WGCNA Pipeline - part two: Selected lipids based on maternal diet

Rachel Richardson

Adapted from the ANOVAs used for plotting interaction and comparing chain length, this code pulls out specific lipid species with different abundances in groups from the metadata. Because we have so many lipids that can be compared to our gene clusters, this step helps us determine which lipids would be the most likely related to our biological group of interest. For our data, we are intrested in the effects of maternal diet.

From scratch, we would need to load our data to use (if you have not already done so in previous steps).

```
#Read in lipid abundance data to be compared:

DGs1.2 <- read.csv("1.2DGs.changed LOQ.csv", head = TRUE, row.names = 1)
AC <- read.csv("AC.formatted.noloq.csv", head = TRUE, row.names = 1)
Cer <- read.csv("Cerimides.changedLOQ.csv", head = TRUE, row.names = 1)

DGs1.3 <- read.csv("CM033018 1-3DGs.changedLOQ.csv", head = TRUE, row.names = 1)
names(DGs1.3) <- paste(names(DGs1.3), ".1.3DGs", sep = "") #Correction for poor naming

dh <- read.csv("dhCer.changedLOQ.csv", head = TRUE, row.names = 1)
Glu <- read.csv("GluCer.changedLOQ.csv", head = TRUE, row.names = 1)
hex <- read.csv("hexosylCer.changedLOQ.csv", head = TRUE, row.names = 1)
Lac <- read.csv("LacCer.changedLOQ.csv", head = TRUE, row.names = 1)
mye <- read.csv("Sphingomyelins.formatted.noloq.csv", head = TRUE, row.names = 1)
sine <- read.csv("Sphingosine.formatted.noloq.csv", header = TRUE, row.names = 1)
TAG <- read.csv("TAG.changedLOQ.csv", head = TRUE, row.names = 1)
```

Our lipid abundance dataset originated from several excel sheets across different lipid species, which we reformatted as needed and saved as CSVs. We also took into account if any data points fell below the limit of quantification (LOQ) and created seperate csvs with adjusted values based on LOQ (noted as changed LOQ) or noted that data points were unmodified since all points were within range of the experiment (no LOQ).

Collected metadata contains the information for different experimental groups across samples. However, metadata was collected on a larger group of subjects than the ones tested for lipid abundance, therefore we filter the data by the our subjects of interest.

```
#Read in metadata, location of groupings to be tested:

  #Selects for subjects with lipid abundance samples, as based on sine dataset
metadata <- read.table("metadata.txt", header = TRUE)[row.names(sine),]

  #Creates a new column with all groupings of interest as a factor. Format:  MaternalBodyMass.MaternalDiet-Postwea
nDiet
metadata$combination <- as.factor(paste(metadata$M_BM, paste(metadata$M_Diet,metadata$PW_Diet, sep = "-"), sep =
"."))
```

The resulting lipid data CSVs are arranged with different lipid species as columns and different subjects as rows. The metadata is formatted similarly, where columns are different categorical variables and rows are subjects.

```
library(knitr)

#Lipid data format
kable(head(AC[1:7]))
```

	X10.0.AC..pmol.	X12.0.AC..pmol.	X14.0.AC..pmol.	X16.0.AC..pmol.	X16.1.AC..pmol.	X18.0.AC..pmol.	X18.1.AC..pmol.
T101	58.9	38.1	154.3	1195.3	203.9	646.1	1359.3
T103	36.6	26.5	71.7	425.4	78.5	274.9	411.4
T104	40.5	31.8	129.7	1177.7	156.0	518.7	1171.6
T105	37.7	16.7	58.5	250.2	71.7	145.0	271.2
T106	58.8	34.1	123.9	953.2	169.9	442.4	1058.8
T107	22.2	22.9	83.0	636.6	132.0	379.0	821.4

```
#Metadata format
kable(head(metadata))
```

	Group	M_BM	M_Diet	PW_Diet	Sex	combination
T101	1	Ln	CTR	CTR	M	Ln.CTR-CTR
T103	1	Ln	CTR	CTR	F	Ln.CTR-CTR
T104	1	Ln	CTR	CTR	F	Ln.CTR-CTR
T105	1	Ln	CTR	CTR	F	Ln.CTR-CTR
T106	1	Ln	CTR	CTR	M	Ln.CTR-CTR
T107	1	Ln	CTR	CTR	F	Ln.CTR-CTR

In order to determine the effect of diet, we are most interested in the lipid species with significantly different abundances between the groups. One common way would be to run ANOVA or t-test using the groups notated in the metadata.

However, ANOVA testing assumes normality of the residuals in our data, which can be approximated by overall distribution when sample size is not small. We know from our previous distribution analyses that the raw data does not appear to be normally distributed, but we can double-check our eye-ball rejection of normality with a Shapiro-Wilk (test of normality where the null hypothesis is that samples come from a normal distribution).

```
#Select p-values for testing purposes.

shapvalue <- 0.05

#Contain all data sets in a list as variable and a string list

datanames <- list(DGs1.2, AC, Cer, DGs1.3, dh, Glu, hex, mye, Lac, sine, TAG)
datanamesstr <- c('DGs1.2', 'AC', 'Cer', 'DGs1.3', 'dh', 'Glu', 'hex', 'mye', 'Lac', 'sine', 'TAG')

#Apply Shapiro-Wilk across datasets

totalprint <- c()   # Vector to hold outcomes across datasets
iter <- 1           # iterable for which dataset is being processed

for (datan in datanames){ #For each lipid dataset read in,
  testresult <- array()            #create a new array for holding normality test results.

  for(x in 1:length(datan)){                          #For every column in that dataset,
    test <- shapiro.test(unlist(datan[,x]))     #perform an ANOVA based on the groups specified in the metadata,
    testresult[x] <- test$p.value                #and save the isolated significance value from the anova output
in the sig array.
  }


  total <- length(testresult)                                  # Counts all species in processed dataset
  rejectnull <- length(testresult[testresult < shapvalue])    # Counts all species not from a normal distribution
in a processed dataset

  totalprint <- c(totalprint,                                  # Adds to totalprint vector
                  sprintf("From the %s dataset, %d/%d reject the null hypothesis that samples are drawn from a nor
mal distribution.",
                  datanamesstr[iter], rejectnull,total))
  iter <- iter + 1
}

totalprint
```

```
##  [1] "From the DGs1.2 dataset, 23/24 reject the null hypothesis that samples are drawn from a normal distributi
on."
##  [2] "From the AC dataset, 12/12 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [3] "From the Cer dataset, 5/9 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [4] "From the DGs1.3 dataset, 8/8 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [5] "From the dh dataset, 2/5 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [6] "From the Glu dataset, 7/7 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [7] "From the hex dataset, 8/8 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [8] "From the mye dataset, 0/13 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [9] "From the Lac dataset, 6/7 reject the null hypothesis that samples are drawn from a normal distributio
n."
## [10] "From the sine dataset, 1/1 reject the null hypothesis that samples are drawn from a normal distributio
n."
## [11] "From the TAG dataset, 27/27 reject the null hypothesis that samples are drawn from a normal distributio
n."
```

Similarly, we can check if transformations are roughly sufficient. We used a log10 transformation here:

```
#Select p-values for testing purposes.

shapvalue <- 0.05

#Contain all data sets in a list as variable and a string list

datanames <- list(DGs1.2, AC, Cer, DGs1.3, dh, Glu, hex, mye, Lac, sine, TAG)
datanamesstr <- c('DGs1.2', 'AC', 'Cer', 'DGs1.3', 'dh', 'Glu', 'hex', 'mye', 'Lac', 'sine', 'TAG')

#Apply Shapiro-Wilk across datasets

totalprint <- c()   # Vector to hold outcomes across datasets
iter <- 1           # iterable for which dataset is being processed

for (datan in datanames){ #For each lipid dataset read in,
  testresult <- array()           #create a new array for holding normality test results.

  for(x in 1:length(datan)){                          #For every column in that dataset,

    #######

    test <- shapiro.test(unlist(log10(datan[,x])))     #perform an ANOVA based on the groups specified in the meta
data, use this line for transformation changes

    #######

    testresult[x] <- test$p.value                #and save the isolated significance value from the anova output
in the sig array.
  }


  total <- length(testresult)                                  # Counts all species in processed dataset
  rejectnull <- length(testresult[testresult < shapvalue])    # Counts all species not from a normal distribution
in a processed dataset

  totalprint <- c(totalprint,                                  # Adds to totalprint vector
                  sprintf("From the %s dataset, %d/%d reject the null hypothesis that samples are drawn from a nor
mal distribution.",
                  datanamesstr[iter], rejectnull,total))
  iter <- iter + 1
}

totalprint
```

```
##  [1] "From the DGs1.2 dataset, 11/24 reject the null hypothesis that samples are drawn from a normal distributi
on."
##  [2] "From the AC dataset, 0/12 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [3] "From the Cer dataset, 5/9 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [4] "From the DGs1.3 dataset, 5/8 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [5] "From the dh dataset, 4/5 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [6] "From the Glu dataset, 1/7 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [7] "From the hex dataset, 1/8 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [8] "From the mye dataset, 3/13 reject the null hypothesis that samples are drawn from a normal distributio
n."
##  [9] "From the Lac dataset, 4/7 reject the null hypothesis that samples are drawn from a normal distributio
n."
## [10] "From the sine dataset, 0/1 reject the null hypothesis that samples are drawn from a normal distributio
n."
## [11] "From the TAG dataset, 3/27 reject the null hypothesis that samples are drawn from a normal distribution."
```

Our raw data does not seem to meet the assumptions of ANOVA due to non-normality and our log-transform data seems questionable. **An alternative method to the ANOVA that doesn't assume normality is the Mann-Whitney-Wilcoxon Test, which we will use here to analyze differences based on maternal diet and post-wean diet. Because Mann-Whitney-Wilcoxon Test only accepts two levels for grouping factors, we use Kruskal-Wallis test (also non-parametric comparison of distributions) to test the combined factor that includes maternal body mass, maternal diet, and post-wean diet (five levels, one for each experimental group).**

Note: Significance rejects the null hypothisis, where both datasets are sampled from the same distribution

```
library(exactRankTests)
```

```
#Contain all data sets in a list

datanames <- list(DGs1.2, AC, Cer, DGs1.3, dh, Glu, hex, mye, Lac, sine, TAG)

#Define significance p-value; selects for lipids of interest

pvalue <- 0.05

################################
#All combonation Kruskal-Wallis


# New vector to hold anova lipid species and if abundances are significantly different among all five groups
ALL <- c()


for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    kru <- kruskal.test(datan[,x]~unlist(metadata[,6]))     #perform test based on the groups specified in the met
adata,
    sig[x] <- kru$p.value                                  #and save the isolated significance value from the an
ova output in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  ALL <- c(ALL, sig[sig < pvalue])    #Add to vector, ALL, that is a subset of all sig arrays with significance le
ss than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
ALL <- data.frame(ALL)
colnames(ALL) <-  "Significance"

 #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(ALL))
rnames <- gsub("X", "", rnames)
row.names(ALL) <- rnames


#View data format
kable(head(ALL))
```

	Significance
14.0.16.0.1.2DGs	0.0001666
14.0.16.1.1.2DGs	0.0209245
14.0.18.0.1.2DGs	0.0000338
14.0.18.1.1.2DGs	0.0001290
16.0.16.1.1.2DGs	0.0007229
16.0.18.0.1.2DGs	0.0159889

```
#######################

#Maternal diet Wilcox test

Mat <- c() # New vector to hold anova lipid species and if abundances are significantly different among all five g
roups

for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    wil <- wilcox.exact(datan[,x]~unlist(metadata[,3]))     #perform an wilcox test based on the groups specified
in the metadata,
    sig[x] <- wil$p.value                                  #and save the isolated significance value from the anov
a output in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  Mat <- c(Mat, sig[sig < pvalue])     #Add to vector, Mat, that is a subset of all sig arrays with significance l
ess than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
Mat <- data.frame(Mat)
colnames(Mat) <-  "Significance"

  #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(Mat))
rnames <- gsub("X", "", rnames)
row.names(Mat) <- rnames

#View data format
kable(head(Mat))
```

	Significance
di16.0.1.2DGs	0.0399700
10.0.AC	0.0241747
12.0.AC	0.0117546
14.0.AC	0.0098190
16.0.AC	0.0071347
16.1.AC	0.0118730

```
#######################

#Offspring diet Wilcox test

PW <- c()

for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    wil <- wilcox.exact(datan[,x]~unlist(metadata[,4]))     #perform an wilcox test based on the groups specified
in the metadata,
    sig[x] <- wil$p.value                                  #and save the isolated significance value from the anov
a output in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  PW <- c(PW, sig[sig < pvalue])     #Add to vector, PW, that is a subset of all sig arrays with significance les
s than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
PW <- data.frame(PW)
colnames(PW) <-  "Significance"

  #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(PW))
rnames <- gsub("X", "", rnames)
row.names(PW) <- rnames


#View data format
kable(head(PW))
```

	Significance
14.0.16.0.1.2DGs	0.0000094
14.0.16.1.1.2DGs	0.0424469
14.0.18.0.1.2DGs	0.0000001
14.0.18.1.1.2DGs	0.0000008
16.0.16.1.1.2DGs	0.0000266
16.0.18.0.1.2DGs	0.0192579

```
#Save as method specific; we will be looking at an ANOVA/t-test example with similar code as well

ALL.K <- ALL
Mat.W <- Mat
PW.W <- PW
```

If our transformed samples would meet normality the assumptions, below is an example of how you would use ANOVA and t-test to analyze differences based on maternal diet, post-wean diet, and a combined factor that includes maternal body mass, maternal diet, and post-wean diet.

```
#Define significance p-value; selects for lipids of interest

pvalue <- 0.05

################################
#All combonation ANOVA

# New vector to hold anova lipid species and if abundances are significantly different among all five groups
ALL <- c()


for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding ANOVA significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    ano <- anova(aov(log10(datan[,x])~unlist(metadata[,6])))     #perform an ANOVA based on the groups specified i
n the metadata,
    sig[x] <- as.double(ano$`Pr(>F)`[1])                  #and save the isolated significance value from the anov
a output in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  ALL <- c(ALL, sig[sig < pvalue])    #Add to vector, ALL, that is a subset of all sig arrays with significance le
ss than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
ALL <- data.frame(ALL)
colnames(ALL) <-  "Significance"

 #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(ALL))
rnames <- gsub("X", "", rnames)
row.names(ALL) <- rnames


#View data format
kable(head(ALL))
```

	Significance
14.0.16.0.1.2DGs	0.0000030
14.0.16.1.1.2DGs	0.0264376
14.0.18.0.1.2DGs	0.0000000
14.0.18.1.1.2DGs	0.0000001
16.0.16.1.1.2DGs	0.0000345
16.0.18.0.1.2DGs	0.0146928

```
######################

#Maternal diet t-test

Mat <- c() # New vector to hold anova lipid species and if abundances are significantly different among all five g
roups

for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding ANOVA significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    ttest <- t.test(log10(datan[,x])~unlist(metadata[,3]))     #perform test based on the groups specified in the
metadata,
    sig[x] <- ttest$p.value                              #and save the isolated significance value from the anova
output in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  Mat <- c(Mat, sig[sig < pvalue])     #Add to vector, Mat, that is a subset of all sig arrays with significance l
ess than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
Mat <- data.frame(Mat)
colnames(Mat) <-  "Significance"

  #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(Mat))
rnames <- gsub("X", "", rnames)
row.names(Mat) <- rnames

#View data format
kable(head(Mat))
```

	Significance
14.0.16.0.1.2DGs	0.0286764
di16.0.1.2DGs	0.0383824
10.0.AC	0.0200857
12.0.AC	0.0099445
14.0.AC	0.0043946
16.0.AC	0.0036193

```
#######################

#Offspring diet t-test

PW <- c()

for (datan in datanames){ #For each lipid dataset read in,
  sig <- array()            #create a new array for holding ANOVA significance values.

  for(x in 1:length(datan)){                            #For every column in that dataset,
    ttest <- t.test(log10(datan[,x])~unlist(metadata[,4]))     #perform test based on the groups specified in the
metadata,
    sig[x] <- ttest$p.value                            #and save the isolated significance value from the anova ou
tput in the sig array.
  }

  #Sig array will have all significance values for each lipid species in the dataset
  names(sig) <- colnames(datan)     #Label sig values with the column names of the dataset
  PW <- c(PW, sig[sig < pvalue])     #Add to vector, PW, that is a subset of all sig arrays with significance les
s than defined p-value
}

#Edit column names and row names for clarity; specific to our dataset
  #Rename column as "Significance"
PW <- data.frame(PW)
colnames(PW) <-  "Significance"

  #Global substitute for row name format
rnames <- gsub("..pmol.", "", row.names(PW))
rnames <- gsub("X", "", rnames)
row.names(PW) <- rnames


#View data format
kable(head(PW))
```

	Significance
14.0.16.0.1.2DGs	0.0000008
14.0.16.1.1.2DGs	0.0339856
14.0.18.0.1.2DGs	0.0000000
14.0.18.1.1.2DGs	0.0000000
16.0.16.1.1.2DGs	0.0000031
16.0.18.0.1.2DGs	0.0275390

```
#Save as method specific for comparison to non-parametric

ALL.A <- ALL
Mat.T <- Mat
PW.T <- PW
```

Comparing the two methods, we can see that there are different lipids identified as significant. We can also tell which lipids are identified as significant by both methods, a more conservative approach for finding lipids of interest.

```
#Generate booleans for indexes of lipids shared between lists and use to find overlapping lipids of significance

AllAinK <- row.names(ALL.A) %in% row.names(ALL.K)
AllKinA <- row.names(ALL.K) %in% row.names(ALL.A)

cat("ALL overlap:  \n", row.names(ALL.A)[AllAinK[AllKinA]]) #All shared genes between two methods
```

```
## ALL overlap:
##  14.0.16.0.1.2DGs 14.0.16.1.1.2DGs 14.0.18.0.1.2DGs 14.0.18.1.1.2DGs 16.0.16.1.1.2DGs 16.0.18.0.1.2DGs 16.0.18.
1.1.2DGs 16.0.18.2.1.2DGs 16.0.20.1.1.2DGs 16.1.18.1.1.2DGs 18.0.16.1.1.2DGs 18.0.18.1.1.2DGs 18.0.18.2.1.2DGs 18.
0.22.6.1.2DGs di14.0.1.2DGs di16.0.1.2DGs di16.1.1.2DGs 10.0.AC 12.0.AC 14.0.AC 16.0.AC 16.1.AC 18.0.AC 18.1.AC 1
8.2.AC 18.3.AC 6.0.AC 22.0Cer 23.0Cer 24.0Cer 24.1.Cer 18.0.20.4.1.3DGs 18.0.18.0.1.3DGs 22.0dhCer 16.0GluCer 22.0
GluCer 18.0.SPM 22.0.SPM 23.0.SPM 24.0.SPM 24.2.SPM 24.3.SPM 22.0.LacCer 23.0.LacCer 24.0.LacCer 24.1.LacCer 48.0.
TAG 48.1.TAG 48.2.TAG 50.0.TAG 50.1.TAG 50.2.TAG 52.0.TAG 52.2.TAG 52.3.TAG 54.1.TAG 54.2.TAG 54.3.TAG 54.6.TAG 5
6.3.TAG 56.4.TAG 56.5.TAG 56.6.TAG 56.7.TAG
```

```
PWTinW <- row.names(PW.T) %in% row.names(PW.W)
PWWinT <- row.names(PW.W) %in% row.names(PW.T)


cat("PW overlap: \n", row.names(PW.T)[PWTinW[PWWinT]]) #All shared genes between two methods
```

```
## PW overlap:
##  14.0.16.0.1.2DGs 14.0.16.1.1.2DGs 14.0.18.0.1.2DGs 14.0.18.1.1.2DGs 16.0.16.1.1.2DGs 16.0.18.0.1.2DGs 16.0.18.
1.1.2DGs 16.0.20.1.1.2DGs 16.0.20.4.1.2DGs 16.1.18.1.1.2DGs 18.0.16.1.1.2DGs 18.0.18.1.1.2DGs 18.0.18.2.1.2DGs 18.
0.18.3.1.2DGs 18.0.22.6.1.2DGs di14.0.1.2DGs di16.0.1.2DGs di16.1.1.2DGs 10.0.AC 12.0.AC 14.0.AC 16.0.AC 16.1.AC 1
8.0.AC 18.1.AC 8.0.AC 22.0Cer 23.0Cer 24.0Cer 24.1.Cer 24.2Cer 18.2.18.2.1.3DGs 18.2.18.1.1.3DGs 24.0dhCer 24.1dhC
er 14.0.SPM 20.0.SPM 22.0.SPM 23.0.SPM 24.0.SPM 24.2.SPM 24.3.SPM 22.0.LacCer 23.0.LacCer 24.0.LacCer 24.1.LacCer
48.0.TAG 48.1.TAG 48.2.TAG 50.0.TAG 50.1.TAG 50.2.TAG 50.3.TAG 52.0.TAG 52.1.TAG 52.3.TAG 52.5.TAG 54.1.TAG 54.2.T
AG 54.3.TAG 54.4.TAG 54.6.TAG 56.3.TAG 56.4.TAG 56.5.TAG 56.6.TAG 56.7.TAG
```

```
#Comparison of interest for maternal diet; conveniently the overlap between both methods is equal to Mat.W
#i.e., all lipids identified as significant in the Wilcoxon test were also significant by t-test.

MatTinW <- row.names(Mat.T) %in% row.names(Mat.W)
MatWinT <- row.names(Mat.W) %in% row.names(Mat.T)


cat("Indexes of Mat.T in Mat.W:  \n", which(MatTinW), "\n")
```

```
## Indexes of Mat.T in Mat.W:
##  2 3 4 5 6 7 8 9 10 11 12 13 15 16 21 22
```

```
cat("Indexes of Mat.W in Mat.T:  \n", which(MatWinT), "\n")
```

```
## Indexes of Mat.W in Mat.T:
##  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
```

```
cat("Mat.W:   \n", row.names(Mat.W))
```

```
## Mat.W:
##  di16.0.1.2DGs 10.0.AC 12.0.AC 14.0.AC 16.0.AC 16.1.AC 18.0.AC 18.1.AC 18.2.AC 18.3.AC 4.0.AC 6.0.AC 18.0.18.0.
1.3DGs 20.0GluCer 48.3.TAG 52.2.TAG
```

Since in our comparison of interest (Maternal Diet) all non-parametric significant lipids are also in the list of parametric significant lipids, we will use this list going forward in our WGCNA analysis.

Note: We also evaluated for differences between sexes (not shown here), resulting in only two significant lipid species (16.0.20.4.1.2DGs, 0.04930763; 24.0dhCer, 0.02698315). Evaluation for maternal bodymass can also be tested, however results would not be conclusive given the imbalence of obese and lean maternal subjects.

For a easily communicated graphic, a barchart can adequately display abundance comparisons between groups. Here, we use ggplot2, reshape2, and ggsignif on a log-scale for visual clarity.

```
library(ggplot2)
library(reshape2)
library(ggsignif)
```

```
Alllipid <- data.frame(DGs1.2, AC, Cer, DGs1.3, dh, Glu, hex, mye, Lac, sine, TAG)

 #Global substitute for column name format, keeps consistancy with tests for significance
cnames <- gsub("..pmol.", "", colnames(Alllipid))
cnames <- gsub("X", "", cnames)
colnames(Alllipid) <- cnames


#Pull out columns containing lipids of interest based on significance tests
Intlipid <- Alllipid[,row.names(Mat.W)]
Intlipid$M_Diet <- metadata[,3] #Add grouping of interest from significance test to dataframe


CTR <- subset(Intlipid, M_Diet=="CTR") #Subset of CTR maternal group
HFD <- subset(Intlipid, M_Diet=="HFD") #Subset of HFD maternal group

CTR <- CTR[1:length(CTR)-1] #Remove maternal diet from subset data
HFD <- HFD[1:length(HFD)-1] #Remove maternal diet from subset data


#Create a dataframe with the means of each lipid species subset (HFD, CTR)
plotter <- cbind(lapply(CTR, mean), lapply(HFD, mean))
plotter <- cbind(plotter, row.names(plotter))
plotter <- data.frame(unlist(plotter[,1]), unlist(plotter[,2]), row.names(plotter))
colnames(plotter)<-c("CTR", "HFD", "Lipid species")

#View Format
kable(head(plotter))
```

	CTR	HFD	Lipid species
di16.0.1.2DGs	309.11538	257.36522	di16.0.1.2DGs
10.0.AC	43.04615	29.77826	10.0.AC
12.0.AC	29.72308	20.09130	12.0.AC
14.0.AC	99.15385	60.56522	14.0.AC

	CTR	HFD	Lipid species
16.0.AC	667.49231	348.45652	16.0.AC
16.1.AC	138.80769	83.46522	16.1.AC

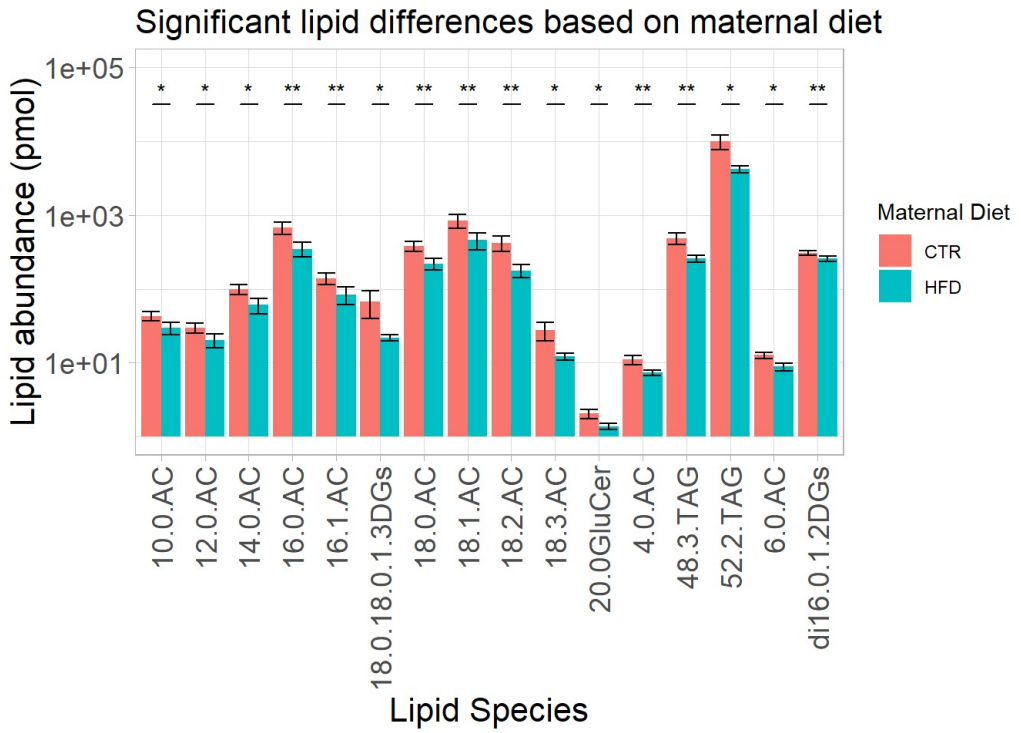
```
#Create a dataframe with the standard deviation of each lipid species subset (HFD, CTR)
plottersd <- cbind(lapply(CTR, sd), lapply(HFD, sd))
plottersd <- cbind(plottersd, row.names(plottersd))
plottersd <- data.frame(unlist(plottersd[,1]), unlist(plottersd[,2]), row.names(plottersd))
colnames(plottersd)<-c("CTR", "HFD", "Lipid species")


#Create a dataframe with the standard error of each lipid species subset (HFD, CTR)
#Standard error, standard deviation divided by square root of sample size

plotterse <- data.frame(plotter$`Lipid species`)
plotterse$CTR <- plottersd$CTR/sqrt((nrow(CTR)))
plotterse$HFD <- plottersd$HFD/sqrt((nrow(HFD)))
colnames(plotterse)<-c("Lipid species", "CTR", "HFD")

#View format
kable(head(plotterse))
```

Lipid species	CTR	HFD
di16.0.1.2DGs	21.962879	23.211479
10.0.AC	6.059984	5.543700
12.0.AC	4.641012	4.309507
14.0.AC	16.231710	14.201568
16.0.AC	122.255000	78.821969
16.1.AC	25.104386	22.237258

```
#Melt into appropriate data structure for ggplot2
plotterm <- melt(plotter[,c("Lipid species","CTR", "HFD")],id.vars = 1)
plotterse <- melt(plotterse[,c("Lipid species","CTR", "HFD")],id.vars = 1)

plotterm$se <- plotterse$value #Add SE to original plotter for ease of graphing

#View format
kable(head(plotterm))
```

Lipid species	variable	value	se
di16.0.1.2DGs	CTR	309.11538	21.962879
10.0.AC	CTR	43.04615	6.059984
12.0.AC	CTR	29.72308	4.641012
14.0.AC	CTR	99.15385	16.231710
16.0.AC	CTR	667.49231	122.255000
16.1.AC	CTR	138.80769	25.104386

```
#Plot data

plot <- ggplot(plotterm,aes(x =`Lipid species`, y = value)) +
    geom_bar(aes(fill = variable),stat = "identity",position = "dodge") +

    #Use log scale and allow for room considering error and significance bars
    scale_y_log10(limits = c(1, 100000)) +

    #Light theme
    theme_light() +

    #Adjust text size and angle
    theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5, size = 14),
        axis.text.y = element_text(size = 14), axis.title.x = element_text(size = 16), axis.title.y = element_text
(size = 16),
        plot.title = element_text(size = 16)) +

    #Adjust labels
    labs(title = "Significant lipid differences based on maternal diet", x = "Lipid Species", y = "Lipid abundanc
e (pmol)", fill = "Maternal Diet") +

    #Add error bars based on standard error
    geom_errorbar(aes(ymin=value-se, ymax=value+se), position = position_dodge2(width=.2))

#Add and position significance bars
plot1 <- plot + geom_signif(y_position=4.5,
                  xmin=c(seq(from = 0.8, to = length(row.names(plotter))-0.2, by =1 )),
                  xmax=c(seq(from = 1.2, to = length(row.names(plotter))+0.2, by =1 )),

                  # Set astrisks; * = <0.05 (based on initial cutoff), ** = < 0.01, *** = < 0.001, etc
                  annotation=c(strrep("*", floor(log10(1/Mat.W$Significance)))), tip_length=0)

plot1
```



Save your plot if you'd like and save your lipids of interest to use in WGCNA!

(Below code saves as an R object)

```
#Readjusting for size and height
png("MatSigbarplot.png", height = 900, width = 900)
plot +

#Adjust text size and angle
theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5, size = 26),
        axis.text.y = element_text(size = 26), axis.title.x = element_text(size = 30), axis.title.y = element_text
(size = 30),
        plot.title = element_text(color = "white"), legend.text = element_text(size = 26), legend.title = element_
text(size = 30)) +  #Changed to blank title


#Add and position significance bars
geom_signif(y_position=4.5,
              xmin=c(seq(from = 0.8, to = length(row.names(plotter))-0.2, by =1 )),
              xmax=c(seq(from = 1.2, to = length(row.names(plotter))+0.2, by =1 )),

              # Set astrisks; * = <0.05 (based on initial cutoff), ** = < 0.01, *** = < 0.001, etc
              annotation=c(strrep("*", floor(log10(1/Mat.W$Significance)))), tip_length=0, textsize = 10)

dev.off()
```

```
## png
##   2
```

```
# Save an object to a file
saveRDS(Mat.W, file = "Mat.W.rds")
```