

Network Concepts and their geometric Interpretation
R Tutorial
Motivational Example:
weighted gene co-expression networks
in different gender/tissue combinations

Jun Dong, Steve Horvath

Correspondence: shorvath@mednet.ucla.edu, <http://www.ph.ucla.edu/biostat/people/horvath.htm>

This is a self-contained R software tutorial that illustrates how to compute network concepts and their eigengene based analogues. This R tutorial shows how we arrived at Figure 1 and Table 3 in Horvath and Dong (2008).

The microarray data sets correspond to gene expression measurements in the mouse tissues of male and female mice of an F2 mouse cross. Some familiarity with the R software is desirable but the document is fairly self-contained.

To cite the methods and results of this article, please use the following references

- *Horvath S, Dong J (2008) Geometric Interpretation of Gene Coexpression Network Analysis. PLoS Comput Biol 4(8): e1000117*
- *Dong J, Horvath S (2007) Understanding Network Concepts in Modules, BMC Systems Biology 2007, 1:24*
- *The WGCNA R package is described in: Langfelder P, Horvath S (2008) WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics, 2008 9:559*

Background: Weighted gene co-expression network construction.

Since microarray data can be noisy and the number of samples is often small, we and others have found it useful to emphasize strong correlations and to punish weak correlations. It is natural to define the adjacency between 2 genes as a high power of the absolute value of the correlation coefficient [Zhang and Horvath 2005, Horvath et al 2006]:

$$a_{ij} = |cor(x_i, x_j)|^\beta, \quad (2)$$

with $\beta \geq 1$. This soft thresholding approach leads to a weighted gene co-expression network.

In this article, we are particularly interested in gene significance measures that are based on a microarray sample trait, which is also known as response or outcome. The microarray sample trait $T = (T_1, \dots, T_m)$ may be quantitative (e.g. body weight) or it may be binary (e.g. case control status). Since our goal is to provide a simple geometric interpretation of co-expression network analysis, we define the **trait-based gene significance measure** by raising the Pearson correlation between the i -th gene expression profile x_i and the clinical trait T to a power β :

$$GS_i = |cor(x_i, T)|^\beta. \quad (3)$$

Although any power β could be used in Equation 3, here we use the same power as in Equation 2 to facilitate a simple geometric interpretation.

Fundamental Network Concepts

Next it shows how to compute several important network concepts (see also Dong and Horvath 2007).

The connectivity (also known as degree) of the i -th gene is defined by

$$k_i = \sum_{j \neq i} a_{ij}. \quad (4)$$

The maximum connectivity is defined as

$$k_{max} = \max_j (k_j). \quad (5)$$

The scaled connectivity K_i of the i -th gene is defined by

$$K_i = \frac{k_i}{k_{max}}. \quad (6)$$

We define the **maximum adjacency ratio** of gene i as follows

$$MAR_i = \frac{\sum_{j \neq i} (a_{ij})^2}{\sum_{j \neq i} a_{ij}}.$$

The **line density** is defined as the mean off-diagonal adjacency and is closely related to the mean connectivity.

$$Density = \frac{\sum_i \sum_{j \neq i} a_{ij}}{n(n-1)} = \frac{S_1(k)}{n(n-1)} \approx \frac{S_1(k)}{(n)^2}, \quad (8)$$

where $k = (k_1, \dots, k_n)$ denotes the vector of connectivities and the function vector v is defined by $S_p(v) = \sum_i v_i^p$.

The normalized connectivity centralization, also known as degree centralization is given by

$$\begin{aligned} Centralization &= \frac{n}{n-2} \left(\frac{k_{max}}{n-1} - Density \right) \\ &\approx \frac{k_{max}}{n} - Density. \end{aligned} \quad (9)$$

The centralization is 1 for a network with star topology; by contrast, it is 0 for a network where each node has the same connectivity. The centralization index has been used to describe structural differences of metabolic networks.

The network **heterogeneity** measure is based on the variance of the connectivity. Authors differ on how to scale the variance. Our definition equals the coefficient of variation of the connectivity distribution, i.e.

$$Heterogeneity = \frac{\sqrt{\text{var}(k)}}{\text{mean}(k)} = \sqrt{\frac{nS_2(k)}{S_1(k)^2} - 1}. \quad (10)$$

This heterogeneity measure is invariant with respect to multiplying the connectivity by a scalar. Complex, scale-free networks tend to be very heterogeneous: while some ‘hub’ genes are highly connected, the majority of genes tend to have very few connections.

The **clustering coefficient** of gene i is a density measure of local connections, or ‘cliquishness’. Specifically,

$$ClusterCoef_i = \frac{\sum_{l \neq i} \sum_{m \neq i, l} a_{il} a_{lm} a_{mi}}{\left\{ \left(\sum_{l \neq i} a_{il} \right)^2 - \sum_{l \neq i} (a_{il})^2 \right\}}. \quad (11)$$

In unweighted networks, $ClusterCoef_i$ equals 1 if and only if all neighbors of i are also connected to each other.

To measure the association between connectivity and gene significance, we propose the following measure of **hub gene significance**.

$$HubGeneSignif = \frac{\sum_i GS_i K_i}{\sum_i (K_i)^2}.$$

Animal husbandry and physiological trait measurements.

C57BL/6J apoE null (B6.apoE^{-/-}) mice were purchased from the Jackson Laboratory (Bar Harbor, Maine, United States) and C3H/HeJ apoE null (C3H.apoE^{-/-}) mice were bred by backcrossing B6.apoE^{-/-} to C3H/HeJ for ten generations with selection at each generation for the targeted ApoE^{-/-} alleles on Chromosome 7. All mice were fed ad libitum and maintained on a 12-h light/dark cycle. F2 mice were generated by crossing B6.apoE^{-/-} with C3H.apoE^{-/-} and subsequently intercrossing the F1 mice. F2 mice were fed Purina Chow (Ralston-Purina Co., St. Louis, Missouri, United States)

At the time of euthanasia, all mice were weighed and measured from the tip of the nose to the anus. Fat depots, plasma lipids (free fatty acids and triglycerides), plasma high-density lipoprotein (HDL) cholesterol and total cholesterol, and plasma insulin levels were measured as previously described. Very low-density lipoprotein (VLDL)/LDL cholesterol levels were calculated by subtracting HDL cholesterol from total cholesterol levels. Plasma glucose concentrations were measured using a glucose kit (#315-100; Sigma, St. Louis, Missouri, United States). Plasma leptin, adiponectin, and MCP-1 levels were measured using mouse enzyme-linked immunoabsorbent (ELISA) kits (#MOB00, #MRP300, and MJE00; R&D Systems, Minneapolis, Minnesota, United States).

Microarray analysis.

RNA preparation and array hybridizations were performed at Rosetta Inpharmatics (Seattle, Washington, United States). The custom ink-jet microarrays used in this study (Agilent Technologies [Palo Alto, California, United States], previously described [3,45]) contain 2,186 control probes and 23,574 non-control oligonucleotides extracted from mouse Unigene clusters and combined with RefSeq sequences and RIKEN full-length clones. Mouse livers were homogenized and total RNA extracted using Trizol reagent (Invitrogen, Carlsbad, California, United States) according to manufacturer's protocol. Three µg of total RNA was reverse transcribed and labeled with either Cy3 or Cy5 fluorochromes. Purified Cy3 or Cy5 complementary RNA was hybridized to at least two microarray slides with fluor reversal for 24 h in a hybridization chamber, washed, and scanned using a laser confocal scanner. Arrays were quantified on the basis of spot

intensity relative to background, adjusted for experimental variation between arrays using average intensity over multiple channels, and fit to an error model to determine significance (type I error). Gene expression is reported as the ratio of the mean log10 intensity (mlratio) relative to the pool derived from 150 mice randomly selected from the F2 population.

Microarray data reduction.

In order to minimize noise in the gene expression dataset, several data-filtering steps were taken. First, preliminary evidence showed major differences in gene expression levels between sexes among the F2 mice used, and therefore only female mice were used for network construction. The construction and comparison of the male network will be reported elsewhere. Only those mice with complete phenotype, genotype, and array data were used. This gave a final experimental sample of 135 female mouse liver arrays used for the female liver network construction.

As a motivational example, we study the pair-wise correlations between 498 genes that had previously been found to form the Blue module in the female liver network, which we had found to be related to mouse body weight [Ghazalpour et al 2006].

The 498 genes were a subset of the Blue module that did not have missing values.

We used multiple tissue samples from male and female mice of an this F2 intercross. For each gender/tissue combination approximately 100 tissue samples were available. The biological significance of this sub-network is described in [Ghazalpour et al 2006, Fuller et al 2007]. Here we focus on the mathematical and topological properties of the pair-wise absolute correlations $a_{ij} = |cor(x_i, x_j)|$ between the genes. For each gender and tissue type (liver, adipose, brain, muscle), we construct a separate gene co-expression network.

Module Eigengene

The singular value decomposition (SVD) of $X^{(q)}$ is given by

$$X^{(q)} = U^{(q)} D^{(q)} (V^{(q)})^T,$$

where $U^{(q)}$ is an $n^{(q)} \times m$ orthogonal matrix, $V^{(q)}$ is an $m \times m$ orthogonal matrix, and $D^{(q)}$ is an $m \times m$ diagonal matrix of the singular values $\{|d_i^{(q)}|\}$. Specifically, $V^{(q)}$ and $D^{(q)}$ are given by

$$\begin{aligned} V^{(q)} &= (v_1^{(q)} \quad v_2^{(q)} \quad \dots \quad v_m^{(q)}), \\ D^{(q)} &= \text{diag}\{|d_1^{(q)}|, |d_2^{(q)}|, \dots, |d_m^{(q)}|\}. \end{aligned} \tag{19}$$

We refer to the first column of $V^{(q)}$ as the **Module Eigengene**:

The tutorial, R functions, data etc can be found at the following webpage
<http://www.genetics.ucla.edu/labs/horvath/ModuleConformity/GeometricInterpretation/>

```
# Downloading the R software
# 1) Go to http://www.R-project.org, download R and install it on your computer
# After installing R, you need to install several additional R library packages:
# For example to install Hmisc, open R,
# go to menu "Packages\Install package(s) from CRAN",
# then choose Hmisc. R will automatically install the package.
# When asked "Delete downloaded files (y/N)? ", answer "y".
# Do the same for some of the other libraries mentioned below. But note that
# several libraries are already present in the software so there is no need to re-install
them.
```

```
# Download the zip file containing the data
# Unzip all the files into the same directory.
```

Please also download the **WGCNA R library** from
<http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/Softwares/WGCNA/>
To install it in R use the command "Install package(s) from local zip file" which can be found in the tab "Packages" The WGCNA package contains our most recent R software code and accompanying software tutorials.

Disclaimer: Absolutely no warranty on the code. Please contact Steve Horvath (shorvath@mednet.ucla.edu), Peter Langfelder (peter.langfelder@gmail.com) or Jun Dong with suggestions.

START of the R session

```
# Copy and paste the following commands into the R session.
# Text after "#" is a comment and is automatically ignored by R.
```

```
# this cleans the R session
rm(list=ls())
```

```
# Please adapt this path to your setting. Note the path contains backslashes / instead of \.
```

```
setwd("C:/Documents and Settings/Steve Horvath/My
Documents/ADAG/JunDong/WebpageAugust2009")
```

```
#Comment: if the above output gives you an error consider changing the path.
# Also make sure that you use straight quotes and a back slash /
```

```
# the installation of the WGCNA is describe above
library(WGCNA)
# install the following R packages from the cran webpage using the packages tab on R
library(sma)
library(impute)
```

```
#Here we read in the data and create a list of clinical traits (body weight) and of
# gene expression data. Each component of a list corresponds to one of the 8
# tissue/gender combinations
```

```
dat1=read.table("MouseWeightBlue.xls", sep="\t", quote="", header=T);
dim(dat1);
datalab.list=as.vector(unique(dat1$datalab))
n= table(dat1$datalab) [match(datalab.list , names(table(dat1$datalab)) )]
n1=0
for(i in 1:8) n1=c(n1, sum(n[1:i]))
n1
Trait.list=list(NULL)
datExpr.list=list(NULL)
```

```
# Note that the vector datalab.list provides a label for each gender/tissue combination
datalab.list
```

```
[1] "LiverFemale" "LiverMale" "AdiposeFemale" "AdiposeMale"
[5] "BrainFemale" "BrainMale" "MuscleFemale" "MuscleMale"
```

```
# Here we shorten these labels as follows
```

```
datalab.list=c("Fem. Liver" , "Male Liver" , "Fem. Adipose" , "Male Adipose" , "Fem.
Brain" , "Male Brain" , "Fem. Muscle" , "Male Muscle")
```

```
# the variable dat.index indexes the 8 networks (2 genders x 4 tissues)
```

```
for(dat.index in 1:8){
Trait.list[[dat.index]]=dat1[ (n1[dat.index]+1): n1[dat.index+1], 4:25 ]
datExpr.list[[dat.index]]=dat1[ (n1[dat.index]+1): n1[dat.index+1], 26:559 ]
print (paste(datalab.list[dat.index], ": number of mice with trait data =
",dim(Trait.list[[dat.index]])[[1]] ) );
print (paste(datalab.list[dat.index], ": number of traits =
",dim(Trait.list[[dat.index]])[[2]] ) );
print (paste(datalab.list[dat.index], ": number of Blue Module Genes =
",dim(datExpr.list[[dat.index]])[[2]] ) );
```

```

print (paste(datalab.list[dat.index], ": number of mice with expression data = ",
dim(datExpr.list[[dat.index]])[[1]]) );
} # end of for

```

```

[1] "Fem. Liver : number of mice with trait data = 135"
[1] "Fem. Liver : number of traits = 22"
[1] "Fem. Liver : number of Blue Module Genes = 534"
[1] "Fem. Liver : number of mice with expression data = 135"
[1] "Male Liver : number of mice with trait data = 124"
[1] "Male Liver : number of traits = 22"
[1] "Male Liver : number of Blue Module Genes = 534"
[1] "Male Liver : number of mice with expression data = 124"
[1] "Fem. Adipose : number of mice with trait data = 123"
[1] "Fem. Adipose : number of traits = 22"
[1] "Fem. Adipose : number of Blue Module Genes = 534"
[1] "Fem. Adipose : number of mice with expression data = 123"
[1] "Male Adipose : number of mice with trait data = 85"
[1] "Male Adipose : number of traits = 22"
[1] "Male Adipose : number of Blue Module Genes = 534"
[1] "Male Adipose : number of mice with expression data = 85"
[1] "Fem. Brain : number of mice with trait data = 110"
[1] "Fem. Brain : number of traits = 22"
[1] "Fem. Brain : number of Blue Module Genes = 534"
[1] "Fem. Brain : number of mice with expression data = 110"
[1] "Male Brain : number of mice with trait data = 98"
[1] "Male Brain : number of traits = 22"
[1] "Male Brain : number of Blue Module Genes = 534"
[1] "Male Brain : number of mice with expression data = 98"
[1] "Fem. Muscle : number of mice with trait data = 143"
[1] "Fem. Muscle : number of traits = 22"
[1] "Fem. Muscle : number of Blue Module Genes = 534"
[1] "Fem. Muscle : number of mice with expression data = 143"
[1] "Male Muscle : number of mice with trait data = 127"
[1] "Male Muscle : number of traits = 22"
[1] "Male Muscle : number of Blue Module Genes = 534"
[1] "Male Muscle : number of mice with expression data = 127"

```

Check the number of missing values in each dataset

```

for(dat.index in 1:8){
datExpr=datExpr.list[[dat.index]]
print(sum(is.na(datExpr)))
}

```

```

[1] 350
[1] 298
[1] 1416
[1] 1002
[1] 641
[1] 549
[1] 403
[1] 389

```

Remove genes/probesets that have more than n.rm (e.g 10) missing values across

#samples

n.rm=10

gene.rm=rep(F, dim(datExpr.list[[1]])[2])

for(dat.index in 1:8){

datExpr=datExpr.list[[dat.index]]

gene.rm=gene.rm | (apply(is.na(datExpr), 2, sum) > n.rm)

```

print (paste(datalab.list[dat.index], ": Number of genes that will be removed = ",
sum(gene.rm)  ) )
}
[1] "Fem. Liver : Number of genes that will be removed = 7"
[1] "Male Liver : Number of genes that will be removed = 7"
[1] "Fem. Adipose : Number of genes that will be removed = 28"
[1] "Male Adipose : Number of genes that will be removed = 28"
[1] "Fem. Brain : Number of genes that will be removed = 34"
[1] "Male Brain : Number of genes that will be removed = 34"
[1] "Fem. Muscle : Number of genes that will be removed = 36"
[1] "Male Muscle : Number of genes that will be removed = 36"
# here we remove the genes with too many missing values across the arrays
for(dat.index in 1:8){
datExpr.list[[dat.index]]= datExpr.list[[dat.index]][, !gene.rm]
print(dim(datExpr.list[[dat.index]]))
}

# this is the power used for constructing the weighted network
power1=1

# Now we compute the network concepts for each of the 8 networks
for(dat.index in 1:8){
datalab=datalab.list[dat.index]
datExpr=datExpr.list[[dat.index]]
Trait=Trait.list[[dat.index]][,1] ### The first column for weight
colorh1=rep("turquoise", dim(datExpr)[2])
colorlevel1=levels(factor(colorh1))
# the following computes network concepts for each of the 8 co-expression networks
assign(paste("NC", dat.index,sep=""), networkConcepts(datExpr, trait=Trait))
# ADJ hierarchical plots based on Adjacency matrix
ADJ = adjacency(datExpr,power=power1)
assign(paste("ADJ", dat.index,sep=""), ADJ)
assign(paste("hierADJ", dat.index,sep=""), hclust(as.dist(1-ADJ),method="average"))
} # end of for loop

# The following will give us network concepts
# that only involve the adjacency matrix
NC1$Summary
NC2$Summary
NC3$Summary
NC4$Summary
NC5$Summary
NC6$Summary
NC7$Summary
NC8$Summary

# The following will give us network concepts
# that make use of the gene significance measure.
NC1$Significance
NC2$Significance
NC3$Significance

```

NC4\$Significance
NC5\$Significance
NC6\$Significance
NC7\$Significance
NC8\$Significance

Now we report the Factorizability and VarExplained by ME

c(NC1\$Factorizability, NC1\$VarExplained [1])
c(NC2\$Factorizability, NC2\$VarExplained [1])
c(NC3\$Factorizability, NC3\$VarExplained [1])
c(NC4\$Factorizability, NC4\$VarExplained [1])
c(NC5\$Factorizability, NC5\$VarExplained [1])
c(NC6\$Factorizability, NC6\$VarExplained [1])
c(NC7\$Factorizability, NC7\$VarExplained [1])
c(NC8\$Factorizability, NC8\$VarExplained [1])

```

> NC1$Summary
Fundamental Eigengene-based Conformity-Based
Density      0.3864613      0.3859529      0.3861584
Centralization 0.1876239      0.1914378      0.1899320
Heterogeneity 0.1803588      0.1925771      0.1846080
Mean ClusterCoef 0.4166819      0.4142770      0.4129817
Mean Connectivity 192.0712906      191.8186050      191.9207178
Approximate Conformity-based
Density      0.3869620
Centralization 0.1908601
Heterogeneity 0.1849447
Mean ClusterCoef 0.4130553
Mean Connectivity 192.3201120
> NC2$Summary
Fundamental Eigengene-based Conformity-Based
Density      0.3615877      0.3549528      0.3596691
Centralization 0.1944611      0.2053842      0.2020271
Heterogeneity 0.2818349      0.3238952      0.3006533
Mean ClusterCoef 0.4215018      0.4324640      0.4277697
Mean Connectivity 179.7090622      176.4115376      178.7555338
Approximate Conformity-based
Density      0.3604585
Centralization 0.2030051
Heterogeneity 0.3010887
Mean ClusterCoef 0.4279141
Mean Connectivity 179.1478800
> NC3$Summary
Fundamental Eigengene-based Conformity-Based
Density      0.2265701      0.1473462      0.2251163
Centralization 0.1142552      0.1914770      0.1340146
Heterogeneity 0.2307228      0.5822623      0.2562006
Mean ClusterCoef 0.2655973      0.2636612      0.2557017
Mean Connectivity 112.6053505      73.2310537      111.8827890
Approximate Conformity-based
Density      0.2255991
Centralization 0.1346864
Heterogeneity 0.2566515
Mean ClusterCoef 0.2557837
Mean Connectivity 112.1227655
> NC4$Summary
Fundamental Eigengene-based Conformity-Based
Density      0.2238550      0.1542119      0.2186147
Centralization 0.1402861      0.2062844      0.1743470
Heterogeneity 0.2865083      0.6550260      0.3579970
Mean ClusterCoef 0.2700680      0.3143006      0.2783765
Mean Connectivity 111.2559157      76.6432936      108.6515046
Approximate Conformity-based
Density      0.2191113
Centralization 0.1752745
Heterogeneity 0.3587005
Mean ClusterCoef 0.2785625
Mean Connectivity 108.8983115
> NC5$Summary
Fundamental Eigengene-based Conformity-Based
Density      0.3117242      0.2645489      0.3041086
Centralization 0.1813869      0.2287904      0.2100198
Heterogeneity 0.3470537      0.5521844      0.4148131
Mean ClusterCoef 0.3898889      0.4495649      0.4180301
Mean Connectivity 154.9269258      131.4807800      151.1419834

```

```

Approximate Conformity-based
Density 0.3048264
Centralization 0.2110535
Heterogeneity 0.4154847
Mean ClusterCoef 0.4183115
Mean Connectivity 151.4987134
> NC6$Summary
Fundamental Eigengene-based Conformity-Based
Density 0.2108353 0.1266350 0.2088422
Centralization 0.1213481 0.2026615 0.1433124
Heterogeneity 0.2606747 0.6629905 0.2941844
Mean ClusterCoef 0.2504440 0.2619016 0.2466352
Mean Connectivity 104.7851509 62.9376008 103.7945757
Approximate Conformity-based
Density 0.2092990
Centralization 0.1440526
Heterogeneity 0.2947285
Mean ClusterCoef 0.2467433
Mean Connectivity 104.0215987
> NC7$Summary
Fundamental Eigengene-based Conformity-Based
Density 0.2364807 0.1825793 0.2323857
Centralization 0.1602586 0.2137564 0.1864895
Heterogeneity 0.3130432 0.5749789 0.3651659
Mean ClusterCoef 0.2938815 0.3226072 0.2986401
Mean Connectivity 117.5309058 90.7419028 115.4956970
Approximate Conformity-based
Density 0.2329160
Centralization 0.1874815
Heterogeneity 0.3658197
Mean ClusterCoef 0.2988255
Mean Connectivity 115.7592525
> NC8$Summary
Fundamental Eigengene-based Conformity-Based
Density 0.2461070 0.1577524 0.2440186
Centralization 0.1266224 0.2028885 0.1436422
Heterogeneity 0.3011017 0.6533764 0.3287144
Mean ClusterCoef 0.3065029 0.3205461 0.2997190
Mean Connectivity 122.3151701 78.4029395 121.2772348
Approximate Conformity-based
Density 0.2445629
Centralization 0.1443388
Heterogeneity 0.3292142
Mean ClusterCoef 0.2998448
Mean Connectivity 121.5477596
>
> NC1$Significance
Fundamental Eigengene-based
ModuleSignificance 0.3946043 0.3932602
HubGeneSignificance 0.5872317 0.5875391
EigengeneSignificance 0.6336498 NA
> NC2$Significance
Fundamental Eigengene-based
ModuleSignificance 0.2985054 0.2927207
HubGeneSignificance 0.4578544 0.4614157
EigengeneSignificance 0.4918183 NA
> NC3$Significance
Fundamental Eigengene-based
ModuleSignificance 0.2637418 0.1678485
HubGeneSignificance 0.4009576 0.3850923
EigengeneSignificance 0.4377079 NA

```

```

> NC4$Significance
      Fundamental Eigengene-based
ModuleSignificance 0.2467058      0.2258509
HubGeneSignificance 0.4249948      0.5267512
EigengeneSignificance 0.5757041      NA
> NC5$Significance
      Fundamental Eigengene-based
ModuleSignificance 0.06709798      0.02906898
HubGeneSignificance 0.09535625      0.05410781
EigengeneSignificance 0.05657353      NA
> NC6$Significance
      Fundamental Eigengene-based
ModuleSignificance 0.083757805      0.001312904
HubGeneSignificance 0.119330108      0.003405583
EigengeneSignificance 0.003693112      NA
> NC7$Significance
      Fundamental Eigengene-based
ModuleSignificance 0.12411409      0.02826496
HubGeneSignificance 0.19436261      0.06122353
EigengeneSignificance 0.06621542      NA
> NC8$Significance
      Fundamental Eigengene-based
ModuleSignificance 0.1087172      0.03602491
HubGeneSignificance 0.1557615      0.08217120
EigengeneSignificance 0.0907928      NA
>

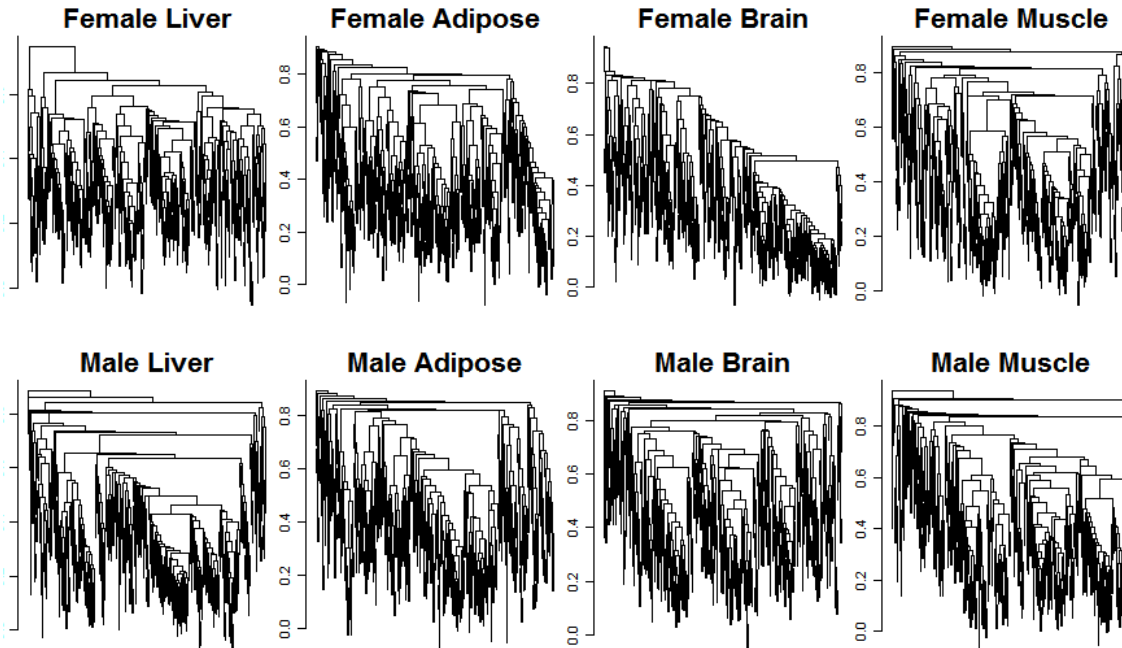
> # Now we report the Factorizability and VarExplained by ME
> c(NC1$Factorizability, NC1$VarExplained [1])
[1] 0.9151752 0.6404968
> c(NC2$Factorizability, NC2$VarExplained [1])
[1] 0.92633890 0.09095694
> c(NC3$Factorizability, NC3$VarExplained [1])
[1] 0.718941524 0.005105357
> c(NC4$Factorizability, NC4$VarExplained [1])
[1] 0.7607919 -0.1391425
> c(NC5$Factorizability, NC5$VarExplained [1])
[1] 0.87443770 -0.02027542
> c(NC6$Factorizability, NC6$VarExplained [1])
[1] 0.7313355 -0.2166925
> c(NC7$Factorizability, NC7$VarExplained [1])
[1] 0.7850412 -0.1061443
> c(NC8$Factorizability, NC8$VarExplained [1])
[1] 0.7673299 -0.2112926

```

```

# Now we visualize the average linkage cluster trees for each network
par(mar=c(1,2.5,5,1), mfrow=c(2,4))
for(i in c(1,3,5,7,2,4,6,8) ){
plot(eval(as.name(paste("hierADJ",i,sep="")))), main= paste(datalab.list[i]), labels=F,
xlab="", sub="", cex.main=2);
}

```

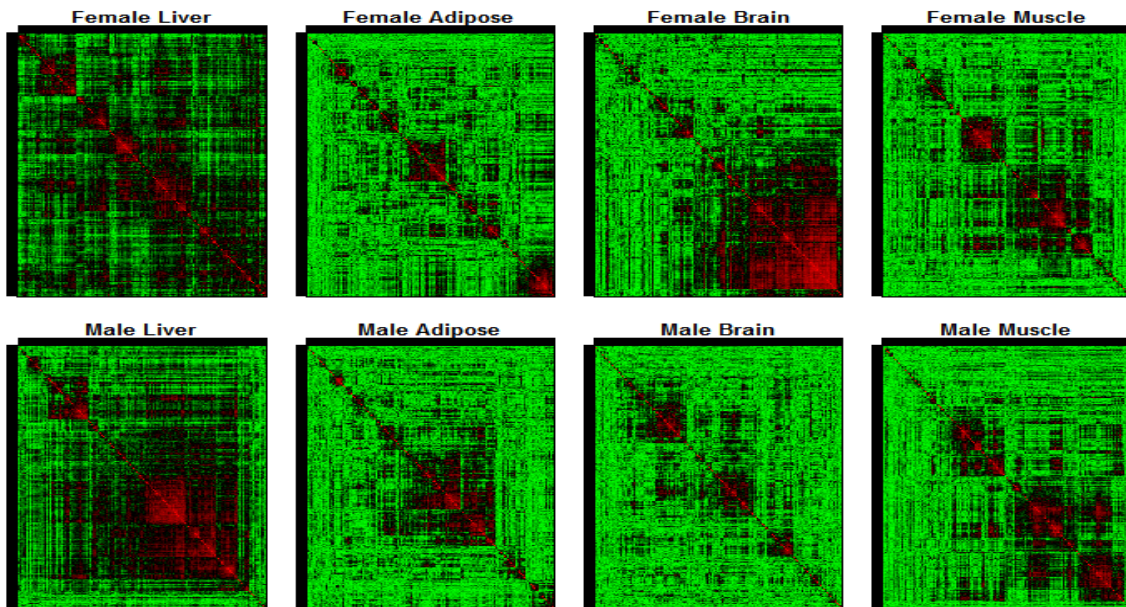


Caption: This is Figures 1A in (Horvath and Dong 2008). For each gender/tissue combination, the Figure depicts an average linkage hierarchical cluster tree of the genes. As an example for the use of network concepts, compare the cluster tree of the female brain network with that of the male brain network. The cluster tree of the female network appears to be comprised of a single large branch, i.e., gene at tip of the branch appear to be highly correlated with most other genes in the network. In contrast, the cluster tree corresponding to the male brain network appears to split into multiple smaller branches. To measure whether some gene appears to be highly connected to most other genes in one network one can use the concept of centralization. The female brain and the male brain networks have centralization 0.34 and 0.21, respectively.

```

# Now we create the heatmap plots
par(mfrow=c(2,4) ,mar=c(1,1, 2.5,1))
i=1; plot.mat(ADJ1[hierADJ1$order, hierADJ1$order],main= datalab.list[i], cex.main=2)
i=3; plot.mat(ADJ3[hierADJ3$order, hierADJ3$order], main= datalab.list[i]
,cex.main=2)
i=5; plot.mat(ADJ5[hierADJ5$order, hierADJ5$order], main= datalab.list[i],
cex.main=2)
i=7; plot.mat(ADJ7[hierADJ7$order, hierADJ7$order], main= datalab.list[i],
cex.main=2)
i=2; plot.mat(ADJ2[hierADJ2$order, hierADJ2$order], main= datalab.list[i],
cex.main=2)
i=4; plot.mat(ADJ4[hierADJ4$order, hierADJ4$order], main= datalab.list[i],
cex.main=2)
i=6; plot.mat(ADJ6[hierADJ6$order, hierADJ6$order], main= datalab.list[i],
cex.main=2)
i=8; plot.mat(ADJ8[hierADJ8$order, hierADJ8$order], main= datalab.list[i],
cex.main=2)

```



Caption: This is Figure 1B shows the corresponding heat maps which color-code the absolute pair-wise correlations a_{ij} : red and green in the heat map indicate high and low absolute correlation, respectively. The genes in the rows and columns of each heat map are sorted by the corresponding cluster tree. It is visually obvious that the heat maps and the cluster trees of different gender/tissue combinations can look quite different. Network theory offers a wealth of intuitive concepts for describing the pair-wise relationships between genes that are depicted in cluster trees and heat maps. To illustrate this point, we describe several such concepts in the following. By visual inspection of Figure 1B, genes appear to be more highly correlated in liver than in adipose (a lot of red versus green

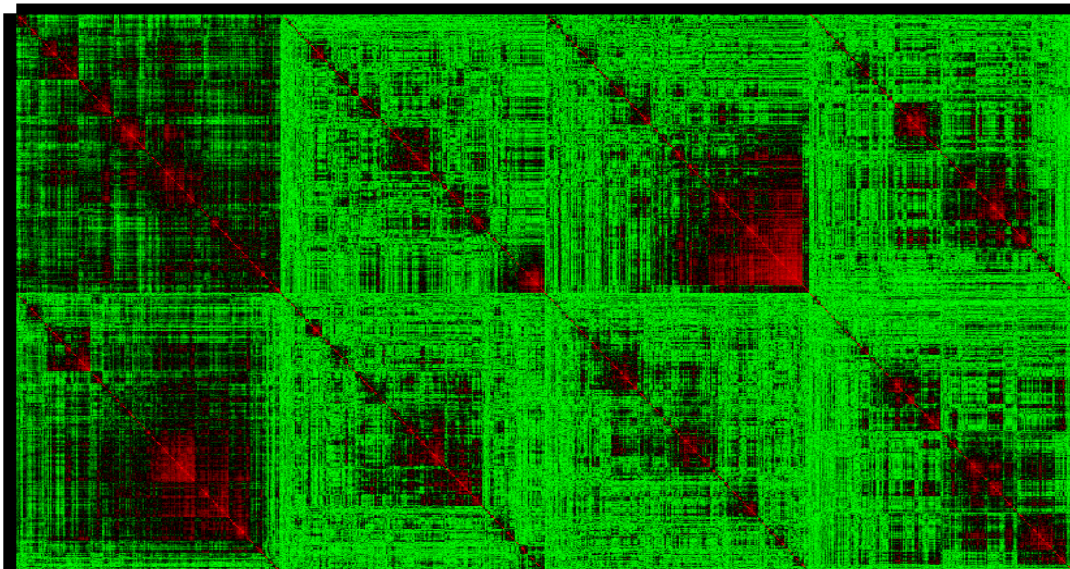
color in the corresponding heat maps). This property can be captured by the concept of network density: the density of the female liver network is 0.39 while it is only 0.23 for the female adipose network. Another example for the use of network concepts is to quantify the extent of cluster (module) structure. In this example, branches of a cluster trees (Figure 1A) correspond to (sub-)modules. The cluster structure is also reflected in the corresponding heat maps: modules correspond to large red squares along the diagonal.

Network theory provides a concept for quantifying the extent of module structure in a network: the mean clustering coefficient. The female liver, male liver, and female brain have high mean clustering coefficients (mean *ClusterCoef* = 0.42, 0.43, 0.41, respectively). In contrast, female adipose, male adipose, and male brain have lower mean clustering coefficients (mean *ClusterCoef* = 0.27, 0.27, 0.25, respectively). Difference in module structure may reflect technical artifacts or true biological differences.

The following is the same as Figure 1B but there are no white margins
 # between the quadrants.

```
par(mar=c(1,2.5,5,1))
par(mar=c(1,3, 4,1))
plot.mat(rbind(
  cbind(ADJ1[hierADJ1$order, hierADJ1$order],
  ADJ3[hierADJ3$order, hierADJ3$order],
  ADJ5[hierADJ5$order, hierADJ5$order],
  ADJ7[hierADJ7$order, hierADJ7$order]
  ),
  cbind(
  ADJ2[hierADJ2$order, hierADJ2$order],
  ADJ4[hierADJ4$order, hierADJ4$order],
  ADJ6[hierADJ6$order, hierADJ6$order],
  ADJ8[hierADJ8$order, hierADJ8$order]
  )), title="Ordered by Hierarchical Tree")
```

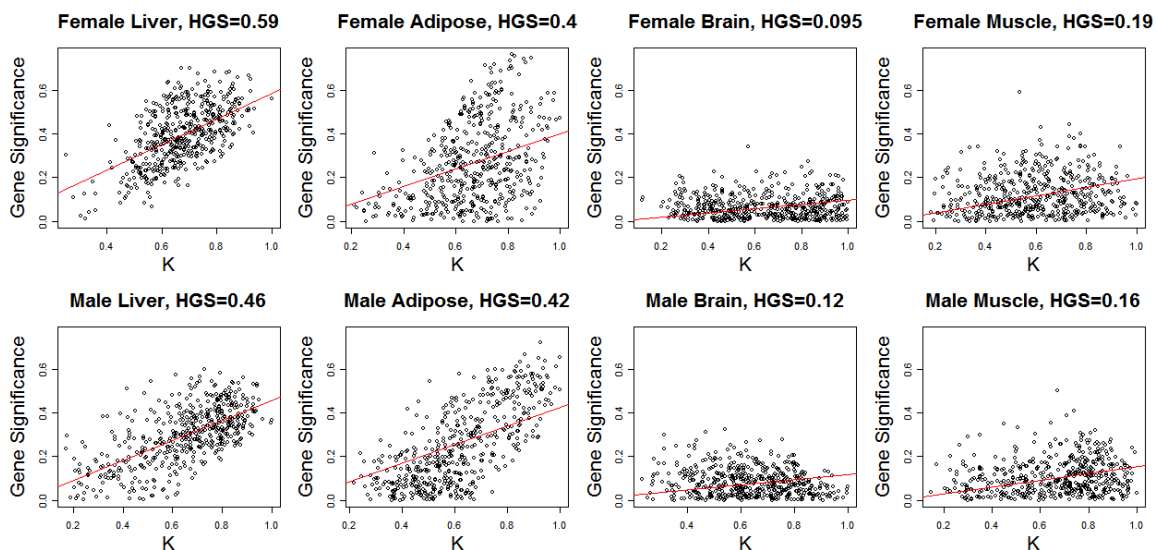
Ordered by Hierarchical Tree



Body weight based gene significance and hub gene significance

Numerous network articles have pointed out that highly connected "hub" nodes are central to the network architecture but several authors have pointed out that hub genes may not always be biologically significant. Network theorists have long studied the relationship between gene significance and connectivity. We define a gene significance measure as the absolute correlation between the gene expression profile and body weight. The higher the gene significance, the higher is its absolute correlation with body weight.

```
GS.max=0
for(i in c(1,3,5,7,2,4,6,8) ){
temp= eval(as.name(paste("NC",i,sep="")))
GS.max=max(GS.max, temp$GS )
}
par(mar=c(4,5,5,1), mfrow=c(2,4))
for(i in c(1,3,5,7,2,4,6,8) ){
temp= eval(as.name(paste("NC",i,sep="")))
K=temp$Connectivity/max(temp$Connectivity)
GS=temp$GS
plot(K, temp$GS, main= paste(datalab.list[i], ", HGS=",
signif(temp$Significance[2,1],2), sep=""), xlab="K", ylab="Gene Significance", sub="",
cex.main=1.5, ylim=c(0,GS.max), cex.lab=1.5);
abline(0, temp$Significance[2,1], col=2)
}
}
```



Caption: This is Figure 1C in our paper. The figure shows the relationship between this gene significance measure and connectivity in the different gender/tissue type networks. We find a high correlation between gene significance and connectivity in the female ($r = 0.59$) and male mouse liver network ($r = 0.46$), respectively.

Other References

1. Bin Zhang and Steve Horvath (2005) "A General Framework for Weighted Gene Co-Expression Network Analysis", *Statistical Applications in Genetics and Molecular Biology: Vol. 4: No. 1, Article 17* Technical Report and software code at: www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork.
2. Horvath S, Zhang B, Carlson M, Lu KV, Zhu S, Felciano RM, Laurance MF, Zhao W, Shu, Q, Lee Y, Scheck AC, Liau LM, Wu H, Geschwind DH, Febbo PG, Kornblum HI, Cloughesy TF, Nelson SF, Mischel PS (2006) "Analysis of Oncogenic Signaling Networks in Glioblastoma Identifies ASPM as a Novel Molecular Target", *PNAS* | November 14, 2006 | vol. 103 | no. 46 | 17402-17407

For the generalized topological overlap matrix as applied to *unweighted* networks see

3. Yip A, Horvath S (2007) *Gene network interconnectedness and the generalized topological overlap measure*. *BMC Bioinformatics* 2007, 8:22
<http://www.genetics.ucla.edu/labs/horvath/GTOM/>.

If you like math and theory consider

4. Dong J, Horvath S (2007) *Understanding Network Concepts in Modules*, *BMC Systems Biology* 2007, 1:24

The mouse data are described in

5. Ghazalpour A, Doss S, Zhang B, Wang S, Plaisier C, Castellanos R, Brozell A, Schadt EE, Drake TA, Lusk AJ, Horvath S (2006) *Integrating Genetic and Network Analysis to Characterize Genes Related to Mouse Weight*. *PLoS Genetics*. Volume 2 | Issue 8 | AUGUST 2006
6. Fuller TF, Ghazalpour A, Aten JE, Drake TA, Lusk AJ, Horvath S (2007) "Weighted Gene Co-expression Network Analysis Strategies Applied to Mouse Weight", *Mamm Genome* 18(6):463-472.

THE END