Lab 5 – Visualizing Biological Networks

[Software needed: web access and Cytoscape – see **where to get it** at the end of the lab]

In this lab, we will use a piece of standalone software for the dynamic representation and annotation of protein interaction networks, Cytoscape. We'll also check out D3 for representing networks in a browser.

Typically, proteins do not float around freely in the cell, but rather act in concert with other proteins to create larger cellular systems. Even if a given protein does "float around" this may be during a signal transduction event in which previously it contacted membrane-bound proteins that perceived some signal and subsequently it might then interact with downstream components to bring about a cellular response. Therefore protein-protein interactions are a very important aspect of biology.

A large assortment of protein-protein interaction databases exist and, for the large part, a canonical reference has still to emerge. There are presently over 50 PPI databases online, possessing PPI data for a variety of organisms. We will retrieve data from just one of these, BioGRID. See Kangueane and Nilofer (2018) for a partial list of PPI databases.

Box 1. Identifying Protein-Protein Interactions in the Lab

While the ability to identify protein-protein interactions has existed for many years, the "classical" biochemical and chromatographic methods for doing so are, while robust, decidedly low throughput, and are not readily automatable for data generation in the post-genomic era.

One of the first high throughput methods for detecting protein-protein interactions was the yeast two hybrid (**Y2H**) system, developed by Fields & Song in 1989. Essentially, the protein coding sequences to be tested for interaction are cloned in frame with either the activation domain or binding domain of the yeast GAL4 transcription factor. For a high throughput screen, one protein coding sequence would be used as a "bait" and a library contain many thousands of protein coding sequences as the "prey". If two proteins interact under the conditions of the assay, they effectively reconstitute the activity of GAL4, and transcription of a reporter gene, such as LacZ, occurs. The plasmids in the yeast colonies exhibiting the reporter signal may be recovered and sequenced to determine the identity of the interacting partners.

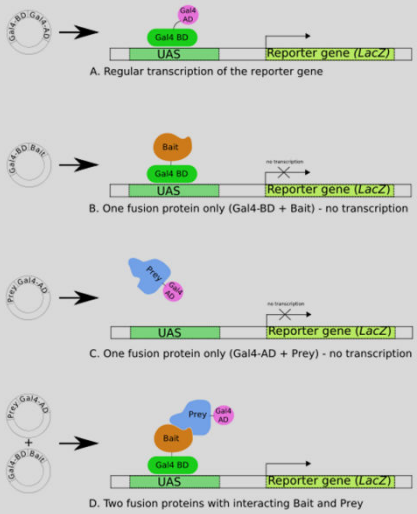
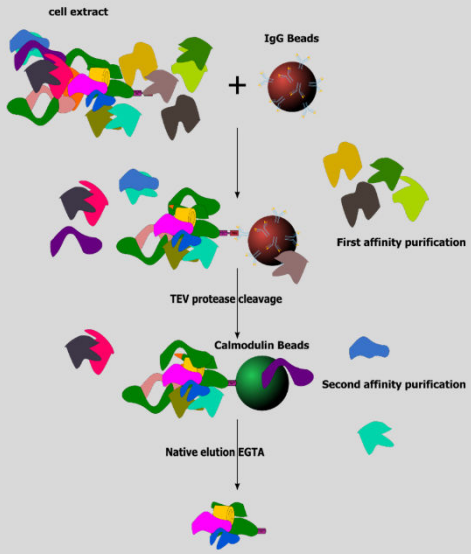


Image courtesy of Anna K., under the GNU Free Documentation License, Version 1.2.

In practice, there are several problems with the yeast two hybrid system. The protein hybrids must be targeted to the yeast nucleus, so membrane-bound and membrane-associated interacting proteins can seldom be identified. Additionally, many proteins are inherently "sticky" so the rate of false positives can be quite high – this is exacerbated because the

hybrid proteins are typically overexpressed in the yeast cells. Adaptations to the original method have been devised to obviate some of these problems, and on the plus side, the Y2H system can be quite sensitive to transient interactions.

To identify protein-protein interactions in their endogenous context, affinity purification may be used. Here antibodies to a given protein of interest can be used to capture that protein and its interactors from cell extracts. Proteins that co-purify with a given target protein are then identified by mass spectrometry. A variation of this method is the tandem affinity purification (**TAP**) tag method, developed in 1999 by Bertrand Séraphin and colleagues at the EMBL Laboratories in Heidelberg. A schematic of this method is shown to the right. Basically, two affinity tags are attached to the protein of interest, expressed under the control of the native or some other promoter. These tags are the calmodulin binding peptide and two IgG binding domains of Protein A from *S. aureus*. These are separated by a TEV protease cleavage site. Two rounds of affinity purification are thus possible, resulting in far fewer false positive interactions. The resultant interacting proteins are then identified by mass spectrometry. Drawbacks include the fact that the introduced tags may disrupt potential interactions. In general, the TAP tag method is less sensitive in detecting transient interactions and is better for identifying proteins in protein complexes.



TAP tagging image by Chandres, under a CC BY-SA 3.0 licence.

Another clever method (generically called "Proximity ID", commonly called "BioID") that also uses mass spec. attaches a biotin ligase to a given protein, such that nearby endogenous proteins will become biotinylated *in vivo*. These can then be easily purified and identified by mass spec. (Roux *et al.* 2016, Trends Cell Biol., doi: 10.1016/j.tcb.2016.09.004). Co-fractionation mass spec. is emerging a promising way to identify complexes in rapid manner, without the need to introduce tags, but this method requires good software (Drew et al., 2017, PLoS Comp. Biol., doi: 10.1371/journal.pcbi.1005625).

Last, it is increasingly possible to identify potential PPIs by computational methods. Proteins from a given species whose orthologs have been identified as interactors can also be assumed to interact. These are sometimes called interologs, for *inter*acting *ortho**logs***. Another computational approach involves docking the 3D structures for two proteins, using algorithms like HEX (Ritchie & Kemp, 2000, Proteins, PMID: 10737939) or Alphafold2 (e.g. Bryant & Noe, 2023, BioRXiv, doi: 10.1101/2023.04.15.536993).

Other high throughput methods have been developed for identifying interactions between membrane proteins, and other proteins that are considered difficult to work with. **It is important to recognize the limitations of each of these methods and to be aware of the quality of PPI data in the databases**. Multiple lines of interaction evidence are desirable. High throughput methods (both for sequencing and Y2H) have enabled determining which parts of proteins interact with one another, sometimes called "reverse edgetics" (Charloteaux *et al.* 2011, Yeast Systems Biology, doi: 10.1007/978-1-61779-173-4_12).

Cytoscape – Visualizing Protein-Protein Interactions

Let's visualize some PPI data available at BioGRID as a graph network using a powerful network viewing tool called Cytoscape. Although you can work with your own tables of protein-protein interactions and easily import these into Cytoscape, we'll be using a newer method of retrieving data on the fly from online repositories via web services. We'll also be using one of many plugins developed by researchers to perform a GO enrichment analysis of interactors of BRCA2, instead of trying to determine in an *ad hoc* manner which GO category is over-represented. Start the Cytoscape application (see **where to get it** at the end of the lab for installation details). Mutations in BRCA2 were discovered as drivers of breast cancer in 1995 by Wooster et al. (Nature 378: 789–792, https://doi.org/10.1038/378789a0) and much work has been done in the past 25 years to elucidate its role in DNA repair. It has been shown to interact with a lot of other proteins, so let's dive in and see what those are.

Start the Cytoscape app, then use the *File>Import>Network>Public Databases...* to retrieve the interactions in BioGRID, as described below (or enter BRCA2 beside the Ψ icon – switch it from the NDex default icon if necessary – in the top left and click on the magnifying glass).

1. Click on *File>Import>Network>Network from Public Databases…*
Configure the Import Network from Web Service dialogue box as shown in **Figure 1**: type BRCA2 into the **1. Enter Search Conditions** search box, and click Search. In **2. Select Databases** select BioGRID. Click Import and then Close when prompted (don't Manually Merge). Cancel out of the Import Networks dialogue box.

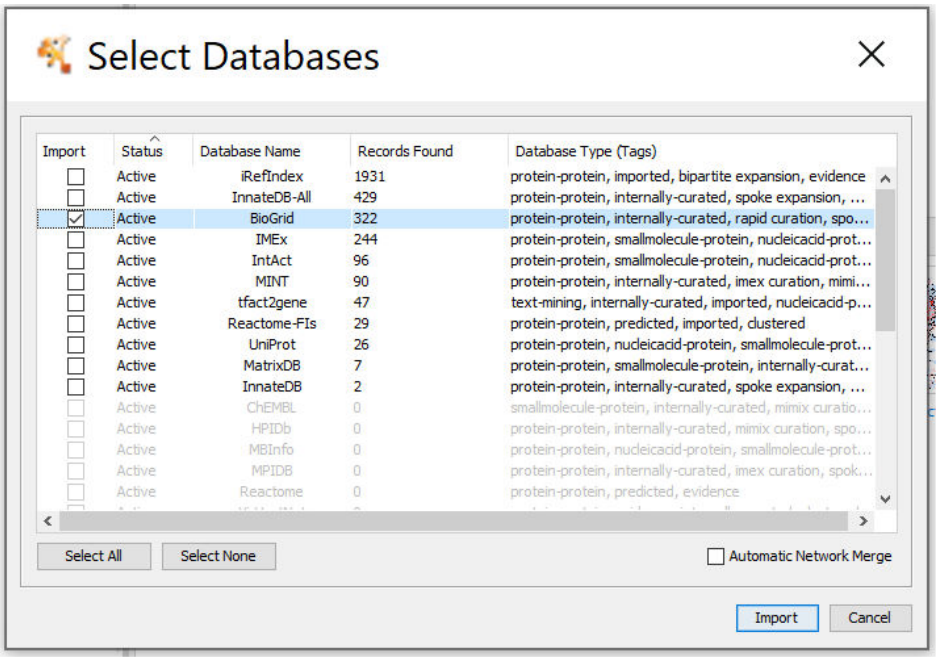


Figure 1: Importing a network into Cytoscape from BioGRID via web services. Search for the desired term in Step 1, select the appropriate database in Step 2 and finally click on Import.

Box 2. Protein-Protein Interaction Networks

Protein-protein interaction networks are typically visualized as "**graph networks**". The "nodes" in the graph network represent the proteins, while an "edge" connecting two nodes denotes a documented protein-protein interaction between the two proteins represented by the nodes.

There is a large body of literature on methods for graph network analysis, much of which has roots in social anthropological studies from the 1960s and '70s. The field of graph network analysis has gained importance in the last 20 years in diverse areas ranging from the study of the world-wide web, through social networks to protein-protein interaction networks in biology.

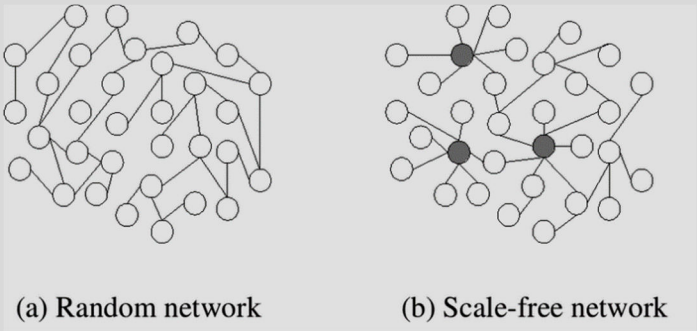


Image courtesy of ChaTo, under the GNU Free Documentation License, Version 1.2.

In many of these systems, including protein-protein interaction networks, the degree of connectivity of the nodes exhibits a "**scale-free**" property. That is, the structure of the network in terms of the distribution of the number of node connections is independent of the number of nodes in the network. What this means in terms of the network structure is that there are a few nodes that are highly connected, while the majority have few connections – see the above figure. The ones that are highly connected are called "**hubs**", and in the case of biological networks these can further be subdivided into "**party hubs**", which exhibit coexpression of the genes encoding the interacting partners, and "**date hubs**", which do not.

It is thought that scale-free networks provide a biological system with a high level of **robustness**, in that the loss of one component in general will not disrupt the system to a great extent as the majority of components do not have many connections. The structure of the Internet is similar, and it was in fact designed to be this way for robustness' sake. Of course, if a major "hub" is affected then one can expect a large effect. This is true both biologically and in the case of the Internet.

Explore Cytoscape's interface. You can zoom in on the network that you retrieve by clicking the magnifying glass icon in the tool bar along the top of the screen. If you click on a node, you will see that information about that node will appear in the Table Panel at the bottom of the screen (**Figure 2**). You will only see a few columns of data, but you can easily add other columns by clicking on the Show Columns button: . You can select multiple nodes by holding the shift key and clicking the other desired nodes, or by holding the left mouse key and drawing a box around the nodes of interest. You can also explore different layout options for the network using the Layout menu option – the "Edge-weighted Spring Embedded" layout is shown in **Figure 2**. You can also select specific edges (which denote interactions) by clicking on them and then switching to the Edge Table tab. Unfortunately, we only get a limited amount of information about how the interactions were determined from the BioGRID web service! You can get more information on how the interactions were determined from BioGRID at https://thebiogrid.org/.

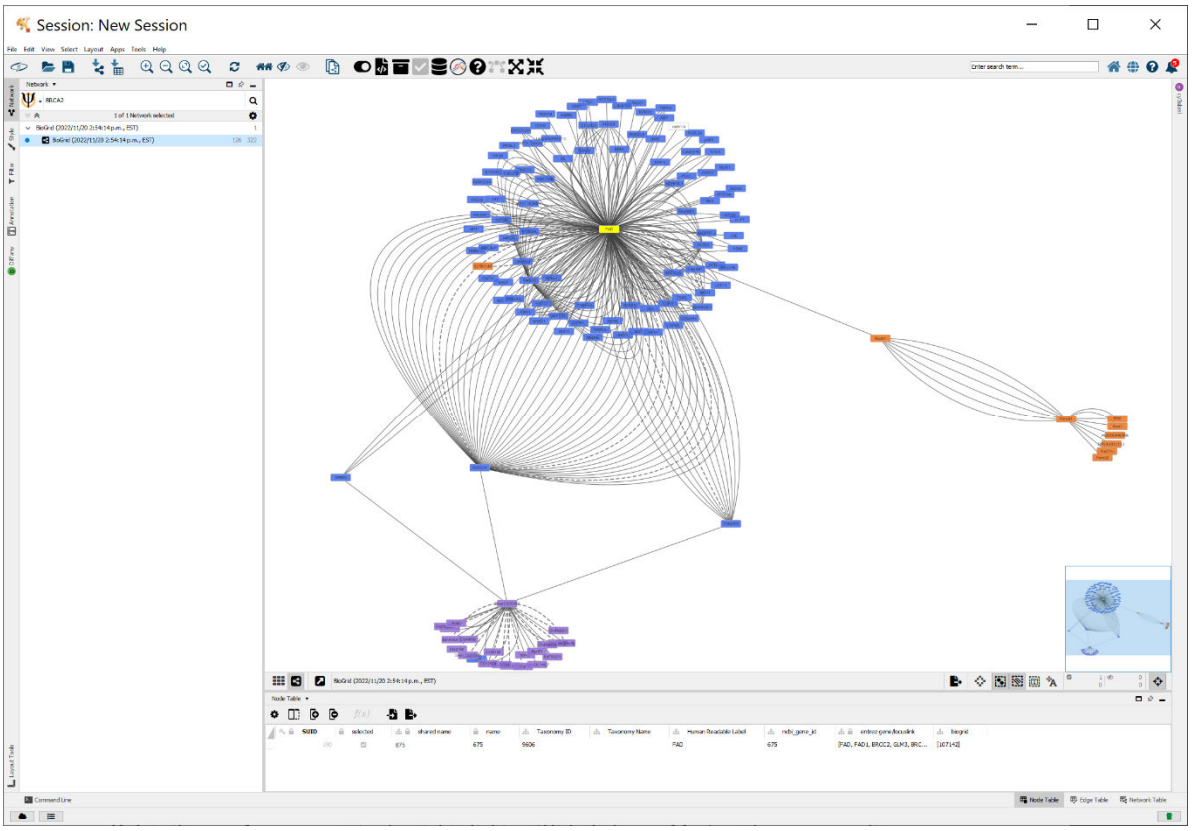


Figure 2: BRCA2 protein-protein interaction network, retrieved from the BioGRID Web Service Client. BRCA2 (here labeled by an alias, FAD) has been selected by clicking on it (yellow node). The Layout was set using the Edge-weighted Spring Embedded layout (a few clusters were moved manually), and the information for the BRCA2 node retrieved by the web service call is shown in the Table Panel below the network diagram. The Show Columns button was used to display all information about the selected node. Use the Style tab indicated with an arrow to change to the "PSIMI 25" style to be able to see nodes coloured by organism (Taxonomy ID).

How many non-human interactors did we retrieve from BioGRID for BRCA2 and what organism(s) are they from? (Hint: the default colour scheme for the nodes for the "PSIMI 25" style is by NCBI's Taxonomy ID; you can find out the corresponding organism by going to http://www.ncbi.nlm.nih.gov/taxonomy/ and entering the Taxonomy ID).

Lab Quiz Question 2

Delete the **non-human** proteins by right clicking (Ctrl-click on a Mac) on the corresponding nodes and choosing *Edit > Cut*.

2. Let's use a powerful feature of Cytoscape to colour the nodes according to their Gene Ontology (GO) categories, the VizMapper/Style feature. First, we'll need to retrieve the GO terms for the proteins in the network and we'll do this by using BioMart at http://useast.ensembl.org/biomart/martview/ (or equivalent mirror), which is a machine-readable repository for many attributes associated with various bits of data stored at the EBI, the European counterpart to the NCBI. We've done a BioMart query using the Entrez Gene IDs (in the shared name column) for the proteins in our network and saved the results as a tab-delimited file on the Coursera website:

- Download the "Coursera_DataViz_Lab05_martquery_tabbed.txt" file from the Coursera Hands-On Lab section for this lab.
- In the Cytoscape main menu, do *File* > *Import* > *Table from File*... (choose the file you just downloaded)
- In the Preview part, scroll across to the "shared name" column and click on the small leftward pointing arrow ‹ to show import options (see **Figure 3**). Select the "**key**" icon in the "Meaning" section, and "**ab**" icon denoting a string (as opposed to numeric) in the "Data Type" section. We're telling Cytoscape to match up, i.e. to key, this column with the shared name column that it already has loaded. Again, refer to **Figure 3**.
- Click OK in the main import panel – if you activate the Show All Columns button you'll see we now have some extra GO columns in the Table Panel (you may need to scroll to the right to see them).

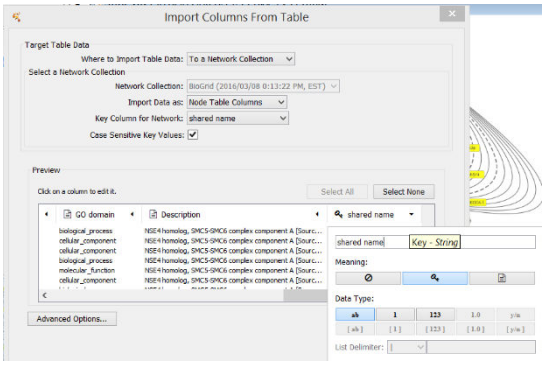


Figure 3: Importing Columns from Table. Here we're importing GO terms from a BioMart query saved as a text file in tab-delimited format and are specifying the the "shared name" column in the file we're importing should be used as the "key" and that it's a character string.

Now let's colour the nodes according to their Gene Ontology description. Click on the Style tab in the Control Panel (see arrow in Figure 2). Open the "Fill Color" to select options (click on the ‹ icon or double click on the "Fill Color" term), then select GO Term Definition as the Column attribute to use for the colouring. You will see all of the GO Term Definitions listed. The Mapping Type should be set to Discrete Mapping. Click on Fill Color again, right-click to select Mapping Value Generators then select Rainbow. Voilà! See **Figure 4**.

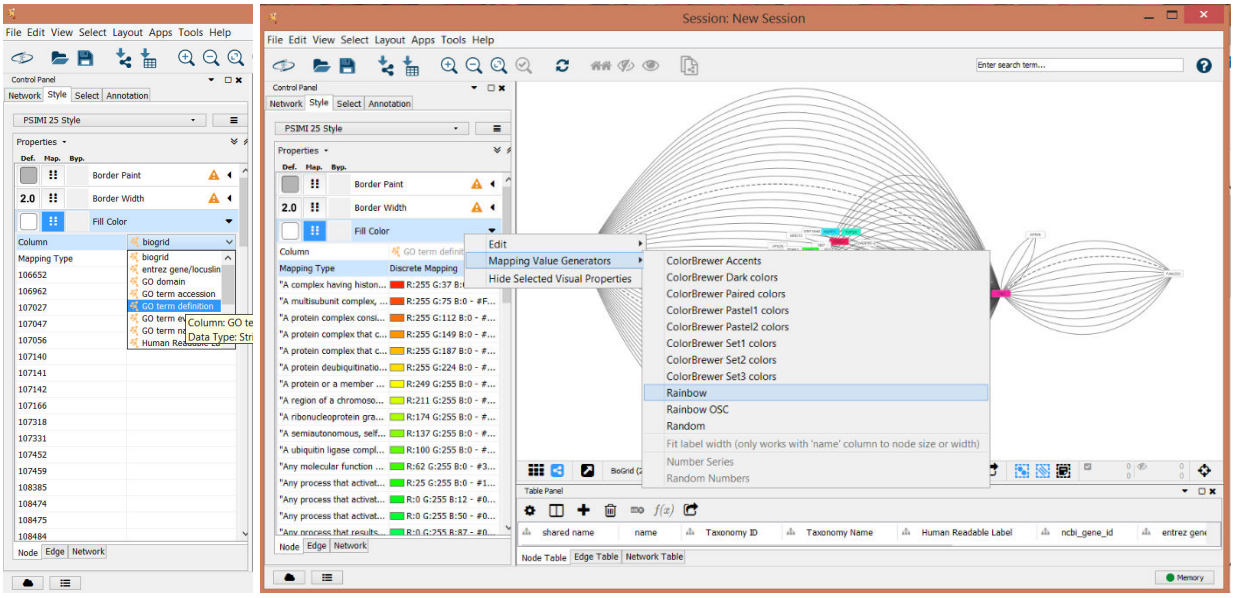
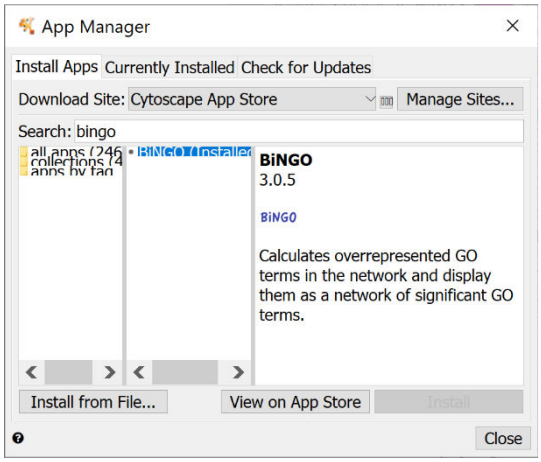


Figure 4: Using Fill Style to colour nodes according to discretized values (e.g. GO Definition).

Now the node colours correspond to the GO Term Definition categories, insofar as they are present. What are the general gene ontology definitions for proteins that interact with BRCA2?

You can imagine that it would be possible to import a tab-delimited table keyed by gene identifiers containing information on any aspect of the biology of your study system, and to change fill colours (or any other node attributes) to help draw attention to certain kinds of proteins etc.

Although you can qualitatively see that there are a lot of BRCA2-interacting proteins that are in the GO Term Description categories similar to "DNA damage response", it would be nice to know if there were any kind of over-representation relative to all genes or their protein products in the genome. Fortunately, there are some Apps that will tell us this exact thing! Go to *Apps* > *App Manager* and find BiNGO (either by name or under the Ontology Analysis tag). Click BiNGO and then Install it. It will take a few minutes to download and install. When the download has completed, close the App Manager window (see small image to the right).



3. Select all of the nodes in the network by doing *Select* > *Nodes* > *All Nodes* (or just hold the left mouse button while dragging the box to highlight all the nodes). Next, activate the BiNGO app by clicking on *BiNGO* under *Apps*. Name the Cluster, be sure to select Homo sapiens as the Organism/Annotation. Unselect the "Save BiNGO Data File option. Start BiNGO!

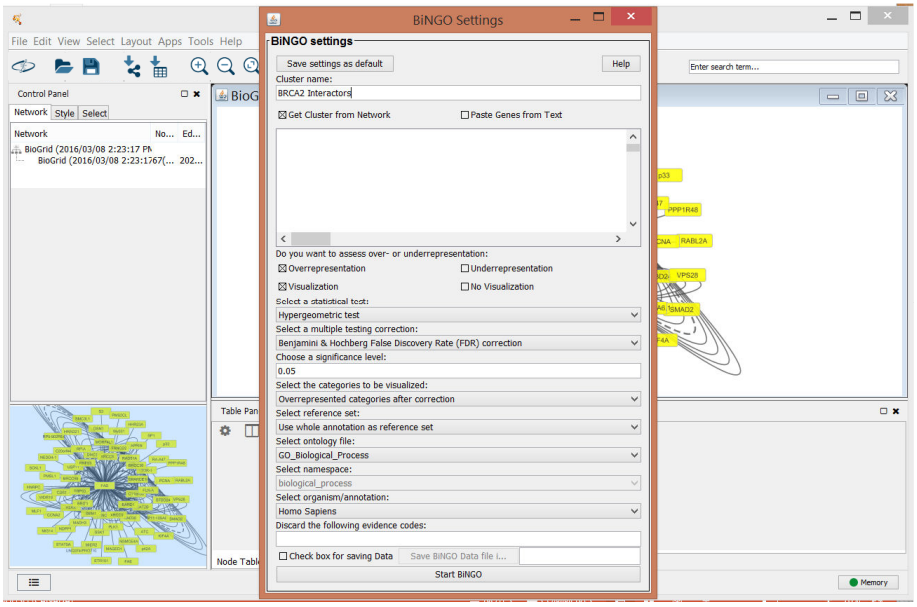


Figure 5: BiNGO Gene Ontology enrichment analysis with BRCA2-interacting proteins from BioGRID. All nodes are selected then it is possible to run a Gene Ontology enrichment analysis for GO Biological Process terms with the BiNGO app using "Homo sapiens" as the organism/annotation. You may need to adjust the "Scale and Layout" in your systems settings in order to be able to see the entire BiNGO dialog box. Alternatively, with the cursor in the "Cluster name" box, hold the Shift key and hit the tab key 4 times to move to the "Start BiNGO" button (which may not be visible). Hit "Enter" to start BiNGO.

After a couple of minutes a table of GO BP terms appears along with their p-values for over-representation, along with a network which represents the GO term graph. The graph represents the underlying directed acyclic graph structure of GO that we talked about last week. The nodes are the parent and child terms, while the node colour represents the level to which that term is enriched, with red denoting more enrichment…this is yet another way to visualize GO term enrichment! Here we'll focus, however, on the table (**Figure 6**). The smaller the p-value, the more significantly enriched is the GO BP category.

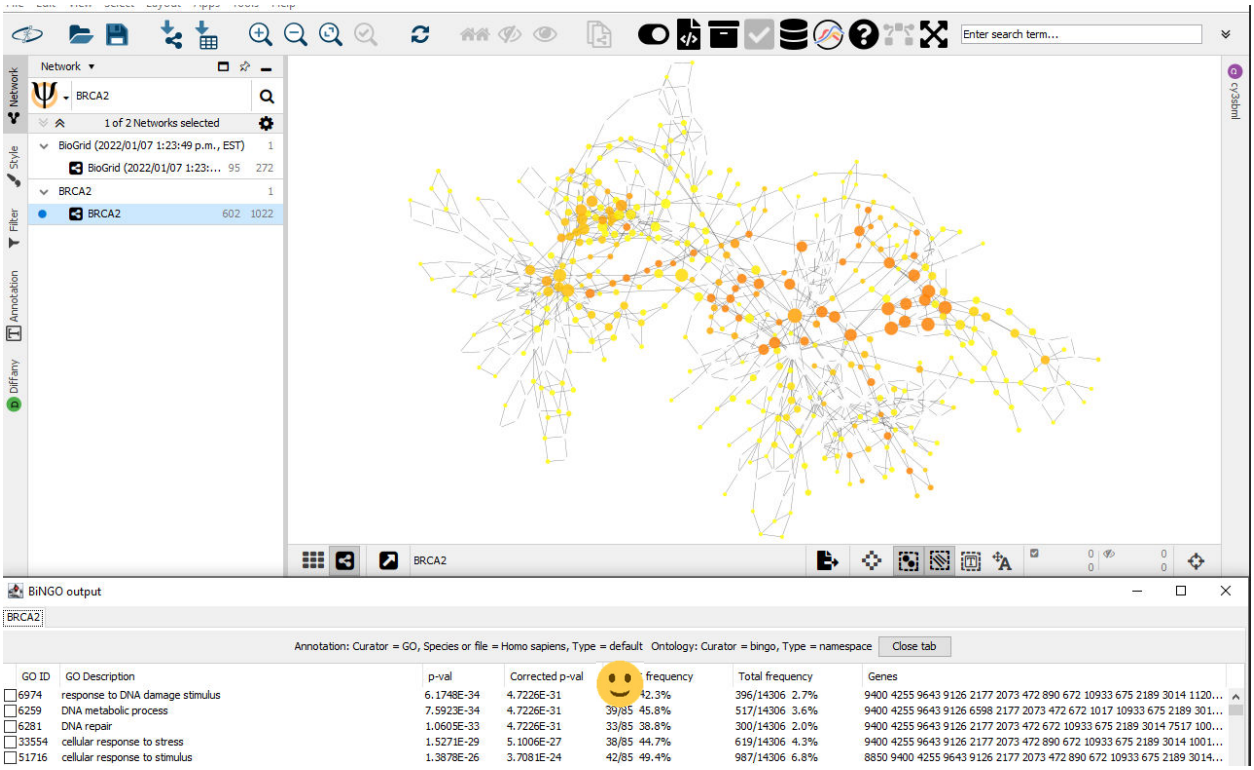


Figure 6: BRCA2-interacting proteins' BiNGO results. Highlight genes to be able to copy them.

Which GO Biological Process category is most over-represented in our network, relative to the GO terms for all of the genes (proteins) in human? (It's the first item in the list). What is the p-value for enrichment?

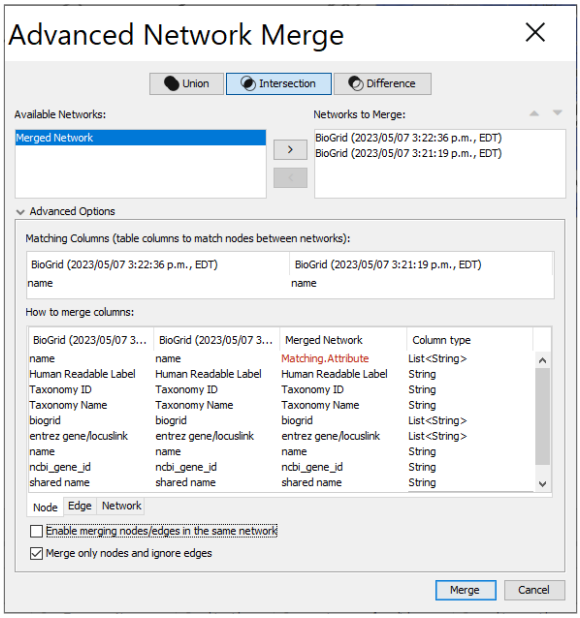
Lab Quiz Question 2

Exploring Cytoscape Layouts

Cytoscape offers many layout algorithms, which can help understand the often times large numbers of interactions in a network. When you use these in combination with built-in filtering options, it is easy to create subgraphs of the network, focusing in on say proteins with 2 or more interactions.

We'll supplement the BRCA2 interactor network with interactions *between* BRCA2's interactors. We can do this by loading all the interactions for the ~94 nodes (as defined by the Shared Name) using the same approach as in Step 1, except we'd paste in all node identifiers.

We could then use the Tools > Merge > Networks, choosing the original BRCA2 network and the new network with all the interactors of the BRCA2 interactors (with around 22,000 edges!). If we choose the Intersection option, and the "Merge only nodes and ignore edges" option, we'll generate a network of around 94 nodes and 1,503 edges. Alternately, download the Cytoscape session file called **BRCA2_and_interators_recursive.cys** and open it to view this network. With the provided layout, we can see that BRCA2/FAD has a lot of interactors, but that some of the other interactors of BRCA2 are also quite highly connected. Which ones have higher node degrees? We'll need some network statistics to be able to assess this. Before we start, remove duplicated edges by doing *Edit* > *Remove Duplicated Edges* and > *Remove Self Loops*. This leaves around 386 edges.



4. First do *Tools* > *Analyze Network* to generate network statistics that we can filter on. Treat the network as undirected, and combine pair edges if prompted. The kinds of network statistics that were generated include node degree, the number of edges emanating from each node. If we want to focus on proteins with more than 2 interactions, we could click on the Filter tab in the Control Panel. Click on the + symbol to add a new filter, and choose the Degree option. Set the values such that only the nodes with 1 edge are highlighted ("between 1 and 1 inclusive" – simply enter 1 in both boxes). Click Apply, then do *Edit* > *Delete Selected Nodes and Edges*. This simplifies our network considerably! *With the node degree information now available, can you figure out how to use the Style options to size the nodes according to their node degree?*

Lab Quiz Question 3

5. Next, explore the different layout options available in Cytoscape – see options here: https://manual.cytoscape.org/en/stable/Navigation_and_Layout.html#automatic-layout-algorithms. Four layouts are presented below in **Figure 7**. The hierarchical layout algorithm is good for representing the main flow of information within a network, while the circular layout emphasizes group and tree structures within a network. The grid layout's advantage is that it is very fast, and some of force-directed or spring-embedded layouts provide visually pleasing layouts.

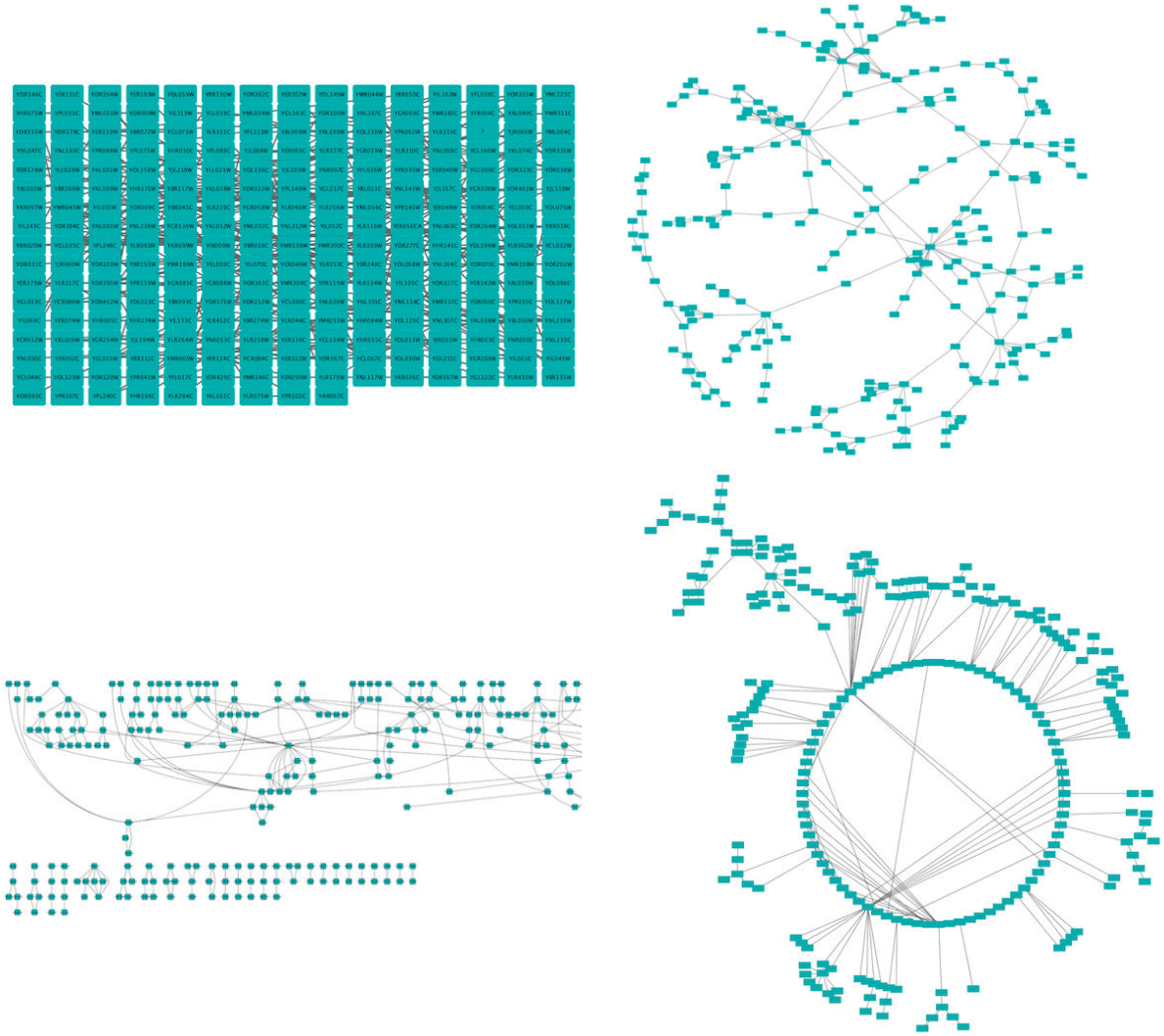


Figure 7. Different Cytoscape network layout options. Clockwise from top left: grid layout, edge-weighted spring-embedded layout, circle layout, and hierarchical layout (images from Cytoscape.org website).

D3

Let's move on to another method for creating network graphs, as well as virtually any other type of chart you can think of, all in an interactive framework, using D3.

D3.js is one of the most popular tools for making data visualizations that can be displayed in a web browser. It is a Javascript library that provides functions for loading data, manipulating them, and drawing SVG elements to the screen that don't already come with vanilla Javascript. There is a steep learning curve since you need to use it in combination with HTML and CSS, but it is extremely powerful and the best tool for drawing highly customized and interactive charts.

One of easiest ways to get started with D3 is with the free online notebook platform, Observable. It is similar to Jupyter, however it allows you to build interactive data visualizations with Javascript and D3 instead of R and ggPlot.

1. Create an account with https://observablehq.com/ (you can log in with your Google account)

2. Read the first steps Observable start guide, it's helpful to learn the basics of the interface: https://observablehq.com/guide/first-steps.

3. Let's create a force directed graph (similar to the 2nd layout in Figure 7) with some of our own data. Create a new notebook, starting from scratch with a blank template:

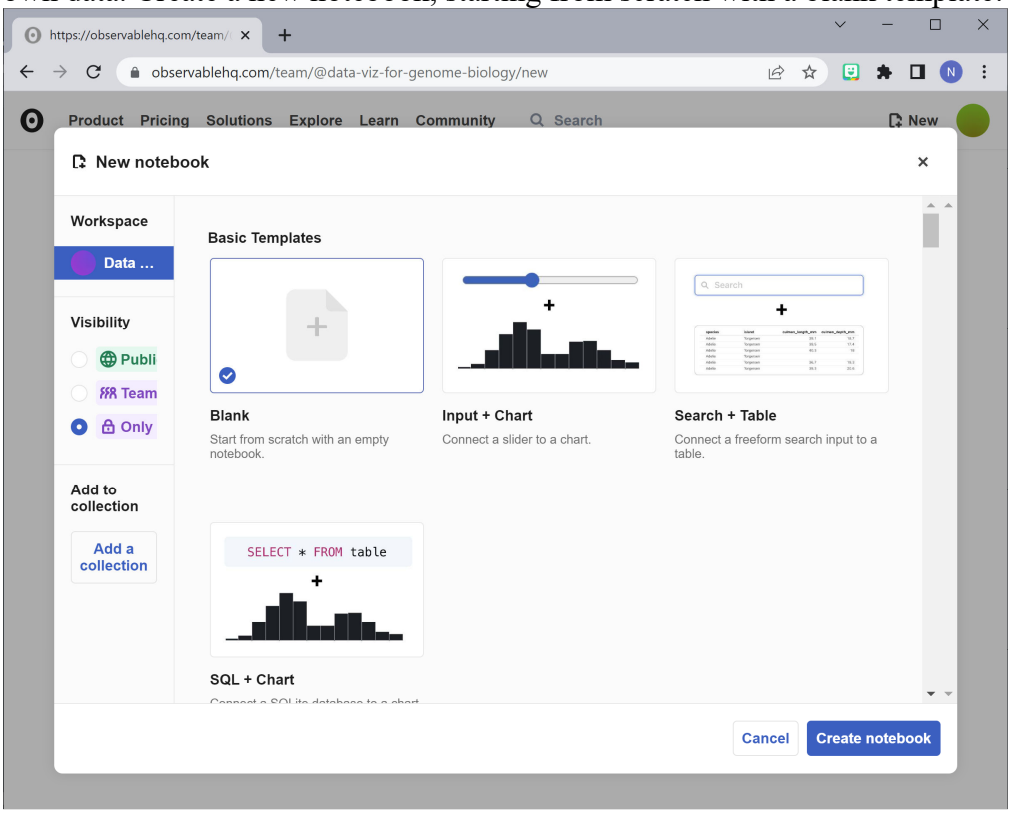


Figure 8. Creating a blank Observable notebook.

4. Upload the "networks.json" file by dragging it onto the Files section of the screen, or by clicking the paper clip icon and selecting it with the file browser. Click the blue Insert into Notebook icon so we can begin working with it (see arrow in **Figure 9**). This is the default network that is provided for the default ForceGraph function, consisting of the interactions between characters in Victor Hugo's 1862 opus *Les Miserables*. Also upload a biological network consisting of the interactions between the direct interactors of BRCA2 that we generated in the Layouts part of the lab, "network_BRCA2.json".

   As an aside, to generate the networks_BRCA2.json data structure, we downloaded the node and edge tables from the Cytoscape part of the lab as .csv files, and used VLOOKUP and CONCATENATE functions to build the nodes and links data objects in the networks_BRCA2.json file in a text editor. Subcellular localization for the nodes was retrieved using BioMART, and edited to be Group = 1 for nucleus, 0 otherwise.

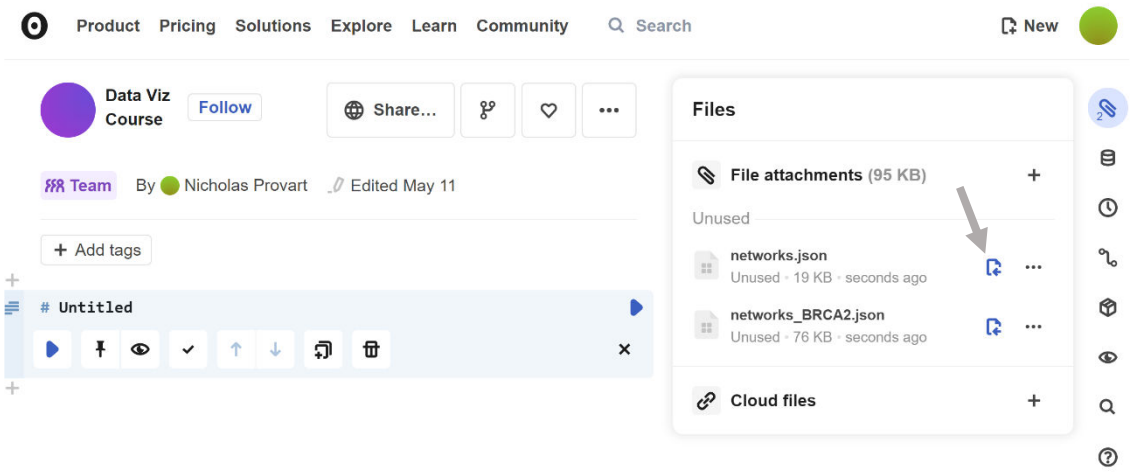
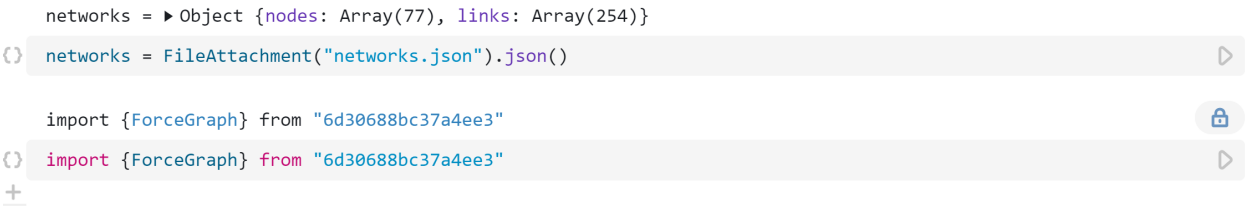


Figure 9: Inserting an attached file into an Observable notebook. Click the blue icon indicated by an arrow to do so.

5. Click the + to add a new cell, choose {} Javascript { }, and enter the following code into the code block: `import {ForceGraph} from "6d30688bc37a4ee3"` . This will import a slightly modified version of the D3 ForceGraph code that we've modified to allow for nodes to be sized according to degree, and for labels to be appended. Click the blue ▶ arrow to import the modified code. Once the block has run, the arrow will become just an outline:



6. Click the + to add another new cell and enter the following code (you can copy-paste from this document). Click the blue ▶ button in the top right of the code block to run it.

```
chart = ForceGraph(networks, {
  nodeId: d => d.id,
  nodeGroup: d => d.group,
  nodeTitle: d => `${d.id}`,
  linkStrokeWidth: l => Math.sqrt(l.value),
  nodeRadius: d => Math.sqrt(d.degree)*3,
  width,
  height: 600,
  nodeStrength: -50,
  invalidation // a promise to stop the simulation when the cell is re-run

})
```


7. The default network doesn't actually have any node degree information. *Examine the networks.json and networks_BRCA.json files in a text editor to confirm this*.

8. Let's add a slider to allow us to adjust the forces repelling the nodes from one another to get a good layout. What's the best setting? Click the + button and type "slider" and select the Range slider option.

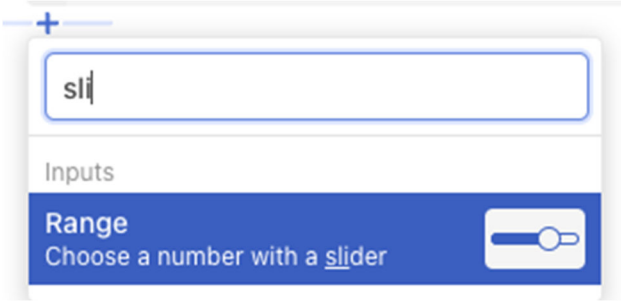


Figure 10. Adding an interactive slider option.

9. Adjust the range to be between –100 and 0, and the step to be 1 by editing the code block (see **Figure 11**).

10. Next, in the ForceGraph options, instead of hard coding nodeStrength as –50, tell Observable to get the value from the range slider by setting `nodeStrength: range`

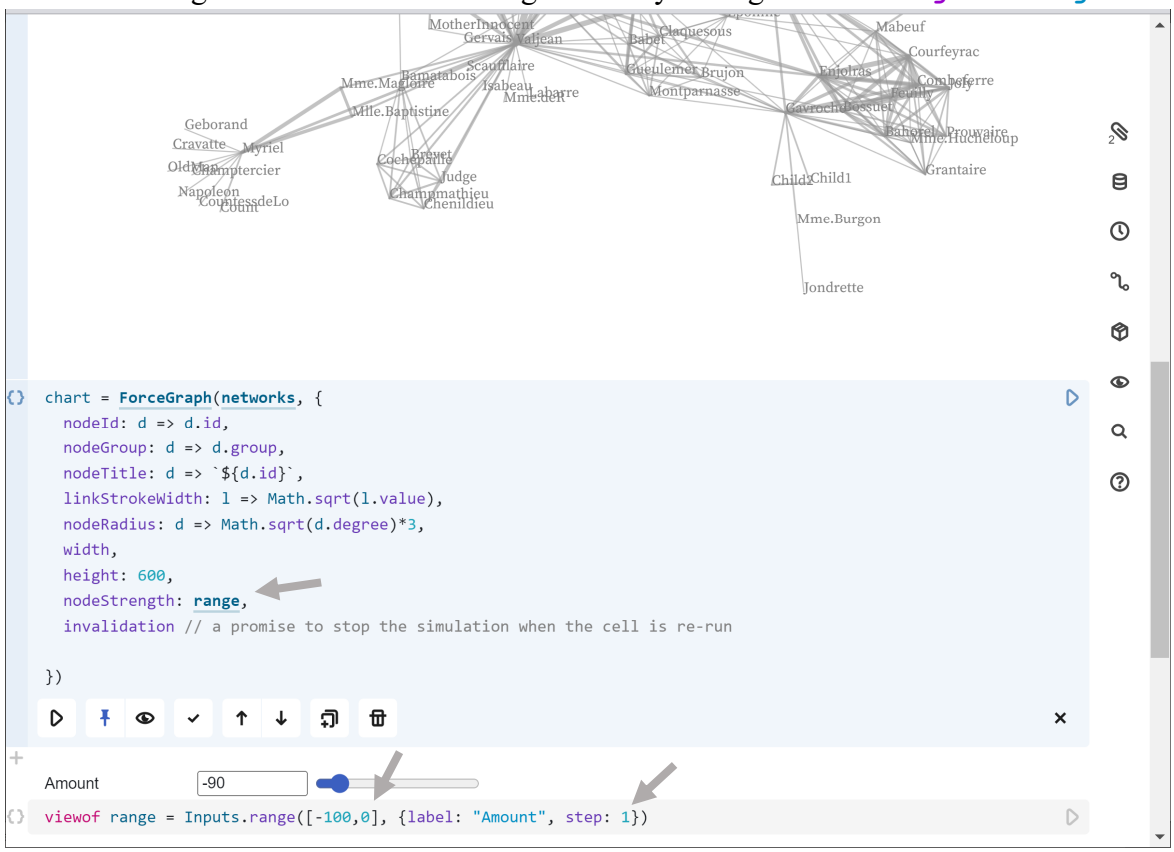


Figure 11. Adjusting the Range slider options, and adding the `range` variable for NodeStrength.

11. Try adjusting the size of the graph to width: 1024, height: 1024.

12. Last, let's load our networks_BRCA2.json data set by clicking on the 2nd data set's "insert into notebook" icon as in Step 4. Edit the data object name to be used in the ForceGraph function (see **Figure 12**) to `networks_brca2`. Rerun the code by clicking the blue arrow. You may need to adjust the slider to a more negative value to get better node spacing. The nodes are coloured orange if the gene product is in the nucleus, and blue otherwise. *Do interactors of BRCA2/FAD appear to be found in the nucleus? How could we modify the data file to include other node attributes like node stress centrality (a measure of the number of shortest paths passing through a node)?*

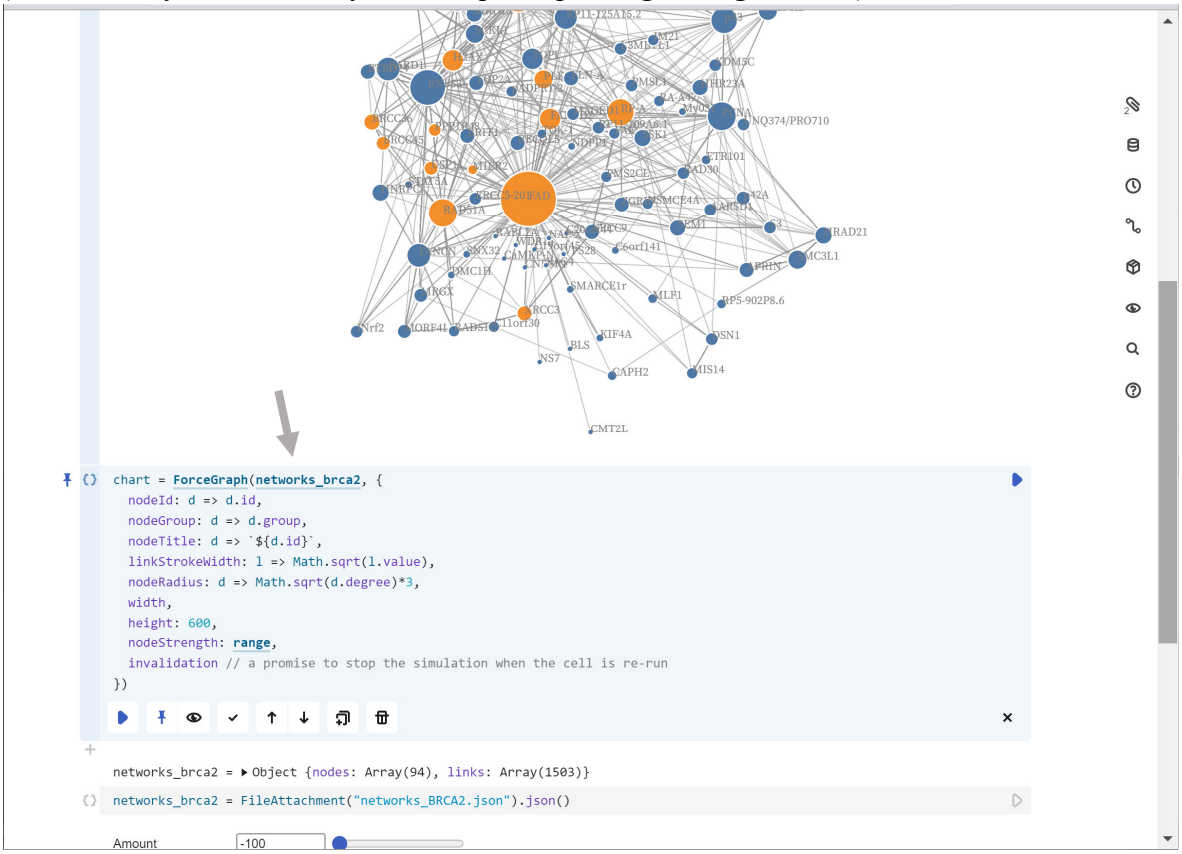


Figure 12. Modifying the code to use the networks_brca2 object we inserted into our notebook.

13. Obervable code can be embedded into a regular webpage in two ways, see https://towardsdatascience.com/how-to-insert-an-observablehq-graph-into-a-html-page-57a9f4546ecf. A Javascript-based alternative for viewing networks that has a lot of built-in functionality is Cytoscape.js…check out some fun uses at https://js.cytoscape.org/.

14. For one last wow, check out this Observable notebook, showing what you can do with it! https://observablehq.com/@mootari/force-graph-trails

In this lab, we've seen that protein-protein interactions (PPIs) can be easily visualized with the powerful desktop app, Cytoscape. Cytoscape allows virtually unlimited possibilities for the representation of a network in terms of layout options, node appearance, etc. But, at least with the web service we used, the ability to identify how those interactions were determined and to be

able to access the primary literature concerning them is limited, so it sometimes makes sense to return to the parent database to understand how the interactions were determined. Of course, if you have generated your own PPI data, then you'll already know how you determined the interactions.

End of Lab!

Where to get it:

Download the Cytoscape executable from http://www.cytoscape.org/download.html. Use the Platform-Specific Installers to install Cytoscape 3.9.1. You will need to have the appropriate Java Runtime Environment installed (usually 64-bit) first, which you can get from http://www.java.com. During the set-up/start-up of Cytoscape, permit access with private networks. *Note: this lab has been tested and works with Cytoscape 3.9.1 on Windows, Mac, and Linux machines (version 3.6.0 and higher will give similar outputs).*

Lab 5 Objectives

By the end of Lab 5 (comprising the labs including their boxes, and the lectures), you should:

- understand why protein-protein interactions are important biologically, and also how they may be determined experimentally;
- be able to assess the advantages and disadvantages of the methods for determining protein-protein interactions;
- know the terminology associated with protein-protein interaction graphs;
- be able to use Cytoscape to identify interacting proteins for your gene product of interest and to filter and decorate networks based on additional information;
- be aware of different Cytoscape layouts and what kind of information each is best suited to displaying;
- be able to identify the type of support for a given interaction in a given database;
- be able to interpret the other types of information (GO categories) provided by the software tools.

Do not hesitate to use the Coursera discussion forums if you do not understand any of the above after reading the relevant material.

Further Reading

Cline MS, Smoot M, Cerami E, Kuchinsky A, Landys N, Workman C, Christmas R, Avila-Campilo I, Creech M, Gross B, Hanspers K, Isserlin R, Kelley R, Killcoyne S, Lotia S, Maere S, Morris J, Ono K, Pavlovic V, Pico AR, Vailaya A, Wang PL, Adler A, Conklin BR, Hood L, Kuiper M, Sander C, Schmulevich I, Schwikowski B, Warner GJ, Ideker T, Bader GD (2007). Integration of biological networks and gene expression data using Cytoscape. Nat Protoc. 2(10):2366-82.

Kangueane P and Nilofer C (2018). Databases for Protein-Protein Interaction. In: Protein-Protein and Domain-Domain Interactions. Springer, Singapore. https://doi.org/10.1007/978-981-10-7347-2_9.

Stark C, Breitkreutz BJ, Reguly T, Boucher L, Breitkreutz A, Tyers M (2006). BioGRID: A General Repository for Interaction Datasets. Nucleic Acids Res. 34:D535-9.

