005 0006.

105. Mangan S, Itzkovitz S, Zaslaver A, Alon U: **The incoherent feed-forward loop accelerates the response-time of the gal system of Escherichia coli**. *Journal of molecular biology* 2006, **356**(5):1073-1081.

106. Entus R, Aufderheide B, Sauro HM: **Design and implementation of three incoherent feed-forward motif based biological concentration sensors**. *Syst Synth Biol* 2007, **1**(3):119-128.

107. Glossop NR, Lyons LC, Hardin PE: **Interlocked feedback loops within the Drosophila circadian oscillator**. *Science* 1999, **286**(5440):766-768.

108. Hau LD, Kwon YK: **The effects of feedback loops on disease comorbidity in human signaling networks**. *Bioinformatics* 2011.

109. Ferrazzi F, Engel FB, Wu E, Moseman AP, Kohane IS, Bellazzi R, Ramoni MF: **Inferring cell cycle feedback regulation from gene expression data**. *J Biomed Inform* 2011.

110. Sevim V, Gong X, Socolar JE: **Reliability of transcriptional cycles and the yeast cell-cycle oscillator**. *PLoS computational biology* 2010, **6**(7):e1000842.

111. Eichenberger P, Fujita M, Jensen ST, Conlon EM, Rudner DZ, Wang ST, Ferguson C, Haga K, Sato T, Liu JS *et al*: **The program of gene transcription for a single differentiating cell type during sporulation in Bacillus subtilis**. *PLoS biology* 2004, **2**(10):e328.

112. Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Hannett NM, Harbison CT, Thompson CM, Simon I *et al*: **Transcriptional regulatory networks in Saccharomyces cerevisiae**. *Science* 2002, **298**(5594):799-804.

113. Boyer LA, Lee TI, Cole MF, Johnstone SE, Levine SS, Zucker JP, Guenther MG, Kumar RM, Murray HL, Jenner RG *et al*: **Core transcriptional regulatory circuitry in human embryonic stem cells**. *Cell* 2005, **122**(6):947-956.

114. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT *et al*: **Gene ontology: tool for the unification of biology. The Gene Ontology Consortium**. *Nature genetics* 2000, **25**(1):25-29.

115. Draghici S, Khatri P, Martins RP, Ostermeier GC, Krawetz SA: **Global functional profiling of gene expression**. *Genomics* 2003, **81**(2):98-104.

116. Cho RJ, Huang M, Campbell MJ, Dong H, Steinmetz L, Sapinoso L, Hampton G, Elledge SJ, Davis RW, Lockhart DJ: **Transcriptional regulation and function during the human cell cycle**. *Nature genetics* 2001, **27**(1):48-54.

117. Khatri P, Draghici S, Ostermeier GC, Krawetz SA: **Profiling gene expression using onto-express**. *Genomics* 2002, **79**(2):266-270.

118. Zhang S, Cao J, Kong YM, Scheuermann RH: **GO-Bayes: Gene Ontology-based overrepresentation analysis using a Bayesian approach**. *Bioinformatics* 2010, **26**(7):905-911.

119. Dunn OJ: **Multiple Comparisons Among Means**. *Journal of the Americal Statistical Association* 1961, **56**:52-64.

120. Benjamini Y, Hochberg, Y.: **Controlling the false discovery rate: a practical and powerful approach to multiple testing.** *Journal of the Royal Statistical Society* 1995, **57**(1):289-300.

121. Maere S, Heymans K, Kuiper M: **BiNGO: a Cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks**. *Bioinformatics* 2005, **21**(16):3448-3449.

122. Ideker T, Thorsson V, Ranish JA, Christmas R, Buhler J, Eng JK, Bumgarner R, Goodlett DR, Aebersold R, Hood L: **Integrated genomic and proteomic analyses of a systematically perturbed metabolic network**. *Science* 2001, **292**(5518):929-934.

123. Schuldiner M, Collins SR, Thompson NJ, Denic V, Bhamidipati A, Punna T, Ihmels J, Andrews B, Boone C, Greenblatt JF *et al*: **Exploration of the function and organization of the yeast early secretory pathway through an epistatic miniarray profile**. *Cell* 2005, **123**(3):507-519.