Methods for network visualization and gene enrichment analysis
February 22, 2012

Jeremy Miller
Postdoctoral Fellow - Computational Data Analysis
Outline

• Visualizing networks using R

• Visualizing networks using outside programs
  – Using VisANT to graph modules

• Gene enrichment analyses using R

• Gene enrichment analyses using outside programs
Visualizing networks using R

• First, run WGCNA and assign modules
  – This process involves creating a dendrogram
  – A dendrogram shows the topology of a network but doesn’t directly show gene expression relationships or module correlations.
• But what do these modules represent?
• Which modules are distinct? Do some have similar patterns?
Visualizing module relationships

- Calculate the module eigengenes

```r
ME = moduleEigengenes(DATA, colors=MODULES)$eigengenes
```

- Many of the visualization and enrichment strategies require this value.
- Think of this as a representative value for each module.

- Module eigengenes can be visualized in dendrograms just like genes:

```r
distance = 1-(1+cor(ME_1A, use="p")(2)
cluster = hclust(as.dist(distance), method="a")
plot(cluster, [parameters])
```

Modules in the same branch contain genes with relatively similar expression patterns (*but note that the genes within a module have higher co-expression than genes between similar modules*).
Visualizing module relationships

- A multidimensional scaling plot can show similar information about module relationships in two dimensions
  - This plots the first two principal components of the distance matrix

```r
MDS = cmdscale(as.dist(distance), 2)
plot(MDS, col=MODULES, [parameters])
```

Modules that group together on this plot tend to contain genes with similar expression patterns.

For example, in a WGCNA study of Alzheimer’s disease, we found four main groups of modules, most of which could be distinguished from one another based on gene expression, enrichment analyses, etc.
Visualizing module relationships

• The module eigengenes can be directly plotted using graphs
  – In this case, each bar is a sample.
  – If you first order these samples in a biologically-meaningful way, you can learn a lot about a module just by looking at the eigengene!

\[ \text{barplot}(\text{ME}_\text{moduleX}, [\text{parameters}]) \]

With some minor adjustments, we can find modules related to brain subregions just by looking!
Other network visualizations using R

There are several other types of network visualizations which I will not discuss in detail here.

**Heat maps**: there are many standard ways of making these plots.

**Scatter plots**: these are particularly useful for between-study analyses.

**Box plots**: these are useful for displaying differential expression.
Visualizing networks using outside programs

There are many useful programs outside R for visualizing networks:

• Programs with both enrichment and visualization components
  – ChiliBot, Ingenuity, STRING, etc.
  – (These will be discussed with the enrichment analysis section.)

• Cytoscape
  – For plotting modules (like in VisANT)
  – See “exportNetworkToCytoscape” function in the WGCNA library.

• VisANT
  – This is available for both PC and Mac, but I find that the PC version works much better (particularly for reading in the data).
Visualizing networks using outside programs - VisANT

A step by step tutorial for how to use VisANT (with screen shots) is available for anyone who is interested!

In short:

• Download and install VisANT (http://visant.bu.edu/) and Java
• Create a file with your interactions in the appropriate format
• Read the interaction data into VisANT
• Format your interaction map in VisANT as desired
Visualizing networks using outside programs - VisANT

There are three ways of making the input file for VisANT:

1. On your own. It must be in the proper format.
2. Using “exportNetworkToVisANT” in the WGCNA library
3. Using “visantPrepOverall” (which goes along with the meta-analysis discussed later today).

From there you just copy and paste the interactions you want to show directly into VisANT.

<table>
<thead>
<tr>
<th>gene1</th>
<th>gene2</th>
<th>zero</th>
<th>color</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREB3</td>
<td>LSP5</td>
<td>0</td>
<td>M1003</td>
<td>0.452197</td>
</tr>
<tr>
<td>USP5</td>
<td>CREB3</td>
<td>0</td>
<td>M1003</td>
<td>0.452197</td>
</tr>
<tr>
<td>CREB3</td>
<td>TESSP2</td>
<td>0</td>
<td>M1003</td>
<td>0.438122</td>
</tr>
<tr>
<td>TESSP2</td>
<td>CREB3</td>
<td>0</td>
<td>M1003</td>
<td>0.438122</td>
</tr>
<tr>
<td>CREB3</td>
<td>PSMD13</td>
<td>0</td>
<td>M1003</td>
<td>0.436564</td>
</tr>
<tr>
<td>PSMD13</td>
<td>CREB3</td>
<td>0</td>
<td>M1003</td>
<td>0.436564</td>
</tr>
<tr>
<td>CREB3</td>
<td>LRRC1</td>
<td>0</td>
<td>M1003</td>
<td>0.428405</td>
</tr>
<tr>
<td>LRRC1</td>
<td>CREB3</td>
<td>0</td>
<td>M1003</td>
<td>0.428405</td>
</tr>
</tbody>
</table>

Character vector representing the edge (M1003 = orange). (Note that node color is set within VisANT itself.)

A numeric value for sorting interactions. In this case, topological overlap is used (not strictly necessary, although a number between 0 and 1 must be here).
Visualizing networks using outside programs - VisANT

The best way to learn how to use VisANT is just to try it!

Some helpful hints:

(2) Turn off “fine arts”
(3) After highlighting the nodes of interest, changes the color and size by clicking “Nodes” ➔ “Properties”
(4) Choose one of the “relaxing” options to make the nodes group in a hub-and-spoke manner. After this, you will have to move nodes manually…
(5) This will allow you to display only certain connections. Use this option LAST, if you use it at all.
(6) Finish up by saving your file AND by saving your image (SVG file will give the highest-quality image).

(1) FIRST, copy data here and click “Add”
Gene enrichment analyses using R

There are two basic methods for gene enrichment analysis in R:

1. Enrichment for published or user-defined lists
   - **userListEnrichment** in the WGCNA library
   - This function performs hypergeometric tests for all of your modules against any user-defined lists.
   - It also includes pre-loaded lists from brain, blood, and stem cell data sets.
     - Cell type markers from many publications
     - Genes from modules found in several WGCNA analyses
     - Known and predicted lists of disease genes
     - Lists of genes enriched in particular brain areas
     - Immune-related gene lists

2. Gene Ontology Enrichment
   - **GOenrichmentAnalysis** in the WGCNA library
   - **enrichGO** in the clusterProfiler library
   - *In my experience the results from DAVID/EASE are better*
Gene enrichment analyses using outside programs

There are several outside programs available for annotating modules.

I will only be discussing a small subset of these programs:

- EASE:  [http://david.abcc.ncifcrf.gov/ease/ease1.htm](http://david.abcc.ncifcrf.gov/ease/ease1.htm)
- WebGestalt:  [http://bioinfo.vanderbilt.edu/webgestalt/](http://bioinfo.vanderbilt.edu/webgestalt/)
- GSEA:  [http://www.broadinstitute.org/gsea/index.jsp](http://www.broadinstitute.org/gsea/index.jsp)
- UGET:  [http://genome.ucla.edu/projects/UGET](http://genome.ucla.edu/projects/UGET)
- STRING:  [http://string.embl.de/](http://string.embl.de/)
EASE – A GO (etc.) enrichment analysis tool

EASE (Enrichment Analysis Systematic Explorer) is a standalone version of DAVID that can be used to find enrichment of GO, KEGG, etc. in a list, given both the test list and the reference list.

Typical output from EASE:

<table>
<thead>
<tr>
<th>System</th>
<th>Gene Category</th>
<th>List Hits</th>
<th>List Total</th>
<th>Population Hits</th>
<th>Population Total</th>
<th>EASE score</th>
<th>Bonferroni</th>
<th>Gene identifiers</th>
<th>Gene Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>GO Biological Process</td>
<td>synaptic transmission</td>
<td>19</td>
<td>140</td>
<td>225</td>
<td>8539</td>
<td>2.09E-08</td>
<td>2.78E-05</td>
<td>348; 1742; 255; APOE; DLG4; DLG6</td>
<td></td>
</tr>
<tr>
<td>GO Biological Process</td>
<td>transmission of nerve impulse</td>
<td>19</td>
<td>140</td>
<td>231</td>
<td>8539</td>
<td>3.15E-08</td>
<td>4.19E-05</td>
<td>348; 1742; 255; APOE; DLG4; DLG6</td>
<td></td>
</tr>
<tr>
<td>GenMAPP pathway</td>
<td>Hs_Peptide GPCRs</td>
<td>8</td>
<td>19</td>
<td>52</td>
<td>1144</td>
<td>5.78E-06</td>
<td>7.70E-03</td>
<td>2587; 2925; 481; ATP8A1; GALR1; G1</td>
<td></td>
</tr>
<tr>
<td>GO Biological Process</td>
<td>cell-cell signaling</td>
<td>23</td>
<td>140</td>
<td>483</td>
<td>8539</td>
<td>9.80E-06</td>
<td>1.30E-02</td>
<td>152; 348; 1742; ADRA2C; APOE; DLG4</td>
<td></td>
</tr>
<tr>
<td>SwissProt keyword</td>
<td>Glycoprotein</td>
<td>43</td>
<td>97</td>
<td>1700</td>
<td>6875</td>
<td>3.19E-05</td>
<td>4.24E-02</td>
<td>152; 348; 1001; ADRA2C; APOE; CD1</td>
<td></td>
</tr>
<tr>
<td>SwissProt keyword</td>
<td>Neuropeptide</td>
<td>5</td>
<td>97</td>
<td>14</td>
<td>6875</td>
<td>3.21E-05</td>
<td>4.27E-02</td>
<td>5173; 5368; 681; NMU; NPFF; PDYN</td>
<td></td>
</tr>
<tr>
<td>SwissProt keyword</td>
<td>Signal</td>
<td>39</td>
<td>97</td>
<td>1521</td>
<td>6875</td>
<td>7.78E-05</td>
<td>1.03E-01</td>
<td>348; 885; 1001; APOE; CCK; CDH3</td>
<td></td>
</tr>
<tr>
<td>GO Biological Process</td>
<td>cell communication</td>
<td>62</td>
<td>140</td>
<td>2471</td>
<td>8539</td>
<td>0.0001111</td>
<td>1.48E-01</td>
<td>152; 348; 885; ADRA2C; APOE; B1</td>
<td></td>
</tr>
</tbody>
</table>
ChiliBot – A literature search tool

Chilibot will take a list of up to 50 genes, search the literature for co-occurrences of these terms, then output an interactive plot of the literature connections between these terms. For example:

If you click on the connection between two genes it will show you text where both terms are presented.

**Word of caution:** since this is a literature search, you should check the references carefully!
WebGestalt – A toolkit of enrichment analyses

WebGestalt can perform several enrichment analyses from a relatively straightforward web-based interface. The output is not as user-friendly as EASE, but the results can sometimes be more informative.
Ingenuity – A hand-curated list of interactions

Ingenuity is a comprehensive program for both annotation and visualization. It requires training and an expensive subscription to use.

An example output plot looks like this:
GSEA – A powerful method for gene enrichment

GSEA takes as input a sorted list of all genes with respect to a parameter (i.e., correlation with age, module membership, etc.), and asks whether an a priori defined set of genes is significantly enriched at one end of this distribution.

GSEA is very powerful since it uses all of the data, not just the best subset for enrichments, but the software has a high learning curve and is VERY particular about data formats.
UGET – A tool for finding other co-expressed genes

UGET isn’t an enrichment analysis method itself, but it can help find other gene correlated with your genes of interest across thousands of microarray samples in the Celsius database. This could be useful either before or after enrichment analysis, depending on your goal.

For example, it can find other genes highly correlated with ribosomal proteins, and likely involved in translational machinery:
STRING – A tool for finding networks of 1 gene

STRING takes a single gene as input and returns a list and plot of predicted functional partners based on several lines of evidence. STRING isn’t an enrichment analysis method itself, but is still very useful, particularly for following up on hub genes.
Summary

• Once you have your network, it is useful to visualize it.

• Once you have your modules, it is useful to visualize and annotate them to get a better understand of what these gene lists represent.

• There are many different ways of visualizing and annotating modules, both within R and by using additional programs.

• Many of these methods will work with any gene list, regardless of origin (not just modules).
Acknowledgements

We wish to thank the Allen Institute founders, Paul G. Allen and Jody Allen, for their vision, encouragement, and support.

Any questions?