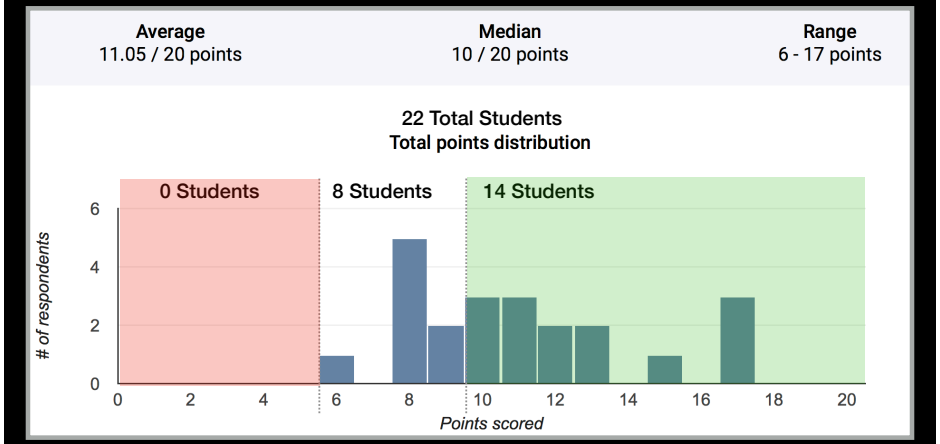
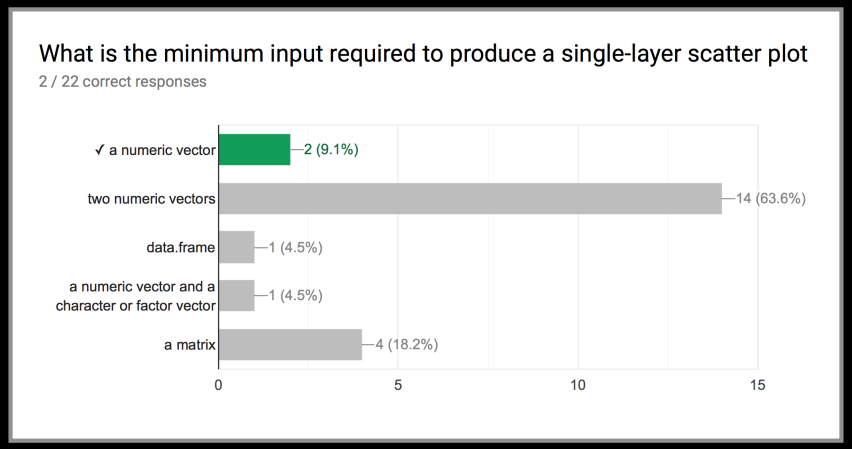
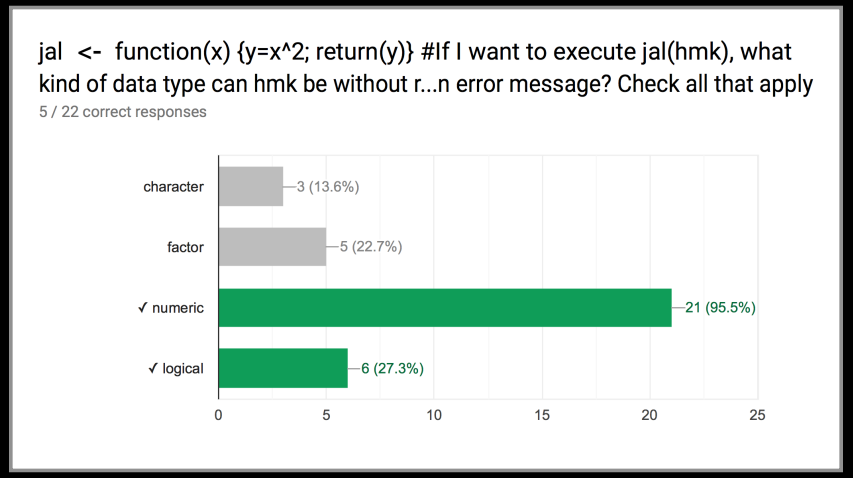
R 'Knowledge Check' Quiz Results



Frequently Missed Questions



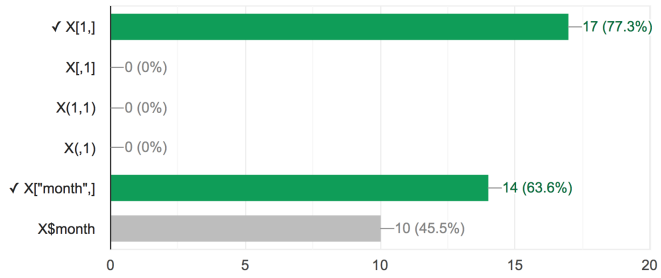
Frequently Missed Questions



Frequently Missed Questions

Select the correct way(s) to extract a vector from a row in the data frame X. The name of the first row is month.

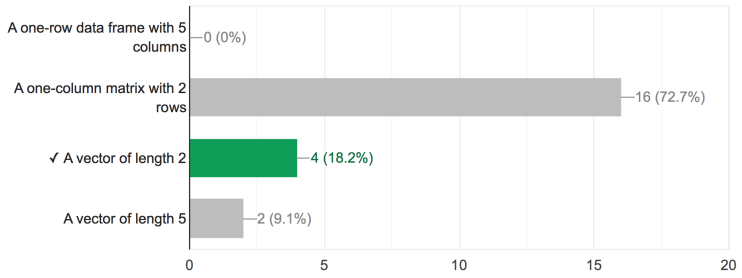
9 / 22 correct responses



Frequently Missed Questions

If I have a data frame "df" with 2 rows and 5 columns, what will df[,1] return?

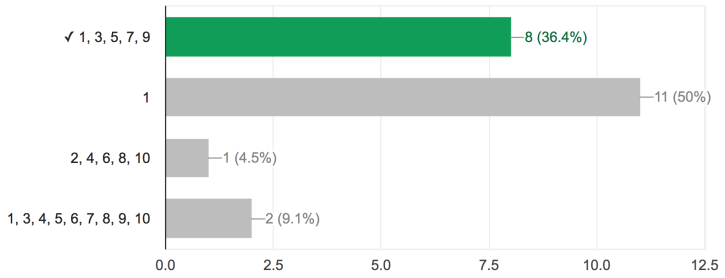
4 / 22 correct responses



Frequently Missed Questions

If x <- 1:10 Predict without using R the result of: x[c(TRUE, FALSE)]

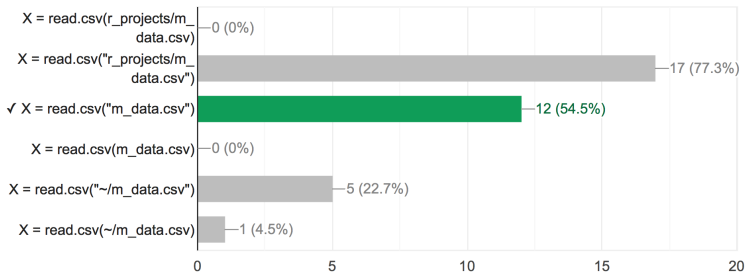
8 / 22 correct responses



Frequently Missed Questions

The working directory is in the "r_projects" folder, and it contains the file m_data.csv. Select the correct way(s) to read m_data into X.

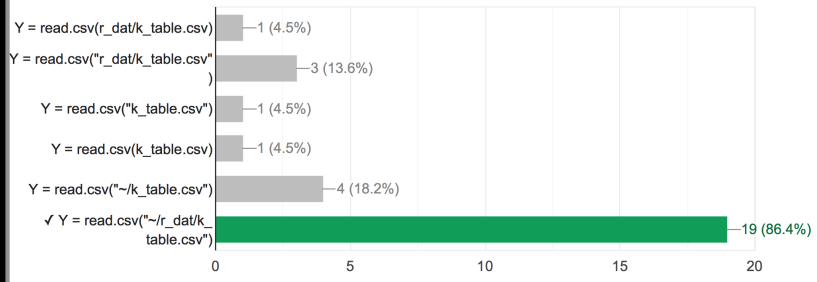
4 / 22 correct responses



Frequently Missed Questions

The "r_dat" folder is not in your working directory but it is in your home directory. It contains the file k_table.cs...rrect way(s) to read k_table.csv into Y.

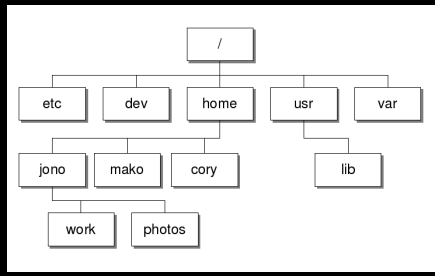
15 / 22 correct responses



7 of 7

File System Structure

- Information in the file system is stored in files, which are stored in directories (folders). Directories can also store other directories, which forms a directory tree.



- The forward slash character '**/**' is used to represent the root directory of the whole file system, and is also used to separate directory names. E.g. **/home/jono/work/bggn213_notes.txt**

UNIX Basics: Using the filesystem

ls	List files and directories
cd	Change directory (i.e. move to a different 'folder')
pwd	Print working directory (which folder are you in)
mkdir	MaKe a new DIRectories
cp	CoPy a file or directory to somewhere else
mv	MoVe a file or directory (basically rename)
rm	ReMove a file or directory

Side Note: File Paths

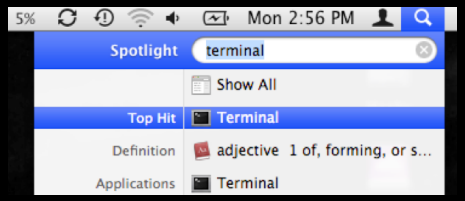
- An absolute path specifies a location from the root of the file system. E.g. **/home/jono/work/bggn213_notes.txt**
- A relative path specifies a location starting from the current location. E.g. **../bggn213_notes.txt**

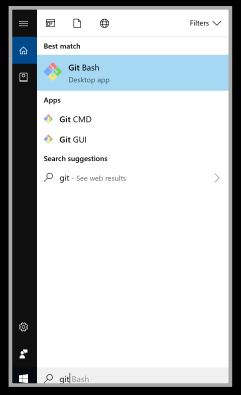
.	Single dot '**.**' (for current directory)
..	Double dot '**..**' (for parent directory)
~	Tilda '**~**' (for your home directory)
[Tab]	Pressing the tab key can autocomplete names

Lets get started…

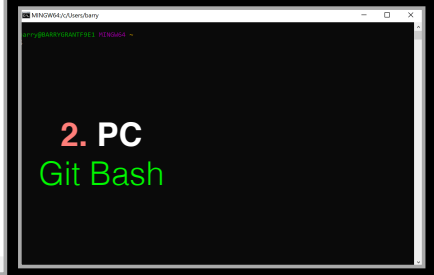
Do it Yourself!

1. Mac
Terminal





2. PC
Git Bash



3. OR:
RStudio Terminal

In RStudio: Tools > Terminal > New Terminal

Do it Yourself!

2. class-material (bash)

```
# Print Working Directory:  a.k.a.  where the hell am I?
> pwd

# List out the files and directories where you are
> ls
```

This is a comment line

This is our first UNIX command :-)

Don't type the ">" bit it is the "shell prompt"!

Q. What do you see after each command?

Q. Does it make sense if you compare to your Mac: *Finder* or Windows: *File Explorer*?

---

2. class-material (bash)

```
# Change to your Desktop directory
> cd ~/Desktop
> pwd

# Download an online file to your current directory
> curl -O https://bioboot.github.io/bggn213_W19/class-material/bggn213_01_unix.zip
```

Q. Does what you see at each step make sense if you compare to your Mac: *Finder* or Windows: *File Explorer*?

---

2. class-material (bash)

```
# Download any file to your current directory/folder
> curl -O https://bioboot.github.io/bggn213_S18/class-material/bggn213_01_unix.zip

# List out the files and directories where you are (NB: Use TAB for auto-complete)
> ls
> ls bggn213_01_unix.zip

# Un-zip your downloaded file
> unzip bggn213_01_unix.zip
> ls

# Change directory (i.e. move to the folder named bggn213_01_unix)
> cd bggn213_01_unix
> ls
> pwd
```

Q. Does what you see at each step make sense if you compare to your Mac: *Finder* or Windows: *File Explorer*?

2. class-material (bash)

```
# Practice moving around the file system…
> cd projects
> ls
> pwd
> cd ..
```

Basics: Using the filesystem

ls	List files and directories
cd	Change directory (i.e. move to a different 'folder')
pwd	Print working directory (which folder are you in)
mkdir	MaKe a new DIRectories
cp	CoPy a file or directory to somewhere else
mv	MoVe a file or directory (basically rename)
rm	ReMove a file or directory

Basics	File Control	Viewing & Editing Files	Misc. useful	Power commands	Process related
ls	mv	less	curl	grep	top
cd	cp	head	chmod	find	ps
pwd	mkdir	tail	wc	sed	kill
man	rm	nano	echo	sudo	Crl-c
ssh	| (pipe)	touch	source	git	Crl-z
scp	> (write to file)		cat	R	bg
	< (read from file)		tmux	python	fg

Inspecting text files

- **less** - visualize a text file:
  - use arrow keys
  - page down/page up with "space"/"b" keys
  - search by typing "/"
  - quit by typing "q"
- Also see: **head**, **tail**, **cat**, **more**

Creating text files

Creating files can be done in a few ways:

- With a **text editor** (such as **nano**, emacs, or vi)
- With the **touch** command (> touch a_file)
- From the command line with **cat** or **echo** and **redirection** (more on this later)

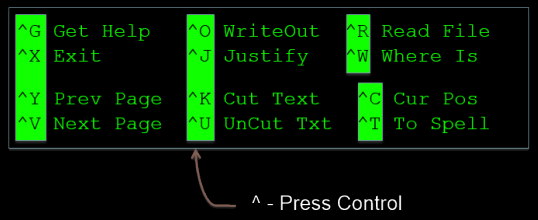
- **nano** is a simple text editor that is recommended for first-time users. Other text editors have more powerful features but also steep learning curves

Creating and editing text files with **nano**

Do it Yourself!

In the terminal type:

> nano yourfilename.txt



- There are many other text file editors (e.g. vim, emacs and sublime text, etc.)

Connecting to remote machines (with **ssh**)

Most high-performance computing (HPC) resources can only be accessed by ssh (Secure SHell)

> ssh [user@host.address]

For example:

> ssh barry@bio3d.ucsd.edu
>
> User Host address

> ssh -i ~/bggn213_private_key tb170077@IP_ADDRESS
>
> Optional key file User Host address

Copying to and from remote machines (**scp**)

- The **scp** (Secure CoPy) command can be used to copy files and directories from one computer to another.

> scp [file] [user@host]:[destination]
> scp localfile.txt barry@bigcomputer.net:/remotedir/.

Basics	File Control	Viewing & Editing Files	Misc. useful	Power commands	Process related
ls	mv	less	chmod	grep	top
cd	cp	head	echo	find	ps
pwd	mkdir	tail	wc	sed	kill
man	rm	nano	curl	sudo	Crl-c
ssh	| (pipe)	touch	source	git	Crl-z
scp	> (write to file)		cat	R	bg
	< (read from file)		tmux	python	fg

Process refers to a running instance of a program

top	Provides a real-time view of all running processes
ps	Report a snapshot of the current processes
kill	Terminate a process (the “force quit” of the unix world)
Crl-c	Stop a job
Crl-z	Suspend a job
bg	Resume a suspended job in the background
fg	Resume a suspended job in the foreground
&	Start a job in the background

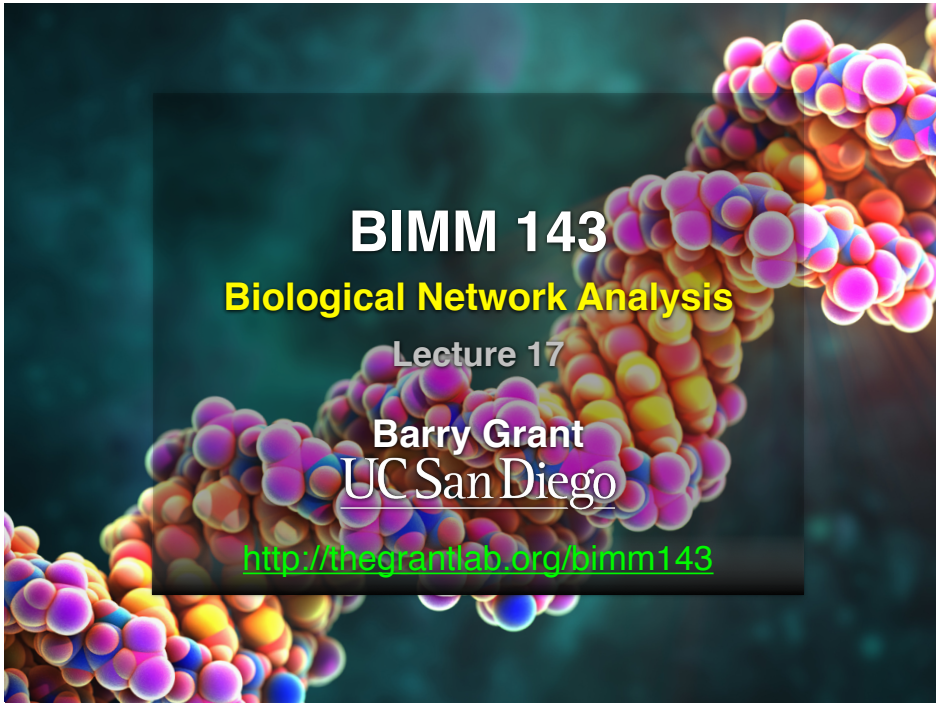
Do it Yourself!

Hands-on time

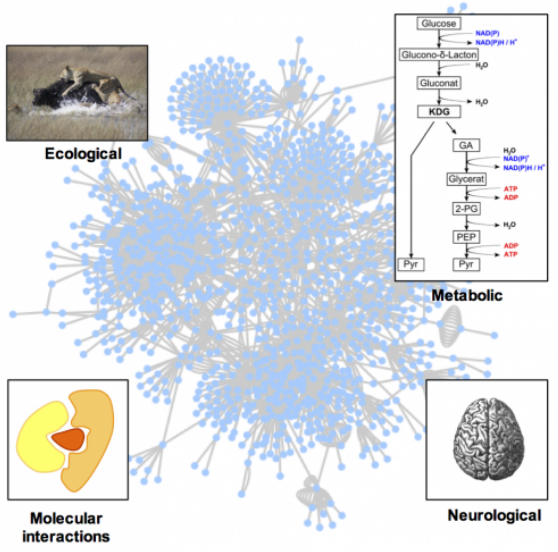
Sections 1 to 3 of software carpentry UNIX lesson

https://swcarpentry.github.io/shell-novice/

https://explainshell.com ~20 mins



Networks can be used to model many types of biological data



TODAYS MENU:

- Network introduction
- Network visualization
- Network analysis
- Hands-on:
  Cytoscape and R (igraph) software tools for network visualization and analysis

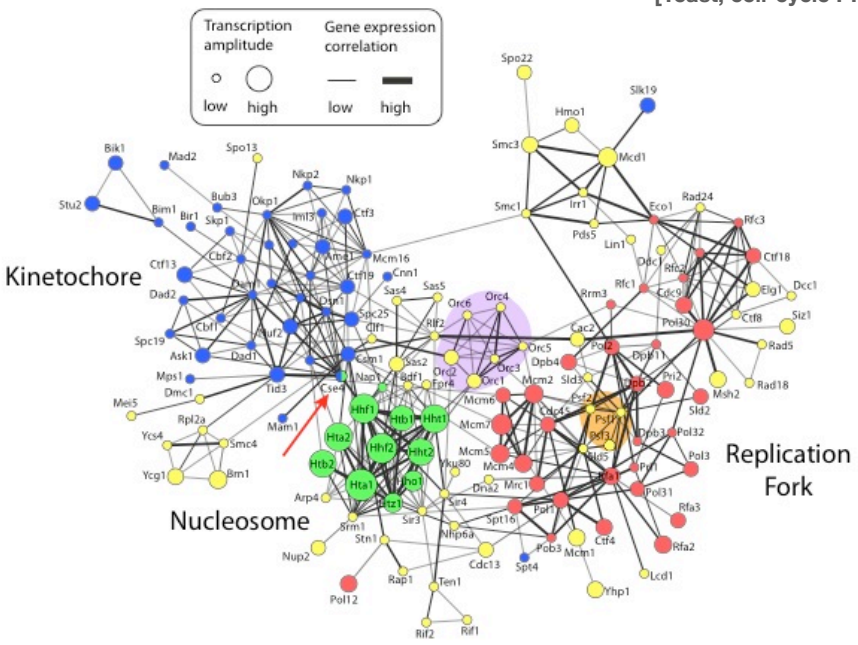
TODAYS MENU:

- Network introduction
- Network visualization
- Network analysis
- Hands-on:
  Cytoscape and R (igraph) software tools for network visualization and analysis

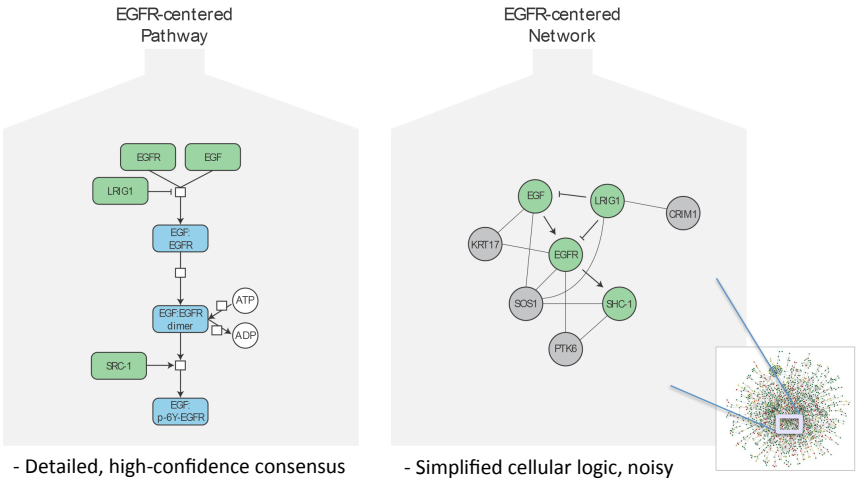
Biological Networks

- **Represent biological interactions**
  - ➡ Physical, regulatory, genetic, functional, etc.
- **Useful for discovering relationships in big data**
  - ➡ Better than tables in Excel
- **Visualize multiple heterogenous data types together**
  - ➡ Help highlight and see interesting patterns
- **Network analysis**
  - ➡ Well established quantitive metrics from graph theory

[Yeast, cell-cycle PPIs]

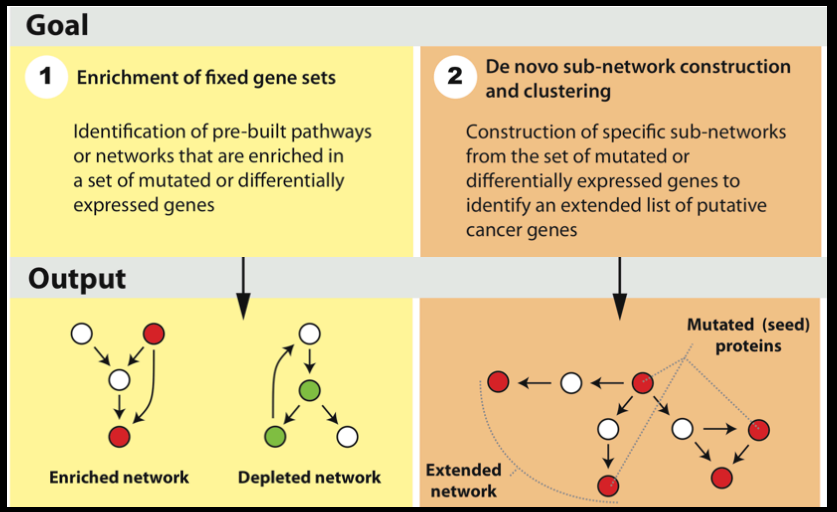


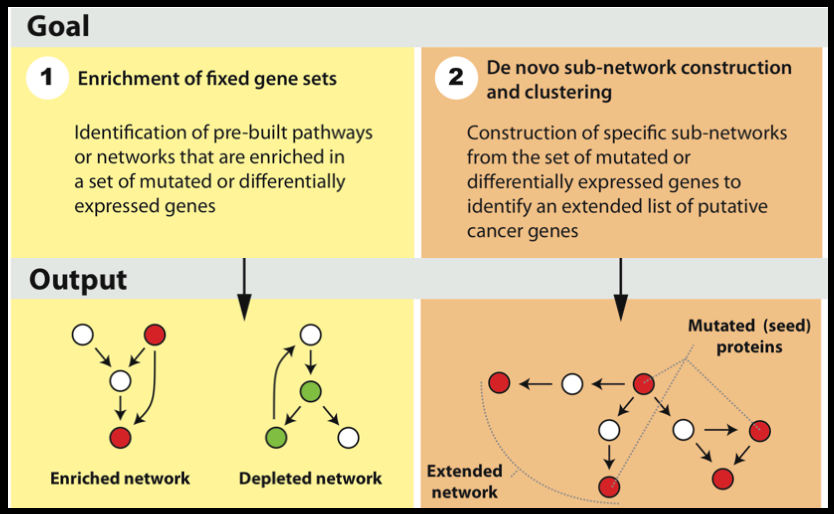
Pathways vs Networks



- Detailed, high-confidence consensus
- Biochemical reactions
- Small-scale, fewer genes
- Concentrated from decades of literature

- Simplified cellular logic, noisy
- Abstractions: directed, undirected
- Large-scale, genome-wide
- Constructed from *omics* data integration





What biological process is altered in this cancer?

Are NEW pathways altered in this cancer? Are there clinically relevant tumor subtypes?

Network analysis is complementary to pathway analysis and can be used to show how key components of different pathways interact.

This can be useful for identifying regulatory events that influence multiple biological processes and pathways

Next Class

Network analysis approaches

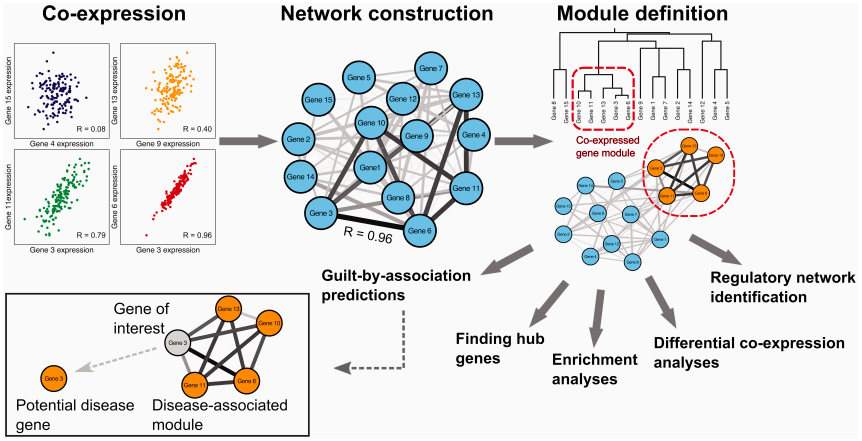
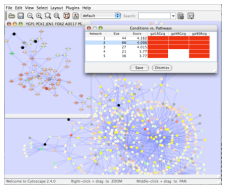
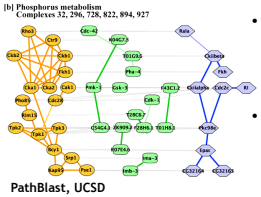


Image from: van Dam et al. (2017) https://doi.org/10.1093/bib/bbw139

Applications of Network Biology



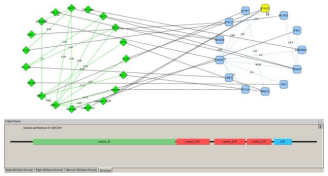
jActiveModules, UCSD



PathBlast, UCSD

- **Gene Function Prediction** – shows connections to sets of genes/proteins involved in same biological process
- **Detection of protein complexes/other modular structures** – discover modularity & higher order organization (motifs, feedback loops)
- **Network evolution** – biological process(es) conservation across species
- **Prediction of new interactions and functional associations** – Statistically significant domain-domain correlations in protein interaction network to predict protein-protein or genetic interaction; allostery in molecular networks

MCODE, University of Toronto



DomainGraph, Max Planck Institute

Slide from: humangenetics-amc.nl

What's missing

- **Dynamics**
  - ➡ Pathways/networks represented as static processes
  - ➡ Difficult to represent a calcium wave or a feedback loop
  - ➡ More detailed mathematical representations exist that handle these e.g. Stoichiometric modeling, Kinetic modeling (VirtualCell, E-cell, …)
- **Detail** – atomic structures & exclusivity of interactions.
- **Context** – cell type, developmental stage

What have we learned so far…

- **Networks are useful for seeing relationships in large data sets**
  - ➡ Important to understand what the nodes and edges mean
  - ➡ Important to define the biological question - know what you want to do with your gene list or network
- **Many methods available for network analysis**
  - ➡ Good to determine your question and search for a solution
  - ➡ Or get to know many methods and see how they can be applied to your data

TODAYS MENU:

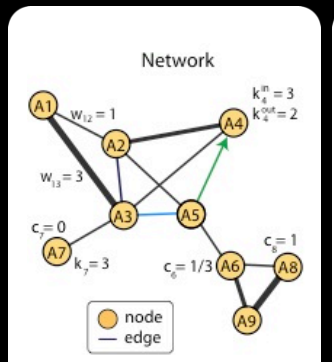
- Network introduction
- **Network visualization**
- **Network analysis**
- **Hands-on:**
  Cytoscape and R (igraph) software tools for network visualization and analysis

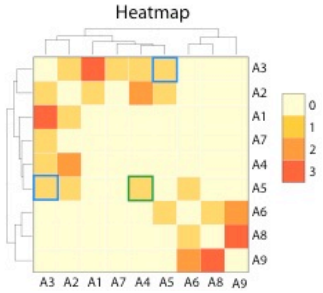
Network Visualization Outline

- Network representations
- Automatic network layout
- Visual features
- Visually interpreting a network

Network representations

Relationships	Optional weight
A1 ↔ A2	1
A1 ↔ A3	3
A2 ↔ A3	1
A2 ↔ A4	2
A2 ↔ A5	1
A3 ↔ A4	1
A3 ↔ A5	1
A3 ↔ A7	1
A5 → A4	1
A5 ↔ A6	1
A6 ↔ A8	1
A6 ↔ A9	2
A8 ↔ A9	3





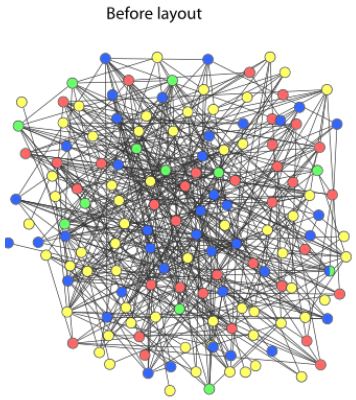
1 List of relationships

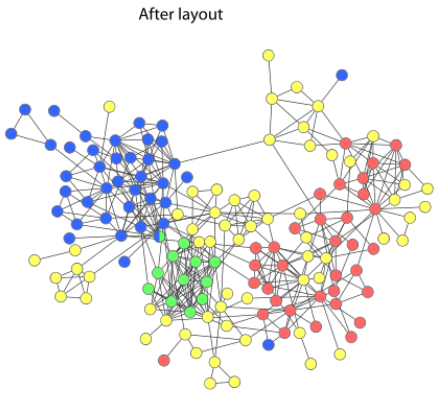
2 Network view

3 Adjacency matrix view

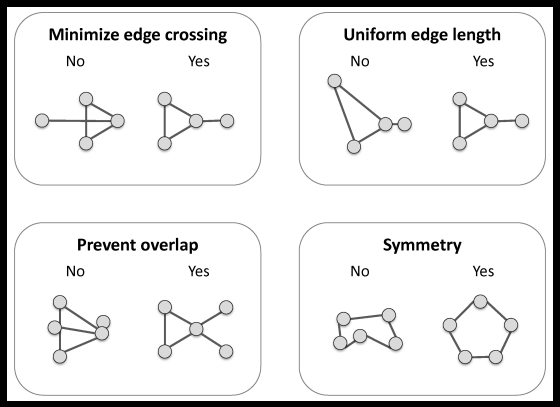
Network view is most useful when network is sparse!

Automatic network layout





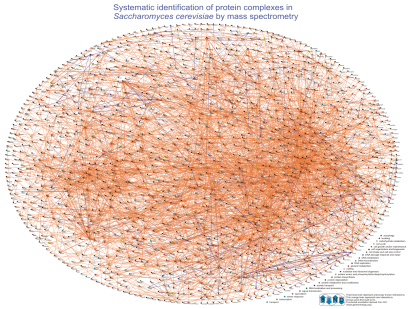
- Modern **graph layouts** are optimized for speed and aesthetics. In particular, they seek to minimize overlaps and edge crossing, and ensure similar edge length across the graph.



Force-directed layout:
Nodes repel and edges pull

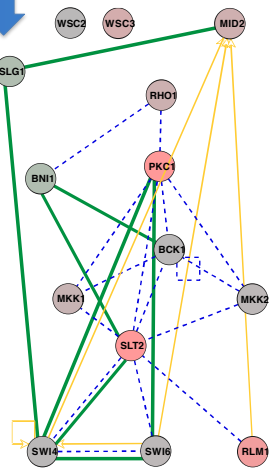
- Good for up to 500 nodes
  - ➡ Bigger networks give hairballs
  - ➡ Reduce number of edges
  - ➡ Or just use a heatmap for dense networks
- Advice: try force directed first, or hierarchical for tree-like networks
- Tips for better looking networks
  - ➡ Manually adjust layout
  - ➡ Load network into a drawing program (e.g. Illustrator) and adjust labels

Dealing with 'hairballs': zoom or filter



Zoom

Focus



PKC Cell Wall Integrity

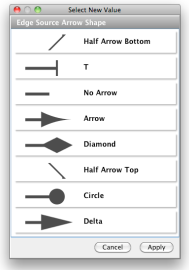
Synthetic Lethal
Transcription Factor Regulation
Protein-Protein Interaction

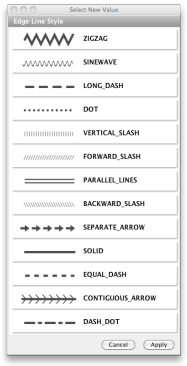
Up Regulated Gene Expression

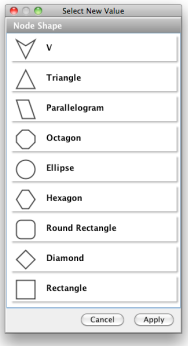
Down Regulated Gene Expression

Visual Features

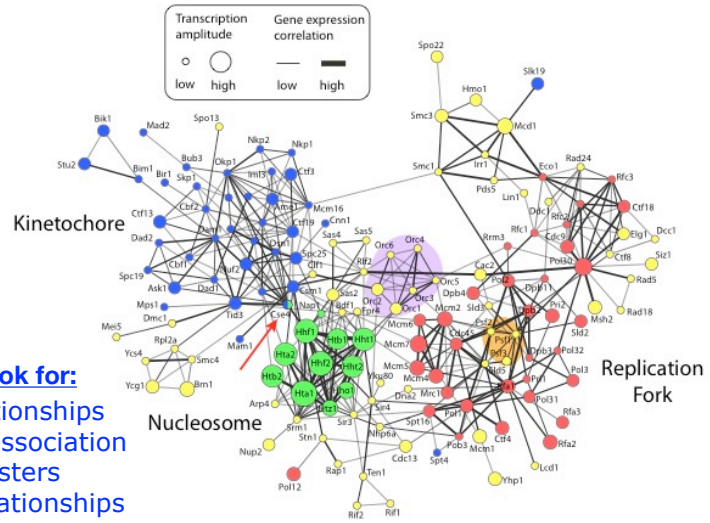
- Node and edge attributes
  - Text (string), integer, float, Boolean, list
  - E.g. represent gene, interaction attributes
- Visual attributes
  - Node, edge visual properties
  - Color, shape, size, borders, opacity...







Visually Interpreting a Network



What to look for:

Data relationships
Guilt-by-association
Dense clusters
Global relationships

What have we learned so far…

- Automatic layout is required to visualize networks
- Networks help you visualize interesting relationships in your data
- Avoid hairballs by focusing analysis
- Visual attributes enable multiple types of data to be shown at once – useful to see their relationships

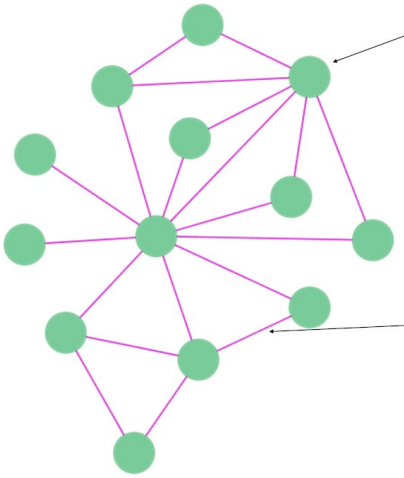
TODAYS MENU:

- Network introduction
- Network visualization
- **Network analysis**
- **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

Introduction to graph theory

- Biological network analysis historically originated from the tools and concepts of social network analysis and the application of graph theory to the social sciences.
- Wikipedia defines graph theory as:
  - ➡ "[…] the study of graphs used to model pairwise relations between objects. A graph in this context is made up of **vertices** connected by **edges**".
- In practical terms, it is the set of concepts and methods that can be used to visualize and analyze networks

Network or graph

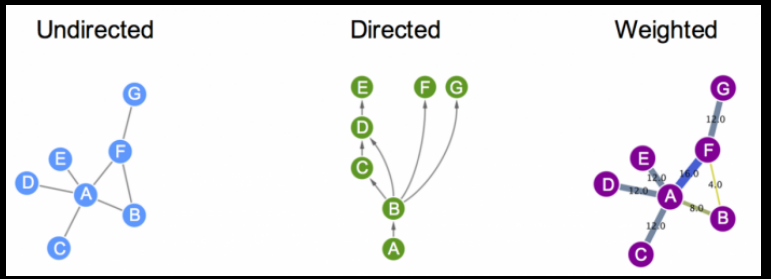


Node or vertex: protein, gene, drug, disease

Edge or link: relation between nodes

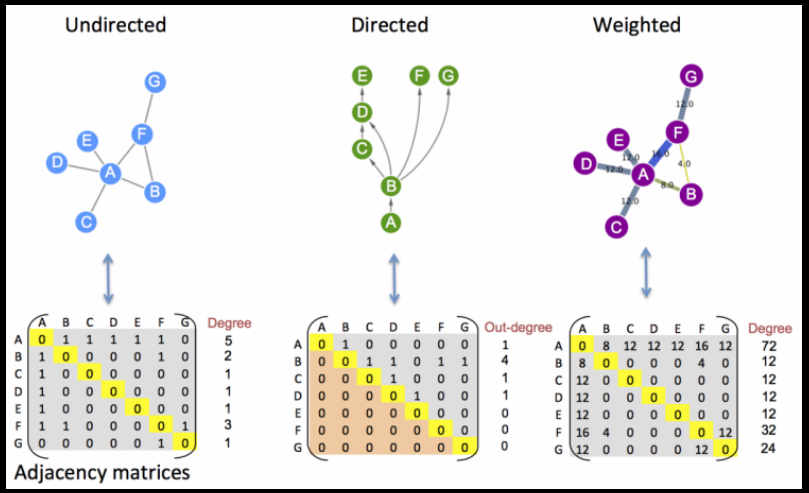
- Binary or continuous
- Directed or indirected
- Edge types

Types of network edges



Undirected	Directed	Weighted
Connection, without a given 'flow' implied (e.g. protein A binds protein B)	There is directional flow/signal implied (e.g. metabolic or gene networks)	Edges can also have weight (i.e. a 'strength' of interaction).

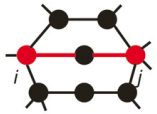
- Every network can be expressed mathematically in the form of an adjacency matrix



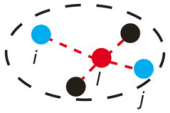
Network topology

- Topology is the way in which the nodes and edges are arranged within a network.
- The most used topological properties and concepts include:
  - **Degree** (i.e. how may node neighbors)
  - **Communities** (i.e. clusters of well connected nodes)
  - **Shortest Paths** (i.e. shortest distance between 2 nodes)
  - **Centralities** (i.e. how 'central' is a given node?)
  - **Betweenness** (a measure of centrality based on shortest paths)

Degree k_i = number of links connected to node i



Distance d_{ij} = shortest path length between node i and j

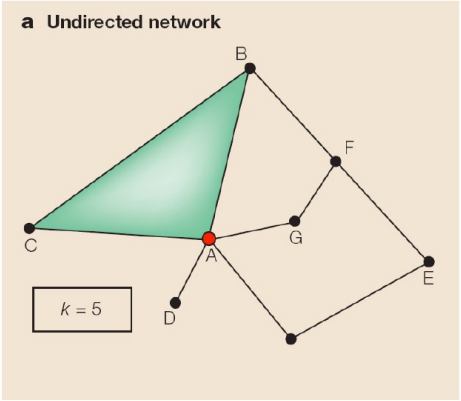


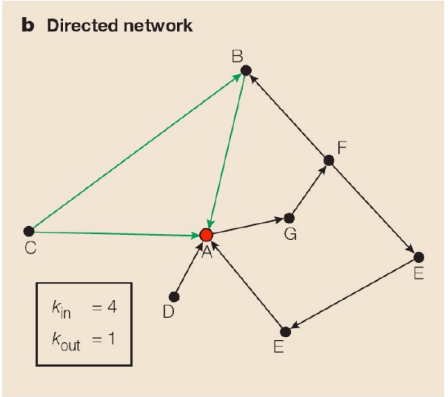
Betweenness $b_l = \sum_{ij} p_{ij}(l)/p_{ij}$

p_{ij} : number of shortest paths between i and j

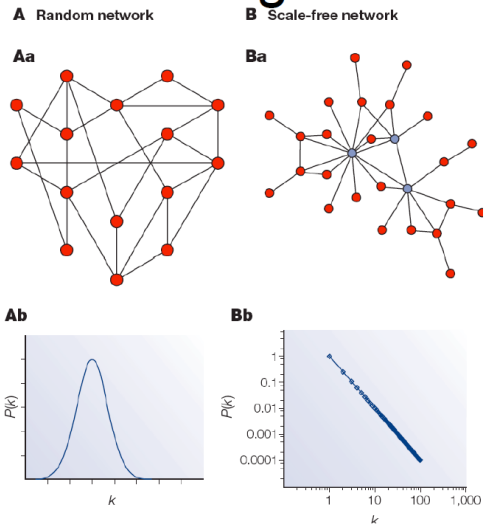
$p_{ij}(l)$: number of shortest paths between i and j going through node l

Network Measures: Degree





Degree Distribution



P(k) is probability of each degree k, i.e fraction of nodes having that degree.

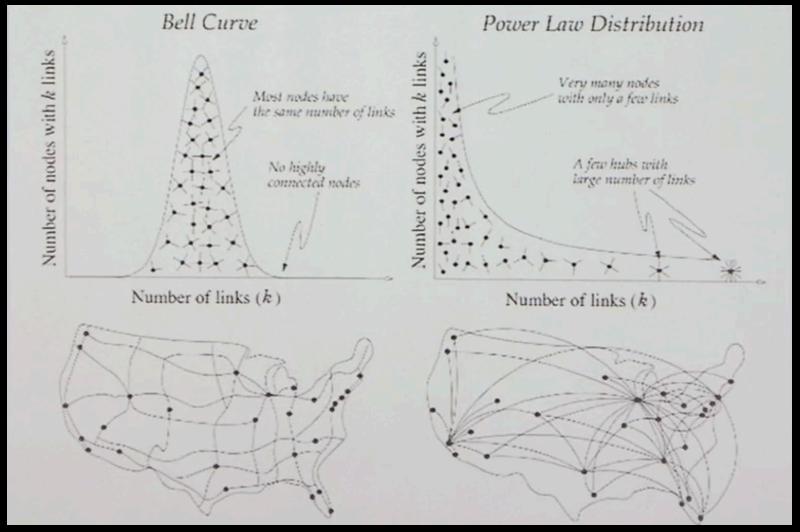
For random networks, P(k) is normally distributed.

For real networks the distribution is often a power-law:

$P(k) \sim k^{-\gamma}$

Such networks are said to be scale-free

Random graphs vs scale free



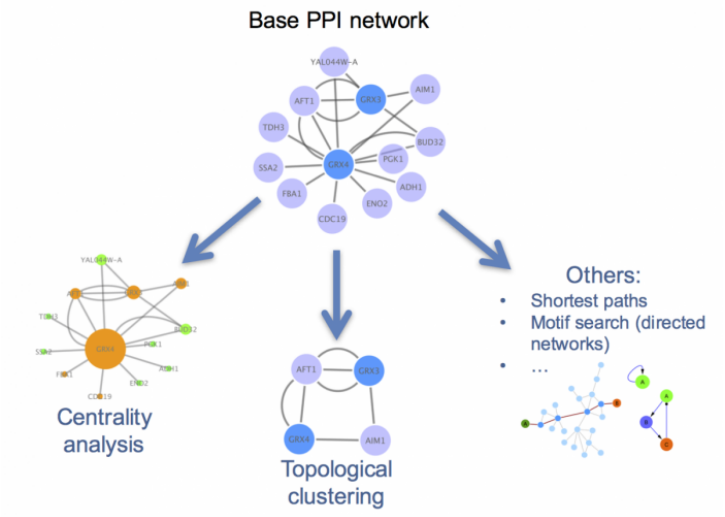
Scale-Free Networks are Robust

- Complex systems (cell, internet, social networks), are resilient to component failure
- Network topology plays an important role in this robustness
  - Even if ~80% of nodes fail, the remaining ~20% still maintain network connectivity
- *Attack vulnerability* if hubs are selectively targeted
- In yeast, only ~20% of proteins are lethal when deleted, and are 5 times more likely to have degree $k>15$ than $k<5$.

Implications

- Many biological networks (protein-protein interaction networks regulatory networks, etc…) are thought to have hubs, or nodes with high degree.
- For protein-protein interaction networks (PPIs) these hubs have been shown to be older [1] and more essential than random proteins [2]
  - [1] Fraser et al. *Science* (2002) 296:750
  - [2] Jeoung et al. *Nature* (2001) 411:41

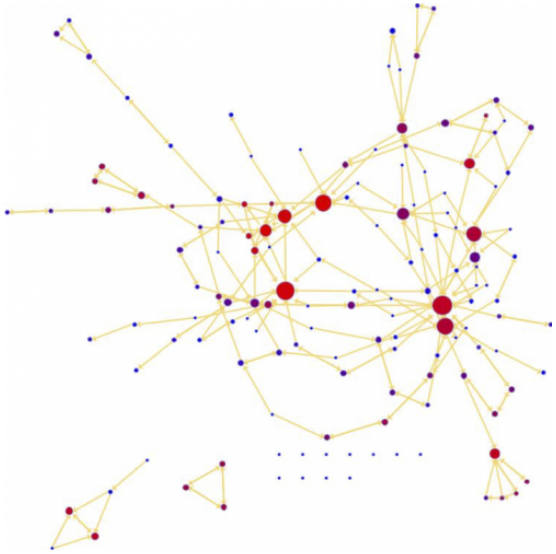
Analyzing the topological features of a network is a useful way of identifying relevant participants and substructures that may be of biological significance.



Centrality analysis

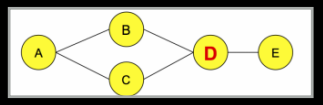
- Centrality gives an estimation on how important a node or edge is for the connectivity or the information flow of the network
- It is a useful parameter in signalling networks and it is often used when trying to find drug targets.
- Centrality analysis in PPINs usually aims to answer the following question:
  - Which protein is the most important and why?

Bigger, redder nodes have higher **centrality values** in this representation.



Betweenness centrality

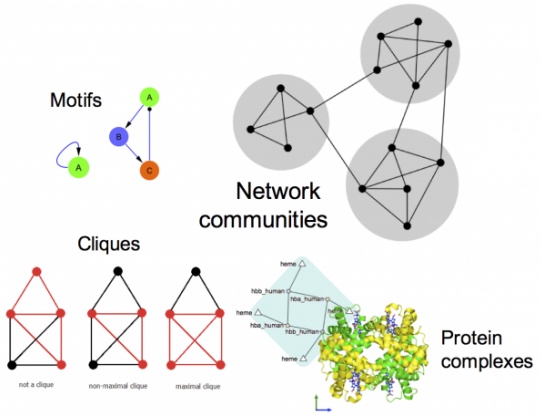
- Nodes with a high betweenness centrality are interesting because they lie on communication paths and can control information flow.
- The number of shortest paths in the graph that pass through the node divided by the total number of shortest paths.
- Betweenness centrality measures how often a node occurs on all shortest paths between two nodes.



Community analysis

- **Community**: A general, catch-all term that can be defined as a group (i.e. *cluster*) of nodes that are more connected within themselves than with the rest of the network. The precise definition for a community will depend on the method or algorithm used to define it.

Looking for communities in a network is a nice strategy for reducing network complexity and extracting functional modules (e.g. protein complexes) that reflect the biology of the network.



TODAYS MENU:

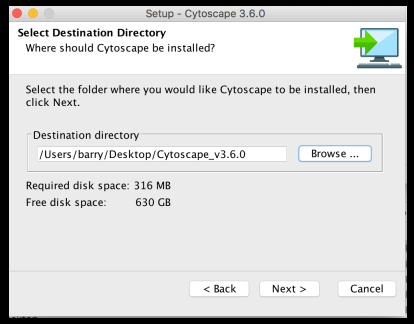
- Network introduction
- Network visualization
- Network analysis
- **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

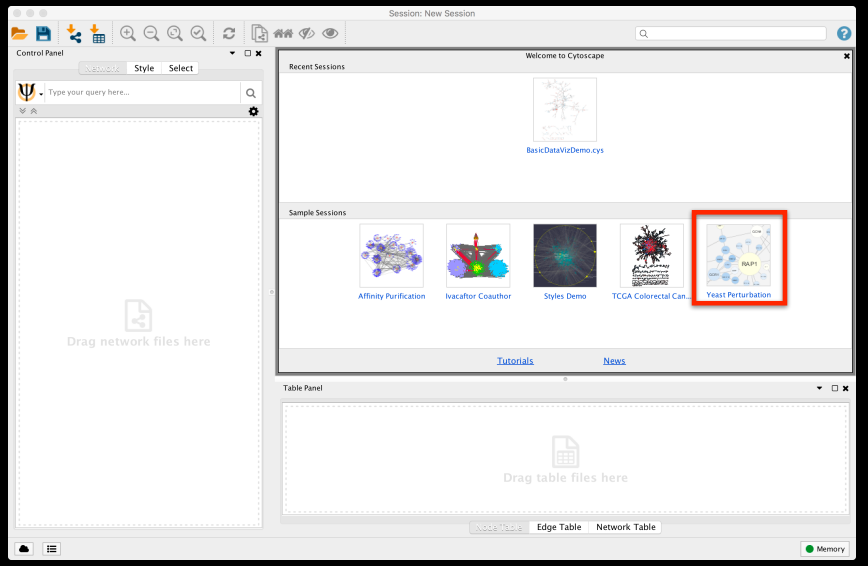
Practical issues

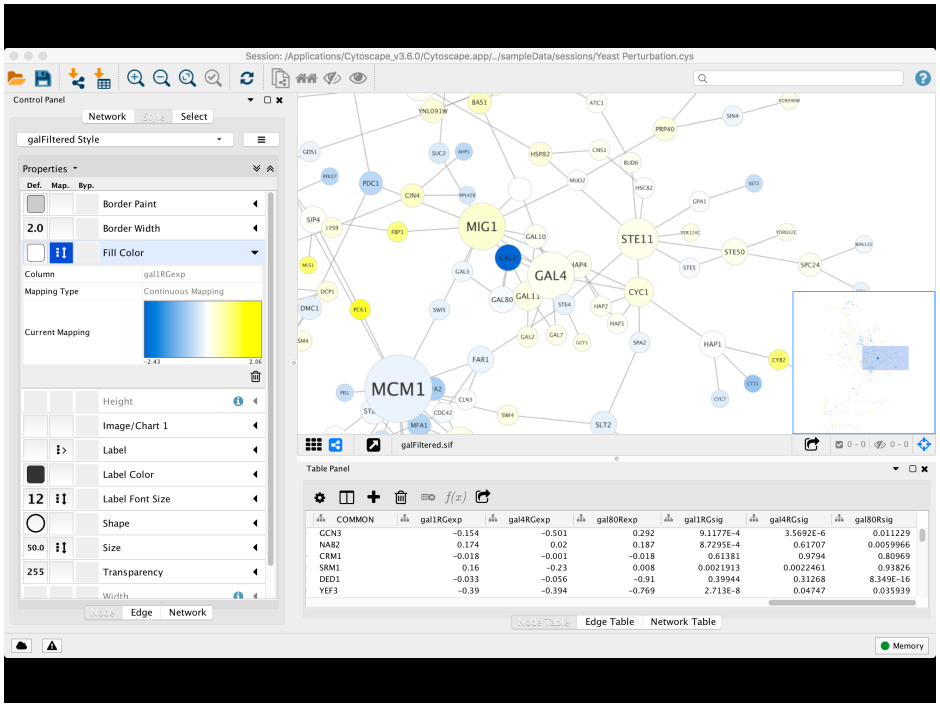
- Major tools for the **creation**, **manipulation** and **visualization** of biological networks include:
  - Cytoscape,
  - Gephi
  - R packages (igraph, graph, tidygraph)
- Tools for network analysis and modeling include:
  - Cytoscape apps/plugins
  - R packages (igraph and others)
  - NetworkX (for Python)
  - ByoDyn, COPASI

http://cytoscape.org/download.php

Note: If you are on a classroom Mac please check if Cytoscape is already installed. If not then please be sure to install to your **Desktop** directory!







Cytoscape Memory Issues

- Cytoscape uses lots of memory and doesn't like to let go of it
  - ➡ An occasional restart when working with large networks is a good thing
  - ➡ Destroy views when you don't need them
- Since version 2.7, Cytoscape does a much better job at "guessing" good default memory sizes than previous versions but it still not great!
  - ➡ Java doesn't give us a good way to get the memory right at start time

Cytoscape 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

Hands-on: Part 1

https://bioboot.github.io/bggn213_W19/lectures/#17

- The data used in **part 1** is from yeast, and the genes Gal1, Gal4, and Gal80 are all yeast transcription factors. The experiments all involve some perturbation of these transcription factor genes.
- In this network view, the following node attributes have been mapped to visual style properties in cytoscape:
  - ➡ The "gal80exp" expression values are used for Node Fill Color.
  - ➡ The Default Node Color, for nodes with no data mapping, is dark grey.
  - ➡ Nodes with expression values that are significant are rendered as rectangles, others are ovals.
  - ➡ The common name for each gene is used as the Node Label.

Hands-on: Part 2

Skip for today!

https://bioboot.github.io/bggn213_W19/lectures/#17

- The data used in **part 2** is from an ocean metagenomic sequencing project - where all the genetic material in a sample of ocean water is sequenced.
- We will use the R package **igraph** and the bioconductor package **RCy3** together with Cytoscape.
- Many of these microbial species in these types of studies have not yet been characterized in the lab.
  - ➡ Thus, to know more about the organisms and their interactions, we can observe which ones occur at the same sites.
  - ➡ One way to do that is by using **co-occurrence networks** where you examine which organisms occur together at which sites.

Revisit Lecture 11

Do this now next day!

https://bioboot.github.io/bggn213_W19/lectures/#11

Muscle setup instructions (at your command line TERMINAL)

Windows install and setup cmd:

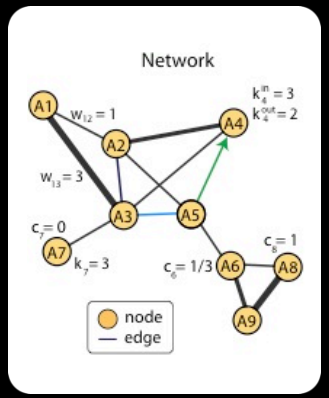
```
cd ~/Desktop
curl -o "muscle.exe" "https://www.drive5.com/muscle/downloads3.8.31/muscle3.8.31_i86win32.exe"
```

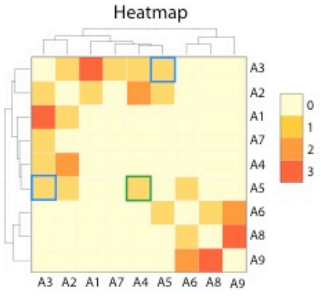
Mac install and setup cmd:

```
curl -o "/usr/local/bin/muscle" "http://thegrantlab.org/misc/muscle"
chmod +x /usr/local/bin/muscle
```

Network representations

Relationships	Optional weight
A1 ↔ A2	1
A1 ↔ A3	3
A2 ↔ A3	1
A2 ↔ A4	2
A2 ↔ A5	1
A3 ↔ A4	1
A3 ↔ A5	1
A3 ↔ A7	1
A5 → A4	1
A5 ↔ A6	1
A6 ↔ A8	1
A6 ↔ A9	2
A8 ↔ A9	3





1. **List of relationships**
2. **Network view**
3. **Adjacency matrix view**

Network view is most useful when network is sparse!

Summary

- Network biology makes use of the tools provided by **graph theory** to represent and analyze complex biological systems.
- Major types of biological networks include: genetic, metabolic, cell signaling etc.
- Networks are represented by **nodes** and **edges**.
- Biological networks have a number of characteristics, mainly:
  - ➡ **Scale-free**: A small number of nodes (hubs) are a lot more connected than the average node.
  - ➡ **Transitivity**: The networks contain communities of nodes that are more connected internally than they are to the rest of the network.
- Major tools for network analysis include: **Cytoscape**, **igraph**, Gephi and NetworkX.
- Two of the most used topological methods to analyze PPINs are:
  - ➡ **Centrality analysis**: Which identifies the most important nodes in a network, using different ways to calculate centrality.
  - ➡ **Community detection:** Which aims to find heavily inter-connected components that may represent protein complexes and machineries

Summary cont…

- **Cytoscape** is a useful, free software tool for network visualization and analysis
  - ➡ Provides basic network manipulation features
  - ➡ Plugins/Apps are available to extend the functionality
- The R **igraph** package has extensive network analysis functionality beyond that in Cytoscape
- The R bioconductor package **RCy3** package allows us to bring networks and associated data from R to Cytoscape so we can have the best of both worlds.

Network Analysis Overview