**Basic expression analysis in Network**

In this project, we will explore the basic structures for applying expression data to a Cytoscape network. This tutorial will introduce you to:

1) Visualizing networks using expression data.

2) Filtering networks based on expression data.

3) Assessing expression data in the context of a biological network.

**I. Loading Network**

* Start Cytoscape and load the network by selecting “File > import > network > file >galFiltered.sif”
* Apply the force-directed layout to organize the layout of the nodes. Select the “Layout->Prefuse ForceDirected Layout" menu.

**II. Loading expression data**

Using your favorite text editor or Excel, open the file galExpData.csv. You should note the following information about the file:

* 1. The first line consists of labels.
	2. All columns are separated by a single comma character.
	3. The first column contains node names, and must match the names of the nodes in your network exactly!
	4. The second column contains common locus names. This column is optional, and the data is not currently used by Cytoscape, but including this column makes the format consistent with the output of many microarray analysis packages, and makes the file easier to read.
	5. The remaining columns contain experimental data, two columns per experiment (one column represents the expression measurement and the second represents the significance value for that measurement), and one line per node. In this case, there are three expression results per node.

Under the File menu, select “Import > table > file >galExpData.csv”. Now we can use the *Table Panel* to browse through the expression data.

**III. Visualizing Expression Data**

Probably the most common use of expression data in Cytoscape is to set the visual attributes of the nodes in a network according to expression data. This creates a powerful visualization, portraying functional relation and experimental response at the same time. Here, we will walk through the steps for doing this.



1. Click “style > options > create new style” in the control panel and name it as “Gal80”.
2. Label the nodes. Click triangle sign in the label line. Select Column = common and Mapping type = Passthrough mapping. Room in on the network so that node labels are visible.
3. Color the nodes. Click triangle sign in the Fill color line. Select Column = gal80Rexp and Mapping type = Continuous mapping. This action will produce a basic black to white color gradient. Click on the color gradient to change the colors. 
4. Note that the default node color of red falls within this spectrum. A useful trick is to choose a color outside this spectrum to distinguish nodes with no defined expression value and those with significantly over-expression. Click the square sign in the left side of *fill color* line and choose a dark gray color.
5. Set the node shape. We imported both expression measurement values and significance values for those measurements. We can use the significance values to change the shape of the nodes so that measurements we have confidence in appear as squares while potentially bad measurements appear as circles. Click triangle sign in the Shape line. Select Column = gal80Rsig and Mapping type = Continuous mapping. Click on the spectrum to change the shapes. 

QUESTION: Can you use different line type to mark different interactions (i.e. pp & pd)?

**IV. A biological analysis scenario**

This section presents one scenario of how expression data can be combined with network data to tell a biological story. Here is some background on your data. You are working with yeast, and the genes Gal1, Gal4, and Gal80 are all yeast transcription factors. Your expression experiments all involve some perturbation of these transcription factor genes. Gal1, Gal4, and Gal80 are also represented in your interaction network, where they are labeled according to yeast locus tags: Gal1 corresponds to YBR020W, Gal4 to YPL248C, and Gal80 to YML051W.

Select the neighborhood of GAL4.

1. Enter GAL4 gene into the selection box and hit return.
2. Select > Nodes > First neighbors of selected node (this gets the immediate neighbors of GAL4).
3. Hit ‘New Network from Selection’ button. 
4. In the new sub-network, apply a graph layout algorithm using the yFiles Organic layout.



Our data show precisely this:

1. Both nodes (GAL4 and GAL11) show fairly small changes in expression, and neither change is statistically significant: they are rendered as light-colored circles. GAL11 is a general transcription co-factor with many interactions.
2. GAL4 interacts with GAL80 (YML051W), which shows a significant level of repression: it is depicted as a green square.
3. Note that while GAL80 shows evidence of significant repression, most nodes interacting with GAL4 show significant levels of induction: they are rendered as pink squares.

Putting all of this together, we see that the transcriptional activation activity of Gal4 is repressed by Gal80. So, repression of Gal80 increases the transcriptional activation activity of Gal4. Even though the expression of Gal4 itself did not change much, the Gal4 transcripts were much more likely to be active transcription factors when Gal80 was repressed. This explains why there is so much up-regulation in the vicinity of Gal4.

Go to the NCBI website (http://www.ncbi.nlm.nih.gov/), and search the Gene database for “YPL248C” (another name for the GAL4 gene). The items returned should include Gal4. Click on the link for Gal4 to get more information.

QUESTION: How do they describe the relation between GAL4 and GAL80?