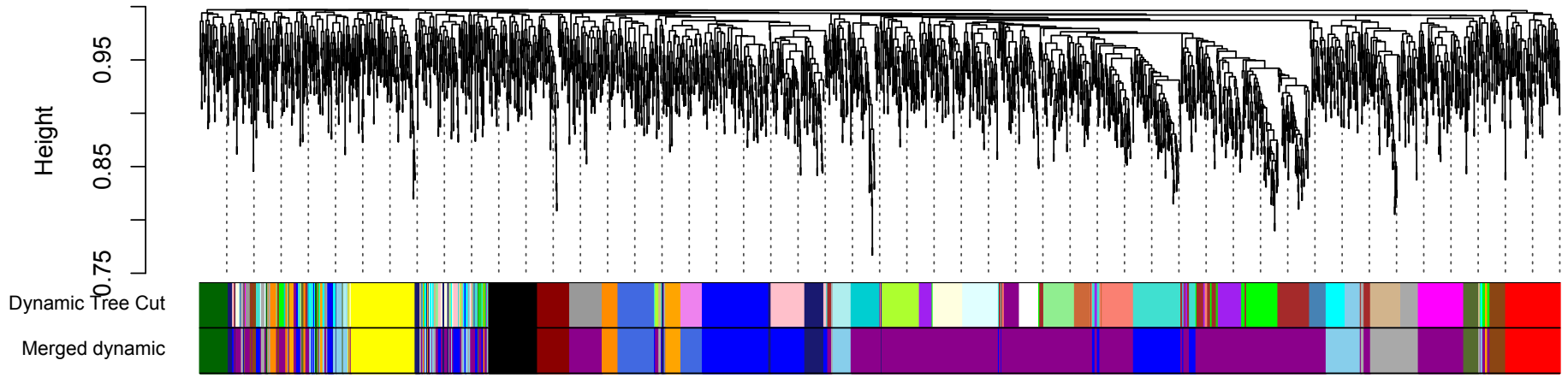


Weighted Gene Co-Expression Network Analysis

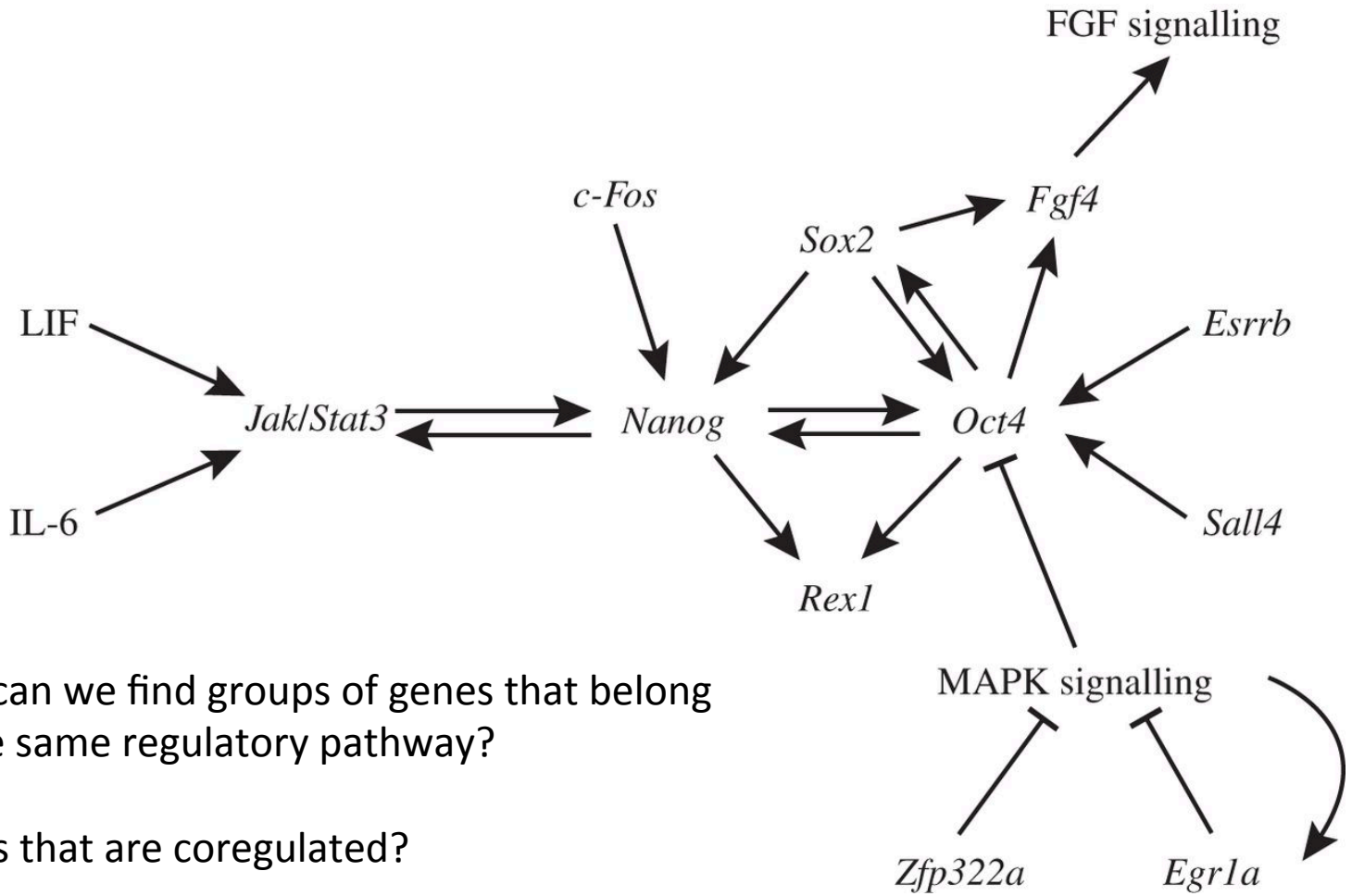
Dept. of Human Genetics, UC Los Angeles (PL, SH), and Dept. of Biostatistics, UC Los Angeles (SH)

Peter (dot) Langfelder (at) gmail (dot) com, SHorvath (at) mednet (dot) ucla (dot) edu

BMC Bioinformatics, 2008 9:559

Thanks to M arie Strader and Rachel Wright for slides

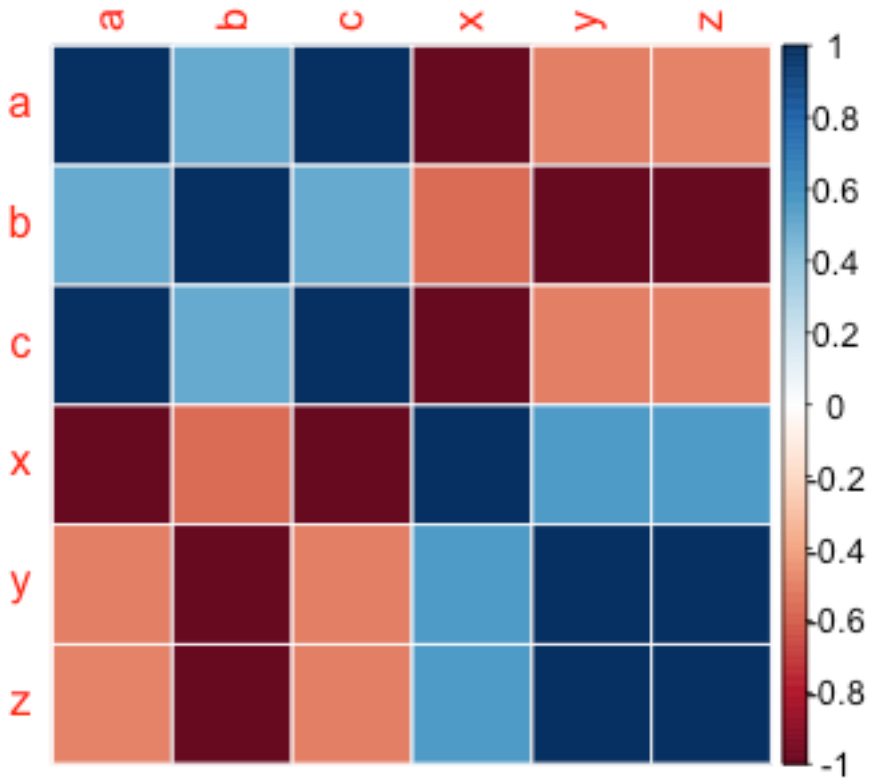
Genes function in regulatory pathways



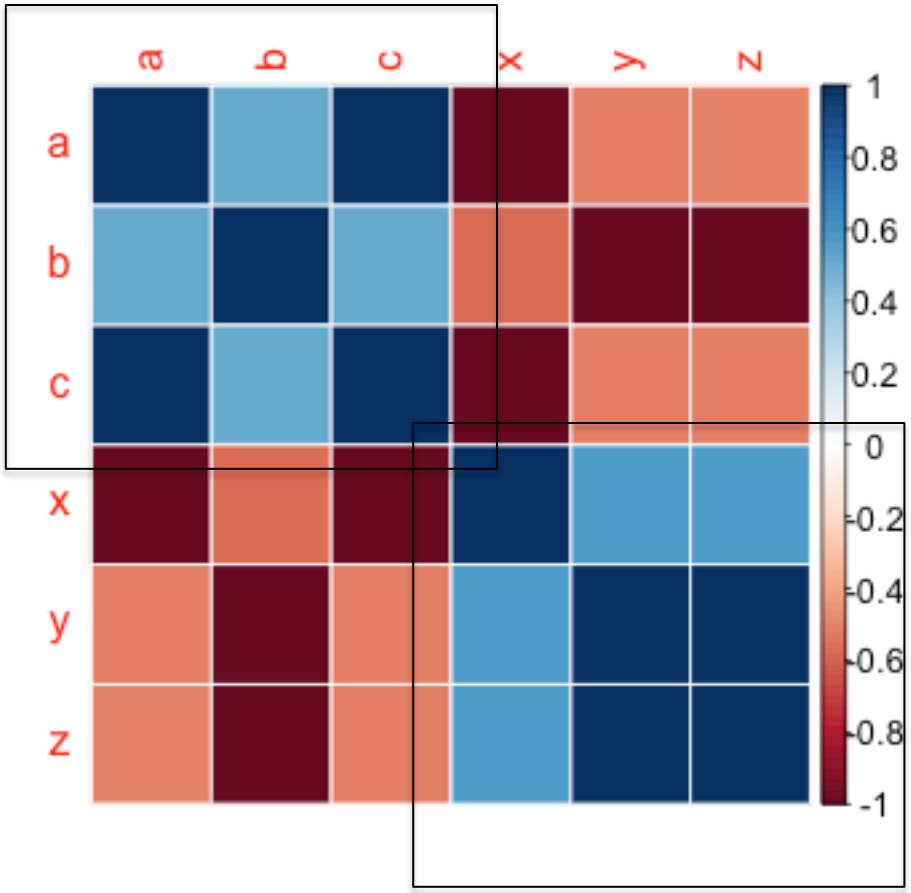
How can we find groups of genes that belong to the same regulatory pathway?

Genes that are coregulated?

Co-regulated genes have correlated expression



Co-regulated genes have correlated expression

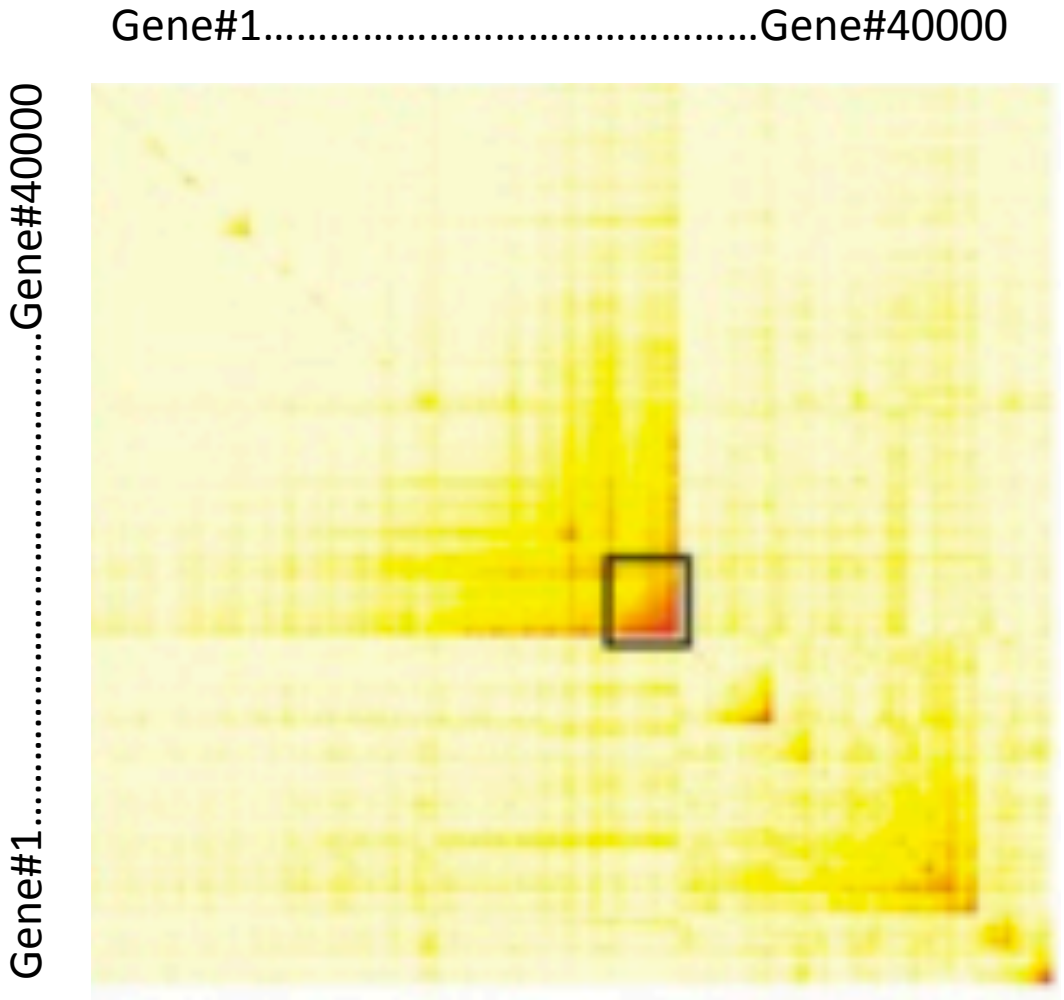


A <-> B <-> C

and

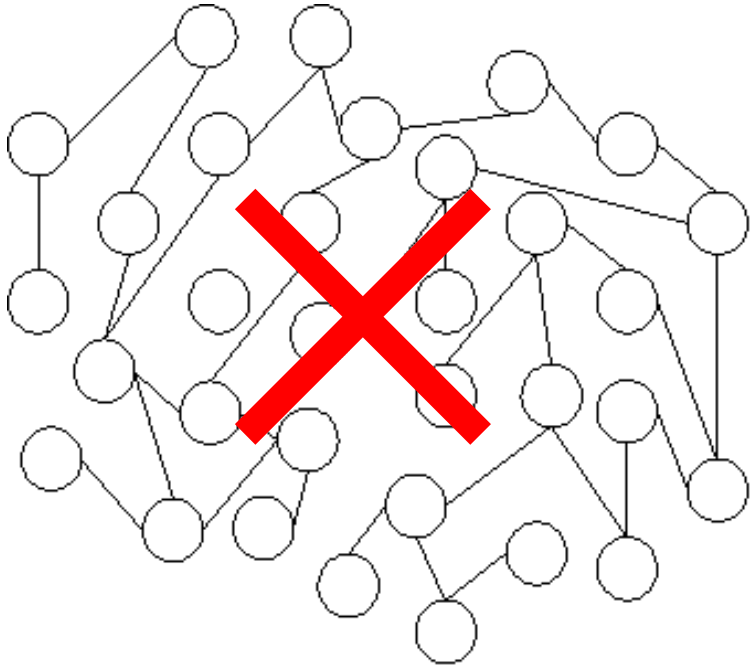
X <-> Y <-> Z

Now with 40,000 genes…

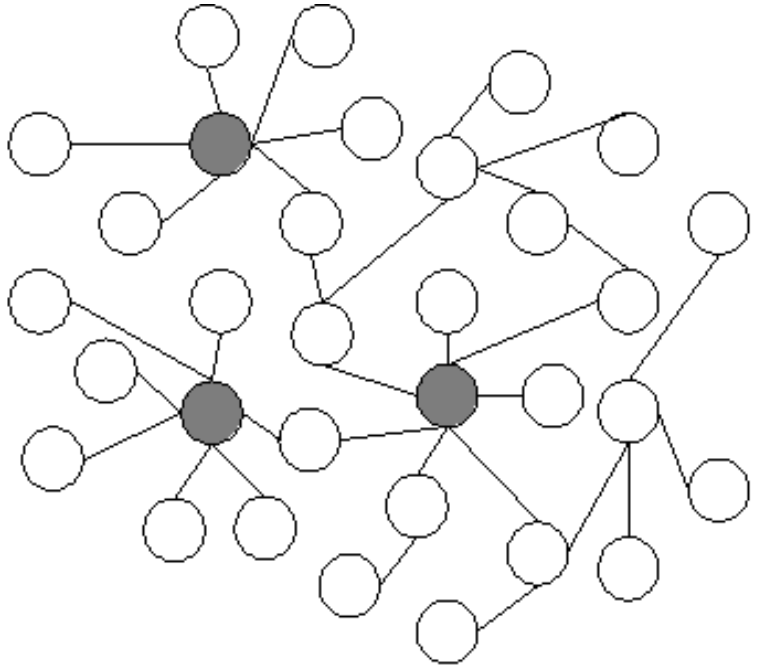


Pro tip: use a supercomputer

We assume a scale-free topology

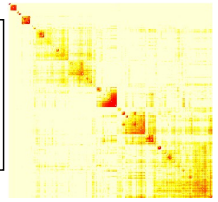


(a) Random network

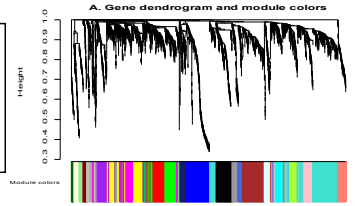


(b) Scale-free network

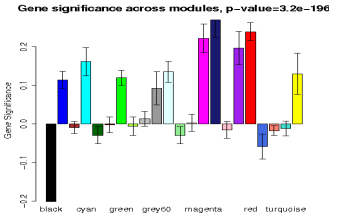
Construct a gene co-expression network
Rationale: make use of interaction patterns among genes
Tools: correlation as a measure of co-expression



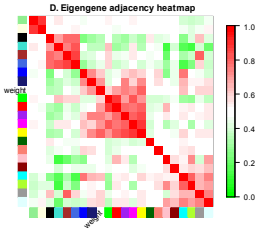
Identify modules
Rationale: module (pathway) based analysis
Tools: hierarchical clustering, Dynamic Tree Cut



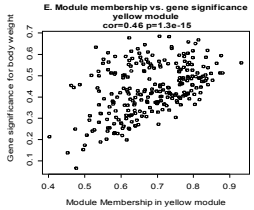
Relate modules to external information
Array Information: clinical data, SNPs, proteomics
Gene Information: ontology, functional enrichement
Rationale: find biologically interesting modules



Study module relationships
Rationale: biological data reduction, systems-level view
Tools: Eigengene Networks

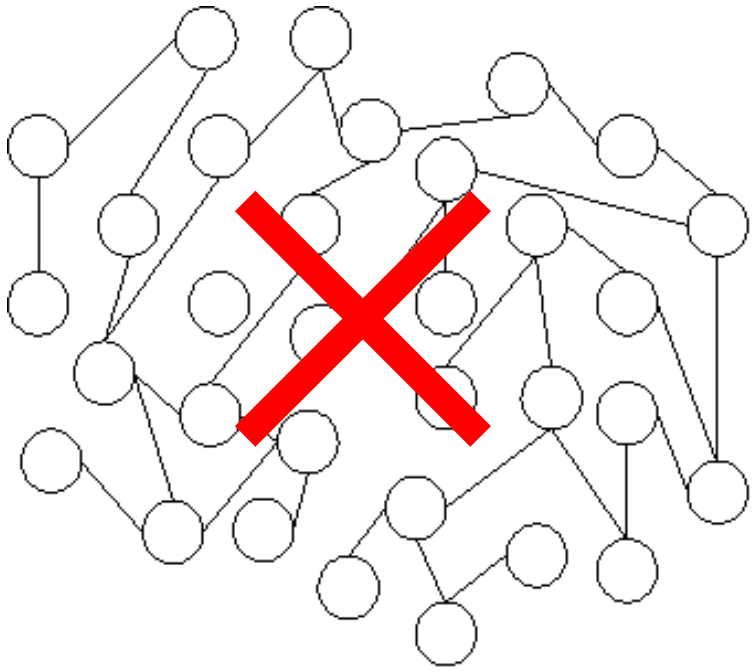


Find the key drivers in *interesting* modules
Rationale: experimental validation, biomarkers
Tools: intramodular connectivity, causality testing

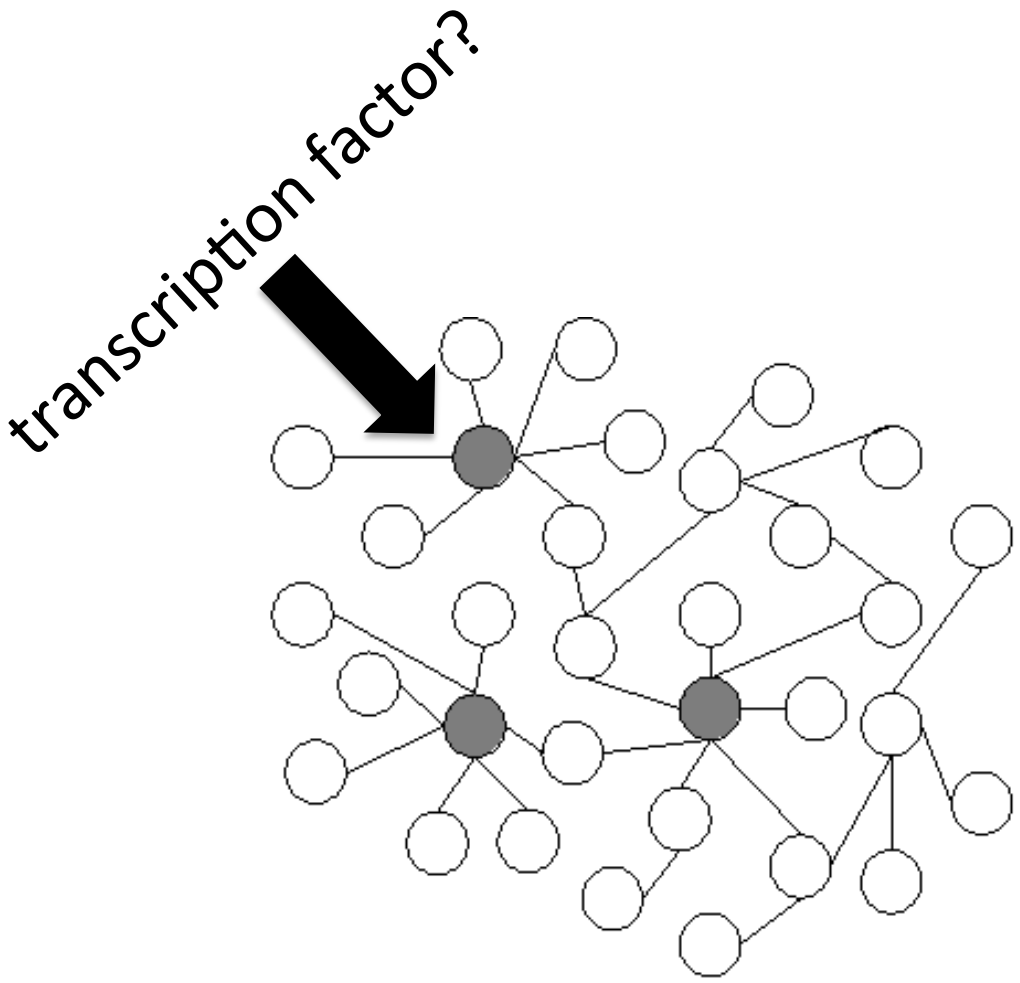


Overview of WGCNA methodology. This flowchart presents a brief overview of the main steps of Weighted Gene Co-expression Network Analysis.

Langfelder and Horvath BMC Bioinformatics 2008 9:559 doi:10.1186/1471-2105-9-559

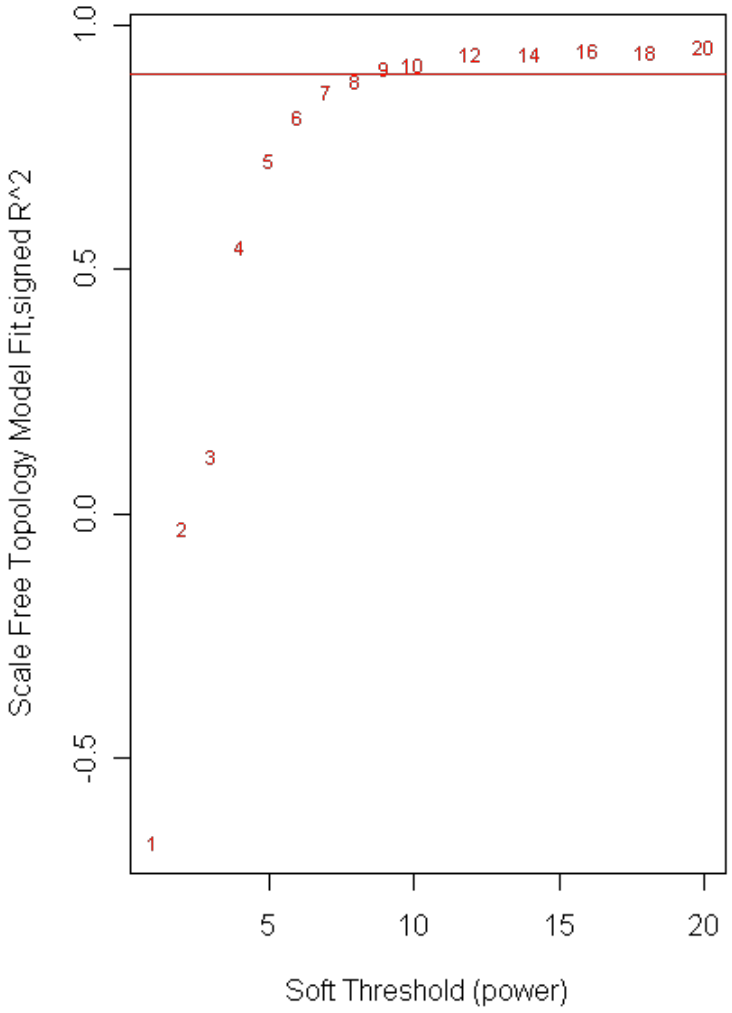


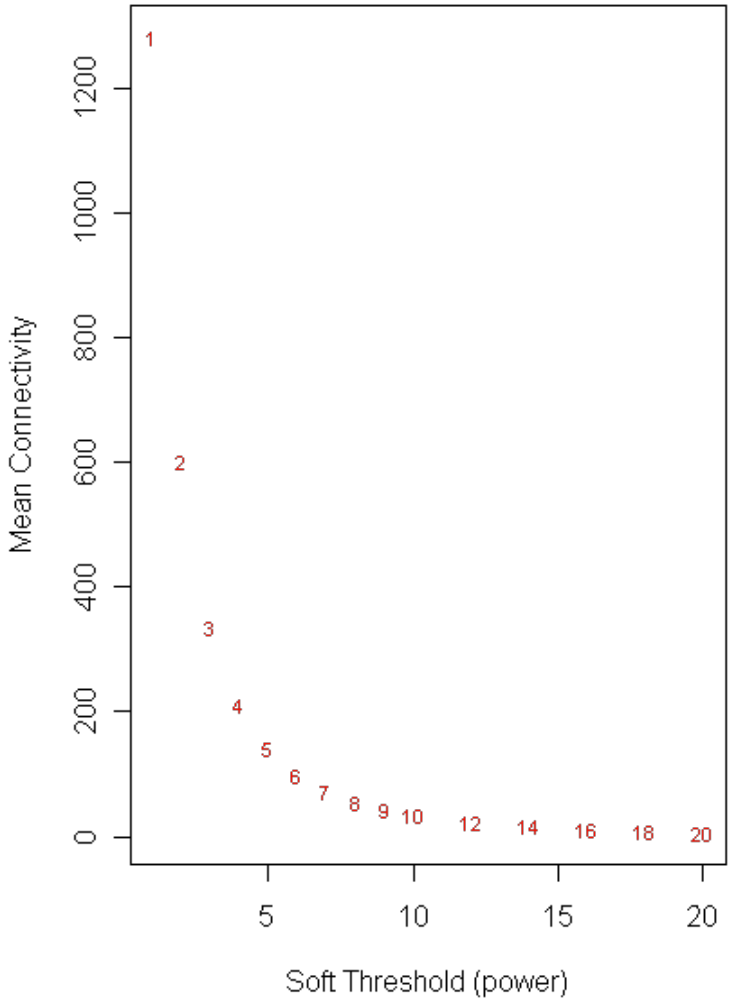
(a) Random network

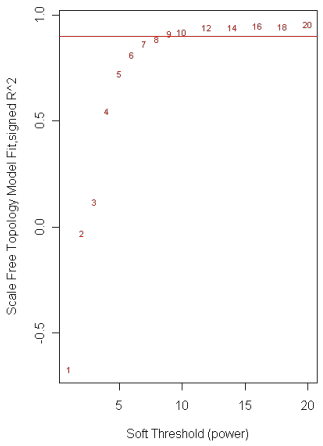


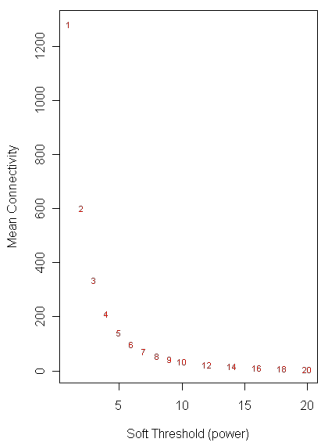
(b) Scale-free network

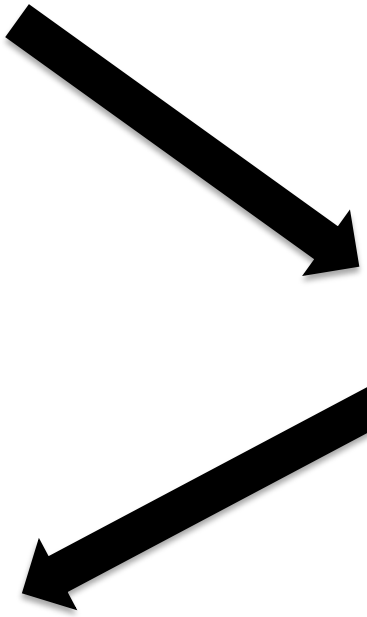
Pick a soft-power threshold

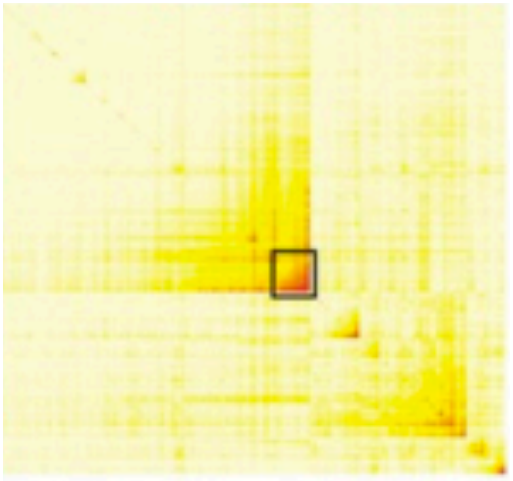








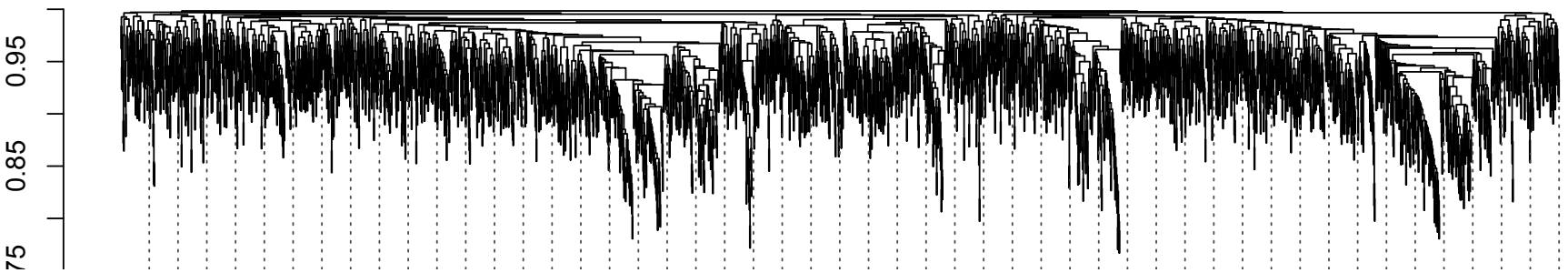




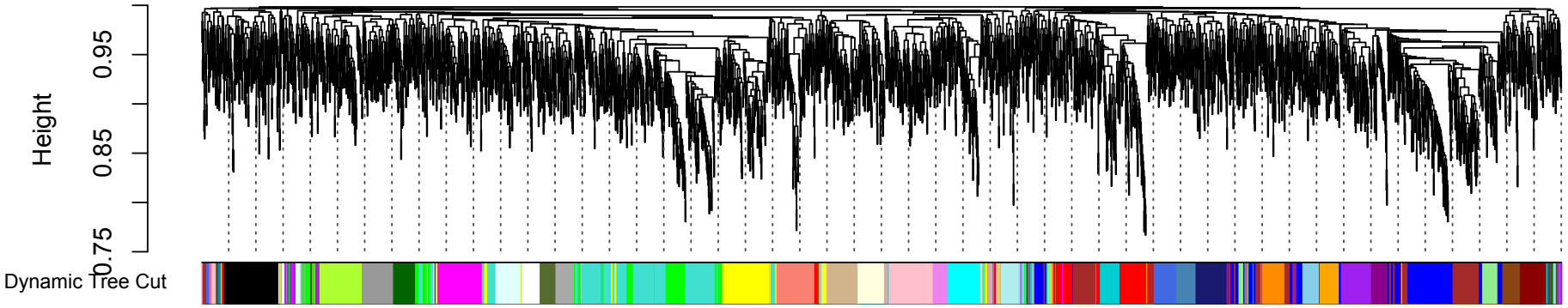
Group genes into “modules”

Cluster genes by dissimilarity

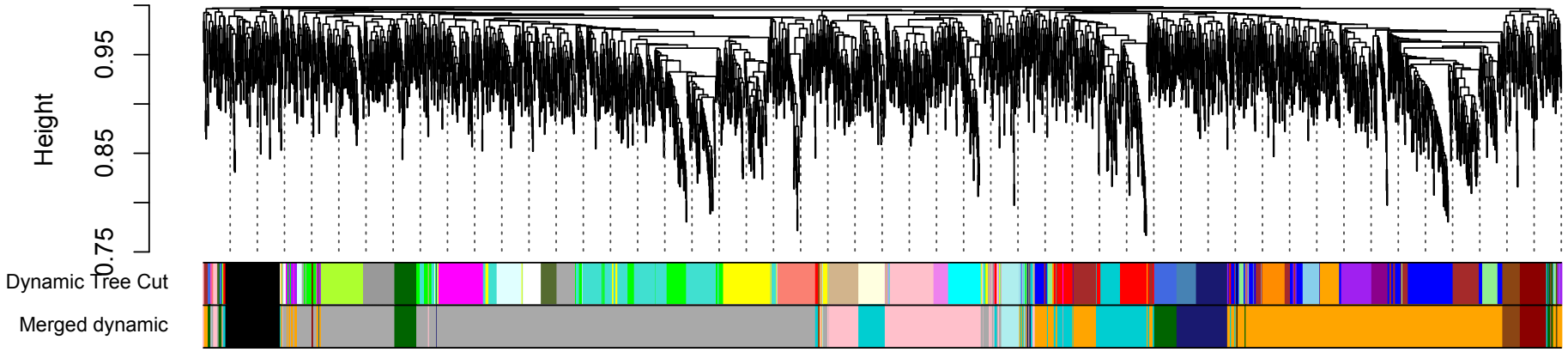
Dissimilarity = (1 – Similarity)



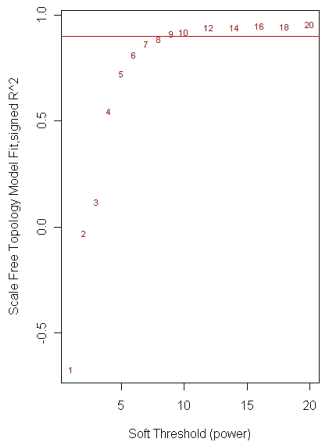
Cluster into “modules”

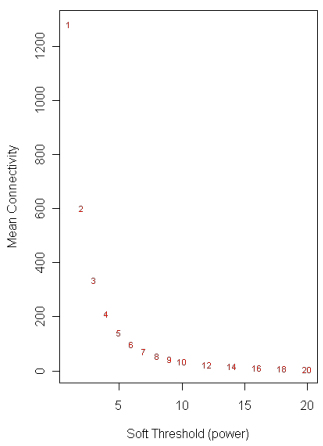


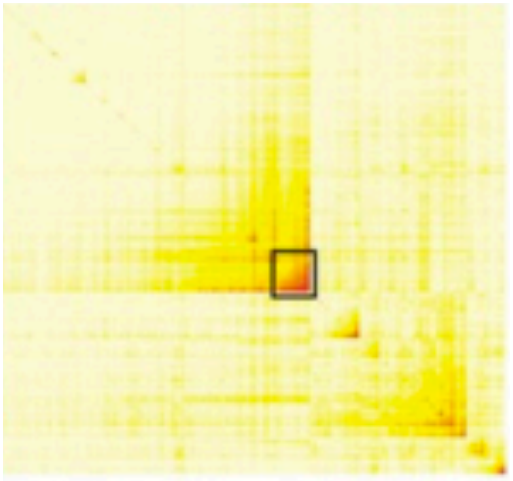
Merge modules as you like*

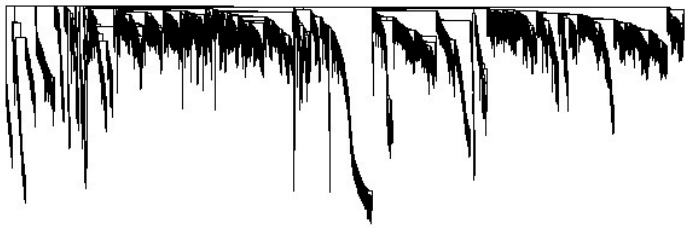


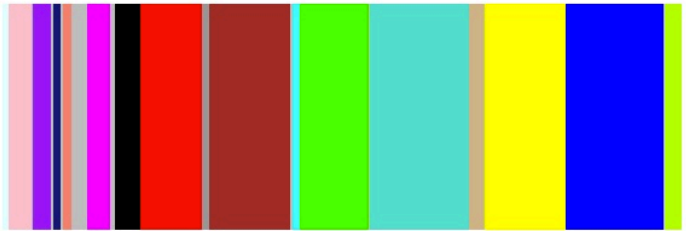
*Merging can be pretty arbitrary.
Devise your own criteria and stick with it.







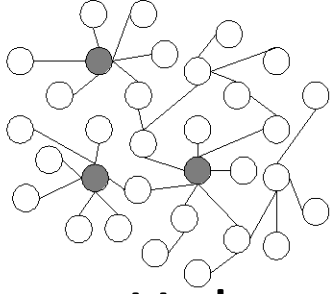




Explore modules

-hub genes

-functional characterization



Hub gene selection

Hub gene = “highly connected” = **High intramodular connectivity** = Their expression profile is very similar to the expression profile representing the entire module.

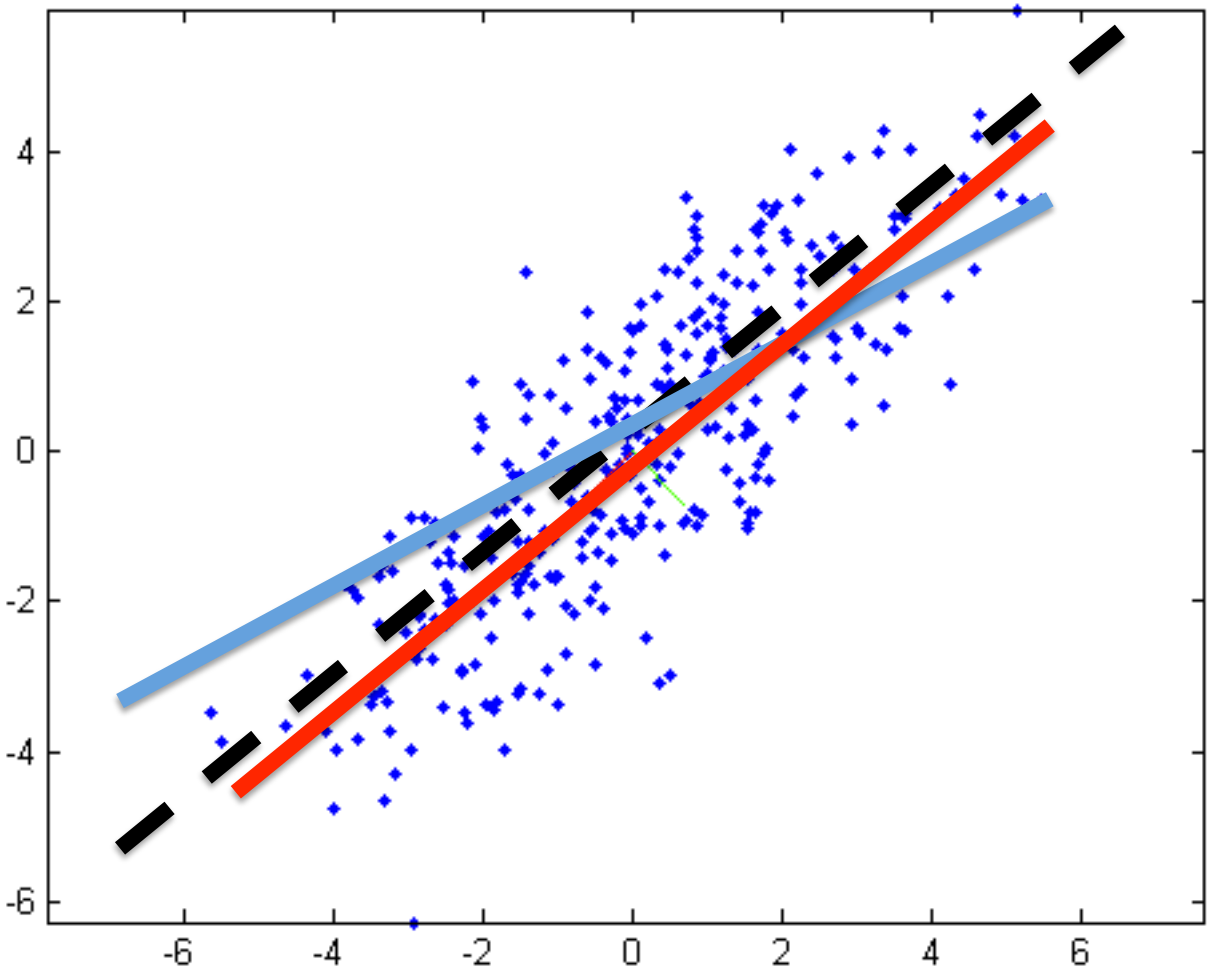
Module Eigengene(ME) = first principal component of the gene expression within a module

Intramodular connectivity (kME) = correlation of the gene of interest with the module eigengene (ME)

Module eigengene (1st PC)

Gene with high kME

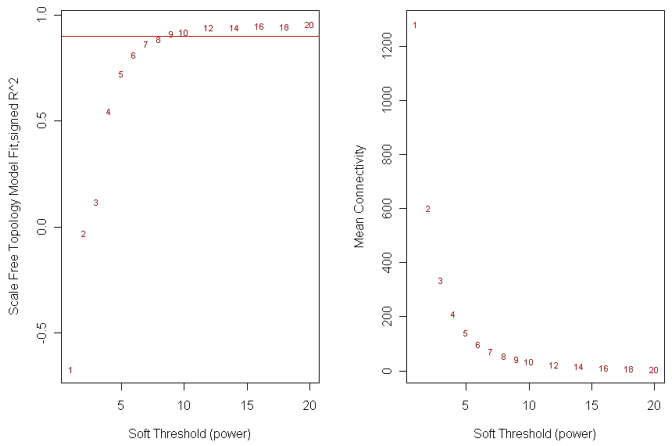
Gene with low kME

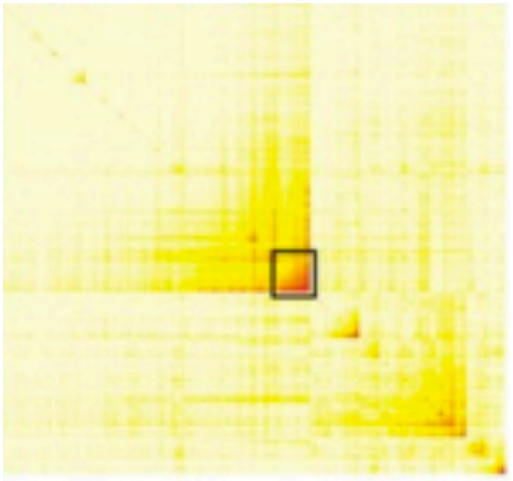


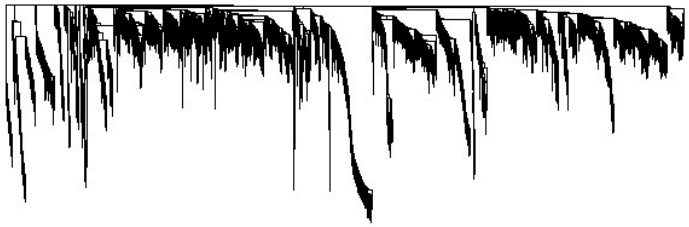
Functional characterization of modules

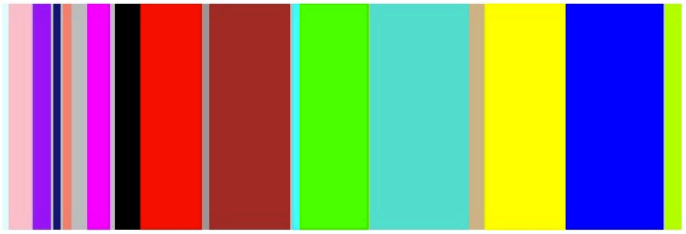
GO enrichment

- Fisher (genes enriched in module compared to genes not in module)
- kME (genes enriched with high connectivity within a module)







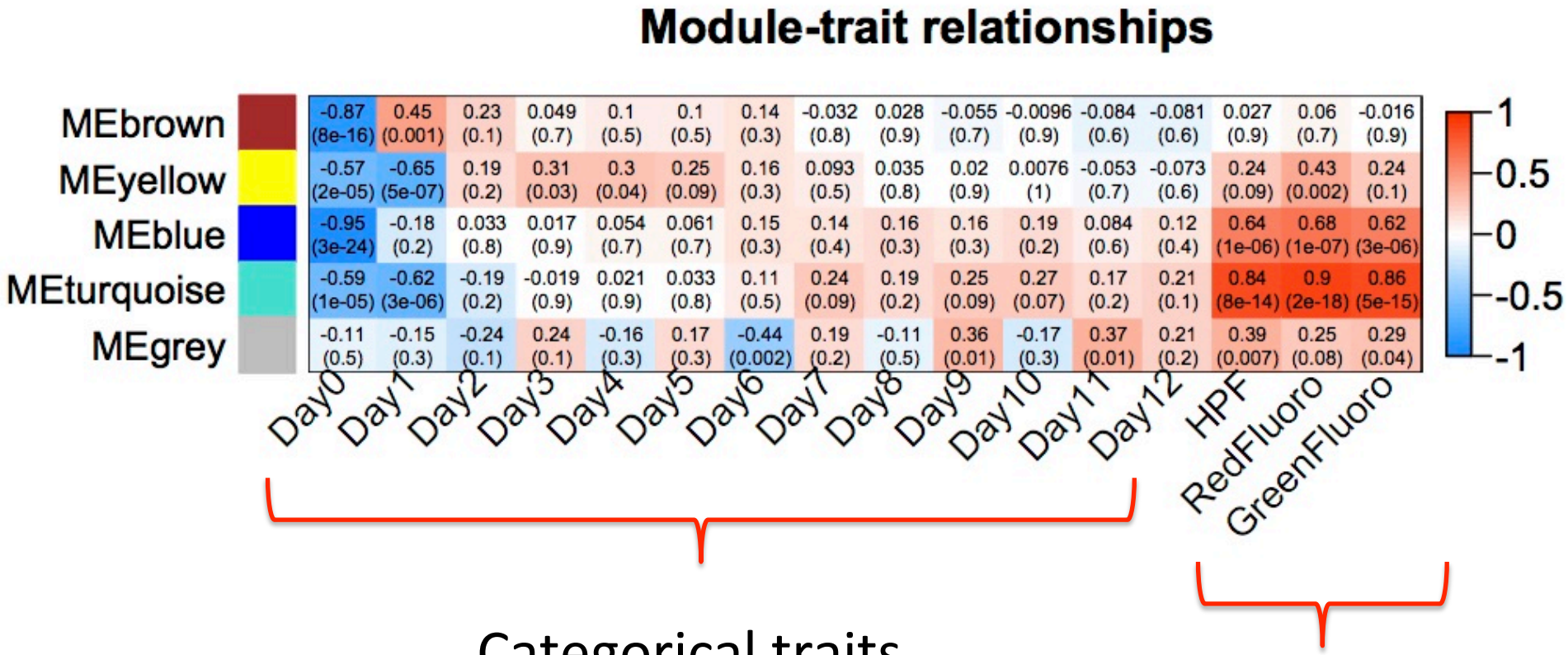


Explore modules

-hub genes

-functional characterization

Relate to traits

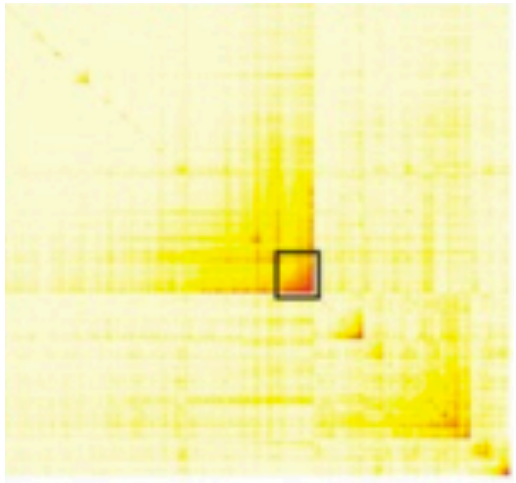


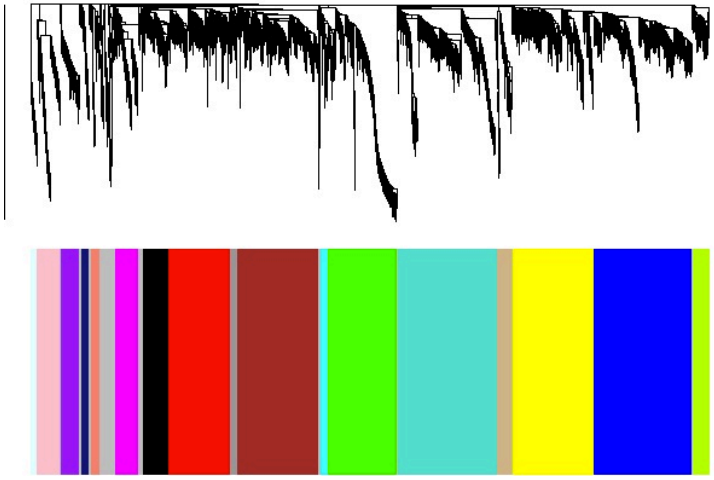
Categorical traits
(e.g., genotype, group)

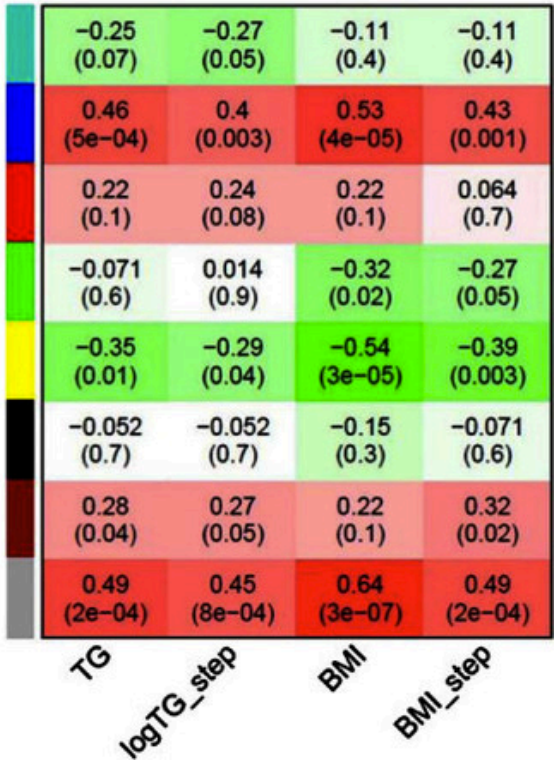
Continuous traits
(e.g., age, blood pressure)

WGCNA: weighted gene coexpression network analysis

- Identify groups (modules) of co-regulated genes
- Correlate module expression values to trait data







Benefits of WGCNA

- Unsupervised and unbiased
- Finds unique co-expression modules that correspond to clusters of correlated transcripts

What kind of data do you need?

- Normalized counts data files (.csv)
  - RLD file: Regularized log transformation
  - VDS file: Variance stabilized transformation
  - Other normalizations…
    - Generated from DESeq2
- Trait Data
  - Made in Excel, saved as .csv

Expression (RLD or VSD) file

Generate in DESeq2 after running models testing for DEG

>rld <- rlogTransformation(dds)

>head(assay(rld))

Sample names

	A0	A1	A2	A4	A5	A6	A7	A8	A9	A10
isogroup15101	2.82549546	4.06759551	4.15015188	3.92425478	3.8752425	4.15720096	4.362588	4.20402253	3.52818774	4.39011452
isogroup1899	3.72944044	3.10146267	3.54232816	4.15086264	3.19906489	4.02911221	3.52304751	4.0822238	3.89230834	3.71526596
isogroup1740	5.54328133	6.18299553	5.33995164	4.89480344	5.26367912	4.43778139	4.83756775	5.07575865	4.78917714	4.46499727
isogroup10619	5.2466388	3.74616265	3.43135709	4.22865372	3.77331132	4.22662229	3.94077908	4.38447932	3.17170044	4.1174373
isogroup17834	4.76228988	5.5904884	5.51641393	5.62605155	5.251188	5.81338891	6.01289306	5.99339396	6.2658312	5.99383711
isogroup9381	2.93847224	3.61373401	4.19689675	5.00706885	4.59319068	4.56363977	4.64371956	4.85606983	5.50908674	4.68913678
isogroup24364	2.00893342	3.53898226	4.52771413	4.12487957	4.35346514	4.17349074	4.42119884	4.16908203	4.67107808	3.40266781
isogroup4609	2.17752437	2.8924102	3.5109953	3.59295348	3.87792018	3.66128958	4.4082792	4.38925095	5.43510798	4.24808138
isogroup1842	5.99710721	5.36598394	5.66620617	5.59889082	5.82472241	5.42928231	5.98803124	5.96858855	5.91991946	5.92887119
isogroup33037	5.66625905	4.77496331	4.38204705	4.26073209	4.16054744	4.30157795	4.1897518	4.41213005	4.29280977	3.67313524
isogroup5105	4.8528762	3.63265174	4.40928524	4.32758803	4.334474	3.82746384	4.50083462	4.70820627	4.55616551	4.33493429
isogroup19649	1.24659062	1.49994797	2.91034086	2.57087966	2.06627208	4.02297538	2.91161036	3.00308279	2.60126955	2.8976201
isogroup15316	0.98077318	1.28237633	2.73568291	2.70618638	2.7856968	2.78304732	3.03534729	3.46513152	2.81370914	2.56991437
isogroup2753	6.58796633	5.81127201	5.98209262	5.65433634	5.70574057	5.84804204	5.7672827	6.02065128	6.20889705	6.26624094

Gene names

Trait Data file (upload in R as .csv)

- Categorical or quantitative traits
- Examples:

Sample	Day0	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Day8	Day9	Day10	Day11	Day12	HPF	RedFluoro	GreenFluoro
A0		1	0	0	0	0	0	0	0	0	0	0	0	3	0	0
A1		0	1	0	0	0	0	0	0	0	0	0	0	22	0	0
A2		0	0	1	0	0	0	0	0	0	0	0	0	46	0	0
A4		0	0	0	0	1	0	0	0	0	0	0	0	56	54.13165	19.4236
A5		0	0	0	0	0	1	0	0	0	0	0	0	80	75.6628	17.8254
A6		0	0	0	0	0	0	1	0	0	0	0	0	104	92.429	56.9575
A7		0	0	0	0	0	0	0	1	0	0	0	0	128	98.276412	63.443118
A8		0	0	0	0	0	0	0	0	1	0	0	0	150	98.828077	64.8785
A9		0	0	0	0	0	0	0	0	0	1	0	0	176	101.976286	75.247286
A10		0	0	0	0	0	0	0	0	0	0	1	0	198	96.972333	74.569667
A11		0	0	0	0	0	0	0	0	0	0	0	1	221	91.075765	78.188059

Extensive Tutorials Online

- http://labs.genetics.ucla.edu/horvath/CoexpressionNetwork/Rpackages/WGCNA/Tutorials/
- http://labs.genetics.ucla.edu/horvath/CoexpressionNetwork/Rpackages/WGCNA/

- R package

```
>source("http://bioconductor.org/biocLite.R")
>biocLite(”WGCNA")
```

End

FAQs (from WGCNA website)

How many samples do I need?
We do not recommend attempting WGCNA on a data set consisting of fewer than 15 samples. In a typical high-throughput setting, correlations on fewer than 15 samples will simply be too noisy for the network to be biologically meaningful. If at all possible, one should have at least 20 samples; as with any analysis methods, more samples usually lead to more robust and refined results.

Should I filter probesets or genes?
Probesets or genes may be filtered by mean expression or variance (or their robust analogs such as median and median absolute deviation, MAD) since low-expressed or non-varying genes usually represent noise. Whether it is better to filter by mean expression or variance is a matter of debate; both have advantages and disadvantages, but more importantly, they tend to filter out similar sets of genes since mean and variance are usually related.

We do not recommend filtering genes by differential expression. WGCNA is designed to be an unsupervised analysis method that clusters genes based on their expression profiles. Filtering genes by differential expression will lead to a set of correlated genes that will essentially form a single (or a few highly correlated) modules. It also completely invalidates the scale-free topology assumption, so choosing soft thresholding power by scale-free topology fit will fail.