Applications of weighted gene coexpression analysis

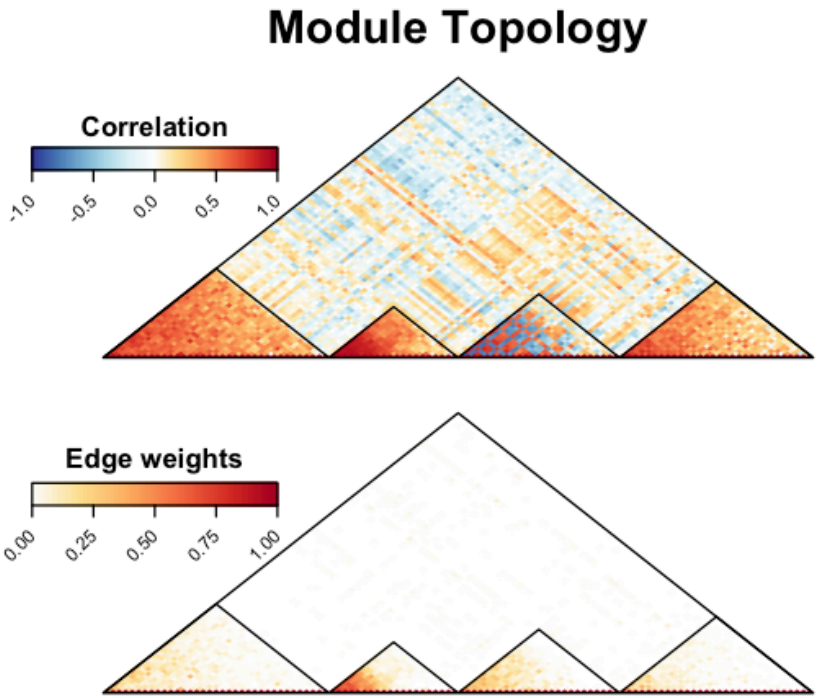
Michael Inouye
Centre for Systems Genomics
University of Melbourne

Summer Institute in Statistical Genetics 2017
Network & Pathway Analysis of Omics Data
Brisbane

@minouye271
inouyelab.org

Gene co-expression networks

- **Weighted, undirected complete gene network**
  - **Nodes**: genes/probes
  - **Edges**: |cor(node_i, node_j)|$^{\gamma}$
    - Scale-free assumption and [0,1]
- **Identify subnets (modules/ clusters)**
  - Typically subnets represent known biological pathways
  - Various methods and tools for clustering



Strategies for testing association of a subnet with a phenotype

- **Univariate**
  - For each subnet gene, perform a test
- **Eigenvector**
  - Calculate 1st principal component
  - With vector of PC1 sample loadings, perform a test
- **Multivariate**
  - Simultaneously test for association of phenotype with all genes
  - Example: Canonical correlation analysis (CCA)
- **Considerations**
  - Multiple testing burden
  - Sensitivity and specificity

Interpretation of subnets

- **Pathway analysis and gene set statistics**
- **If subnet is small enough, manual interpretation is possible (with proper literature support)**
- **Correlation vs Causation**
  - Confounding, causality and reactivity
    - It is more useful (and more difficult) to know the underlying structure of relationships b/n genes than *clusters* of co-regulation
  - How can causality be tested?
    - Perturbation techniques
    - Mendelian randomisation (genetic variation has a special role in determining causality)

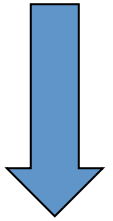
OPEN ACCESS Freely available online PLoS GENETICS

An Immune Response Network Associated with Blood Lipid Levels

Michael Inouye[1,2¤*], Kaisa Silander[3], Eija Hamalainen[1], Veikko Salomaa[4], Kennet Harald[4], Pekka Jousilahti[4], Satu Männistö[4], Johan G. Eriksson[4,5,6,7,8], Janna Saarela[3,9], Samuli Ripatti[3], Markus Perola[3], Gert-Jan B. van Ommen[2], Marja-Riitta Taskinen[10], Aarno Palotie[1,3,11,12], Emmanouil T. Dermitzakis[1,13], Leena Peltonen[1,3,11†]

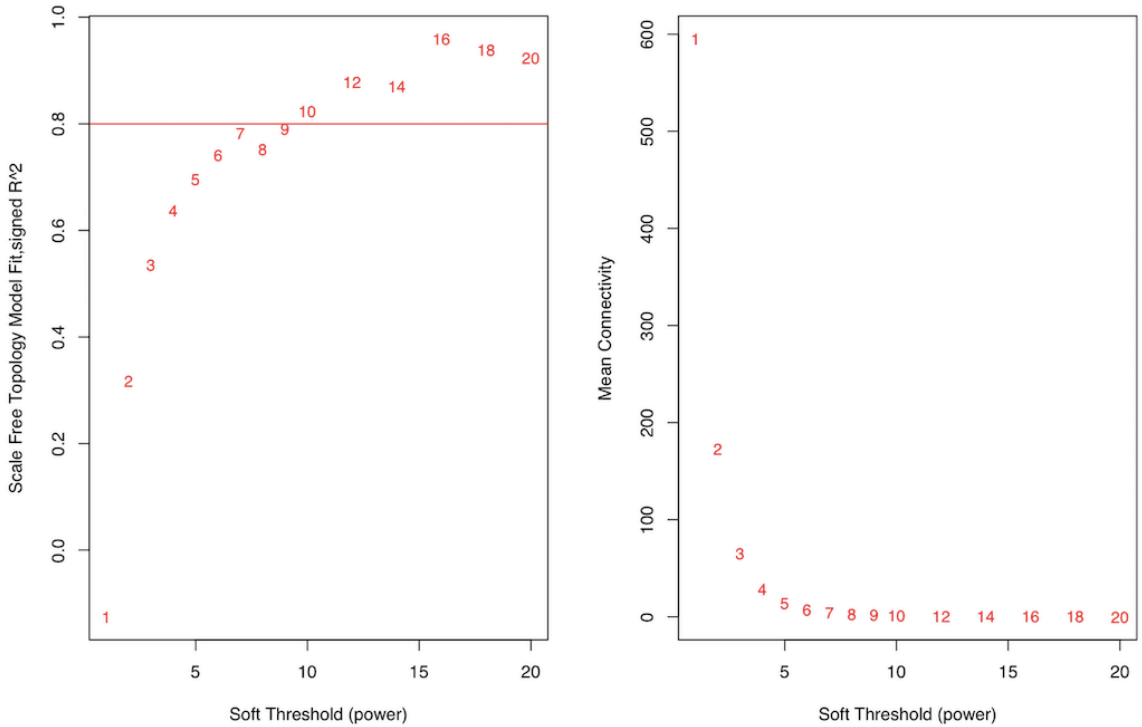
518 randomly sampled individuals

Fasting whole blood



Transcriptome

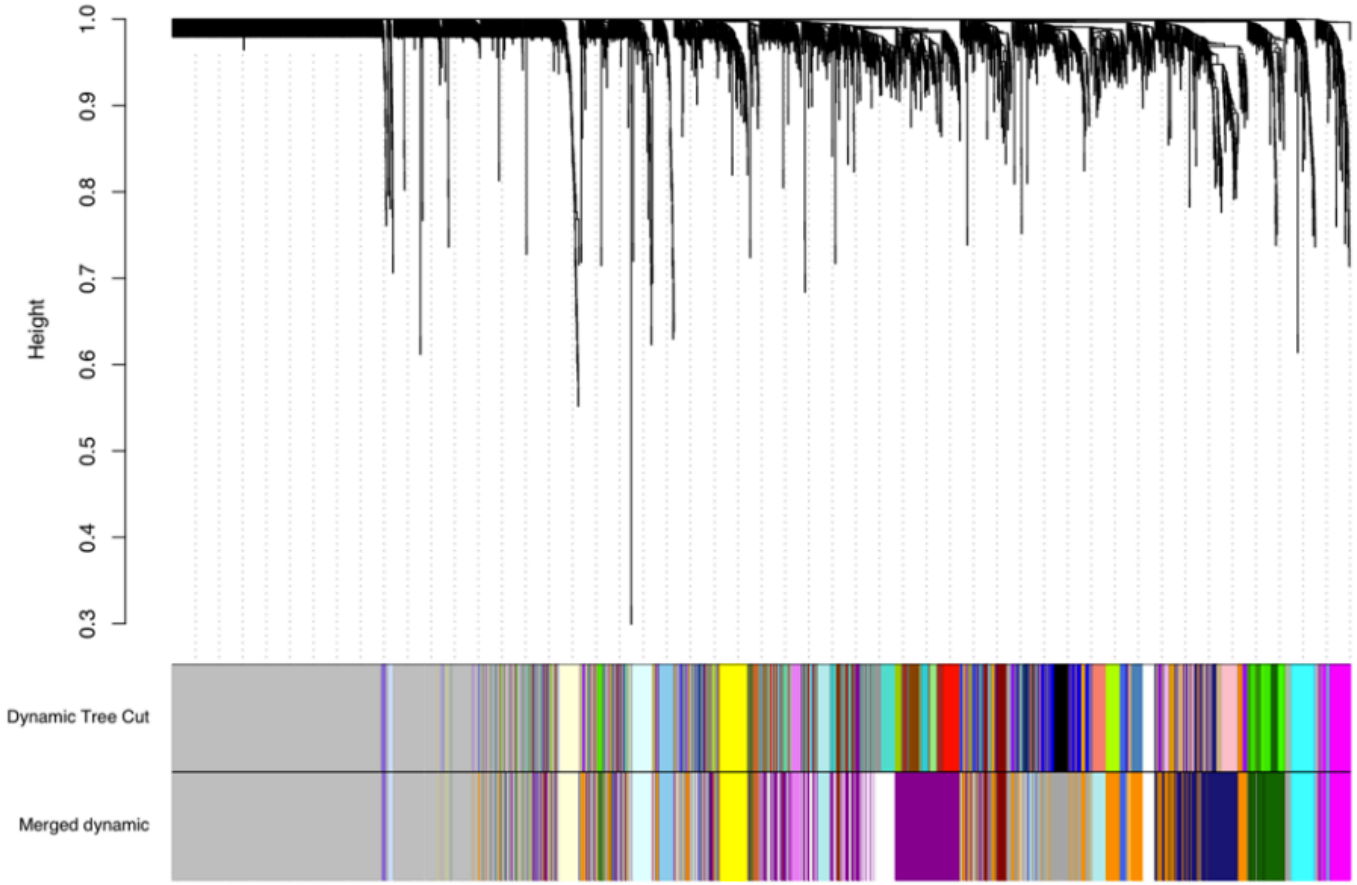
Selection of soft power threshold for adjacency matrix



Better differentiate strong vs weak correlations

Approximate scale-free network topology (signed $R^2 > 0.80$) but maximize connectivity

Identifying gene co-expression networks



Networks and standard clinical lipid measures

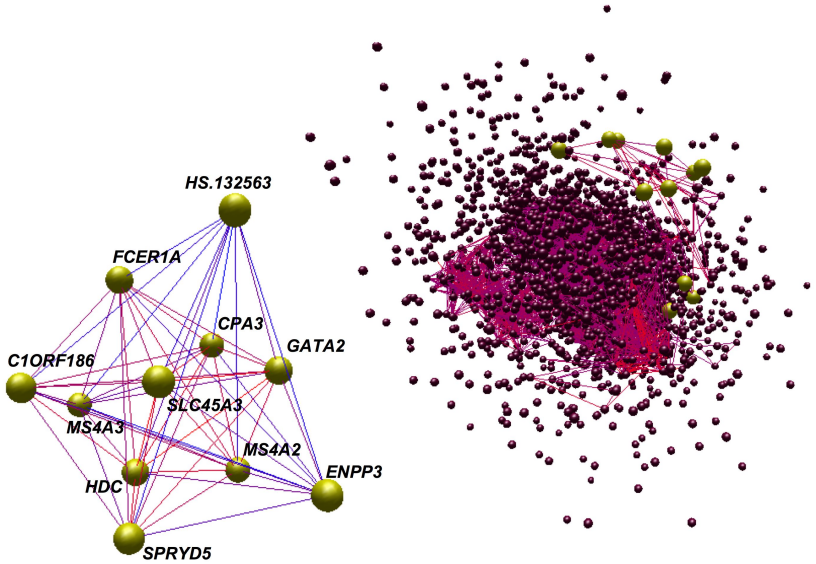
LL module appears to be involved in immune response

Genes

- ***FCER1A*** – high affinity IgE receptor
- ***MS4A2*** – high affinity IgE receptor
- ***HDC*** – enzyme for histamine synthesis
- ***CPA3*** – mast cell secreted peptidase
- ***GATA2*** – TF crucial for mast cell dev
- ***SLC45A3*** - *?*
- ***SPRYD5*** - *?*
- ***MS4A3*** - *?*
- ***ENPP3*** - *?*
- ***C1ORF186*** - *?*
- ***HS.132563*** - *?*

Immune markers

- IL-1ra (P=3.1x10^{-6})
- C-reactive protein (P=2.6x10^{-4})
- HMW adiponectin (P=1.6x10^{-5})
- Total IgE (P>0.05)

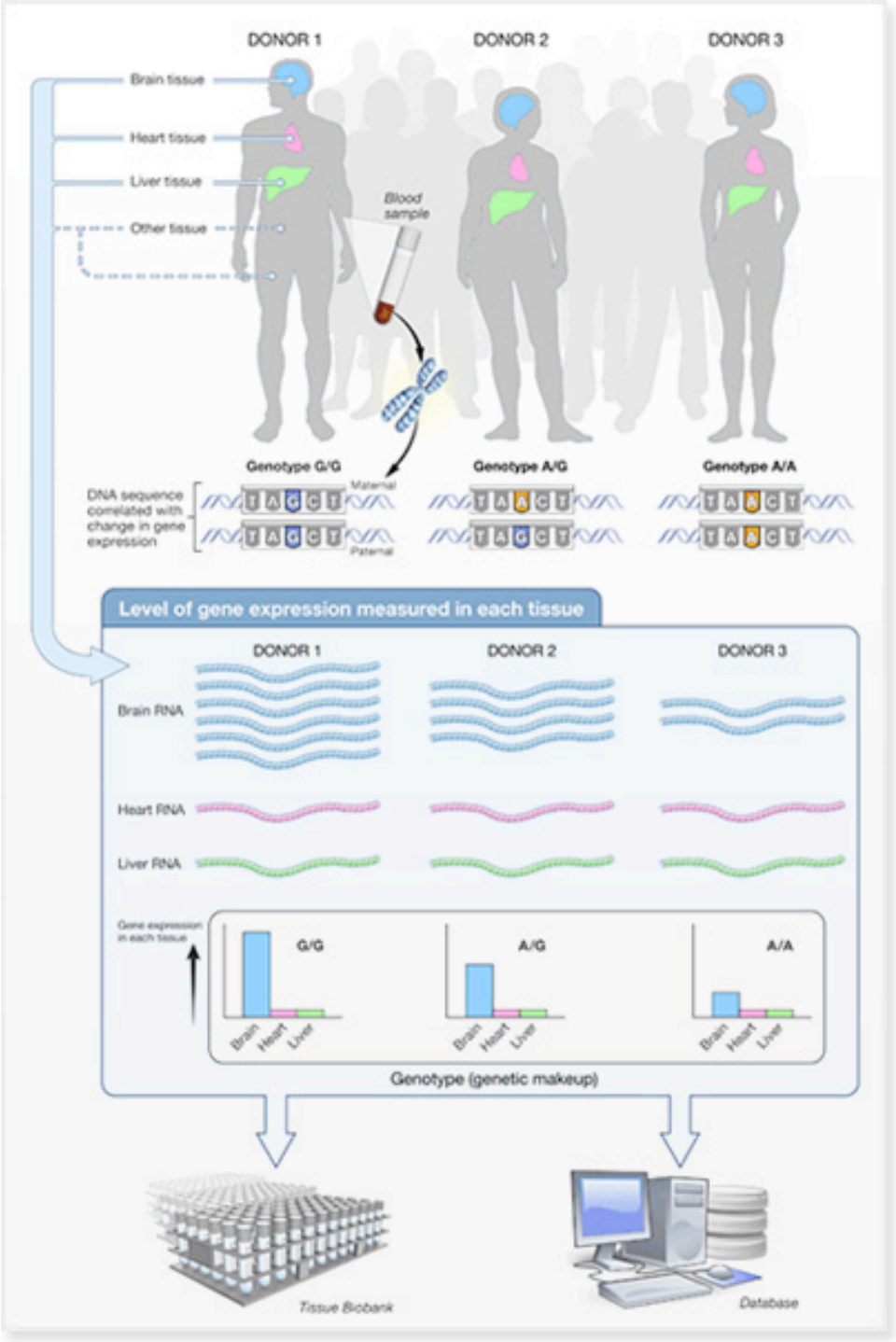


HUMAN GENOMICS

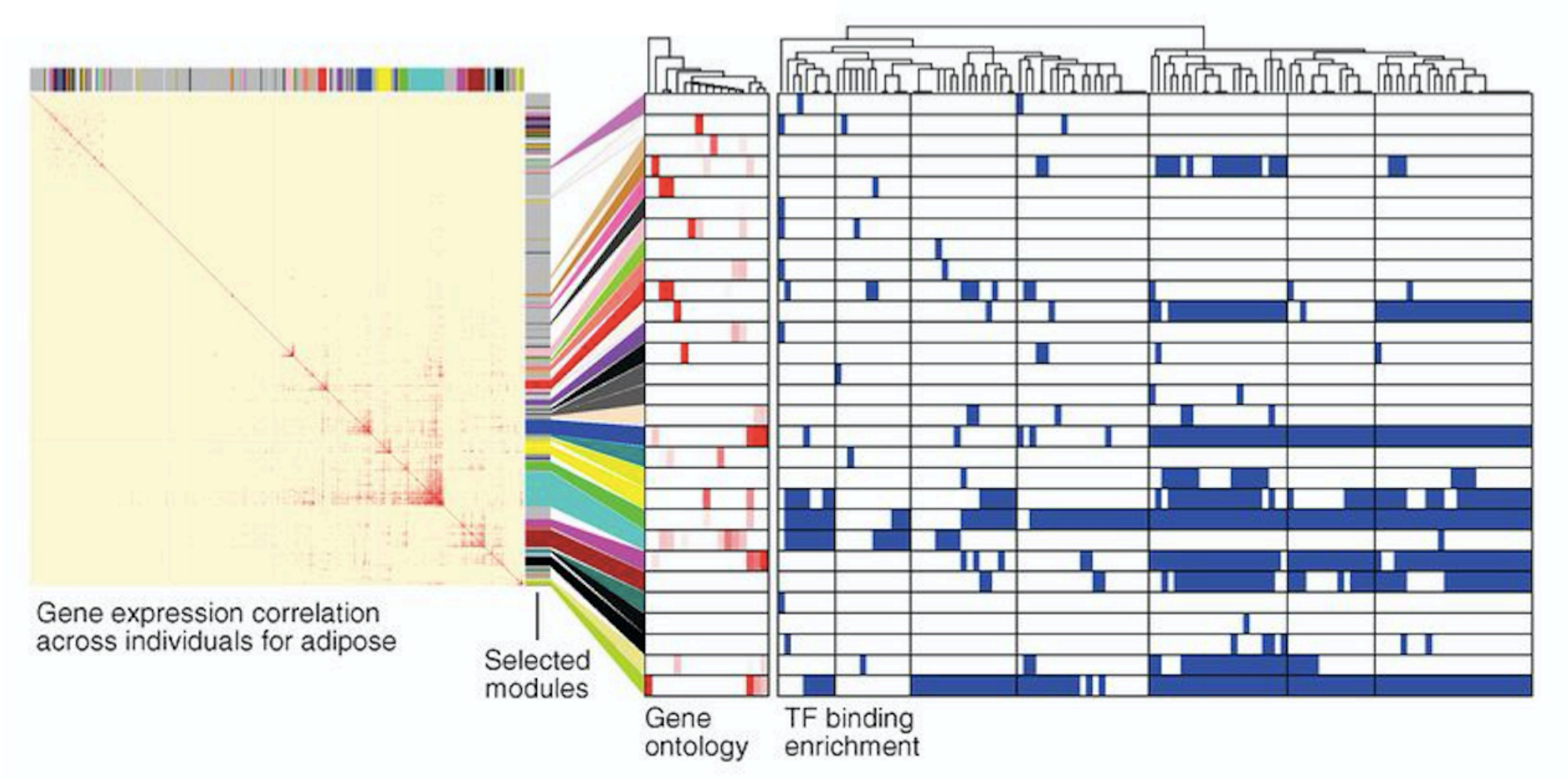
The Genotype-Tissue Expression (GTEx) pilot analysis: Multitissue gene regulation in humans

The GTEx Consortium*†

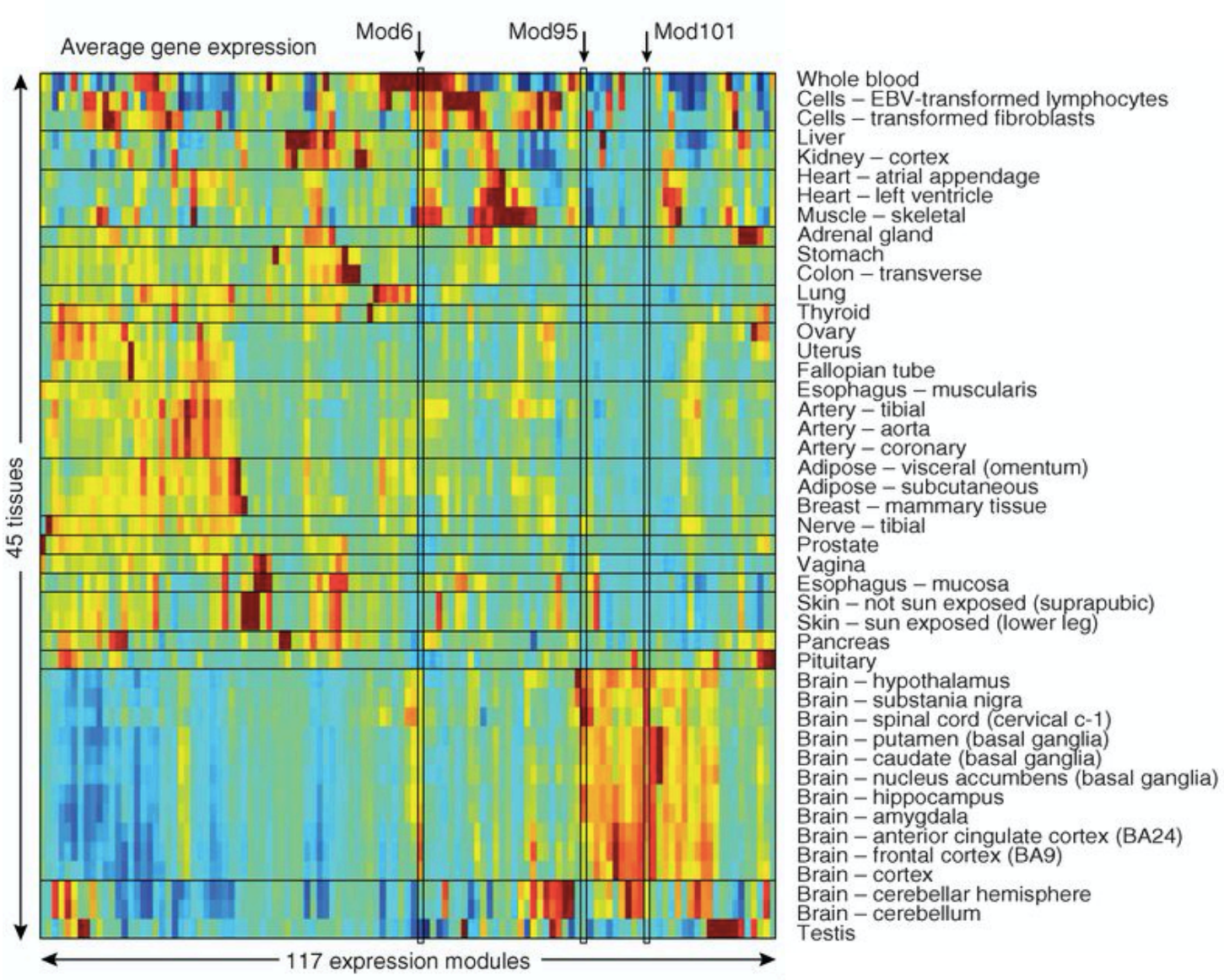
Science 8 MAY 2015 • VOL 348 ISSUE 6235



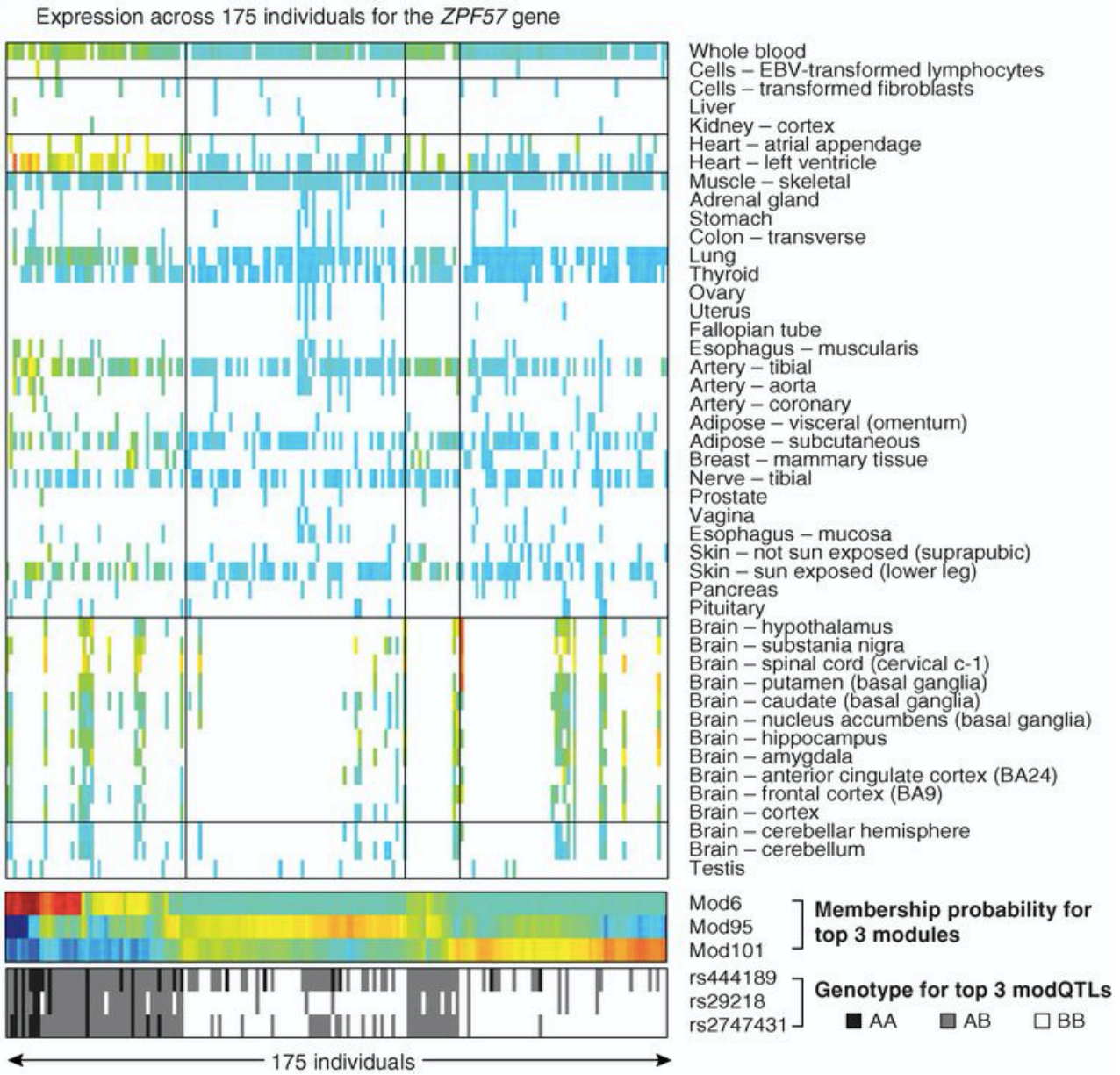
Adipose tissue: Differential pathway enrichment and TF binding profiles



Expression levels of modules across tissues



Expression of a gene (*ZPF57*) between tissues/genotypes



Preservation of subnets

- **Given a subnet (nodes, edges), is to preserved in a separate dataset?**

- **Examples**
  - Replication
    - Given N datasets generated under identical/similar settings, does a subnet 'replicate'?
  - Cross-tissue gene network preservation
    - Is a subnet derived from liver data preserved in adipose data?
  - Microbial communities between body sites
    - Is an operational taxonomic unit (OTU) subnet preserved between skin and upper airway samples?

Approaches to subnet preservation

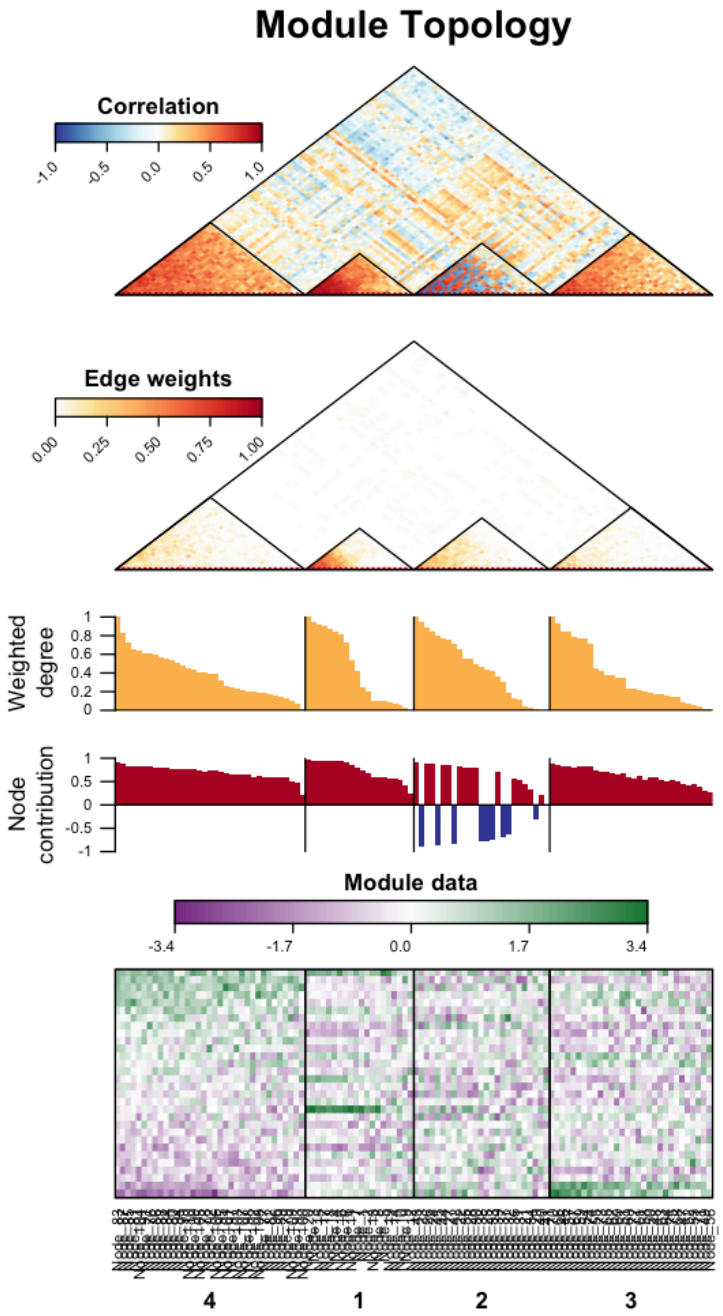
- **Tabulation**
  - Make a table of features in a given subnet and those not. Test for deviation from null (e.g. Fisher Exact Test).

Dataset 2 subnet A \ Dataset 1 subnet A	IN	OUT
IN	a	b
OUT	c	d

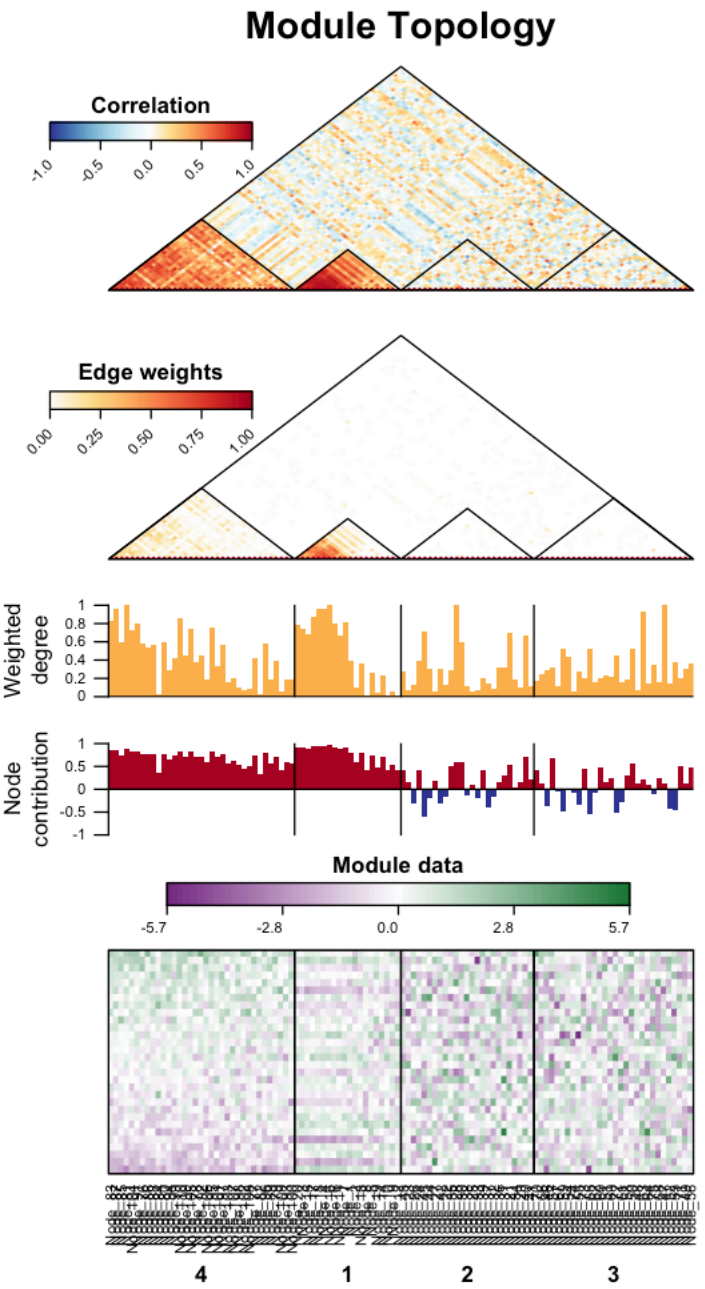
$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}}$$

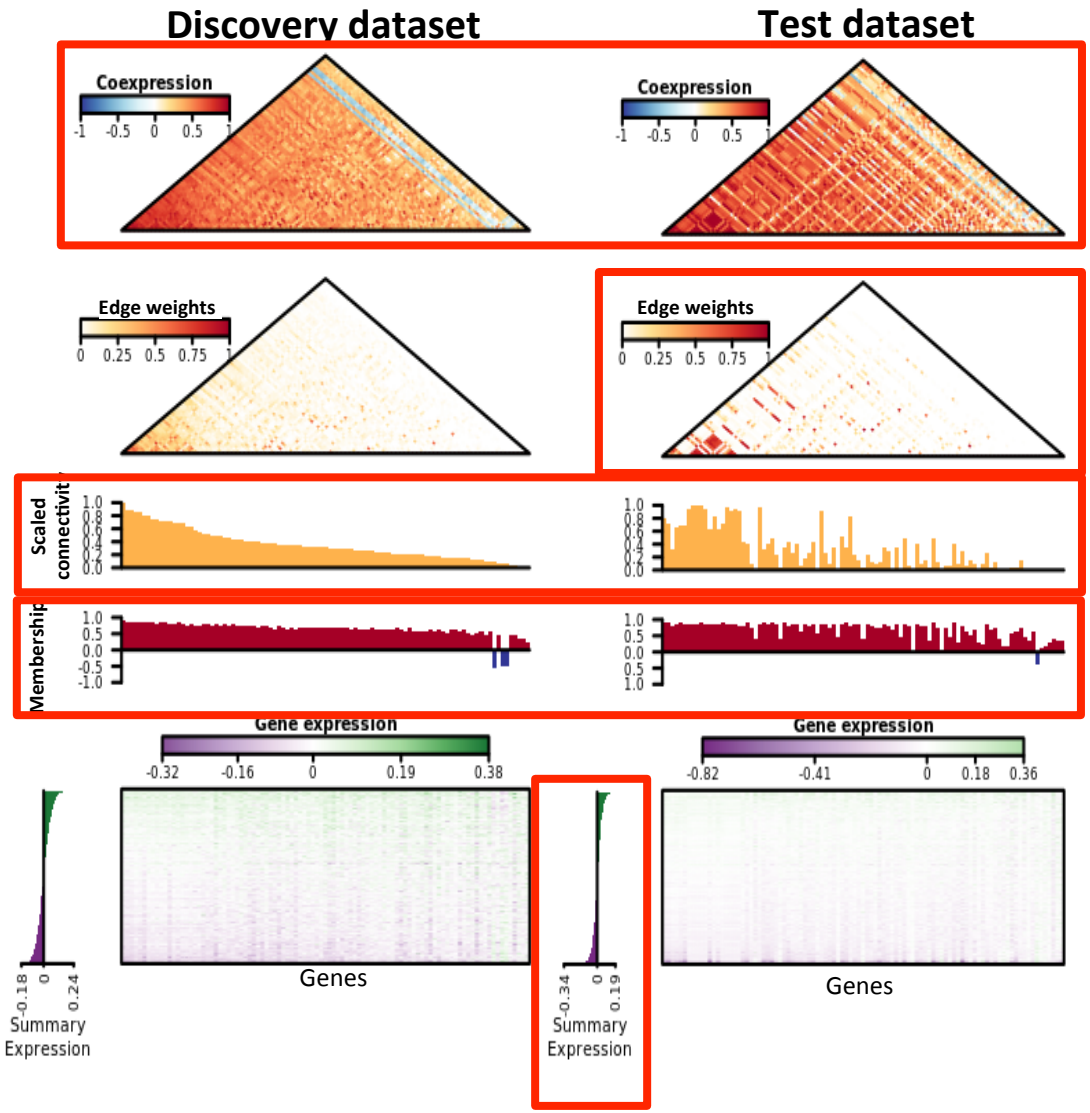
- **Topological properties**
  - Edge patterns (for simplicity, assume no missing nodes)

Dataset 1 (discovery)



Dataset 2 (replication)





Null Hypothesis:

Indistinguishable from comparisons to random gene sets in test dataset.

Module preservation statistics

How distinguishable is the module?

- Density / average edge weight
- Proportion of variance explained

How similar is the module topology?

- Similarity of correlation structure
- Correlation of connectivity / degree
- Correlation of membership / contribution

Combination:

- Mean correlation structure
- Average membership / contribution

Preservation of topology

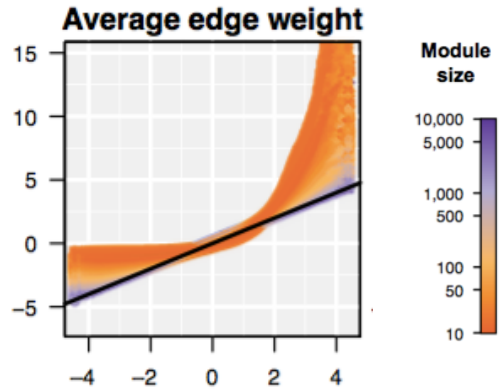
- Langfelder & Horvath, *PLOS Comp Bio* 2011
- Ritchie et al, *Cell Systems* 2016

	General name of test statistic	WGCNA	Calculation
(1)	Module coherence	Proportion of variance explained	$mean\left(\left(cor\left(g_i^{[t](w)}, Eig_1^{[t](w)}\right)\right)^2\right)$
(2)	Average node contribution	Mean sign-aware module membership	$mean\left(sign\left(cor\left(g_i^{[d](w)}, Eig_1^{[d](w)}\right)\right) \cdot cor\left(g_i^{[t](w)}, Eig_1^{[t](w)}\right)\right)$
(3)	Concordance of node contributions	Correlation of module membership	$cor\left(cor\left(g_i^{[d](w)}, Eig_1^{[d](w)}\right), cor\left(g_i^{[t](w)}, Eig_1^{[t](w)}\right)\right)$
(4)	Density of correlation structure	Mean sign-aware coexpression	$mean\left(sign\left(C^{[d](w)}\right) \cdot C^{[t](w)}\right)$
(5)	Concordance of correlation structure	Correlation of coexpression	$cor_{i \neq j}\left(C^{[d](w)}, C^{[t](w)}\right)$
(6)	Average edge weight	Mean adjacency	$mean_{i \neq j}\left(a_{ij}^{[t](w)}\right)$
(7)	Concordance of weighted degree	Correlation of intramodular connectivities	$cor\left(\left(\sum_{i \neq j}^{j} a_i\right)^{[d](w)}, \left(\sum_{i \neq j}^{j} a_i\right)^{[t](w)}\right)$

a	edge weight
g	feature vector
cor	correlation
C	correlation matrix
Sign	+ / -
Eig	1st principal component

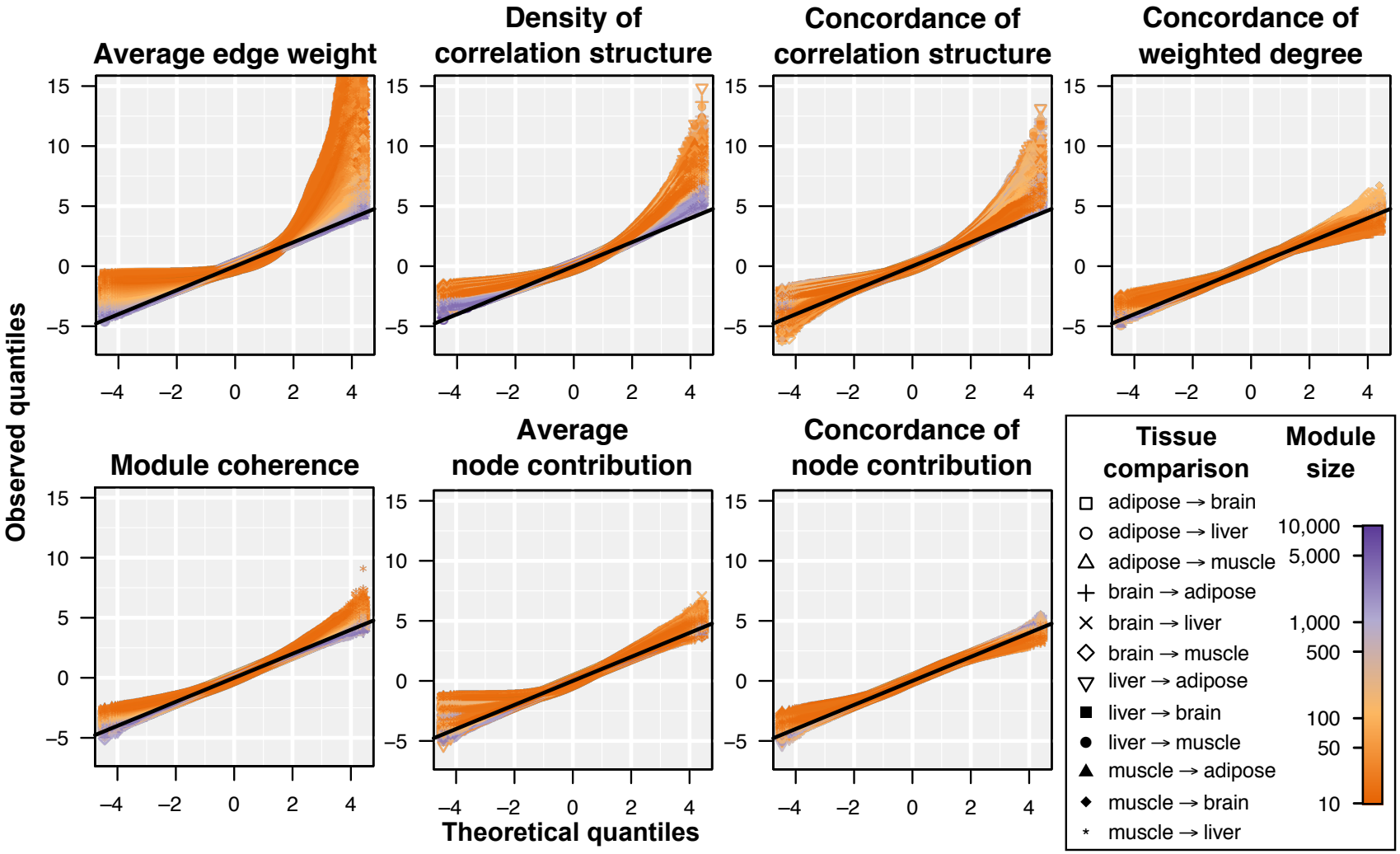
When in doubt, permute the data

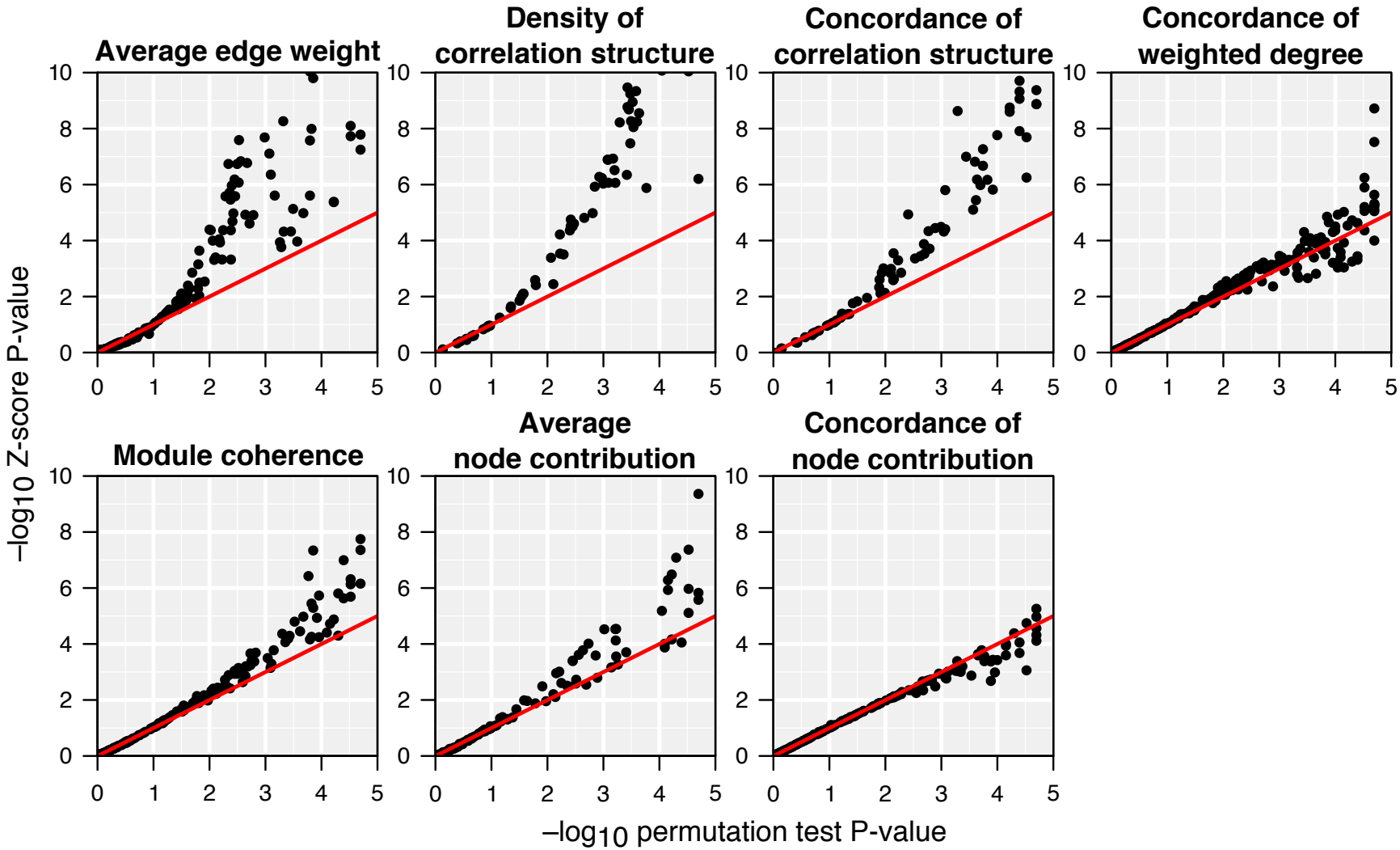
- **In network analysis, the complex relationships amongst nodes can make it difficult to assume a given test statistic follows a particular distribution**



- **It is common (and good practice) to create an empirical (permuted) distribution of the test statistic to assess the original observation's significance**

- **E.g. for a given module of with M nodes, with a given test statistic…**
  - Randomly draw M nodes from the overall network
  - Compute the test statistic of these random M nodes
  - Repeat many times
  - Compare the observed module value to the distribution of permuted values


Average edge weight
Density of correlation structure
Concordance of correlation structure
Concordance of weighted degree
Module coherence
Average node contribution
Concordance of node contribution
Observed quantiles
Theoretical quantiles
15
10
5
0
−5
−4
−2
0
2
4
Tissue comparison
adipose → brain
adipose → liver
adipose → muscle
brain → adipose
brain → liver
brain → muscle
liver → adipose
liver → brain
liver → muscle
muscle → adipose
muscle → brain
muscle → liver
Module size
10,000
5,000
1,000
500
100
50
10


Average edge weight
Density of correlation structure
Concordance of correlation structure
Concordance of weighted degree
Module coherence
Average node contribution
Concordance of node contribution
–log10 Z-score P-value
–log10 permutation test P-value
0
1
2
3
4
5
0
2
4
6
8
10

Effect of scale-free'edness on preservation

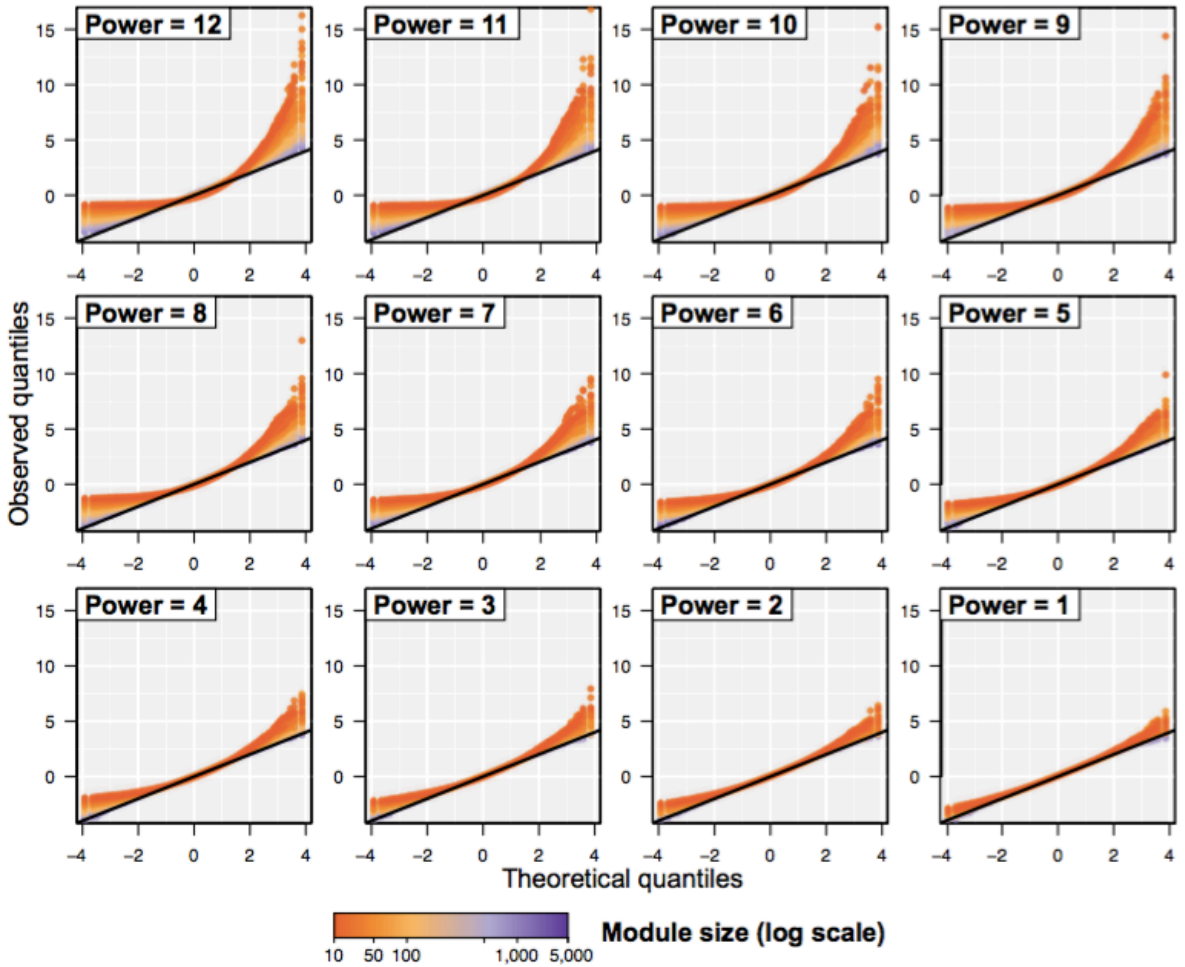
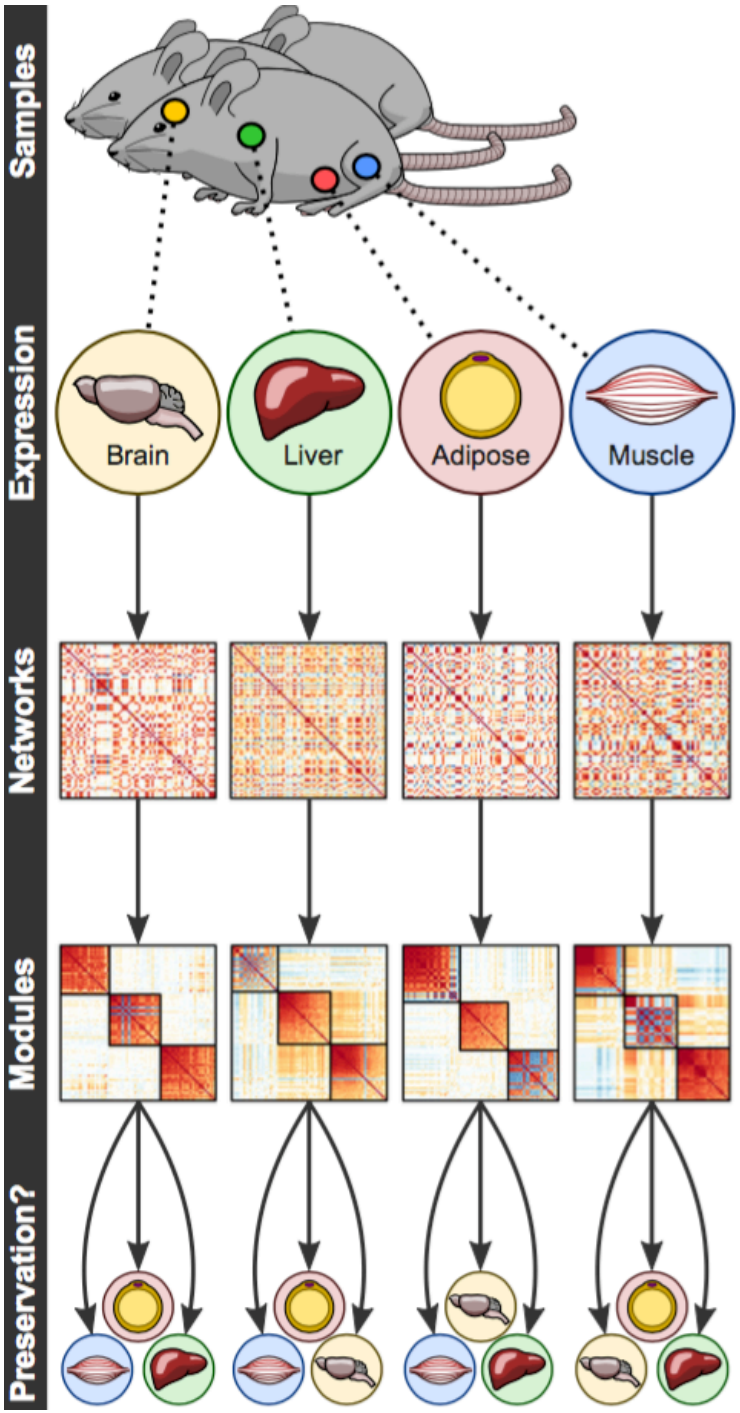
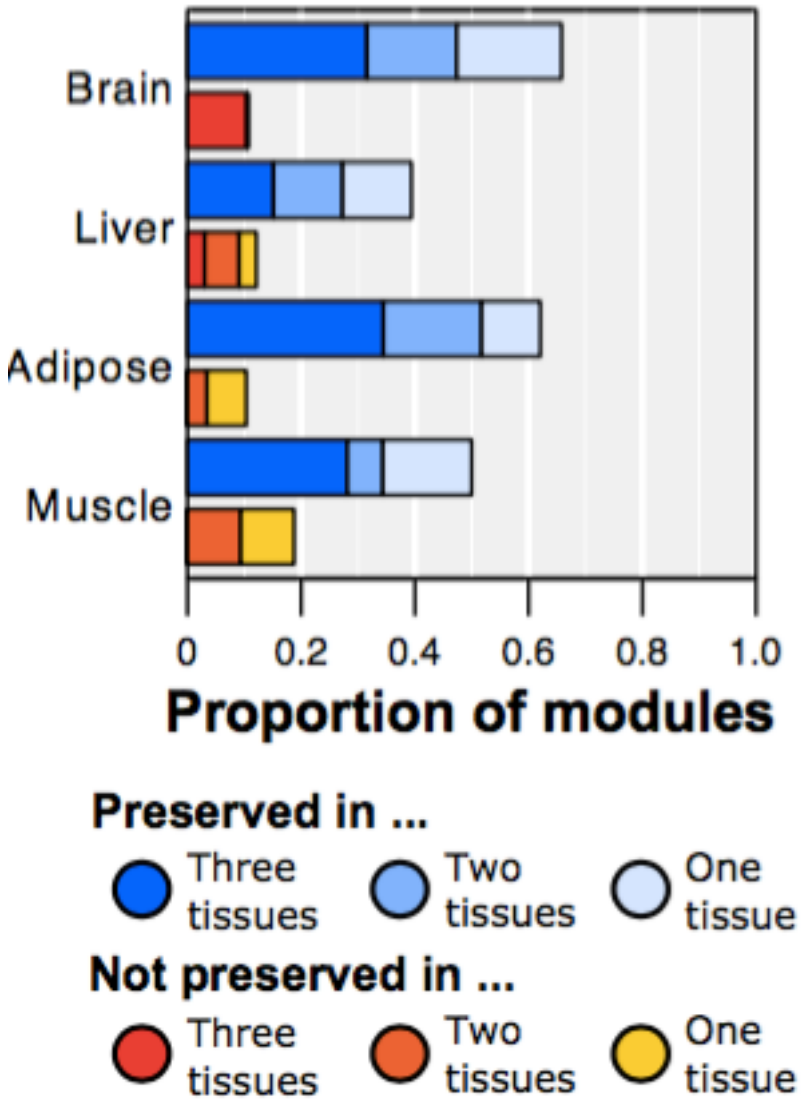
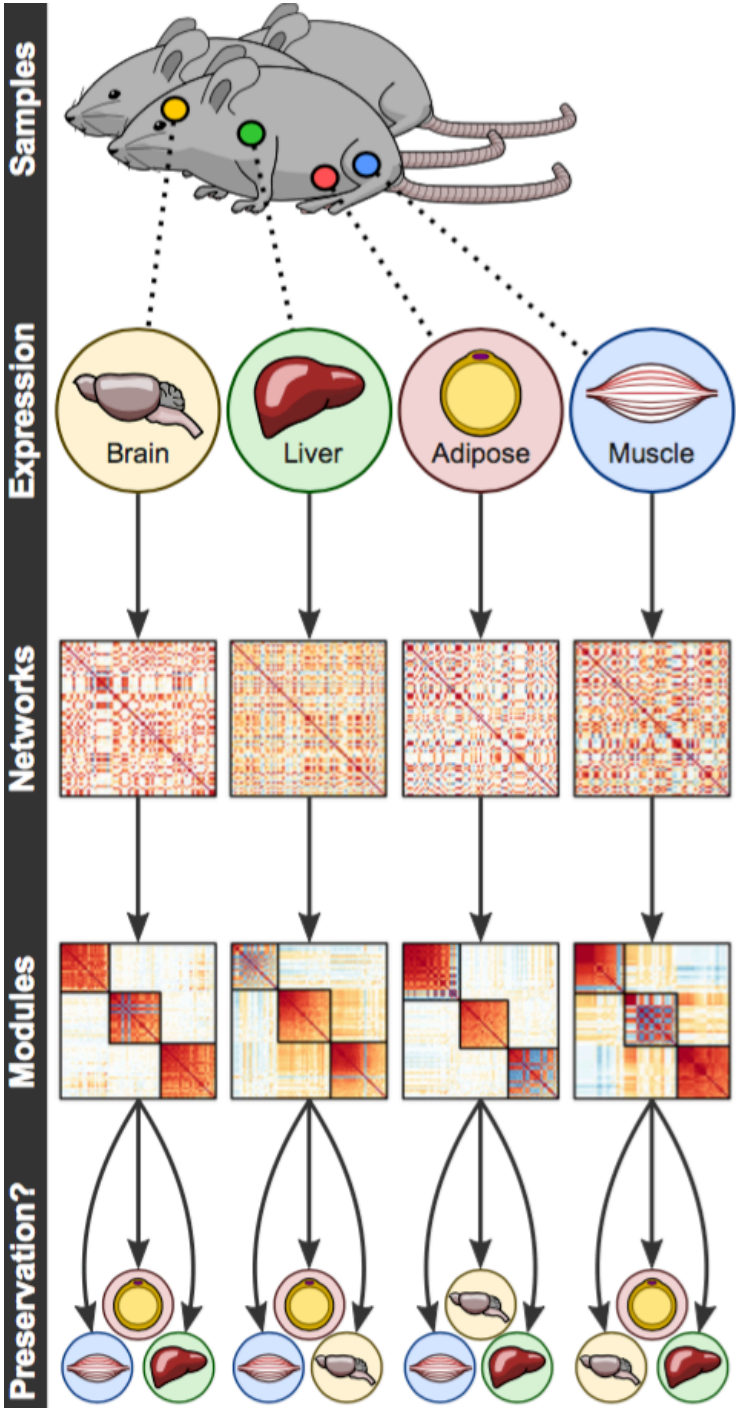


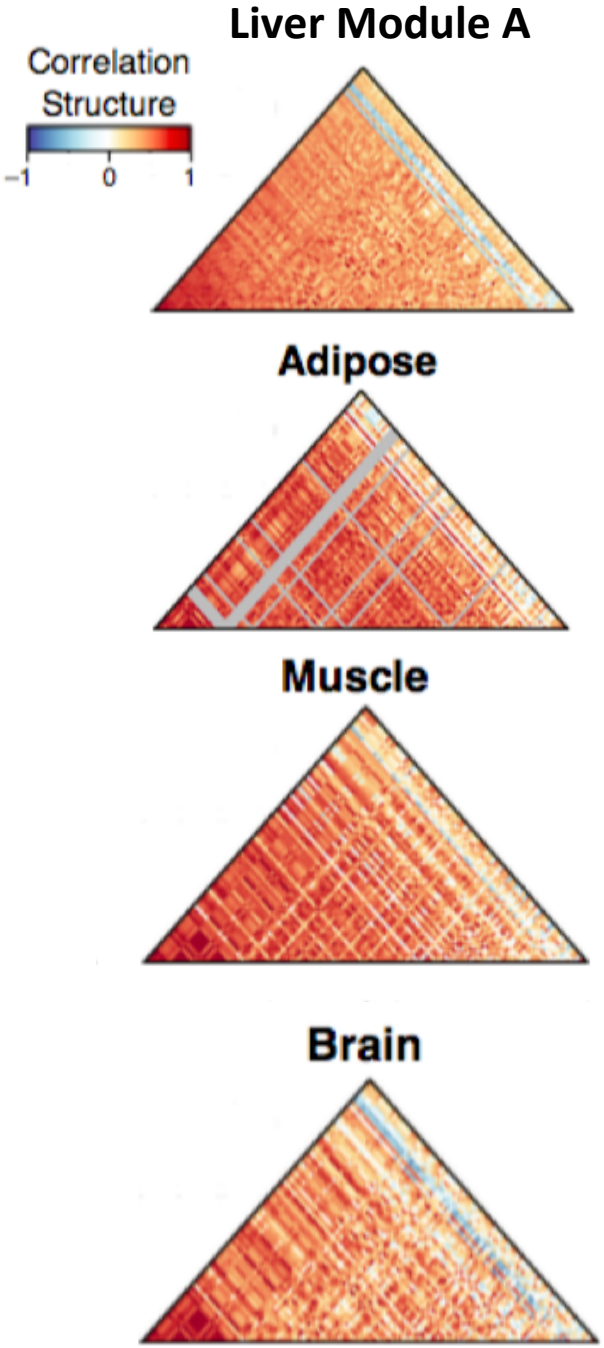
Figure S4, related to the experimental procedures and the main text: The scale-free assumption affects non-normality of the *average edge weight* statistic. Quantile-Quantile plots comparing





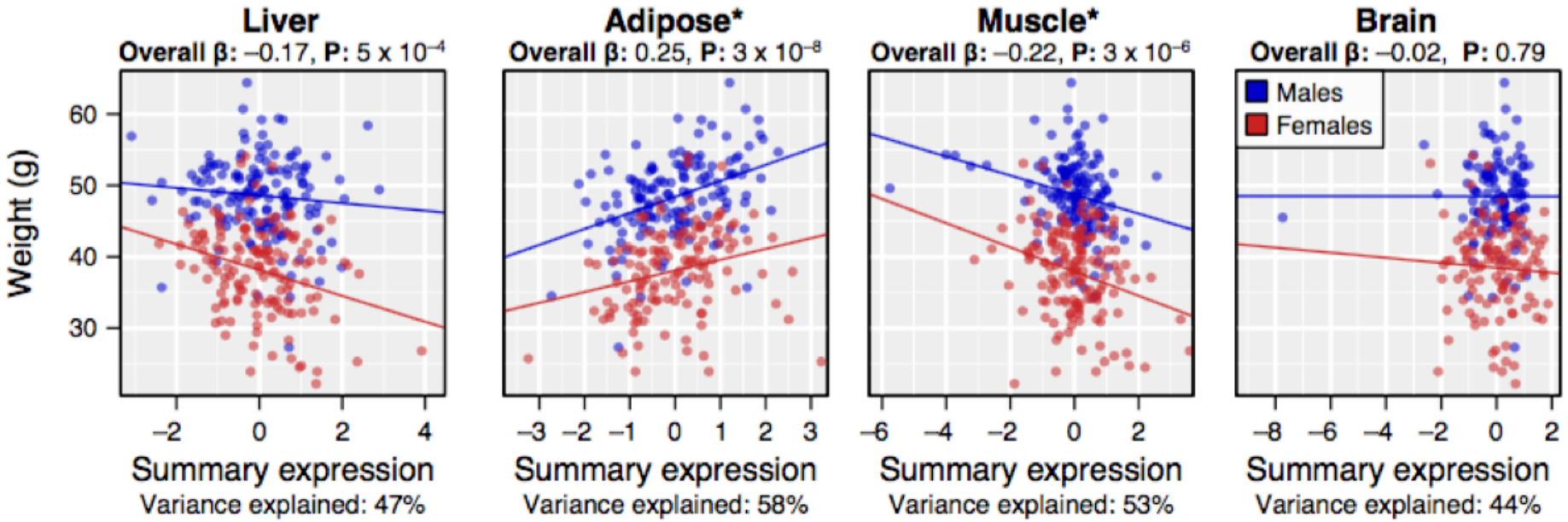
Ritchie et al, *Cell Systems* 2016


Samples
Expression
Networks
Modules
Preservation?
Brain
Liver
Adipose
Muscle


Liver Module A
Correlation Structure
−1
0
1
Adipose
Muscle
Brain

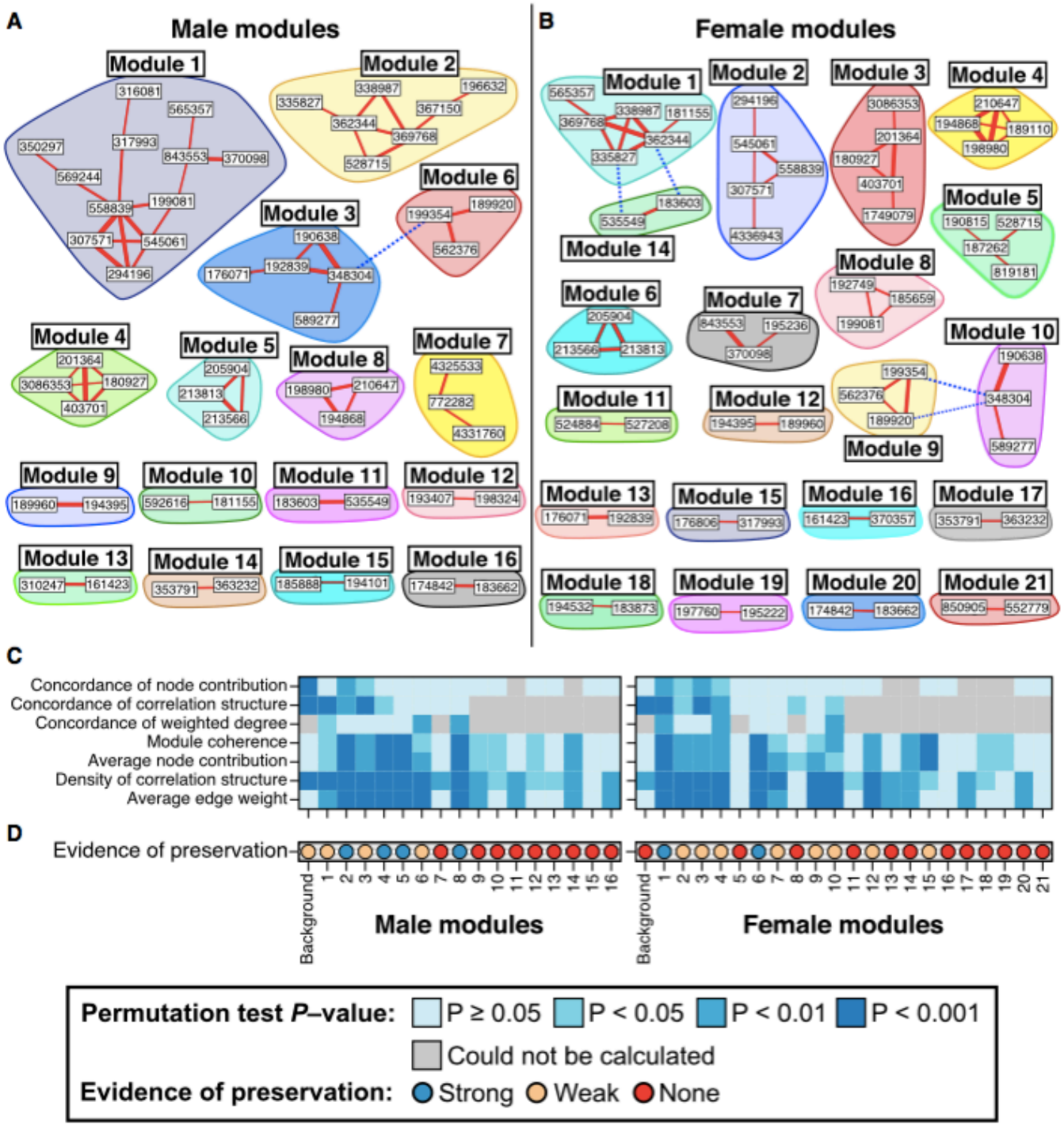
Gene 1 … n

Phenotypic association (body weight)

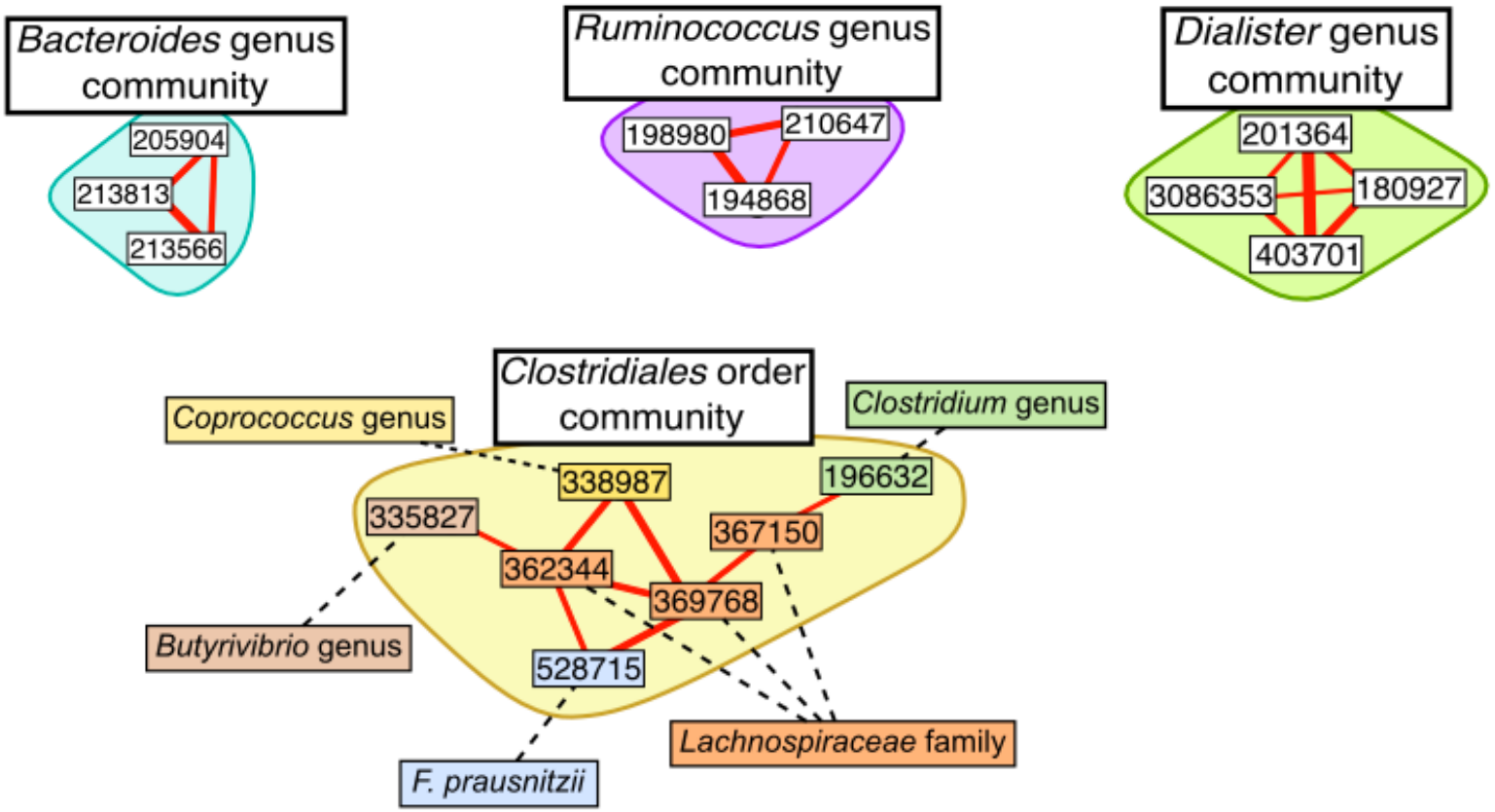


Test tissue	Trait	Effect size	95% confidence interval	*P*-value	*Q*-value
Adipose	**Weight**	**0.25**	**0.16– 0.33**	**3×10^{-8}**	**-**
	Insulin	**0.23**	**0.14– 0.32**	**1×10^{-6}**	**2×10^{-5}**
	Glucose/Insulin	**-0.21**	**-0.30– -0.12**	**7×10^{-6}**	**7×10^{-5}**
	Other fat	0.23	0.11– 0.35	1×10^{-4}	8×10^{-4}
	Total fat	**0.19**	**0.081– 0.30**	**7×10^{-4}**	**0.004**
	Length	0.17	0.069– 0.27	0.001	0.004
	MCP-1 (CCL2)	0.18	0.064– 0.29	0.002	0.007
	Glucose	0.18	0.064– 0.30	0.003	0.007
	Unesterified cholesterol	**0.18**	**0.061– 0.29**	**0.003**	**0.007**
Muscle	**Weight**	**-0.21**	**-0.30– -0.13**	**3×10^{-6}**	**-**
	Unesterified cholesterol	**-0.21**	**-0.34– -0.092**	**6×10^{-4}**	**0.01**
	Insulin	**-0.16**	**-0.25– -0.061**	**0.001**	**0.01**
	Total fat	**-0.19**	**-0.31– -0.072**	**0.002**	**0.01**
	Abdominal fat	-0.17	-0.27– -0.061	0.002	0.01
	Glucose/Insulin	**0.14**	**0.048– 0.24**	**0.003**	**0.01**
	Free fatty acids	-0.18	-0.31– -0.059	0.004	0.01
	LDL+VLDL	-0.18	-0.30– -0.056	0.005	0.01
	HDL/LDL+VLDL	0.17	0.051– 0.29	0.005	0.01
	Total cholesterol	-0.17	-0.29– -0.049	0.006	0.01

Gut 16S microbial community networks (SparCC)

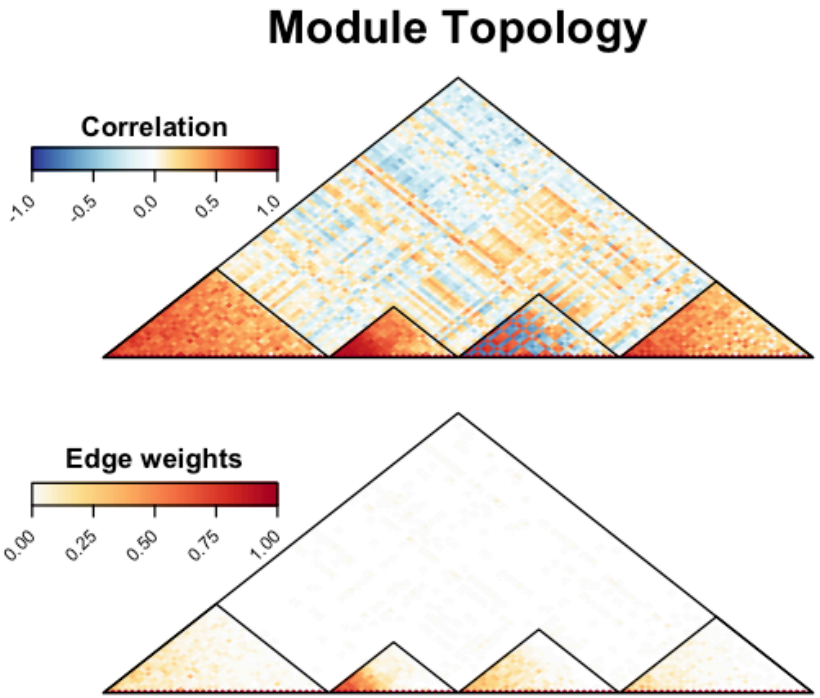


Microbiome communities present in both men and women



Gene co-expression networks

- **Weighted, undirected complete gene network**
  - **Nodes**: genes/probes
  - **Edges**: |cor(node_i, node_j)|$^{\gamma}$
    - Scale-free assumption and [0,1]
- **Identify subnets (modules/ clusters)**
  - Typically subnets represent known biological pathways
  - Various methods and tools for clustering



What we're doing today

- Data management and filtering
- Network construction
- Module detection
- Module association analysis

Getting started
(if you haven't already done so)

Setup

First, we installed the WGCNA package. Some its dependencies are in the BioConductor repository rathern than CRAN. We needed to install these dependencies manually, because `install.packages` will not do it for us:

```
source("https://bioconductor.org/biocLite.R")
# You can answer no ('n') to any prompt that asks to update old packages.
biocLite(c("impute", "preprocessCore", "GO.db", "AnnotationDbi"))

# Now we can install WGCNA from CRAN.
install.packages("WGCNA")
```

Data filtering

First, we will loaded in mouse adipose gene expression data (adapted from Yang X et al, *Genome Res.* 2006 Aug; 16(8): 995–1004, full multi-tissue set freely available from Sage BioNetworks).

```
read.matrix <- function(file) {
  df <- read.table(file, header=TRUE, row.names=1, sep="\t",
                   check.names=FALSE)
  mat <- as.matrix(df)
  # Avoid having numeric IDs
  rownames(mat) <- paste0("Probe_", rownames(mat))
  colnames(mat) <- paste0("Sample_", colnames(mat))
  return(mat)
}

setwd("/Users/minouye/Documents/Courses_Workshops/SISG/2016/prac_WGCNA/
curatedExpressionLiver")

liver_ge <- read.matrix("expression_head2500.txt")
```

Next, we removed remove probes that failed for >5% of samples, and then samples where >5% of their assays failed.

```
# Removes probes that have more the 5% of their observations missing
filter_probes <- function(x) {
  # How many samples are missing for each probe?
  nMissing <- apply(x, 2, function(probe) {
    sum(is.na(probe))
  })
  missingness <- nMissing/nrow(x)
  return(x[, missingness <= 0.05])
}

# Removes samples that failed > 5% of their assays
filter_samples <- function(x) {
  # How many samples are missing for each probe?
  nMissing <- apply(x, 1, function(sample) {
    sum(is.na(sample))
  })
  missingness <- nMissing/ncol(x)
  return(x[missingness <= 0.05,])
}

liver_ge_fil <- filter_samples(filter_probes(liver_ge))
```

Then we imputed the remaining missing observations using the K-nearest neighbours algorithm in the *impute* package:

```
# This package is a dependency of WGCNA so we have installed it already.
library(impute)

if (any(is.na(liver_ge_fil)))
  liver_ge_fil_imp <- impute.knn(liver_ge_fil)$data

anyNA(liver_ge_fil)
anyNA(liver_ge_fil_imp)
```

To reduce the burden of computation for the purposes of software testing we will only analyse the top few thousand most variable and most connected probes. This is standard practice when performing weighted gene coexpression network analysis on computers with limited resources (Ghazalpour *et al.*, 2006).

Following Ghazalpour *et al.*, first we first get the top 1,000 most varying probes in the liver tissue:

```
most_varying <- function(ge, topN=1000) {
  standard_deviation <- apply(ge, 1, sd)
  sorted <- sort(standard_deviation, decreasing=TRUE)
  sorted_names <- names(sorted)
  topN_names <- sorted_names[seq_len(topN)]
  return(topN_names)
}

top_varying <- most_varying(liver_ge_fil_imp)
liver_ge_fil_imp_top1000 <- liver_ge_fil_imp[top_varying,]
```

Network inference

Next we will infer the interaction networks. If we want to use *NetRep*, we need to save both the correlation structure (coexpression), as well as the interaction network (adjacency matrix) inferred through *WGCNA*:

```
library(WGCNA)
## Warning: package 'WGCNA' was built under R version 3.2.3
calculate_coexpression <- function(ge) {
  coexpression <- cor(t(ge), method="pearson")
}

infer_network <- function(coexpression) {
  # First pick the soft threshold to use to define the interaction network
  sft <- WGCNA::pickSoftThreshold(abs(coexpression), dataIsExpr=FALSE)
  if (is.na(sft$powerEstimate)) {
    sft$powerEstimate <- 1
    warning("Could not satisfy the scale-free topology criterion")
  }
  network <- abs(coexpression)^sft$powerEstimate
}

liver_coexpression <- calculate_coexpression(liver_ge_fil_imp_top1000)
liver_network <- infer_network(liver_coexpression)
```

What's it doing?

- Calculate Pearson correlation coefficients between all pairs of genes
- Use a power transform to satisfy scale-free topology criteria (select soft power threshold)
- Infer a network where
  - Nodes: Genes
  - Edges: Pearson correlations raised to the selected power

Next we will identify tightly coexpressed modules in the liver tissue network.

```
detect_modules <- function(ge, network) {
  # Calculate the distance between probes based on their topological similarity:
  # i.e. the strength of their coexpression as well as the similarity of their
  # patterns of coexpression to all other probes
  probe_dist <- WGCNA::TOMdist(network)
  dimnames(probe_dist) <- dimnames(network)

  # Hierarchically cluster based on this distance metric
  dendro <- hclust(as.dist(probe_dist), method="average")

  # Detect modules. `cutreeDynamic` is a function in the `dynamicTreeCut`
  # package, which is loaded in by the `WGCNA` package.
  module_labels <- cutreeDynamic(dendro, distM = probe_dist)
  names(module_labels) <- colnames(network)

  # Merge similar modules
  merged <- mergeCloseModules(t(ge), module_labels, relabel=TRUE)
  module_labels <- merged$colors

  return(module_labels)
}

liver_modules <- detect_modules(liver_ge_fil_imp_top1000, liver_network)
```

What's it doing?

- Goal: Get the most coherent gene subnetworks as possible
- Instead of using the correlation-based edges, WGCNA is calculating a distance measure called topological similarity (TOM):

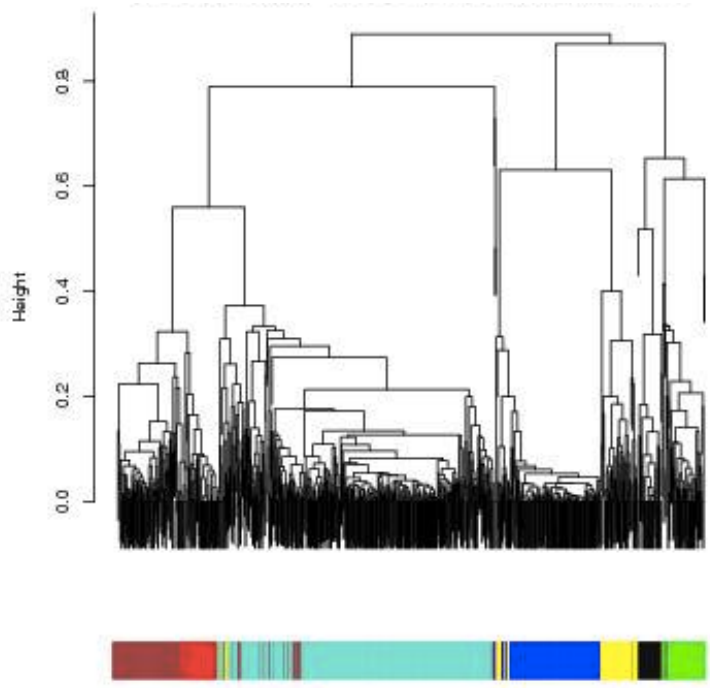
$$t_{ij} = \begin{cases} \dfrac{|N_1(i) \cap N_1(j)| + a_{ij}}{\min\{|N_1(i)|, |N_1(j)|\} + 1 - a_{ij}} & \text{if } i \neq j \\ 1 & \text{if } i = j. \end{cases} \quad (1)$$

where $N_1(i)$ denotes the set of direct neighbors of i excluding i itself and $|\cdot|$ denotes the number of elements (cardinality) in its argument. The quantity $|N_1(i) \cap N_1(j)|$ measures the number of common neighbors that nodes i and j share whereas $|N_1(i)|$ gives the number of neighbors of i. The topological overlap t_{ij} assumes a minimal value of 0 if there is no direct linkage between the two nodes and if they share no common direct neighbors. It assumes a maximum value of 1 if there is a direct link between the two nodes and if one set of direct neighbors is a subset of the other. The fact that t_{ij} is bounded between 0 and 1 is used in the definition of the topological overlap based dissimilarity measure (see Eq. 4).

Yip & Horvath, BMC Bioinf 2007

What's it doing?

- Hierarchical clustering of TOM matrix
- Move through the dendrogram with a dynamic cutting algorithm



Yip & Horvath, BMC Bioinf 2007

Each probe is now assigned a (numeric) module label:

```
table(liver_modules)
## liver_modules
##   0   1   2   3   4   5   6   7
## 395 168 166  93  61  59  32  26
```

Where “0” corresponds to the network “background”: all genes that did not cluster into coexpression module.

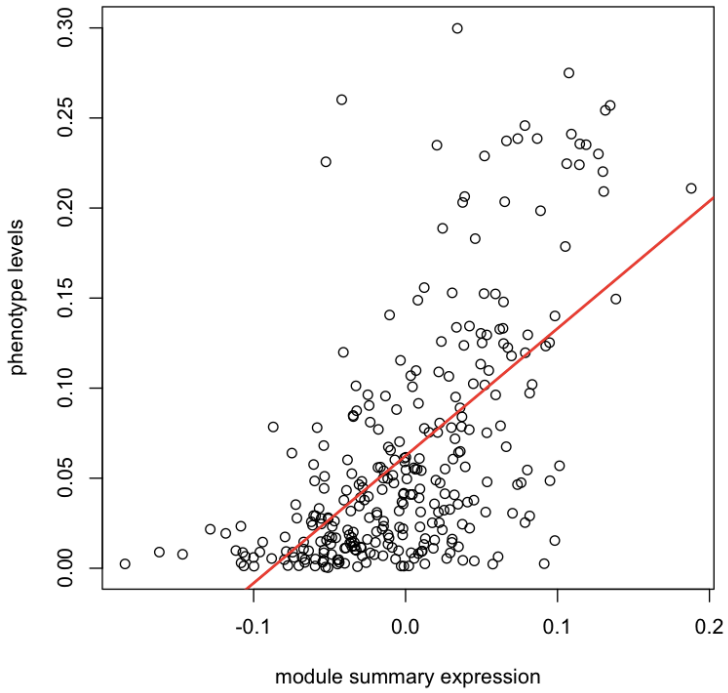
Saving the data

Finally, we will save the data for processing with *NetRep*:

```
# Create the directory to store the data in.
dir.create("test_data")

write.matrix <- function(x, file) {
  write.csv(x, file, quote=FALSE)
}

write.vector <- function(x, file, col.names) {
  column_matrix <- t(t(x))
  colnames(column_matrix) <- col.names
  write.csv(column_matrix, file, quote=FALSE)
}

# NetRep requires the probes to be columns, so we will transpose when
# saving
write.matrix(t(liver_ge_fil_imp_top1000), "test_data/liver_expression.csv")

write.matrix(liver_coexpression, "test_data/liver_coexpression.csv")

write.matrix(liver_network, "test_data/liver_network.csv")

write.vector(liver_modules, "test_data/liver_modules.csv", col.names="Module")
```

Phenotype association analysis

```
MEs <- moduleEigengenes(t(liver_ge_fil_imp_vary), liver_modules)

names(MEs$varExplained) <- colnames(MEs$eigengenes)
MEs$varExplained

# add in sample names to the eigengenes.
rownames(MEs$eigengenes) <- colnames(liver_ge_fil_imp_vary)

pheno<-read.table("pheno.txt", header=TRUE)
data<-cbind(pheno,MEs$eigengenes)

plot(data$ME0,data$pheno)

plot(data$ME6,data$pheno,xlab="module summary expression",ylab="phenotype levels",pch=20)
summary(lm(pheno ~ ME6, data=data))
abline(intercept,slope,col="red",lwd=2)
```



References

Ghazalpour, A., Doss, S., Zhang, B., Wang, S., Plaisier, C., Castellanos, R., Brozell, A., Schadt, E.E., Drake, T.A., Lusis, A.J., et al. (2006). Integrating genetic and network analysis to characterize genes related to mouse weight. PLoS Genet. *2*, 1182–1192.

Langfelder, P., and Horvath, S. (2008). WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics *9*, 559.

Yang, X., Schadt, E.E., Wang, S., Wang, H., Arnold, A.P., Ingram-Drake, L., Drake, T.A., and Lusis, A.J. (2006). Tissue-specific expression and regulation of sexually dimorphic genes in mice. Genome Res. *16*, 995– 1004.

Special thanks to Scott Ritchie
Network inference adapted from his script