

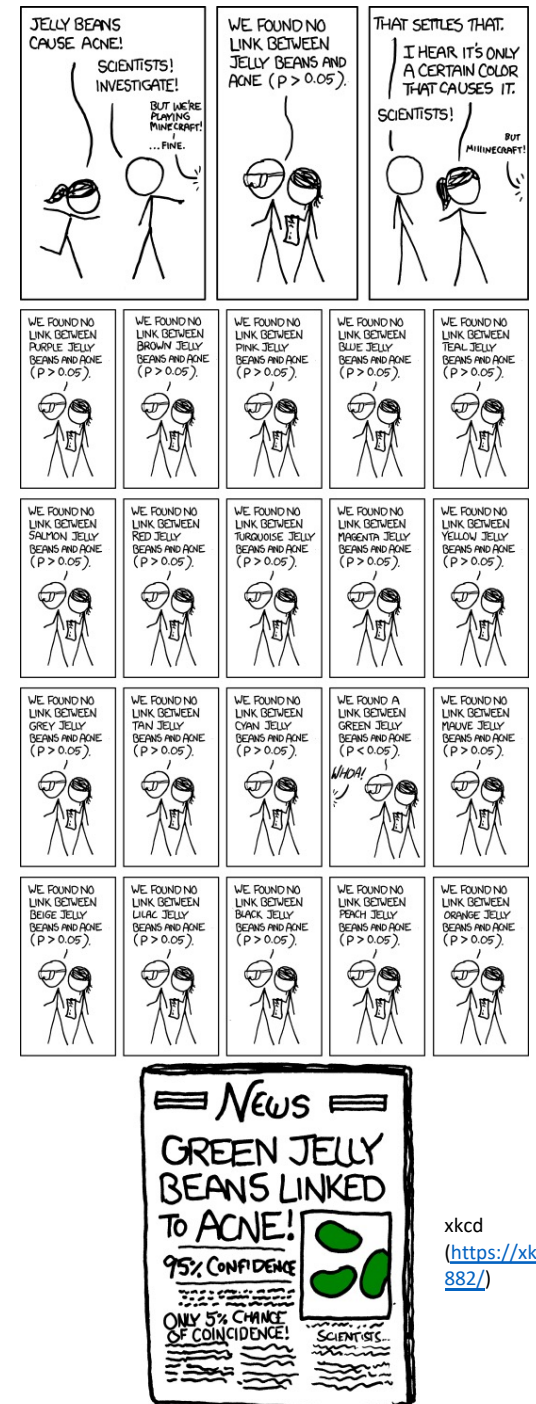
# A Primer on Weighted Gene Co- expression Network Analysis (WGCNA) + Alder Walkthrough

*Lab Meeting May 4<sup>th</sup>, 2020*



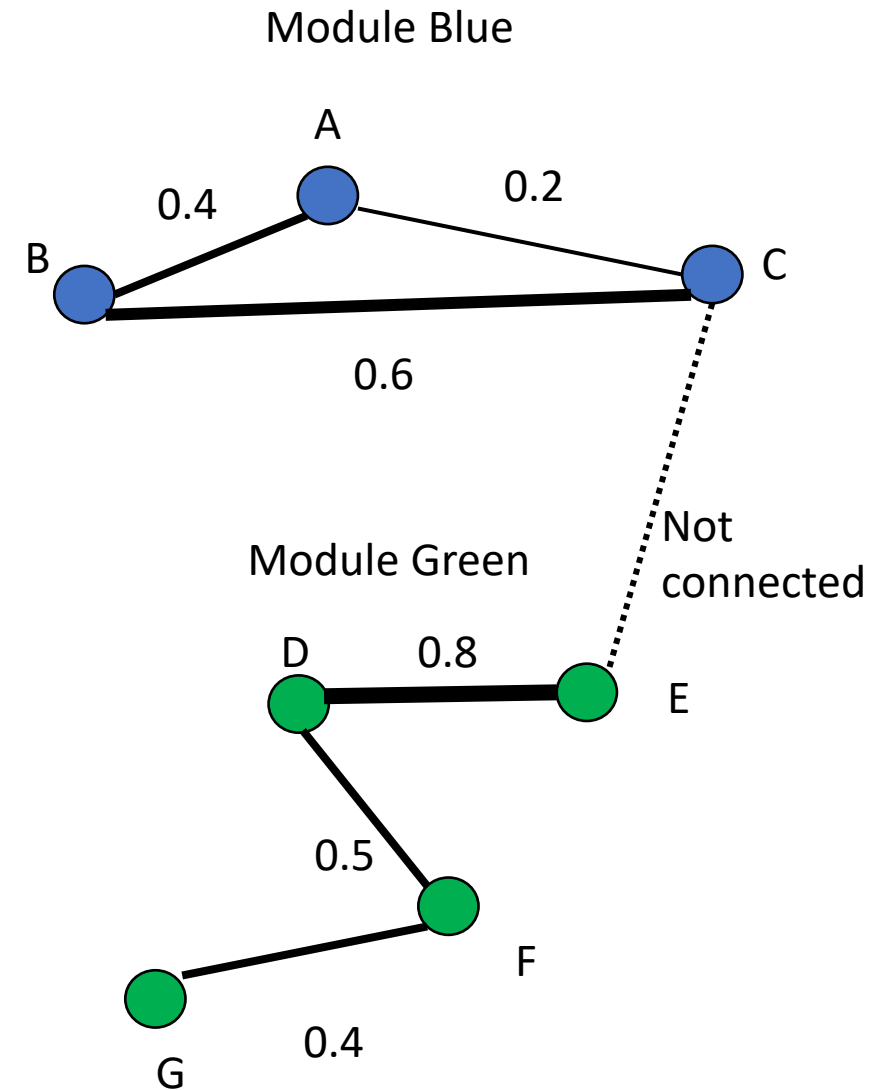
# What is WGCNA?

- Used to quantify relative co-expression of genes across different samples
- Network is constructed to analyze overall connectivity
  - More holistic, systems level picture!
- ‘Weights’ the strength of co-expression
  - More robust!
- Groups genes into modules based on expression similarity, and analyzes by group
  - Solves multiple comparisons problem!

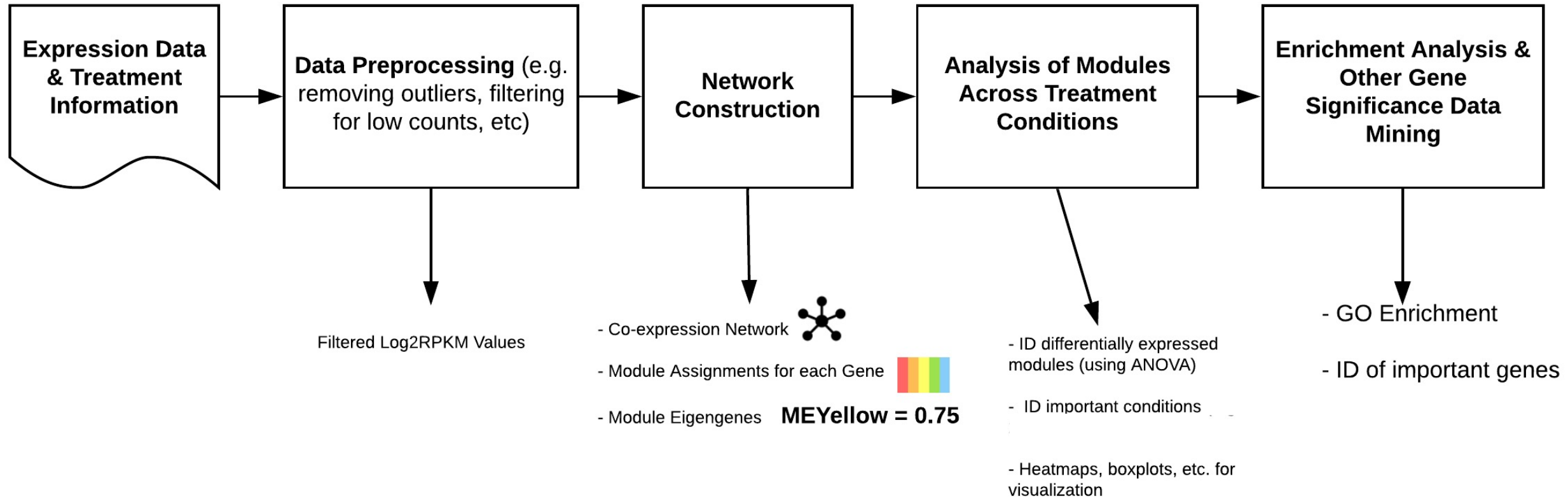


# How does it work?

- To construct the network:
  - Genes are nodes (e.g. A, B, C)
  - the “connection strength” (from 0-1) is calculated based on how strongly the genes are co-expressed
  - This is usually assessed via pairwise correlations
  - We can use this to assign genes into modules
  - We can also ID genes that may be more important than others (e.g. B vs A, or D vs G)



# WGCNA Workflow Overview



# Data Preprocessing

Raw read counts of expressed genes in each sample



Conversion to Counts per Million (CPM)



Filtering for at least one CPM



Conversion to Reads per Kilobase per Million (RPKM)



Removal of Genes with excessive missing values in samples



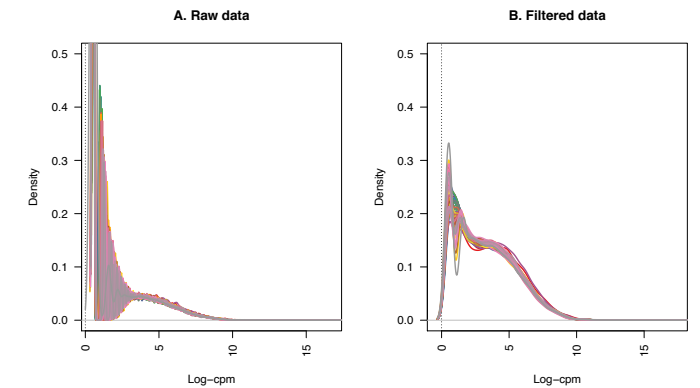
Filtering for genes with  $\text{RPKM} \geq 0.25$



$\text{Log}_2(X + 0.25)$

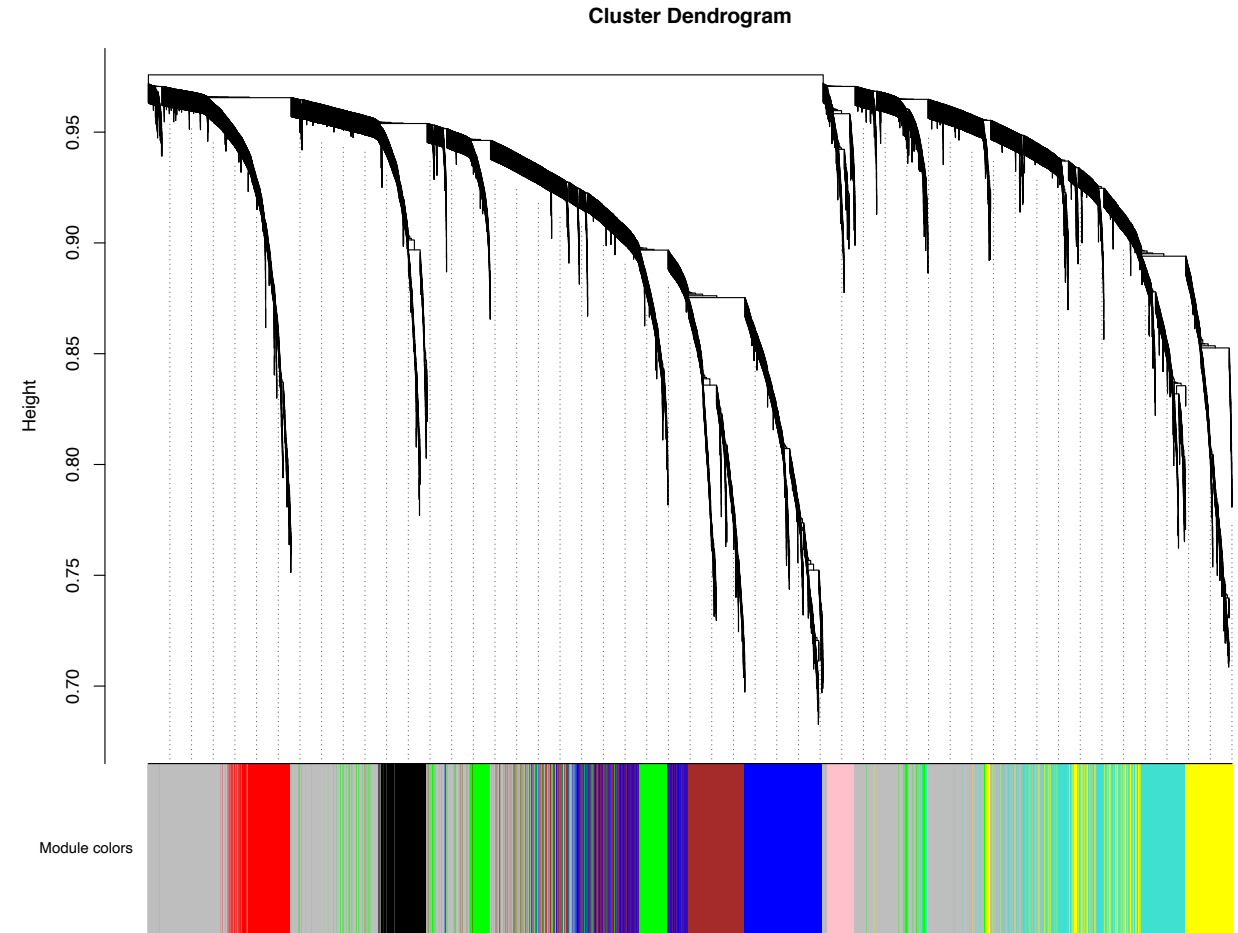


Precluster and remove sample outliers



# Network Construction


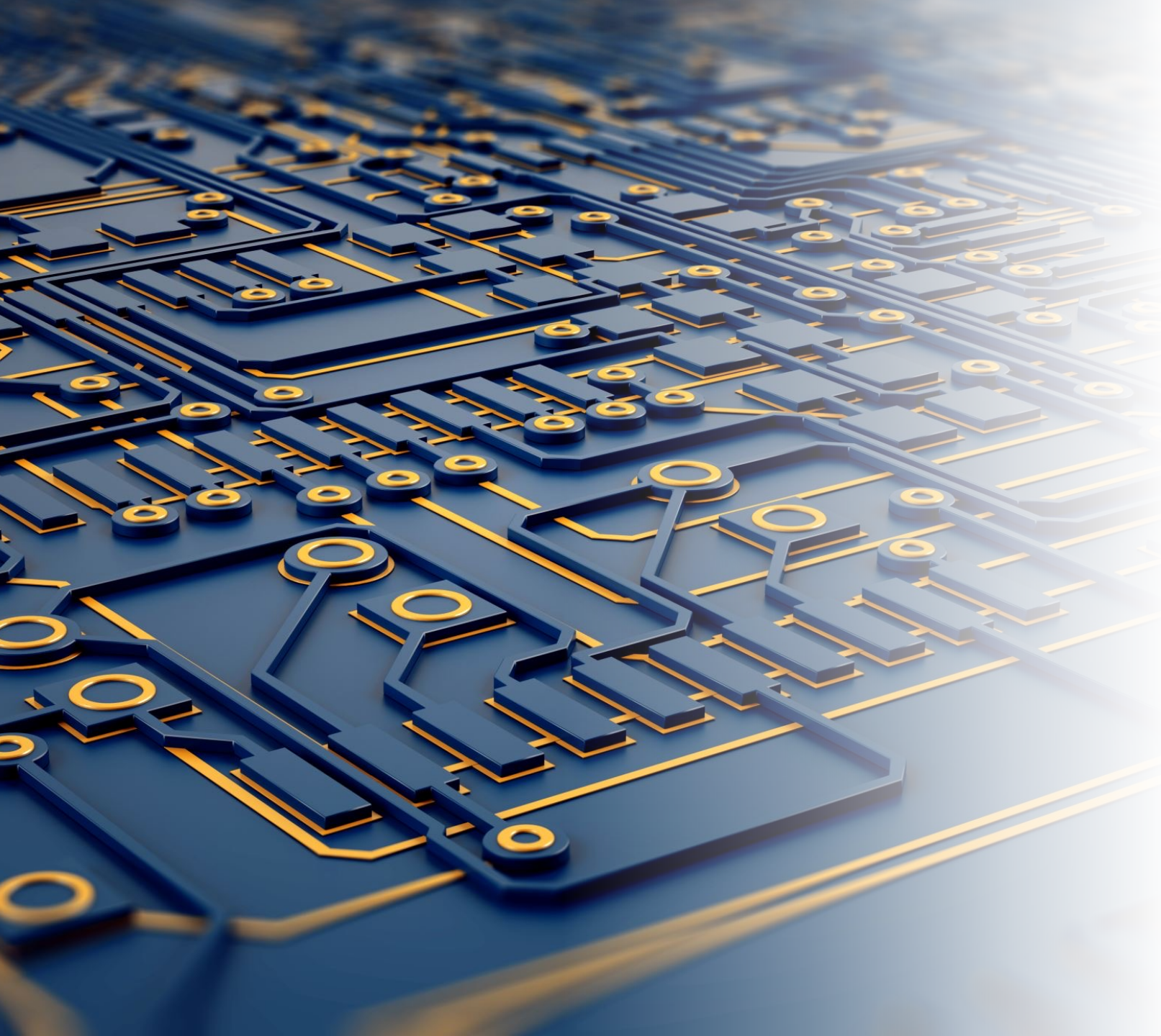
- Soft threshold values (scale it to a power) to ensure results are more robust
- Analyzes connectivity based on co-expression (runs a “topology analysis function”)
- The strength of connection varies based on whether genes are connected or not
- Genes are assigned into modules based on similarity of expression



# Quantifying Module Assignment

- Can analyze relationship between module assignment and treatment information, using correlation
- We can use pearson's correlation to analyze correlation between numerical values (e.g. does module assignment affect weight of the animal)
- In our case, since our treatment information is categorical we can use ANOVA
  - This can tell us whether there is an effect of treatment on module assignment, but not whether a specific treatment condition has an effect (less precise)





# Working with the Alder Cluster Part 2: Submitting Jobs

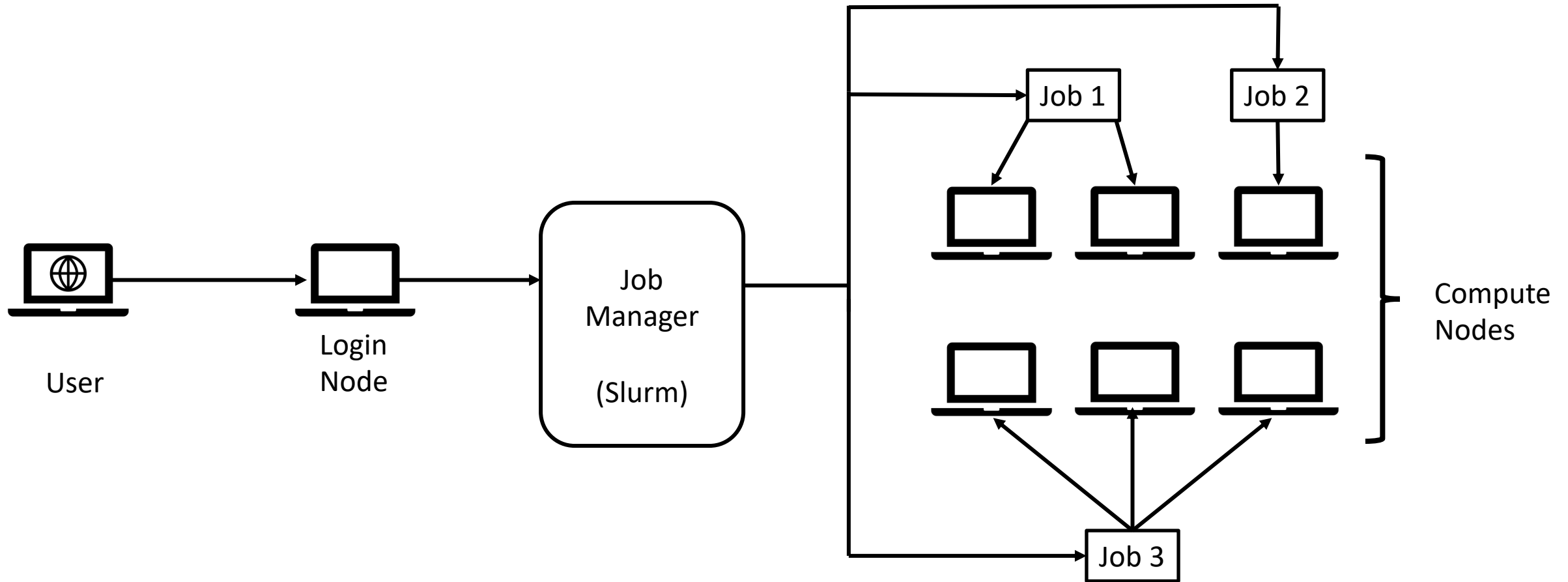
---



# Cores, CPUs, & Memory

- CPUs usually have around 4-16GB of memory (RAM), and 2-4 cores
  - RAM – increases amount of work that can be done at a time
  - Cores – number of “workers” to increase efficiency of processing
- Clusters are essentially a network of computers, that provide increased parallel processing
- Alder has:
  - 1920 GB of RAM
  - 288 cores
  - It has 512 TB of storage

# Outline of High Performance Computing and Clusters



# How do I send a job?

- Move necessary files to the cluster,
- You need to submit a command using a bash script that specifies:

```
1  #!/bin/bash
2  #
3  #SBATCH -c 8 ----- Number of cores needed
4  #SBATCH --mem-per-cpu=4000 ----- Memory needed
5  #SBATCH --job-name= network_analysis_pearson ----- What is the job name? (optional)
6  #SBATCH --output=pearson_v3.out ----- What is the output log named? (optional)
7  #SBATCH --time=12:00:00 ----- How long should it take?
8
9  module load R Load the program "R"
10                                     And run the script located here
11  Rscript --vanilla ~/ Analysis/V3/Pearson/ Network_Construction_pearson.R
12
```

# Important Considerations:

- Note the queue
- Be mindful of the duration and the number of CPUs specified
  - this can delay or even kill your job
- Know where your files are located
- Check output for trouble shooting