2019 General Notebook

Author: Andrew D. Nguyen, Evolutionary Physiologist

Affiliation: University of Florida, Department of Entomology and Nematology

Contact: anbe642@gmail.com

Date started: 2019-01-01

Date end (last modified): 2019-09-02

Introduction:
Notebook for 2019 year. It'll log the rest of my dissertation, post doc projects, meetings, papers I've read, and general project ideas.

List of projects and description

- Hsp rxn norm: Understanding how the local thermal environment shapes thermal tolerance and stress response (using Hsps as a proxy for stress) in forest ants of the genus *Aphaenogaster*. CTmax and rxn norm of Hsp expression measured across forest ants from Fl to Maine.
- Range limits: Identifying the factors/forces that set range limits in common forest ants (*Aphaenogaster picea*). Modelling + measured their cold physiology in forest ants of Maine and Vt.
- Thermal niche paper: Collaborative paper understanding how the environment shapes the ability to withstand cold and hot temperatures. In field and in a common garden, we measured upper and lower thermal limits of ants from GA-Maine (2 species).
- Stress in nature: Are ants stressed under experimental warming that projects climate change? Ants were collected from warming chambers (0-5 C increase from ambient) and we measured their stress response.
- Biological rhythms in *Rhagoletis*: Determining the relationship between behavioral rhythms in adult *Rhagoletis* and diapause exit timing + depth(eclosion and mass specific metabolic rate).
- *Rhagoletis* diapause exit: Determine the physiological parameters that lead to divergent adult emergence patterns between two host races.
- *Rhagoletis cerasi* transcriptome: Determine the adaptive shifts in the transcriptome relating to seasonal timing in low and high altitude populations.
- Proteome stability project in *Drosophila melanogaster*: Determine the physiological tactics at the molecular level that underlie differences in thermal traits and whether they've been shaped by selection at a broad scale.

Table of contents (for 200 entries)

Page 1:

*2018 year goals**

1. Submit and publish range limits, hsp rxn norm, and thermal niche papers.

- still need to submit revisions for range limit, end of feb
- lchick working on thermal niche
- SHC and NJG sent me edits to hsp rxn norm

2. Build a data science course

- lost steam on this, didn't do

3. Solidify meta analysis ideas and start the project

- didn't work on this either

4. Learn and become more proficient at analyzing biological rhythm data; analyze Rhagoletis biological rhythm data to a point where I have a cool story to tell.

- learned wavelet analysis and fourier transformations
- learned how to estimate biological rhythms,

5. Start working on european cornborers:
    - Learn rearing, diapause biology (induction, termination)
    - experiment on behavioral rhythms
    - do experiments to understand the molecular basis of behavioral rhythms, specifically focusing on period
        - didnt have time to start this project
6. Start constructing a teaching and research statement

- started this partly

7. Apply for and secure external funding

- was looking, but didn't find any worth applying to

2019 goals

1. Submit and publish range limits, hsp rxn norm, and thermal niche papers.
2. I started working on Rhagoletis transcriptome project, so solidfy analysis and publish
3. Finish analysis for Rhagoletis biological rhythms data
4. Anlayze proteome stability data. Learn more bayesian stats in the process to account for measurement error.

It looks like goals have changed a lot. I think in 2018, I wanted to set myself up more for teaching jobs and start thinking about how to start my own lab. But for this year, I'm going to focus more on publishing.

Page 2: 2019-01-07 lynda.com data tells a great story

Story with data - don't need

ex: Andrew Moorefield started company ; could not make payroll ; he let some employees decide

He used 5 numbers to illustrate the dilemma of revenue vs salary.

Story telling technique:

1. Don't give them the answer, walked through the facts - presented with begnning middle and ending.
2. Showed, not telling .
3. Let audience draw on their own conclusions. Done telling the story, pause and let the audience to react
4. The converse is statement.

Layout:

Context -> conflict -> resolution

Method 2: Discovery Journey Story method

Walk through the ah-ha moment so that the audience can experience it.

Main character - You, not the company . Walk people through the analysis

Typical story line:

1. Recommendations
2. Reasons
3. Evidence

But you need:

1. story
2. Conclusion
3. Reccomendation

Correlation between sales and profits - good correlation for first 20 years or so(), then correlation stopped (1983-2000)

1. he paused and let the audience answer how to explain data
2. answer: mature market (market saturated)

Strategy diff between a developing market vs mature market . Audience just understood this. The audience were then responsive to the recommendations.

Effectiveness: Humans are more passionate about their own ideas than YOUR ideas.

Let audience struggle with data-- give audience the gift of discovery

---

What data science tools must you konw?

Proxmox, Hadoop, Spark, and Weka.

Intro:

Internet of things (Iot) - tons of data becoming interconnected

Goal: Data fluency - capable of diverse interpretations and creating your own (own what, I have no clue)

Data science : create new information and knowledge ; goal is to provide useful insights for better decisions

You need to know how to do cloud computing.

Build own cloud with ProxMox.

Page 3: 2019-01-10. evolution of Resistance vs tolerance and meeting list with dan

meet with dan:

1. network modules vs time : data too complicated and difficult to interpret. Trying partial correlation , need to scale up the code for each time point and population
2. Review paper on tolerance vs resistance?
3. ANBE goals for this year
    - Data science positions in academia or industry analyzing genomic data
    - publish papers

2019-01-11 meeting with Dan

We want time ordered to observe the checkered patterns(heat map): We'd expect more proportion early on in time and then less heat later in time in the cross comparisons. Who falls together early and who doesn't. Know something about the thresholding (more or less stringent is better?) How are clusters being built? Is WGNCA building clusters that is biologically informative?

Suggestion: We want to know what is diff from beginnning to end

Pairwise diff expression analyses- diff expressed 2 months **detect modules by grouping them with STEMminer ; group by stat effect , meaning take out population effects and time effect; build networks separately**

check dirrecitonality of evolution; phylogeny of cerasi species

Meet on monday, create a dream figure how do we get there? how do we parse the data?

Partial correlation: signs- just add 1 and make everhything on a positive scale !

Tolerance vs resistance

The Evolution of the resistance and tolerance to stress

Set up: Why we need to understand how organisms cope with stress.

Inform whether they are resilient or susceptible to environmental change.

Offsets from ancestral environmental conditions creates mismatches between organism and environment.

Environments can perturb animals, reducing fitness. Specifically, stress damages macromolecules, disrupting cellular activity.

Problem or Need Statement:

The way we refer to how organisms cope with stress severely impacts our understanding of the physiological and molecular strategies/tactics of stress hardiness.

For example: thermal tolerance refers to the ability of an organisms to withstand both low and high temperatures. However, coping with stress can involve not only tolerance mechanisms, but also resistance.

The field would benefit from the herbivore damage literature.

Tolerance - physiological changes in response to environmental perturbations that maintains fitness.

Resistance - physiological mechanisms that reduces damage from environmental perturbations.

A big problem in transcriptomic studies: cant tell the difference between resistance vs tolerance -- need reaction norm approach

Types of Perturbations: Press vs Pulse (Edward Bender 1980)

Press perturbations is consistent damage.

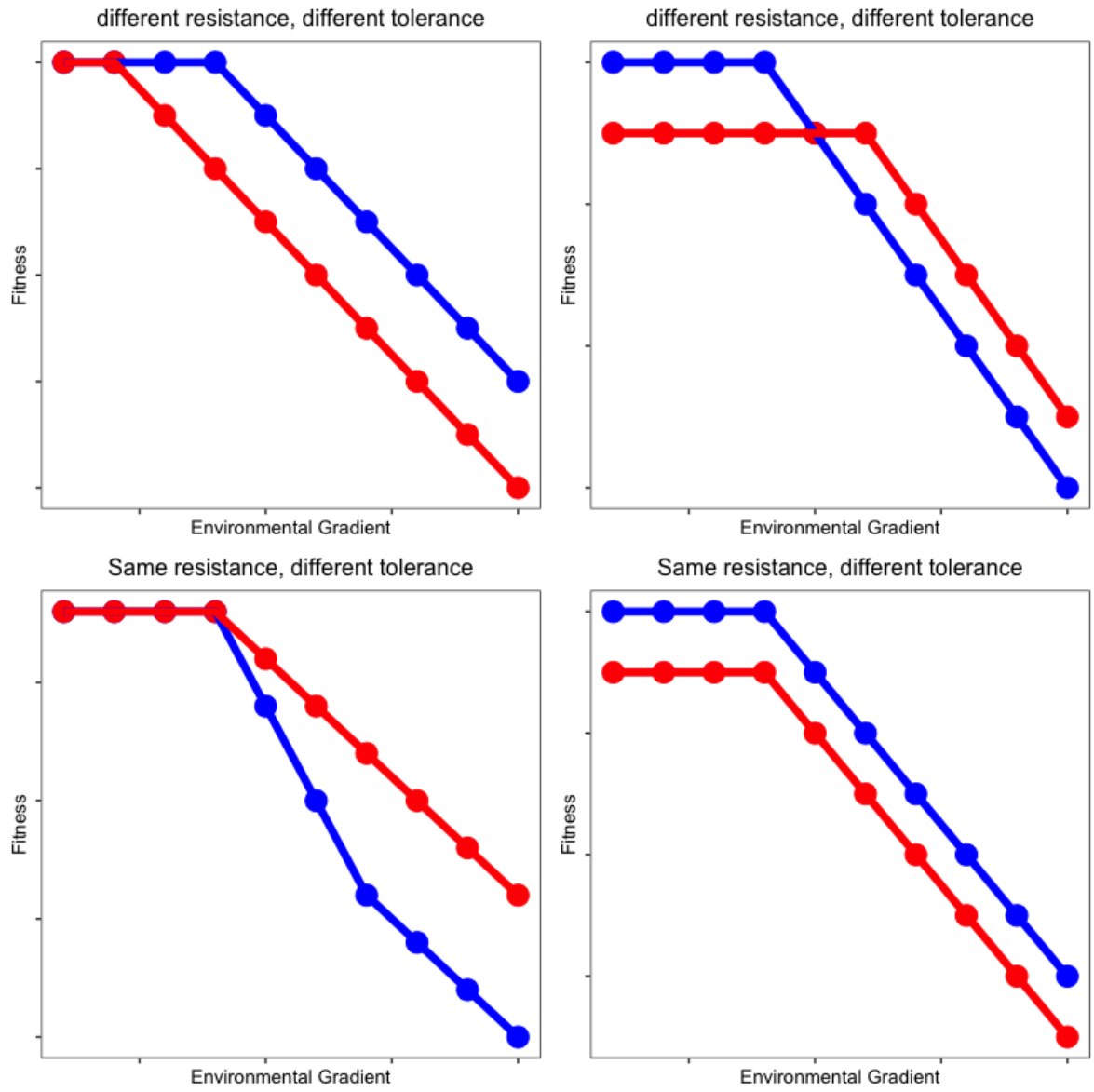
- thermal damage - slow ramping
- performance under herbivore damage
- performance under parasite load

Pulse perturbation is intermittant damage.

- thermal shock, rapid cold/heat treatment - lends itself to investigating recovery mechanisms
- initial herbivore damage
- initial parasite infection

Note: There can be things in between pulse and press: intermediate heat ramping protocols. The central read out for heat damage is proteome stability.

Types of patterns to expect from data, pulse or press



Evolutionary tactics for stress resistance

Selection favors stress resistance mechanisms when stress is constant throughout a lifetime.

Cost- allocation cost whereby investing in defenses or elevating stress resistant mechanisms comes at the cost of other life history traits such as growth.

Benefit - Takes more stress to disrupt biological activity

Tactics

- High baseline ("Front loading") investment in protective molecules.

Evolutionary tactics for stress tolerance

Selection favors stress tolerance mechanisms when stress is variable within a lifetime.

Cost- costs energy to turn on a response

Benefit- the response enables the organism to cope with the environmental condition

Tactics

- When perturbed, increase magnitude of protective molecules.

Molecular level :

Phenotype: environmental limits (thermal limits, drought limits,)

Resistance - Environmental range where key macromolecules do not change

Tolerance - Environmental range wherw macromolecules changing during and after

So should we be thinking about what a particular molecule is doing for the organism? THe molecule itself is resistant or tolerant?

Is this just understanding phenotypic plasticity of a given molecule?

Page 4: 2019-01-10. Partial correlation code to test out

```
#Andrew Nguyen
#partial correlation matrix

#########################################################################
# load libraries

library(ggplot2)
library(data.table)
library(WGCNA)
library(edgeR)
library(tidyr)
library(dplyr)
library(reshape2)
library(igraph)
library("GeneNet")
#library(qgraph)

#load dataset

#myfiles.wide.4<-fread("../Data/CerasiCountsIsos/03_data_set_2018-10-
31_wide_filtered_sig_genes.csv")
myfiles.wide.4<-fread("../Data/03_data_set_2018-10-
31_wide_filtered_sig_genes.csv")
### Calculate FPKM
table.dge<-DGEList(myfiles.wide.4[,3:43],genes=myfiles.wide.4[,1])
table.dge<- calcNormFactors(table.dge) # running this gives FPKM
#https://support.bioconductor.org/p/79379/
```

```
tab.fpkm<-rpkm(y=table.dge,gene.length=myfiles.wide.4$Length)
tab.fpkm<-data.frame(tab.fpkm)
tab.fpkm<-data.frame(Name=myfiles.wide.4$Name,tab.fpkm)
tab.fpkm$from<-seq(1,length(tab.fpkm$Name))
#names(tab.fpkm)
#'
#' # Compute Partial Correlations and Select Relevant Edges

#pcor.dyn = ggm.estimate.pcor(t(log2(tab.fpkm[1110:1130,3:6]+1)))
pcor.dyn = ggm.estimate.pcor(t(log2(tab.fpkm[,3:6]+1)))
arth.edges = network.test.edges(pcor.dyn,direct=TRUE,plot=FALSE)
#arth.net = extract.network(arth.edges, method.ggm="number", cutoff.ggm=50)
arth.net = extract.network(arth.edges, method.ggm="number",
cutoff.ggm=length(tab.fpkm$Name))
dim(arth.edges)

####
#node.labels = as.character(1:nrow(tab.fpkm))
node.labels = as.character(1:length(tab.fpkm$Name))
gr = network.make.graph(arth.edges, node.labels, drop.singles=FALSE)
#adj<-as(gr,"matrix") # convert graphNEL into adjacency matrix

#convert to igraph
gg<-igraph.from.graphNEL(gr)
##############################################
###network node properties
#http://kateto.net/networks-r-igraph
##############################################
#density
edge_density(gg) # proportion of present edges from all possible edges in the
network

#reciprocity - proportion of reciprocated ties ( for a directed network)
#reciprocity(gg)

##global transivity- ratio of triangles to connected triples (direction
disregraded)
#transitivity(gg,type="global")

#diameter: longest geodesic distance (length of shortest parth between two
nodes) in a network
#diameter(gg,)

#degree distribution
#deg.dist <- degree_distribution(gg, cumulative=T, mode="all")

## average path length
#mean_distance(gg,directed=TRUE)


### assortativity
#assortativity_degree(gg, directed=T)

#assortativity_nominal(gg,myfiles.wide.4$sig)


##############################################
###single node properties
```

```
################################################
###local transviity - ratio of triangles to connected triples each vertex is
part of
#transitivity(gg, type="local")

###Grab different measures of centrality

################################################
#eigen centrality - proportional to the sum of connection centralities
#eig<-centr_eigen(gg)$vec
#range(eig)
#strength
#st.all<-strength(gg,mode="all",loops=FALSE) # all directions, in and out
st.in<-strength(gg,mode="in",loops=FALSE) # in degree strength
#st.out<-strength(gg,mode="out",loops=FALSE) #out degree

#degree centrality
#d.all<-degree(gg,mode="all"); range(d.all)
d.in<-degree(gg,mode="in")
#d.out<-degree(gg,mode="out")

d<-data.frame(st.in,d.in)
fwrite(d,"2019-01-10_centrality_partial_networks.csv")
#betweeness
```

running on the hipergator cluster : 04_cluster_script_partial_correlation.sh

```
#!/bin/bash
#SBATCH --job-name=partial_correlation
##SBATCH --mail-user=andrew.nguyen@ufl.edu
##SBATCH --mail-type=ALL
#SBATCH --output=partial_correlation_job-%j.out
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=4
#SBATCH --mem=110gb
#SBATCH --time=96:00:00
#SBATCH --partition=bigmem
#SBATCH --account=dhahn
#SBATCH --qos=dhahn-b
date;hostname;pwd

module load R

cd /ufrc/dhahn/andrew.nguyen/Cerasi_Networks/Script

Rscript 03_2018-12-13_partial_correlation_network_analysis.R

date
```

Page 5: 2019-01-15. Biological Rhythms project: data note on trikinetics experiment

Changes are on github, referencing the the commit ---

power outage last night 2019-01-14 saving raw trik data

Andrew Nguyen committed 6fa7221

```
189134  14 Jan 19   18:28:00 (last reading)
1   15 Jan 19   09:22:00 (start reading after power outage)

How to deal with data:
1. just analyze to 2019-14-19 18:28:00 and make sure the flies arent being
analyzed for mortality

2. Keep it going and use the Lomb-Scargle method.

We can do both if I keep the flies in the trikinetics set up.
```

Page 6: 2019-01-17. module detection script update with WGCNA

Critical changes

- filtering out category based on statistical significance: genes with only population by time interaction from edgeR model
- messing with cutheight = 0.15
- messing with module stringency with "deepSplit" specificatin in the cutreeDynamic() function.

```
#########################################################################
# Author: Andrew Nguyen, post doc
# University of Florida
# Hahn lab
# Initiated : 2018-11-29
# Last updated: 2019-01-17

#This script takes a data set that is in wide format, converts it to long,
#then applies a function to estimate network centrality for different subsets fo
the data

#########################################################################
# load libraries

library(ggplot2)
library(data.table)
library(WGCNA)
library(edgeR)
library(tidyr)
library(dplyr)
library(reshape2)

#########################################################################
#constructing a function to estimate modules of a network for each time point in
a
# time series
#########################################################################
```

```
### WGNCA
#lets write a fucntion that spits out this output

#d<-tab.fpkm.long%>%
#  dcast(formula=Name~sample,value.var="gxp")


wgcna.mod.det<-function(data=data){
  data<-data%>%
    reshape2::dcast(formula=Name~sample,value.var="gxp")
  adjacency = adjacency(t(data[,-1]), power =1); #adjacency matrix
  #adjacency = adjacency(t(dat.wide[,4:7]), power =1); #adjacency matrix
#topologicla overlap matrix
  TOM = TOMsimilarity(adjacency);
  dissTOM = 1-TOM
  geneTree = hclust(as.dist(dissTOM), method = "average");

#identifying modules using dynamic tree cut
  dynamicMods = cutreeDynamic(dendro = geneTree, distM = dissTOM,
                              deepSplit = 4, pamRespectsDendro = FALSE,
                              minClusterSize = 100);

  dynamicColors = labels2colors(dynamicMods)
  table(dynamicColors)
#calculate eigengenes and then cluster modules eigengenes that are similar
  MEList = moduleEigengenes(t(data[,-1]), colors = dynamicColors)
  #MEList = moduleEigengenes(t(dat.wide[,4:7]), colors = dynamicColors)
  MEs = MEList$eigengenes
  #MEs[is.na(MEs)] <-0
  #MEDiss = 1-cor(MEs);
  #METree = hclust(as.dist(MEDiss), method = "average");
#sizeGrWindow(7, 6)
#plot(METree, main = "Clustering of module eigengenes",
 #    xlab = "", sub = "")
# merge modules?
  merge = mergeCloseModules(t(data[,-1]), dynamicColors, cutHeight =.15, verbose
= 3)
  #merge = mergeCloseModules(t(dat.wide[,4:7]), dynamicColors, cutHeight =.25,
verbose = 3)
  mergedColors = data.frame(module=as.factor(merge$colors))

  return(mergedColors)
}



#############################################################################
#data parsing start
#############################################################################

### Load data
tab.fpkm.long<-fread("../Data/03_2019-01-15_tab_fpkm.csv")
#tab.fpkm.long<-fread("../Data/01_2018-12-11_testdata_tab.fpkm.long.csv")
tab.fpkm.long<-tab.fpkm.long[,-1]
#log2 transformation to make gxp normal
tab.fpkm.long$gxp<-log2(tab.fpkm.long$gxp+1)
```

```
third.dat<-fread("2018-10-25_full_sig_list_edgeR_model.csv",header=TRUE)
third.dat<-third.dat%>%
  select("Name","sig")
tab.fpkm.long<-inner_join(tab.fpkm.long,third.dat,by="Name")
tab.fpkm.long<-tab.fpkm.long%>%
  arrange(sig)
glimpse(tab.fpkm.long)
#head(tab.fpkm.long)
###take mean and std of gxp

tab.fpkm.long.mean<-tab.fpkm.long%>%
  filter(sig=="Pop_int_Time_effect")%>%
  group_by(Name,population,time,sig)%>%
  summarize(gxp.mean=mean(gxp),gxp.sd=sd(gxp))

################################################################################
#data parsing end
################################################################################


################################################################################
#### apply function to whole dataset


#modules<-tab.fpkm.long%>%
#  group_by(population,time)%>%
#  do(wgcna.mod.det(data=.))

modules<-tab.fpkm.long%>%
  filter(sig=="Pop_int_Time_effect")%>%
  group_by(population,time)%>%
  do(wgcna.mod.det(data=.))


nn<-tab.fpkm.long.mean%>%
  filter(sig=="Pop_int_Time_effect")
#modules$Name<-rep(tab.fpkm[1:500,1],10)
#adding back in name column so we can join the datasets with the mean and std
dataset
#tab.fpkm.long.mean dataset *
#modules$Name<-rep(unique(tab.fpkm.long[10000,]),10)
#modules$Name<-rep(unique(tab.fpkm.long$Name),10)
modules$Name<-rep(unique(nn$Name),10)

################################################################################



#joining datasets based on name population and time
data.set<-
inner_join(tab.fpkm.long.mean,modules,by=c("Name","population","time"))
# write out the dataset
write.csv(data.set,"05_2019-01-
17_WGCNA_time_series_modules_per_timepoint_log2_transformed_point15cutoff.csv")
```

```
library(plyr)
#library(rlist)
biglist<-dlply(data.set,.(population,time,module),c()) # create a list from the
"data.set"
str(biglist)


#create a list so that we can extract only the Name fromt he list
all_vectors <- list()

for(i in seq_along(biglist)) {
 all_vectors[[i]] <- dplyr::pull(biglist[[i]], Name)
}
names(all_vectors)<-names(biglist) # get the names back to the all_vectors
#str(all_vectors)

#https://codereview.stackexchange.com/questions/17905/compute-intersections-of-
all-combinations-of-vectors-in-a-list-of-vectors-in-r/17931#17931
#overlap function
overlap <- function(l) {
  results <- list()
  # Remove duplicates within each entry of l
  l <- lapply(l, unique)

  # combinations of m elements of list l
  for (m in seq(along=l)) {

    # generate and iterate through combinations of length m
    for (indices in combn(seq(length(l)), m, simplify=FALSE)) {

      # make name by concatenating the names of the elements
      # of l that we're intersecting
      name <- paste(names(l)[indices], collapse="_")

      results[[name]] <- Reduce(intersect, l[indices])
    }
  }
  results
}

#getting pariwise comparisons
nms <- combn( names(all_vectors) , 2 , FUN = paste0 , collapse = "-" , simplify
= FALSE )

# Make the combinations of list elements
ll <- combn( all_vectors , 2 , simplify = FALSE )

# Intersect the list elements
out <- lapply( ll , function(x) length( intersect( x[[1]] , x[[2]] ) ) )
n<- lapply( all_vectors , function(x) length(x ) )# Output with names
tot<-data.frame(n=unlist(n))
tot$ref_treatment<-rownames(tot)
tot2<-data.frame(n2=unlist(n))
tot2$ref_treatment2<-rownames(tot2)
#tot
```

```
###making the dataset for modules
isec.dat<-data.frame(intersection=unlist(setNames( out , nms )))
isec.dat$combo<-rownames(isec.dat)
#isec.dat

isec.dat$ref_treatment<-unlist(lapply(strsplit(isec.dat$combo,split="-
"),`[[`,1))
isec.dat$module1<-
unlist(lapply(strsplit(isec.dat$ref_treatment,split="\\."),`[[`,3))
isec.dat$population<-
unlist(lapply(strsplit(isec.dat$ref_treatment,split="\\."),`[[`,1))
isec.dat$time<-
unlist(lapply(strsplit(isec.dat$ref_treatment,split="\\."),`[[`,2))
isec.dat$ref_treatment2<-unlist(lapply(strsplit(isec.dat$combo,split="-
"),`[[`,2))


isec.dat<-inner_join(isec.dat,tot,by="ref_treatment")
isec.dat<-inner_join(isec.dat,tot2,by="ref_treatment2")
#head(isec.dat)
isec.dat$prop1<-round(isec.dat$intersection/isec.dat$n,2)
isec.dat$prop2<-round(isec.dat$intersection/isec.dat$n2,2)


write.csv(isec.dat,"05_2019-01-
17_modules_intersection_dataset.15cutheight_log2_transformed.csv") # write out
dataset
```

Page 7: 2019-01-21. paper readings for resistance and tolerance

Evolution of plant resistance and tolerance to frost damage; Ecology letters; Agrawal et al. 2004

Background:

2 strategies to combat perturbations

Their definitions:

- Resistance : traits that reduce damage
- Tolerance : traits that reduce the negative fitness impacts of damage

They studied how a frost event late in the season impacted a 75 parental half sibling family experimental quantitative genetic design in an annual wild radish, southern Ontario.

Big Questions

What types of strategies do plants employ to combat frost damage?

Hypotheses:

They can either mainly resist or tolerate frost damage.

Experimental approach :

Frost damage = newly wilted or dead tissue. Proportion leaf damage were visually estimated on each leaf. Ulatimely transformed into a proportion leaf area damage over entire plant.

Measures of -

- Resistance = 1 - damage
- Tolerance = for each parental half sib family as slope of regression between fitness on proportion frost damage.

Main Results

- No trade off in resistance vs tolerance
- Negative correlational selection acting on two traits: selection favored high resistance combined with low tolerance ; low resistance and high tolerance.

Figure 2. fitness is negatively related to tolerance ; so there is a cost to fitness

Conclusion:

Plants can use both strategies. selection operates on maximizing one while not the other. This could be because there is a trade off between the two mechanisms. However, at the population level, there was no trade off, so it could be that frost damage is rare and this selection event doesn't happen often so that the quantitative genetic architecture may not show the trade off at the population level. Or the two different strategies persist and that is how variation is maintained.

EVOLUTIONARY GENETICS OF RESISTANCE AND TOLERANCE TO NATURAL HERBIVORY IN ARABIDOPSIS THALIANA; Evolution; Weinig et al. 2003

Background :

measured resistacne and tolerance to natural apical meristem damage by rabbits in a large field experiment with RILS of arabidopsis. Measured phenological and morphological traits associated with resistance and tolerance

RILs differeed in resistance (proportion of replicates within a RIL that resisted herbivory)

Resistance = reduce amount of damage suffered Tolerance = reducing fitness consequence of damage

They expect a trade-off but empirical evidence shows a mix of results.

Cost of resistance and tolerance may be a way to maintain variation .

Questions:

- (1) Does a genetic trade-off exist between resistance and tolerance to rabbit herbivory in A. thaliana?
- (2) What are the ecological and developmental mechanisms underlying resistance and tolerance to rabbit her- bivory?
- (3) Is tolerance to rabbit herbivory costly in the ab- sence of damage?
- (4) What is the pattern of natural selection acting on resistance and tolerance in this species, and how sensitive are those patterns to the mean level of resistance and tolerance exhibited by the population?

Experimental approach:

Resistance to AMD = 1 -p ; p is the proportion of individuals that suffered AMD. Tolerance = We operationally defined tolerance for each RIL as (WD □- WU), the difference between mean relative fitness of plants with AMD and those without AMD

One issue they ran into: no herbivore exclusion treatment to serve as control.

Results:

Resistance and Tolerance were uncorrelated .

Effect of AMD on phenotypic traits:

- AMD increased branch production, lowered average height of basal branches, and led to later time to senescence.

Numbering the hairs on our heads: The shared challenge and promise of phenomics; PNAS; Houle 2009

medicine and evolution share a common theme in that they both try to understand the genotype-phenotype map.

It is easy to measure genomes, but it is harder to measure all phenotypes.

Imbalance between G space (genotypes) and P space (phenotype space)

Phenomics: the next challenge; Nature Genetics; Houle et al.

> We now define phenomics as the acquisition of high-dimensional phenotypic data on an organism-wide scale

Or...All of the different characteristics of an organism.

Justification of studying:

- trace causal links between genotypes and environmental facts and phenotypes.
    - G-P map.
        - metaphor in which genotypic information influences the phenotype of an organism
        - Jim Burns in 1970 proposed linking population genetic data with biochemical variation *
    - Pleiotropy- how a gene affects multiple phenotypes.
- Genetic basis of complex traits
    - Gwas studies- try to gauge hwo much genetic variance there is on traits
    - predict disease,
- Causal explanation of phenotypes
    - we dont know which traits are important, but if we measure a bunch of them, we can determine that *

Goals and Technical challenges

2 ways to be more comprehensive:

1. sample a wide variety of phenotypes - phenotyping
2. define extensive phenotyping as chracterizing a phenotype in great detail

- Sample gene expression in a tissue through time
- increase quantitative information by phenotype measures

Data challenges

- Overfitting the data: Too many variables that outnumber sample size ; large p, small N problem
- Dimension reduction of phenotypes - pca, discriminant analysis
- Ridge and LASSO regressions, fit models without dimension reduction int he case of large p, small N
    - model complexity is penalized with cross validation
- dimension reduction can discard information
- use machine learning techniques like random forests, regression tree

Causally cohesive models

G-P maps extend across all biological levels of organization and are highly non-linear.

Dynmic models can account for hierarchy, space, and time between genetic variation and phenotypes. ex: tooth shape in mammals.

Phenomic tools:

- Transcriptome and Epigenome.
- Proteomics and metabolics.
- behavior
- Imaging

basically measure stuff at different levels of organization

Page 8: 2019-01-24. Schneider and Ayres 2008; Nature Review Immunology;

Title: Two ways to survive infection, what resistance and tolerance can teach us about treating infectious diseases

Overall, they argue that resistance is well known, but tolerance is less well known

Intro

definitions :

resistance - ability to limit pathogen burden tolerance - ability to limit the health impact of a given pathogen burden

- Tolerance includes all mechanisms that regulate the self-harm that can be caused by an immunse response (aka bystander damage or immunopathology) and other mechanisms not directly related to immune resistance.
- Host's defense capacity = resistance + tolerance
- Well-known : molecular mechanisms that kill pathoges, prevent infection
- less well-known: how hosts regulate production, repair, and avoidance of damage that accumulates during infection
- Big picture - understanding this stuff (resistance and tolerance of pathogen damage) will inform treatment and diagnosis of diseases.

resistance and tolerance in plants

- two-component defense response originated in plant literature .

- Displayed with reaction norms of fitness against environmental gradient.
    - env gradient can be pathogen, damage induced by pathogen, pathogen load, other host response , actual number of pathogens in host
- resistance defined as inverse of parasite burden **Does this make sense**?
    - when resistance increases, pathogen load will decrease
- tolerance is measured by the slope of the reaction norm - the more flat the slope, the more tolerant (host)
- idea of tolerance is under-represented in vertebrate immunology

What are mechanisms of tolerance ?

- people know how to describe resistance mechanisms:
    - immune recognition
    - production of effectors , interaction with immune cells
    - **How does a pathogen cause damage?** What is the read out? Does the pathogen just take resources from the host? It take up space? It doesn't let the host operate optimally?
    - frame issues in terms of how a mechanism impacts tolerance/resistnace
        - consider mechanism that affects tolerance if it impacts the slope of the tolerance curve
        - assuming this is the same for resistance

Class one:

- effector molecules that induce resistance mechanims that cause self harm and result in decrease in tolerance
- resistance and tolerance are opposite
- ex- ROS produced during immune response are important for fighting infection, but can cause immunopathology and lead to death (decreasing tolerance)
- Selection favors less toxic effectors and receptos in the immune response; hosts have less toxic efefctors (i dont get this)
- seelction for receptors that triggure immune responses (Toll-like receptors) - high affinity of receptors for pathogen-associated molecules than for self molecules.
- Take home - the thing that lowers pathogen load (some effector) can decrease tolerance due to self harm.

Class two

- regulators of both resistance and tolerance
- typically signalling molecules

Class three

- Tolerance and resistance can be separated : 5 examples
    - touting as the best type of mechanism for finding new drugs and treatments that modulate tolerance

1. immune response to infection - toxic compounds produced by pathogen must be dealt with to prevent damage to host
2. resistance can be energetically expensive. Fruit flies alter energy use and wasting of body when infected by mycobacterium due to decreased tolerance.
3. preventing physiological damage can affect tolerance. immune responses can induce physiological changes that are deleterious for some organs. Sepsis can induce fatal

chagnes in cardiovascular phys
4. Repair mechanisms: if pathology cannot be prevented, then they must repair tissue damage.
5. genetic traits that increase defenses against malaria.

Tolerance mechanisms in invertebrates

- typically measure pathogen load or antimicrobial activity represent effects of infection on host fitness
- IMD (toll and immune deficiency) signalling pathways in fruit flies.
- no correlation between bacterial titres(resistance) and survival. Genotypes with lowest titres were not necessarily the healthiest ; proceeses other than resistance are used (but if they are already infected, they could already be perturbed) ;

They keep referring to tolerance mechanisms but it really is unclear what the hell they're talking about They're not matching the molecular processes with the framing they had earlier on. For example, they're not walking the reader through how different molecular mechanisms can increase the slope of the relationship between fitness and damage. Very unclear.

Tolerance properties in vertebrate models

- Ex: plasmodium chabaudi ; Råberg et al. applied stat framework of reaction norms by plant ecologists to vertebrate model. Infected 5 strains of mice with plasmodium and tested 3 different clones of plasmodium
    - measured host health by severity of anaemia and weight loss in infected mice ; plotted against peak parasite density(parasite burden) . Variation in resistance between strains
    - the slopes of the reaction norms (tolerance) also differed among strains
    - negative correlation between tolerance and resistance
- tick- lyme disease example ; mouse model gets arthritis
- molecular mechanism example: mice that are dificient in a specific ATP-sensitive potassium channel was found to be more sensitive to LPS--they'd get heart attacks when challenged to it
    - viral infection in K_ATP deficient mice had low cytokine production and abrupt death. The hypothesized mode of action was that the KATP channel prevents vasoconstriction in the arteries of the heart.
    - argued as a tolerance mechanism that prevents damage

Tolerance properties in humans

- Malaria example

Studying tolerance systematically

- argues to broaden perspective
- can broaden with genetic screens
    - some mutant deaths correlates wtih bacterial burden -defects in resistance
    - other mutant deaths had comparable levels of bacteria to wild type flies ; defective in tolerance
    - future screens need to categorize reistance vs tolerance and have gain and loss of function mutations
- genetics only part of story-- test different pathogens

Medicine and Tolerance

- manipulate tolerance mechanisms - administer corticosteroids in combo with antibiotics to decrease risk of mortality and hearing loss
- determine a person's reaction norm

Page 9: 2019-01-31. test script on hipergator

```
#!/bin/bash
#SBATCH --job-name=Test_R_script
##SBATCH --mail-user=andrew.nguyen@ufl.edu
#SBATCH --mail-type=ALL
#SBATCH --output my_job-%j.out
#SBATCH --nodes=4
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=4
#SBATCH --mem=120gb
#SBATCH --time=72:00:00


date;hostname;pwd

module load R


cd /home/andrew.nguyen/Cerasi_Networks/Script

Rscript 03_test.R
```

Page 10: 2019-02-05. Stem Miner analysis with cerasi data

Using STEM-miner http://www.cs.cmu.edu/~jernst/stem/ to identify clusters of gene expression through time for highland and lowland cerasi populations

- paper

It is a GUI based program, so I'll need to log what I'm doing here

Settings

- Log normalized data
- Gene annotation source = flybase for Drosophila melanogaster
- Maximum number of model profiles = 100

I want to try to identify as many clusters as possible

Dataset

Sample data set, where you need the gene ID (using flybase ids for genes) and then the columsn are the different gxp values with different times) . Gxp is already log2 normalized.

```
Gene        2M_0        2_5M         3_5M        4M_0        4_5M
1 FBgn0027495,FBgn0033633,FBgn0036282,FBgn0033427  1.7700410  1.9639728
1.76863459  1.8877797  1.7986530
2                                    FBgn0037941 -0.5193285 -0.1164224
-0.33351716 -0.3594560 -0.4406869
3                                    FBgn0263782  0.8824218  0.8587433
0.69634484  0.7132860  0.7337638
4                                    FBgn0043854 -0.5152793 -0.4016565
-0.34947210 -0.2898082 -0.2252473
5                                    FBgn0262534 -0.5271620  0.0638229
-0.04790487 -0.5071678 -0.1017322
6                                                -0.3525314 -0.5961887
-0.29617612 -0.8444862 -0.4384412
```

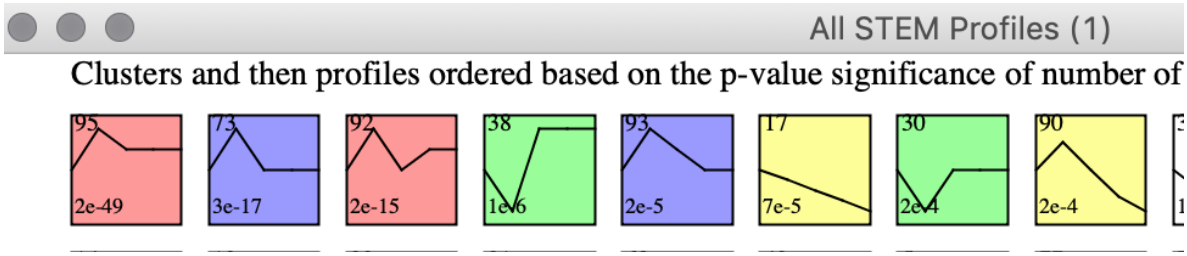
Approach

- get results from Highland
- get results from lowland
- compare highland (reference) with lowland

Results

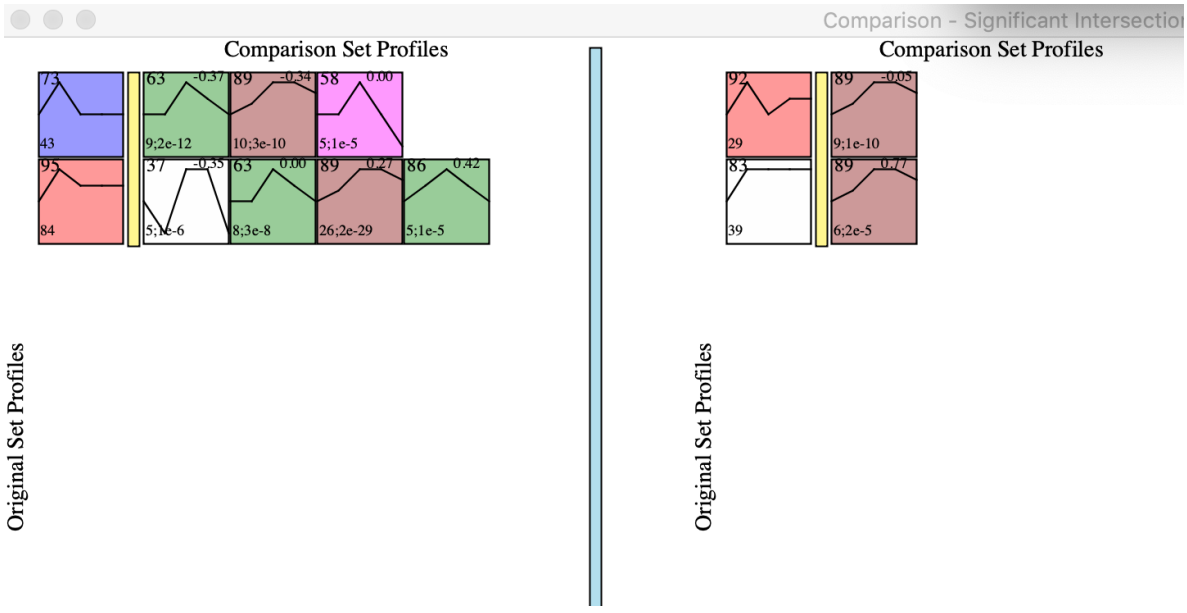
- get results from Highland

sig cluster IDs : 95, 73, 92, 38, 93, 17, 30, 90



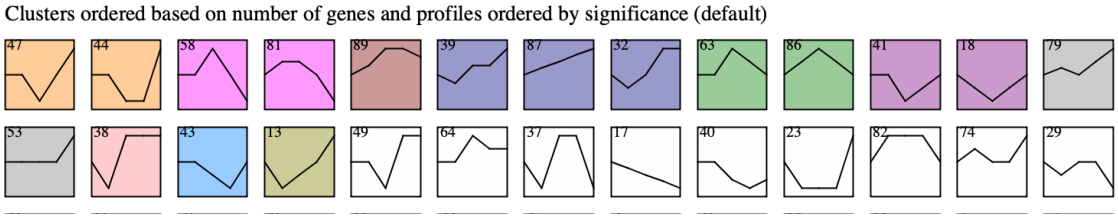
- compare highland (reference) with lowland

high 73: low 58, 63, 89 high 83: low 89 high 92: low 89 high 95: low 37, 63, 86, 89

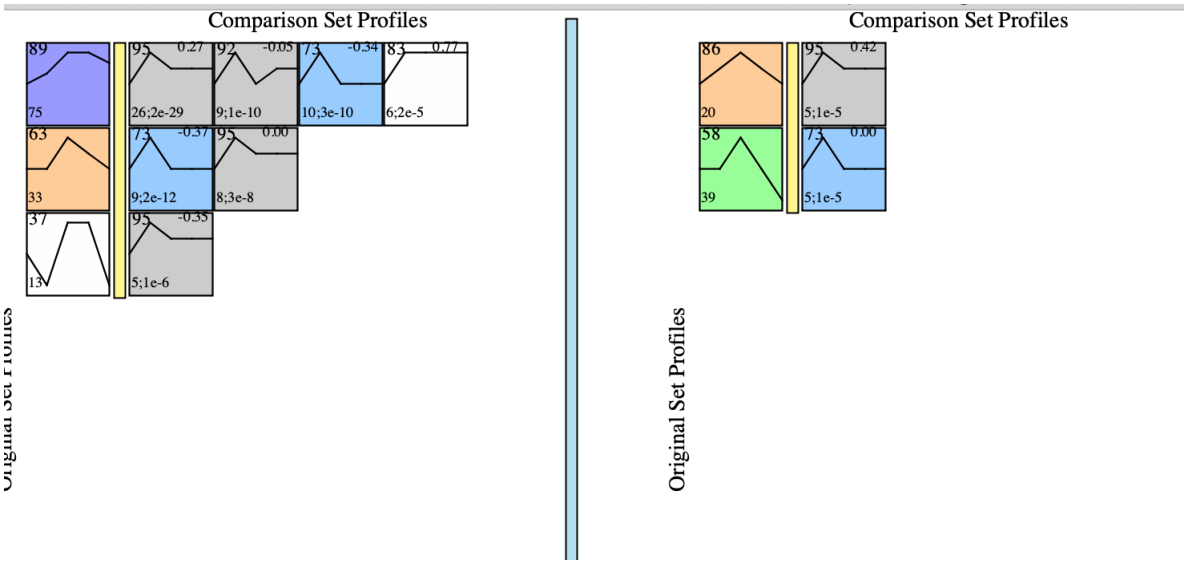


- get results from lowland

47, 89, 63, 58, 44, 79, 81, 41, 53, 39, 13, 86, 87, 38, 32, 18, 43



- compare lowland (reference) with highland



Page 11: 2019-02-11. Meeting notes with Dan, 2019-02-08

Met with Dan and showed the results from STEM-MINER.

Dan's thoughts

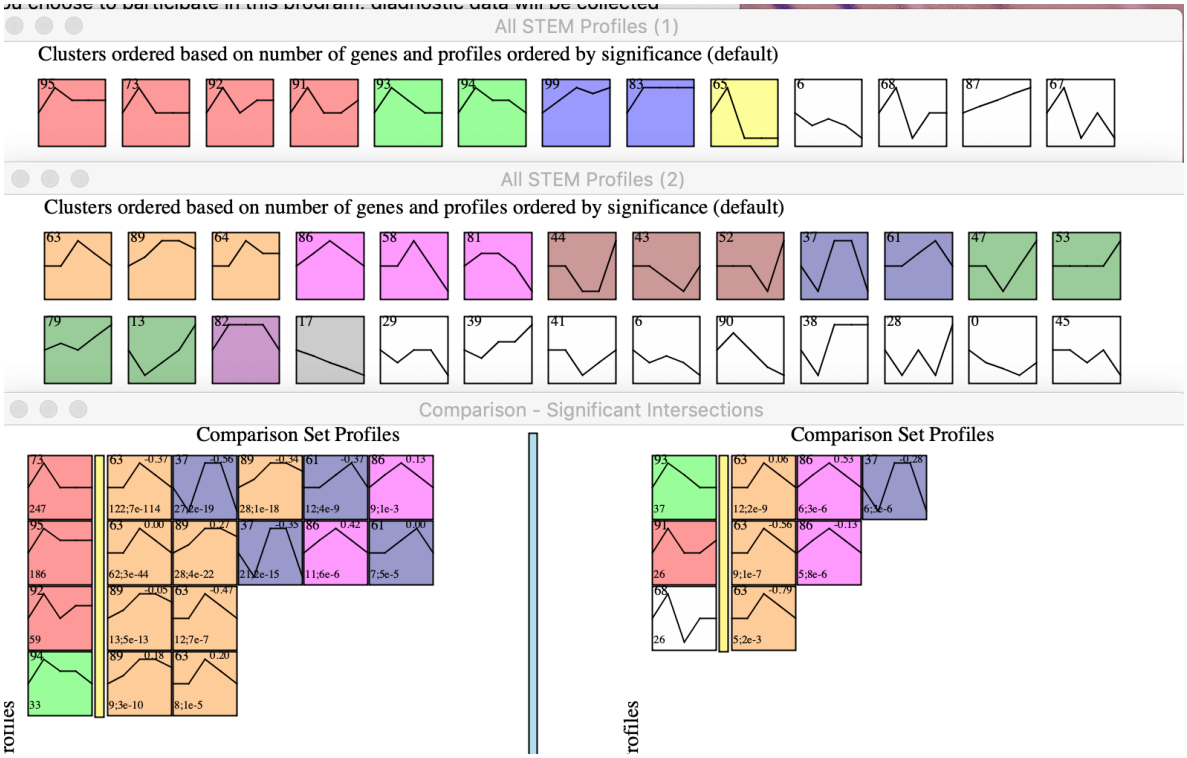
- What are the genes?
- integrate data into the flyer interactome -- which ones are actual hubs and not hubs from those data? Does it match up ?
- Next step: we have gxp vs time, try clustering based on strength vs time.
    - Hubs - gxp has some pattern, strength matches pattern or is consistently high (always hubs)
    - Effectors (downstream?): - gxp has some pattern ; strength always low throughout time

Page 12: 2019-02-13. redo analysis with STEM miner with cerasi data; no flybase annotations

Problem with previous analysis: When I tried to match up the names with the larger annotation file that has flybase IDs, the data shrunk for some reason. So for this run, I'm just taking the genes without the flybase gene IDs. It has 13,857 genes.

Parameters: 100 profiles, no annotations, normalized data option

- The top part is High profile.
- The middle part is Low profile
- bottom part is the comparison



Result: The profiles look overall the same.

Page 13: 2019-02-19. STEM miner analysis on strength dataset

THe data: We have the strength (sum of weights of the edges into a given node, aka, gene in this case) of each gene across 5 time points and for 2 different populations

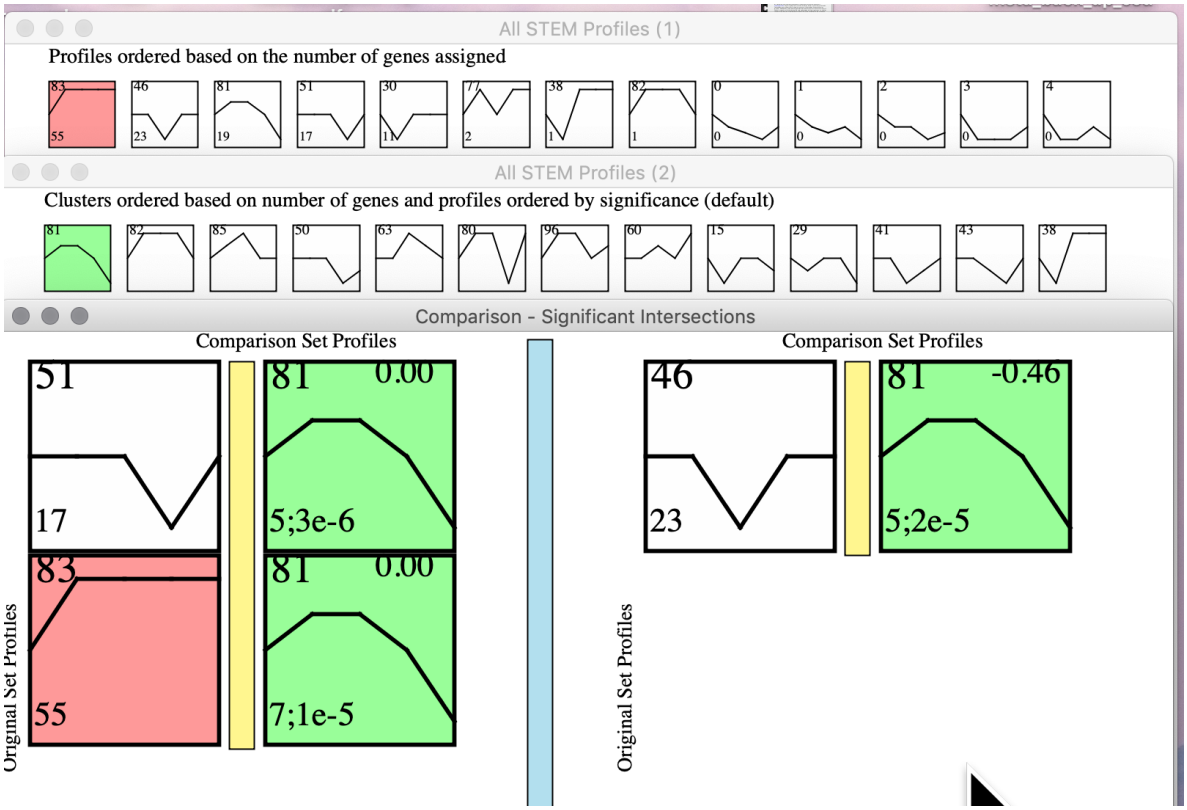
We want to know how sets of genes significantly cluster in their pattern through time.

To calculate strength, we simply calculated the sum of the rows (or columns) of an adjacency matrix (estimated for each time point and each population). Then, we log10 normalized the data.

Used the STEM: miner program:

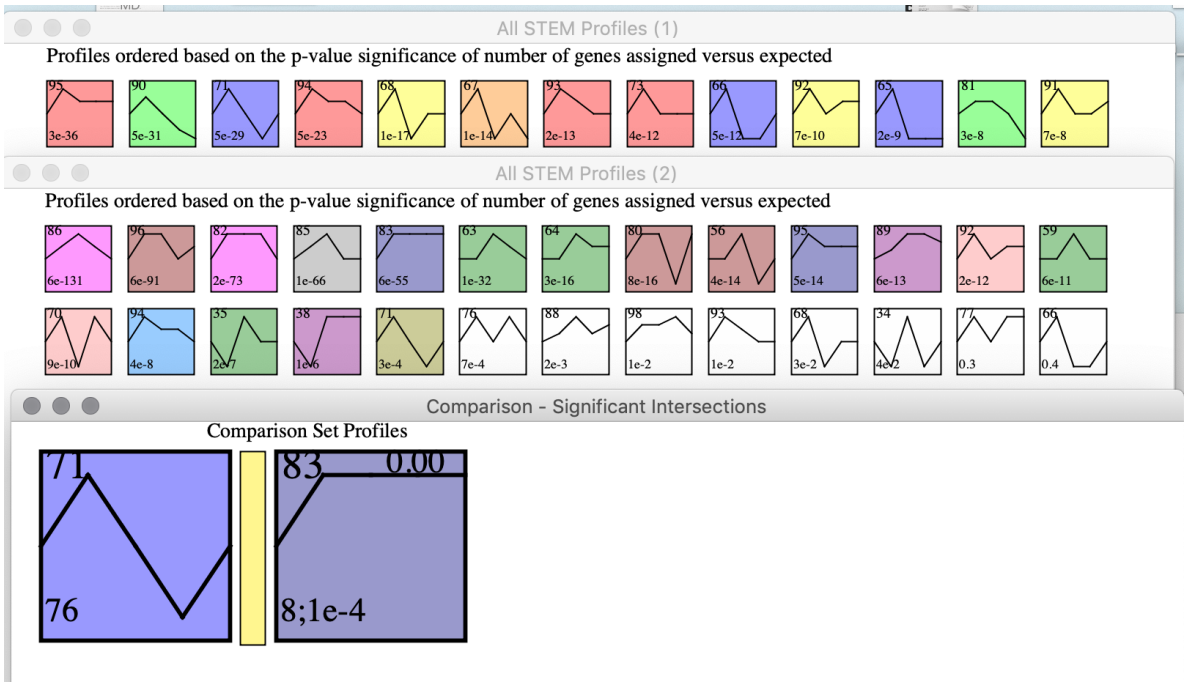
Settings - noramlized data, 100 profiles , no annotations

Results: top is high, middle is low, and the bottom is the comparison (high left, low right)

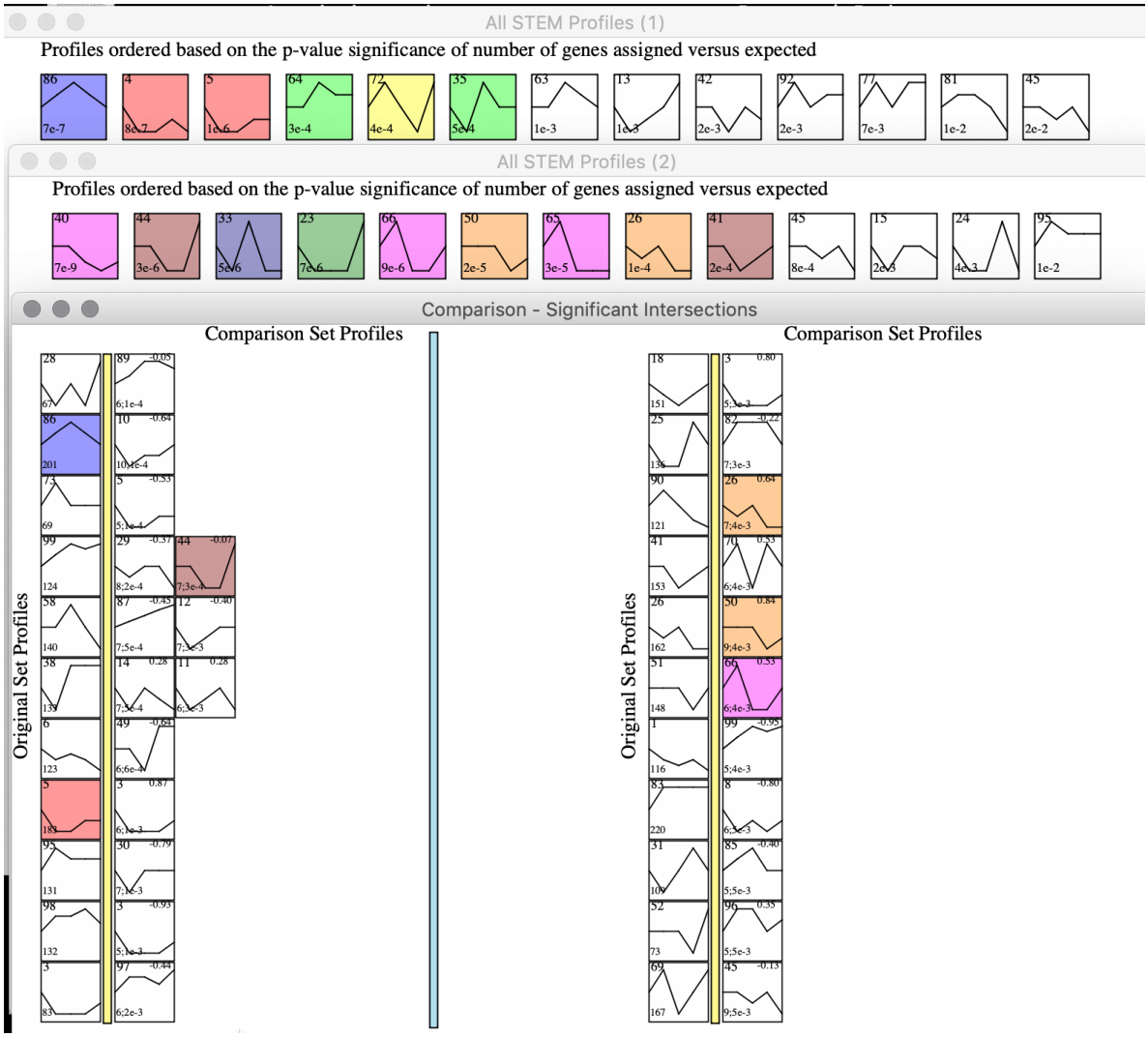


potential issue: there are some time points for genes where the weight was zero, which might skew what profiles are significant. Try removing the zeros.

removing zeros makes no difference; but I did try scaling the data.



When I take out the zeros and scale:



I think this is the best way to go. It gives a more accurate picture of the different types of shapes. The data ranged from 0 to 7000 when zeros left in, but when taken out, data ranged 5000-7000. The 0's may bias expression profiles that are low in abundance and mask effects between 5000-7000 strength.

Page 14: 2019-04-19. Cerasi/pomonella brain transcriptome project: Organizing ideas

Quick update on project: I'm trying to construct a story of the molecular architecture of diapause development. Diapause development is an alternative life history strategy that allows organisms to survive unfavorable conditions by suppressing metabolic rate and increase their stress hardiness. Mainly, there are 3 phases (Kostal et al. 2017): initiation, maintenance, and termination. Because diapause is dynamic, we need to measure gxp through time to capture the progression of phases. We want to compare 2 different species and their ability to shift their seasonal timing. Do they shift their seasonal time with the same molecular architecture? (Cerasi and Pomonella)

So far, I've analyzed cerasi- splitting classes of responses into population differences (population effect), common (time effect), and divergent responses(time by population interaction; also main effect of time and population).

- we find major offsets in the opposite direction, such taht the earlier ecloser actually expresses sets of genes later, suggesting that they upregulate genes prior to termination.

I need to do the same for pomonella. So, far, we find a similar offset.

Some tables for future reference:

Cerasi

Overall break down of diff expressed genes

Source	Significantly differentially expressed
Altitude	452
Time	3014
Altitude + Time	58
Altitude * Time	14564

Overall population level differences; WGCNA power 16, cutuff = .25, deepsplit = 4

Table of the number of genes per module

Var1	Freq
grey	9
turquoise	443

Tables of relabelled names per module and their groupings

color	lett	module
grey	A	MEgrey
turquoise	B	MEturquoise

Common responses , WGCNA power 16, cutuff = .25, deepsplit = 4

Table of the number of genes per module

Var1	Freq
black	127
brown	381
darkgreen	74
darkred	212
darkturquoise	56
greenyellow	108
grey	32
grey60	355
lightyellow	261
magenta	433
midnightblue	102
royalblue	85
turquoise	788

Tables of relabelled names per module and their groupings

module	class	labs
MEgreenyellow	Cycling	A
MEgrey	Cycling	B
MEdarkred	Initiation	C
MEdarkgreen	Maintenance	D
MElightyellow	Maintenance	E
MEroyalblue	Maintenance	F
MEgrey60	Maintenance	G
MEmagenta	Maintenance	H
MEdarkturquoise	Maintenance	I
MEmidnightblue	Maintenance	J
MEblack	Maintenance	K
MEbrown	Termination	L
MEturquoise	Termination	M

Divergent responses , WGCNA power 16, cutuff = .25, deepsplit = 4

Table of the number of genes per module

Var1 | Freq| |:----------|----:| |blue4 | 1810| |blueviolet | 47| |darkseagreen4 | 2442| |darkviolet | 165| |grey | 5| |indianred3 | 5399| |lightpink4 | 1781| |mediumorchid | 1790| |orangered1 | 56| |palevioletred2 | 207| |thistle2 | 92| |yellow4 | 95|

Tables of relabelled names per module and their groupings

module	cluster	laa
MEindianred3	1	A
MEblue4	2	B
MEmediumorchid	2	C
MElightpink4	2	D
MEorangered1	2	E
MEdarkseagreen4	3	F
MEpalevioletred2	4	G
MEthistle2	4	H
MEyellow4	5	I
MEblueviolet	5	J
MEdarkviolet	5	K
MEgrey	6	L

Pomonella

Overall break down of diff expressed genes

Source	Significantly differentially expressed
Population	0
Time	2592
Population + Time	0
Population x Time	2902

Common responses , WGCNA power 12, cutuff = .25, deepsplit = 4

Table of the number of genes per module

Var1	Freq
black	632
cyan	84
green	156
greenyellow	103
grey	2
lightcyan	259
purple	1356

Tables of relabelled names per module and their groupings

module	class	lab
MEgrey	Cycling	A
MElightcyan	Cycling	B
MEblack	Initiation	C
MEcyan	Maintenance	D
MEgreenyellow	Maintenance	E
MEgreen	Termination	F
MEpurple	Termination	G

Divergent responses , WGCNA power 12, cutuff = .25, deepsplit = 4

Table of the number of genes per module

Var1	Freq
black	137
blue	279
brown	337
darkgrey	315
darkturquoise	74
grey	2
grey60	153
purple	253
red	764
royalblue	83
turquoise	194
yellow	311

Tables of relabelled names per module and their groupings

module	order	order2
MEturquoise	1	A
MEgrey60	1	B
MEdarkturquoise	1	C
MEdarkgrey	1	D
MEbrown	2	E
MEyellow	2	F
MEblack	2	G
MEblue	3	H
MEpurple	4	I
MEred	4	J
MEroyalblue	4	K

notes: the red module (J), has genes in the hippo, mtor, dorso-ventral in kegg pathways, but not significant

Introduction notes:

Set up drama about what organisms are doing in diapause-they have to be dormant, but also eventually have to be responsive to favorable cues

Especially in light of climate change, but species seem to be evolutionarily flexible

We know clinal variation SNPs associated with diapause and we know very well which genes may be important for diapause. However, missing gap in knowledge

Result punchline: Common responses have developmental and stress hardiness related genes; divergent responses are focused on growth related genes.

Insects need to be freeze avoiding or freeze tolerant (Bale and Hayward 2009) Seasonal environments promote diapause (Bradford and Roff 1997)

Comparative studies of time course (heterochronic) are lacking

Rough sketch of paragraphs

1. Life history timing is important, they differ especially in variable environments. For seasonal environments, organisms match their
2. Timing is important especially for univoltine because they have a single window to get it right
3. Diapause development- dynamic process,
4. Molecular processes driving diapause development; end with Problem: we don't know what genes are conserved vs which ones become adaptively modulated
5. Rhagoletis system succinct
6. In this study

Highly seasonal environments impose significant challenges for species persistence (). Although winter months appear depauperate of most living things, spring and summer months teem with species abundance and diversity (). Taken aside migration from warmer environments, how do species persist in situ during variable seasonal environments? For insects in temperate environments, they match the timing of life events with the seasons, which is under intense selection (). While the warm spring and summer months promote favorable growing conditions, fall signals insects to initiate dormancy or diapause which proceeds throughout the winter months in order to endure unfavorable growing conditions. Any mismatch between life history timing and seasonal change would lead to negative fitness consequences and subsequent population declines (Van Dyck et al. 2015). Populations constantly face critical life history decisions because seasonal environments themselves vary over space (latitude, longitude) and evolutionary time (Tauber & Tauber 1981). The degree of selection is particularly strong for insect species with only one life cycle per year. Most pronounced over the harsh winter months The ability to "know" how to proceed through life history transitions over the season…

Walk through the developmental diapause process.

Diapause development isn't necessarily discrete and moves in a continuous fashion. The way development transitions depend on the transcriptional trajectories of genes through time. So just knowing differentially expressed genes misses the whole continuous process of diapause development.

-Issue, we ascribe and set up predefined groups already

-Specific tissues are not investigated

Need to add in this idea of the trajectory of diapause development is well characterized (Meyers et al. 2016- immediate termination process; Ragland et al. 2011; Ragland et al. 2010; Kostal et al. 2017) and focus on gene differences. What's missing are comparative studies of transcriptomic profiling across diapause development to identify how organisms may adaptively differentiate their seasonal timing.

Large scale genomic divergence between ancestral and derived populations of Rhagoletis (Egan et al. 2015; Meredith paper; Powell et al. 2013).

Need statement: we need to understand gene identities but also their trajectories too. Because trajectories are an important component of their abundance and in turn, their role in any physiological process.

Predictions:

1. Initiation: genes are important early on to start diapause development and then get turned off
2. Maintenance: genes that play an important role contribute to homeostasis during dormancy or prolonged development; there are actually two general patterns a. U-shaped pattern would signify genes important for physiological processes during permissive conditions because gene abundance matches favorable conditions b. Hump-shaped patterns would signify genes turned on during dormancy and would be important for physiological processes during non-permissive conditions
3. Termination: genes that abruptly or gradually turn on at the later stages of diapause development, we mainly expect genes to be enriched for resumption of growth

In this study, we aim to understand seasonal differences between two distinct population of Rhagoeltis cerasi by determining how these flies progress through diapause development using transcriptional trajectories as landmarks. We first determine the degree of seasonal population divergence in eclosion timing between two to four and a half months while in dormancy. Over the same time course, we assess whether population level divergence in eclosion reflects differences in transcript abundance trajectories as well as transcript identity. For transcripts showing common responses, we expect them to fall into three discrete phases classes of diapause development (Kostal): Initiation, maintenance, and termination based on their pattern of expression over time. Out of the genes related to initiation, we expect genes to be enriched in metabolic suppression and slowdown of development (). Out of the genes related to maintenance, we expect genes to be enriched in stress hardiness pathways and perhaps lipid metabolism (). Out of the genes related to termination, we expect genes to be enriched in overall eye development, morphogenesis, and metabolic activation (). For genes displaying population differences across time, we expect to observe offsets in expression patterns that reflect their eclosion timing. For example, analyzing reaction norms may reveal peak expression values appearing earlier in lowland populations than highland populations. In order to determine whether these class of genes remain conserved or become adaptively modulated for common or divergent gene expression responses between populations, respectively, we compared gene sets and expression trajectories of Rhagoletis cerasi with Rhagoletis pomonella.

Page 15: 2019-05-01. Sorting ideas: How do we know the modules we're finding are robust?

Recap: We used WGCNA to estimate weighted correlated gene networks. Nodes are the genes and the edges are weighted by the degree of correlation between the two nodes. From my understanding, WGCNA takes normalized gene expression data (in this case output of edgeR output, so normalized log fold change) and calculates all pairwise pearsons correlations. Converts this matrix (adjacency) to topological overlap matrix ; 1- TOM = dissimiliarity. Then cluster based on distances and given a cutoff, a group of genes in a cluster is a module.

Dan wants to know how we can tell that these modules don't arise by chance? In other words, are the modules robust? Are the modules real?

I think there are two aspects of whether these modules are real: the number and the degree of genes cluster.

1. the number of modules
2. the degree of genes ending up in a module

Dan is most interested in #2, but they're both related.

Ok, what are the different approaches?

Are these modules assembling more than expected by chance?

- Create a null distribution of random modules by permuting module labels and compare some test statistic from our module to the random distribution. This preserves the number of modules being compared, might be simpler.
    - For each module, compare the degree of shared genes and determine if they're significantly different from empirical module.
- How consistent are the modules?
    - bootstrap the samples and determine how often the same modules arise; ie determine network preservation
- Determine the degree of module overlap between reference module to a randomly generated module (permutation). Compare reference with random generated module and their overalp with a fisher's exact test and determine whether the resultant log odds ratio is different from 1 - can use fisher's exact test. (overlap package, compares to the number of background genes)
- Jack knife- take one sample out, then estimate network and determine module preservation
    - tells you the influence of each sample

Some reference:

Good paper on zsummary and module preservation
Li et al. 2015; Scientific Reports
Title - Quantitative assessment of gene expression network module-validation methods
link (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4607977/#b57)

Different way to assess modules : calculate median z scores

Jia et al. 2012; Plos Computation Biology
Title - Network-Assisted Investigation of Combined Causal Signals from Genome-Wide Association Studies in Schizophrenia
link (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3390381/pdf/pcbi.1002587.pdf)

2019-05-02 update- Workflow:

Rationale

We want to know if the modules we've detected occur by random chance. To test this, we will permute gene names for the given module structure and for each module, ask how often are genes are shared between the reference set (empirically identified modules) and a random set (permuted). For each permuted dataset, we can determine the odds ratio of association (OR) and

generate a distribution of OR across all permutations.

This is the set up for the contingency table: A can be the reference gene set, and B is the permuted gene set

```
`r knitr::kable(data.frame(notA=c("a","c"),inA=c("b","d"),row.names =
c("notB","inB")))`
```

	notA	inA
notB	a	b
inB	c	d

- OR < 1, then no association
- OR = 1, then equal association
- OR > 1, then positive assocation

Then for each module, we want to determine whether the log odds ratio distrubtion is signicantly different from 0 with a one sample t-test.

Expected outcome

If modules are assembling in a non-random pattern, we'd expect the logg odds distribution to be significantly less than 0.

Page 16: 2019-05-07. Code comparing gene lists (anything 2 lists with lots of elements you want to compare in a pairwise fashion)

The GeneOverlap R package helped me do the calculation : https://www.bioconductor.org/packages/release/bioc/vignettes/GeneOverlap/inst/doc/GeneOverlap.pdf

This is how the matrix is set up too : https://www.rdocumentation.org/packages/GeneOverlap/versions/1.8.0/topics/GeneOverlap

Anyway, I want to save some code that calculates the association between two gene sets that is permuted vs a reference set. So if you have a bunch of elements (data subsets with a vector of their own gene names) between 2 lists you want to compare, this is the code:

It is basically a nested for loop which i dont usually do, but I couldnt think of any other way.

the workflow: first set the total possible number of genes among sets (these are genes in common responses for us on the cerasi dataset)

1. first you need 2 lists with elemtns of a vector of gene names
2. loop through the reference set, grab each element first (i)
3. grab the name of the element while you're at it (ref.name)
4. then loop through the permuted gene list (j)
5. grab the name of the element again (per.name)

6. start constructing the contingency table (interesect, union, setdiff, odds ratio)
7. you get a vector, bind the rows together and you get the dataset

```
a1<-
list(magenta=data.frame(a=c("A","Z","B","C")),blue=data.frame(a=c("A","B","C","D
")))
c1<-
list(green=data.frame(a=c("A","Z","B","C")),red=data.frame(a=c("A","B","C","D","
W")))


#total number of genes
n=5000
df=NULL
#ref.names=NULL
#
for(i in seq_along(a1)) {
  #newlist[[i]] <- dplyr::pull(a1[[i]],a)
  ss <- as.character(dplyr::pull(a1[[i]],a))
  ref.name<-names(a1[i])
  #print(ss)
  for(j in seq_along(c1)){
    ss2 <- as.character(dplyr::pull(c1[[j]],a))
    perm.name<-names(c1[j])
    #print(ss2)
    #print(as.vector(length(unlist(lapply(ss2,function(x)
{intersect(ss,x)}))),ss,ss2))
    #print(length(unlist(lapply(ss2,function(x){intersect(ss,x)}))))
    #print(length(unlist(lapply(ss2,function(x){setdiff(ss,x)}))))
    #print(length(unlist(lapply(ss2,function(x){setdiff(x,ss)}))))
    #print(n-length(unlist(lapply(ss2,function(x){union(x,ss)}))))

    inAinB<-length(unlist(lapply(ss2,function(x){intersect(ss,x)})))
    inAnotB<-length(unlist(lapply(ss2,function(x){setdiff(x,ss)})))
    notAinB<-length(unlist(lapply(ss2,function(x){setdiff(ss,x)})))
    notAnotB<-n-length(unlist(lapply(ss2,function(x){union(x,ss)})))
    #calculate odds ratio
    OR<-fisher.test(matrix(c(inAinB,inAnotB,notAinB,notAnotB),nrow=2))[[3]]

    df<-
rbind(df,data.frame(inAinB,inAnotB,notAinB,notAnotB,OR,logOR=log2(OR),ref.name,p
erm.name))

  }
}

df
```

	inAinB	inAnotB	notAinB	notAnotB	OR	logOR	ref.name	perm.name
odds ratio	4	0	0	16	Inf	Inf	magenta	green
odds ratio1	3	2	1	14	16.38836	4.034599	magenta	red
odds ratio2	3	1	1	15	29.62073	4.888535	blue	green
odds ratio3	4	1	0	15	Inf	Inf	blue	red

Set up for the contingency table:

	NotA	inA
notB	total genes- union	setdiff(B,A)
inB	setdiff(A,B)	intersect

Page 17: 2019-05-09. Messing around with code

```
library(ggplot2)
#library(ggthemr)

x<-seq(1:10)
T<-theme_bw()+theme(text=element_text(size=20),axis.text=element_text(size=20),
panel.grid.major=element_blank(), panel.grid.minor.x = element_blank(),
panel.grid = element_blank(), legend.key = element_blank())+
theme(legend.position="none")

dat<-data.frame(x=x[-1:-2],y=x[-1:-2]^2)

ggplot(dat,aes(x=x,y=y))+geom_point(size=5)+geom_line(size=1.5)+T+ylab("Time
spent at desk")+
  scale_x_continuous(labels=c("PhD Year 1",
                              "PhD Year4","Start Postdoc", "Postdoc Year 2"),

breaks=c(3,6,8,10))+xlab("Time")+scale_y_continuous(labels="",breaks=100)+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Page 18: 2019-05-10. Reading Langfelder et al. 2011; Is My Network Module Preserved or Reproducible?

Is my Network Module Preserved and Reproducible? Langfelder et al., PlosOne

Cluster validation = quality; 4 ways to assess:

1. cross-tabulation - statistics comparing cluster assignment in reference and test clusterings
2. edge_density - dont require clustering
3. separability - dont require clustering
4. stability -

To find reproducibility in a module or cluster, take what is found in a one set and apply it to a new case.

They say module preservation is different than cluster preservation but dont tell you how they'll figure this out or do it in the intro.

This paper skips steps in logic, god damn.

Network based stats:

1. Density based- preservation stats can be used to determine whether module nodes remain highly connected in the test network
2. Separability - determine whether network modules remain distinct from one another in the test network
3. Connectivity - determine whether the connectivity pattern between nodes in the reference network is similiar to that in the test network

Zsummary = (Zdensity + Zconnectivity)/2

Z<2 = no preservation Z<10 = weak preservation

medianrank - less dependent on module size , which can influence preservation

module preservation

determine whether a module is present in a reference network can also be found in an independent test network.

Modules between reference and test networks may be preserved if has a high desnity in the test network. Take mean adjacency for module q as the module density of the test network.

Page 19: 2019-05-10. Helping James Brown with some statistic

Make figure and check stats on the differences in wandering day between strain and photoperiod

**loading data and all of the manipulation

```
data=read_excel("../Data/data.xlsx",sheet = "wanderdata")

#file: data
#sheet: wanderdata

#converting wide to long. This code puts all the observations into one column
with the label of your choice (measurements used below)
all=gather(data,day,stage,"82wander_0616":"01wander_0327")
all$strain<-substr(all$tray_id,1,2)
all$strain<-as.factor(all$strain)
all$treat<-substr(all$tray_id,3,4)
all$treat<-as.factor(all$treat)
all$cohort<-substr(all$tray_id,6,9)
all$cohort<-as.factor(all$cohort)
all$day<-as.factor(all$day)
all$stage<-as.factor(all$stage)
all$tray_id<-as.factor(all$tray_id)
all$cell_id<-as.factor(all$cell_id)
```

```
all$fifth_date<-as.factor(all$fifth_date)
all=subset(all,stage!="NA")
```

Checking what the data look like

```
str(all)
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':   3780 obs. of  11 variables:
 $ tray_id   : Factor w/ 8 levels "BE12 0314","BE12 0320",..: 3 3 3 3 3 3 3 3 3
3 ...
 $ fifth_date: Factor w/ 21 levels "20180323","20180324",..: 20 20 19 19 20 20
19 19 20 20 ...
 $ cell_id   : Factor w/ 48 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ fiveday   : num  23 23 22 22 23 23 22 22 23 23 ...
 $ pday      : num  NA 24 20 20 15 19 20 20 24 24 ...
 $ wday      : num  10 10 10 10 10 9 12 10 11 12 ...
 $ day       : Factor w/ 82 levels "01wander_0327",..: 82 82 82 82 82 82 82 82
82 82 ...
 $ stage     : Factor w/ 6 levels "A","D","N","P",..: 6 4 4 4 4 4 4 4 4 4 ...
 $ strain    : Factor w/ 2 levels "BE","UZ": 1 1 1 1 1 1 1 1 1 1 ...
 $ treat     : Factor w/ 2 levels "12","16": 1 1 1 1 1 1 1 1 1 1 ...
 $ cohort    : Factor w/ 4 levels "0302","0314",..: 4 4 4 4 4 4 4 4 4 4 ...

 all%>%
 +   group_by(cohort,strain,treat)%>%
 +   dplyr::summarise(n.cohorts=length(cohort))
 # A tibble: 8 x 4
 # Groups:   cohort, strain [4]
   cohort strain treat n.cohorts
   <fct>  <fct>  <fct>     <int>
 1 0302   UZ     12          630
 2 0302   UZ     16         1056
 3 0314   BE     12          331
 4 0314   BE     16          552
 5 0320   BE     12          184
 6 0320   BE     16          287
 7 0429   BE     12          308
 8 0429   BE     16          432
```

stat models ANOVA

```
mod1<-aov(wday~treat*strain+cohort,data=all)
summary(mod1)
par(mfrow=c(2,2))
plot(mod1)
par(mfrow=c(1,1))

Df Sum Sq Mean Sq F value Pr(>F)
treat           1  17245   17245 6166.03 <2e-16 ***
strain          1    262     262   93.82 <2e-16 ***
cohort          2    642     321  114.72 <2e-16 ***
treat:strain    1    861     861  307.77 <2e-16 ***
Residuals    3380   9453       3
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
394 observations deleted due to missingness
```

Not normal!!

Poisson regression

```
mod2<-glm(wday~treat*strain+cohort,data=all,family=poisson())
summary(mod2)
Call:
glm(formula = wday ~ treat * strain + cohort, family = poisson(),
    data = all)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.65324  -0.31997   0.00955   0.31551   1.83552

Coefficients: (1 not defined because of singularities)
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       2.36027    0.01520 155.314  < 2e-16 ***
treat16          -0.46654    0.01691 -27.597  < 2e-16 ***
strainUZ          0.07460    0.01934   3.858 0.000114 ***
cohort0314       -0.10587    0.01875  -5.648 1.62e-08 ***
cohort0320       -0.23360    0.02379  -9.821  < 2e-16 ***
cohort0429             NA         NA      NA       NA
treat16:strainUZ -0.21913    0.02503  -8.756  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 3596.5  on 3385  degrees of freedom
Residual deviance: 1223.8  on 3380  degrees of freedom
  (394 observations deleted due to missingness)
AIC: 14250

Number of Fisher Scoring iterations: 4
```

Making the boxplot, wandering day vs photoperiod for each strain

```
ggplot(all,aes(x=treat,y=wday,fill=strain))+geom_boxplot()+ylab("Wandering Day
(days)")+
xlab("Photoperiod Treatment (hours)")+scale_fill_manual(labels=c("Long
Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+
T+theme(legend.position = c(0.9,0.85),legend.justification = c("right",
"bottom"))+labs(fill =
"Strain")+scale_y_continuous(limits=c(0,16),breaks=seq(0,16,4),labels=seq(0,16,4
))
```

Lipid accumulation between developmental stages for each strain, and photoperiod

data manipulations

```
##PC
data=read_excel("../data/data.xlsx",sheet = "energy")

#file: data
#sheet: energy

data=subset(data,treat!="NA")
data=subset(data,cohort!="20180131")
data=subset(data,cohort!="20180206")
data=subset(data,sample_day!="W14")
data=subset(data,sample_day!="W19")
data=subset(data,sample_day!="W29")
```

**prep data for generating figure

```
## Plot conparing Lipid Mass of both strains and both treatments
dataLipid=data1
dataLipid=subset(dataLipid,lipid_mass>=0)
#dataLipid=subset(dataLipid,lipid_mass<=0.0200)
count(dataLipid$treat)
dataLipid=subset(dataLipid,sample_day=="1"| sample_day=="W")

dataLipid$phot.title<-paste(dataLipid$photoperiod, "Hour","Photoperiod
Treatment")
```

actual figure in ggplot, pretty

```
ggplot(data=dataLipid,aes(x=sample_day,y=lipid_mass,group=treat,color=strain))+
  stat_summary(aes(y = lipid_mass), fun.y=mean, geom="line",size=1.5)+
  stat_summary(aes(y = lipid_mass), fun.y=mean, geom="point",size=3)+
  stat_summary(fun.data=mean_se, geom="errorbar", width=0.25,size=1.5)+
  #facet_wrap(~photoperiod,nrow=1)+
  facet_wrap(~phot.title,nrow=1)+
  scale_color_manual(labels=c("Long Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+
  theme(axis.text.x  = element_text(vjust=0.5, size=16),axis.text.y  =
element_text(vjust=0.5, size=16))+
  #ggtitle("Treatment Comparison of Larvae lipid Mass (UZ 178 : BE 164
larvae)")+
  ylab("Lipid Mass (g)")+xlab("")+T+scale_x_discrete(labels=c("Day 1 of\n 5th
instar", "Wandering \nStage"))+theme(legend.position =
c(0.9,0.8),legend.justification = c("right", "bottom"))+labs(color = "Strain")
```

data prep for statistics

```
tog<-rbind(dataLW,dataL1)
Observations: 282
Variables: 22
$ cohort                              <fct> 20180302, 20180302, 20180302,
20180302, 20180…
$ rep                                 <dbl> 22, 22, 22, 22, 23, 23, 23, 21, 21,
22, 22, 2…
$ sample_day                          <fct> W, W, W, W, W, W, W, W, W, W, W, W,
W, W, W, …
```

```
$ sample_id                          <fct> 0329-17, 0329-18, 0329-19, 0329-20,
0330-02, …
$ tag_id                             <chr> "0329-17", "0329-18", "0329-19",
"0329-20", "…
$ batch                              <chr> "10", "10", "10", "10", "10", "10",
"10", "10…
$ FAME                               <chr> "F", "F", NA, NA, "F", NA, NA, NA,
NA, "F", "…
$ `5th_date`                         <dbl> 20180323, 20180323, 20180323,
20180323, 20180…
$ treat                              <fct> UZ16, UZ16, UZ16, UZ16, UZ16, UZ16,
UZ16, UZ1…
$ `Microtube wt + Beads`             <dbl> 2.2601, 2.1353, 2.2251, 2.3038,
1.9232, 2.586…
$ `Microtube wt + Beads + WET Larvae` <dbl> 2.3342, 2.2039, 2.3191, 2.4146,
1.9956, 2.674…
$ `Microtube + Beads + DRY Larvae`   <dbl> 2.2798, 2.1543, 2.2521, 2.3358,
1.9429, 2.614…
$ `Microtube + Beads + LEAN Larvae`  <dbl> NA, NA, NA, NA, NA, NA, NA, 1.9097,
2.4334, N…
$ `TV wt`                            <dbl> 14.42190, 15.09370, 14.31270,
15.32050, 15.30…
$ `TV wt + DRY Lipids`               <dbl> 14.4279, 15.1024, 14.3199, 15.3313,
15.3088, …
$ wet_mass                           <dbl> 0.0741, 0.0686, 0.0940, 0.1108,
0.0724, 0.088…
$ dry_mass                           <dbl> 0.0197, 0.0190, 0.0270, 0.0320,
0.0197, 0.027…
$ lean_mass                          <dbl> -2.2601, -2.1353, -2.2251, -2.3038,
-1.9232, …
$ lipid_mass                         <dbl> 0.00600, 0.00870, 0.00720, 0.01080,
0.00580, …
$ tag_mass                           <dbl> 0.0011988, 0.0013124, 0.0015004,
0.0020308, 0…
$ strain                             <fct> UZ, UZ, UZ, UZ, UZ, UZ, UZ, UZ, UZ,
UZ, UZ, U…
$ photoperiod                        <fct> 16, 16, 16, 16, 16, 16, 16, 12, 12,
12, 12, 1…
```

stat model: mixed effects

```
mm1=lmer(lipid_mass ~ photoperiod*strain*sample_day+lean_mass + (1|rep/cohort)
,data=tog, REML = TRUE)

lmerTest::step(mm1)
summary(mm1)
#output
Linear mixed model fit by REML. t-tests use Satterthwaite's method
['lmerModLmerTest']
Formula: lipid_mass ~ photoperiod * strain * sample_day + lean_mass +      (1 |
rep/cohort)
   Data: tog

REML criterion at convergence: -2463.4

Scaled residuals:
```

```
    Min      1Q  Median      3Q     Max
-5.3487 -0.3937 -0.0238  0.4062  3.7083

Random effects:
 Groups     Name        Variance  Std.Dev.
 cohort:rep (Intercept) 0.000e+00 0.0000000
 rep        (Intercept) 6.611e-07 0.0008131
 Residual               5.904e-06 0.0024299
Number of obs: 282, groups:  cohort:rep, 44; rep, 20

Fixed effects:
                                    Estimate Std. Error         df t value
Pr(>|t|)
(Intercept)                         1.951e-03  7.049e-04  7.684e+01   2.768
0.00707 **
photoperiod16                      -7.512e-04  8.116e-04  2.641e+02  -0.926
0.35551
strainUZ                           -8.112e-05  8.023e-04  2.623e+02  -0.101
0.91954
sample_dayW                         8.321e-03  8.195e-04  6.481e+01  10.154
5e-15 ***
lean_mass                          -1.878e-05  2.680e-04  2.421e+01  -0.070
0.94472
photoperiod16:strainUZ              3.612e-04  1.106e-03  2.624e+02   0.327
0.74425
photoperiod16:sample_dayW          -2.909e-03  9.467e-04  2.634e+02  -3.072
0.00235 **
strainUZ:sample_dayW                2.203e-03  9.374e-04  2.612e+02   2.350
0.01949 *
photoperiod16:strainUZ:sample_dayW -1.328e-03  1.305e-03  2.610e+02  -1.018
0.30966
```

figure genration

```
b1<-ggplot(data=dataLeU,aes(x=sample_day,y=lean_mass,group=treat,color=strain))+
  stat_summary(aes(y = lean_mass), fun.y=mean, geom="line",size=1.5)+
  stat_summary(aes(y = lean_mass), fun.y=mean, geom="point",size=3)+
  stat_summary(fun.data=mean_se, geom="errorbar", width=0.25,size=1.5)+
  #facet_wrap(~photoperiod,nrow=1)+
  facet_wrap(~phot.title,nrow=1)+
  scale_color_manual(labels=c("Long Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+
  theme(axis.text.x  = element_text(vjust=0.5, size=16),axis.text.y  =
element_text(vjust=0.5, size=16))+
  #ggtitle("Treatment Comparison of Larvae lipid Mass (UZ 178 : BE 164
larvae)")+
  ylab("Lean Mass (g)")+xlab("")+T+scale_x_discrete(labels=c("Day 1 of\n 5th
instar", "Wandering \nStage"))+theme(legend.position =
c(0.9,0.8),legend.justification = c("right", "bottom"))+labs(color =
"Strain")+scale_y_continuous(breaks=seq(0,.04,.005),labels=seq(0,.04,.005))
```

last part; wet mass vs days for non diapause and diapause

messing around with finding the max value across days

```
datamass16uz<-datamass16uz[-32,]

datamass16uz$predmax<-predict(loess(mass~day,datamass16uz))
max(datamass16uz$predmax)
summary(datamass16uz)

datamass16uz%>%
  dplyr::filter(predmax==0.11237042)

test<-data.frame(x=datamass16uz$day,y=datamass16uz$predmax)

test%>%
  dplyr::filter(y==0.11237042)
  #dplyr::summarise(max=max(predmax))%>%

#day 5
ggplot(test,aes(x,y))+geom_line()+geom_point()+stat_smooth(forumla=y~x^2)
summary(lm(y~x+I(x^2),test))
```

making 2 panel figure

```
## Plot of wet mass peak of BE and UZ strains in long day conditions
c1<-ggplot(data=dataBEUZmass16,aes(x=day,y=mass,color=strain))+
  stat_summary(fun.y=mean, geom="line",shape=25,size=1.5)+
  stat_summary(fun.y=mean, geom="point",shape=19,size=4)+
  stat_summary(fun.data=mean_se, geom="errorbar", width=0.25,size=1.5,shape=21)
+
  theme(axis.text.x = element_text(vjust=0.5, size=16),
        axis.text.y  = element_text(vjust=0.5, size=16))+
  #scale_x_continuous(breaks = seq(0, 10, by=2), limits=c(0,10))+
  scale_fill_manual(values=c("1"="lightblue","2"="black"))+
  #ggtitle("Change in Wet Mass Production: Long Day")+
  ylab("Wet Mass
(g)")+xlab("Day")+T+scale_x_continuous(limits=c(0,10),breaks=seq(0,10,1),labels=
seq(0,10,1))+
  scale_color_manual(labels=c("Long Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))#+theme(
legend.position = c(0.9,0.8),legend.justification = c("right",
"bottom"))+labs(color = "Strain")

c1


## Plot of wet mass peak of BE and UZ strains in short day conditions
d1<-ggplot(data=dataBEUZmass12,aes(x=day,y=mass,color=strain))+
  stat_summary(fun.y=mean, geom="line",shape=25,size=1.5)+
  stat_summary(fun.y=mean, geom="point",shape=19,size=4)+
  stat_summary(fun.data=mean_se, geom="errorbar", width=0.55,size=1,shape=21) +
  #stat_summary(fun.data=mean_se, geom="errorbar", width=0.25,size=1,shape=21) +
  theme(axis.text.x = element_text(vjust=0.5, size=16),
        axis.text.y  = element_text(vjust=0.5, size=16))+
  scale_x_continuous(breaks = seq(0, 40, by=5), limits=c(0,40))+
  #ggtitle("Change in Wet Mass Production: Short Day")+
```

```
  ylab("Wet Mass (g)")+xlab("Day")+scale_color_manual(labels=c("Long
Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+theme(l
egend.position = c(0.9,0.8),legend.justification = c("right",
"bottom"))+labs(color = "Strain")+T

d1

(c1+plot_spacer()+plot_layout(widths=c(.4,1)))/d1
```

Page 20: 2019-05-14. results section cerasi paper before I chop it up

Lowland populations emerge as adults earlier than highland populations

Compared to the highland population, the lowland population shifted their seasonal patterns to earlier adult emergence in two distinct patterns: overall proportion of adult emergence and timing to adult emergence (Figure 2). The lowland population had more proportional adults emerging than the highland population (GLM: population, $p<0.05$) across all increasing levels of overwintering time (GLM: time, $p<0.05$). For timing to adult emergence, lowland populations emerged in less time than highland populations between 2.5 to 4.5 months of overwintering incubation (ANOVA: population x time, $p<0.05$). The greatest difference occurred at 3.5 months, whereby the lowland population emerged on average 7.14 □ 1.46 days sooner than the highland population (ANOVA: $p<0.005$). The propensity to eclose and speed of development may incur a survival cost because the lowland population had higher mortality than the highland population (GLM: population, $p<0.05$).

Most of the differentially expressed genes reflect adaptive differences between populations over time

In total, we found 18,088 out of Y differentially expressed transcripts (Table 1) from our EdgeR model ($p < 0.05$). There was an overall population difference in transcript abundance regardless of time (main effect of population) in 452 transcripts, and both populations shared common responses over time (main effect of time) in 3,014 transcripts ($p < 0.05$). The least number of significant transcripts (58) were found in the main effect of population and the main effect of time ($p < 0.05$). However, the majority of differentially expressed transcripts (14,564) differed over time between high and low populations (time x population interaction, $p < 0.001$).

Gene modules enriched in overall differences between populations

Among the 452 differentially expressed transcripts, we found a single modules showing the overall difference in expression between lowland and highland populations. This module shows higher eigengene values in highland populations compared to lowland populations (Figure 2). Genes were enriched related to protein peptidyl-prolyl isomerization (GO) and peptidyl-proline modification (GO).

Common gene modules expressed over diapause development

Most of the modules were consistent with our predicted pattern of diapause maintenance (stats with contingency table type of analysis; Figure 3). In total, 13 modules were recovered from WGCNA with 11 modules displaying characteristic gene expression trajectories reflecting diapause development (Figure 1). Two modules appear to be cyclical (greenyellow and grey). The greenyellow module was enriched with insecticide metabolic processes (GO) and a KEGG pathway was identified in insect hormone biosynthesis, while the grey module was enriched in response to CO2 (GO). We observed one module following a consistent pattern with Initiation,

and genes in this module were enriched in detection of mechanical stimulus involved in sensory perception of pain (GO:0050966). We observed eight modules following a consistent pattern with Maintenance with two modules displaying U-shaped patterns (black, darkgreen; Figure 1B, Figure 3) and six modules displaying humped shaped patterns (darkturquoise, grey60, lightyellow, magenta, midnightblue, and royalblue; Figure 1A, Figure 3). The genes in the modules with U-shaped patterns were enriched with regulation of retinal cell apoptosis (GO, black module), and HS-mediated polytene chromosome puffing (GO, darkgreen module). The genes in the modules with humped-shaped patterns were enriched in vacuolar acidification (GO, grey60 module), single organism cell-cell adhesion (GO, magenta), system development (GO, midnightblue), and HS-mediated polytene chromosome puffing (GO, royalblue). Finally, we observed two modules (brown and turquoise) following a consistent pattern with Termination. Genes in the brown module were enriched in melanin biosynthesis process (GO), wax metabolic process (GO), and fatty acyl-COA metabolism (GO); the same genes were enriched in lysine degradation according to the KEGG pathway (). Genes in the turquoise module were enriched in rRNA modification (GO), glycolipid biosynthetic process (GO), and xanthine catabolic process (GO).

Divergence in gene modules between populations over time

When comparing the expression modules between populations, we found significant offsets in overall eigengene profiles over time in a total of 12 modules (Figure 4). In contrast to our initial predictions, we found populations cycling in opposite phases with one another in six modules (dark violet, yellow4, thistle2, palevioletred2, grey, blueviolet). These cyclically patterned genes were enriched for xanthine catabolic and flavonoid biologic process (darkviolet), inositol phosphate dephosphorylation (yellow4), amino acid transmembrane transport (thistle2), detection of chemical stimulus (palevioletred2), and neuropeptide signaling pathway (blueviolet). Most of the genes fell within the indianred3 cluster, which appear to be offset in eigengene values over time in the opposite expected pattern: lowland populations had later eigenvalue expression compared to highland populations. Although these trajectories would be indicative of a genes importance for maintenance, genes in this module (indianred3) were enriched for imaginal disc derived morphogenesis (GO), and axon guidance (GO), suggesting a role for early orchestration of diapause termination in lowland populations, while these genes may act as a prepatory role for ongoing maintenance in highland populations. However, we found offsets in eigengene profiles for four modules (lightpink4, mediumorchid, orangered1, blue4) in the expected direction of change: relative to highland populations, lowland populations displayed earlier increases in expression reflective of termination trajectories and gene sets (Figure 1 C,F). Genes in these early termination modules for lowland population were enriched in synapse assembly and neuron cell-cell adhesion (lightpink4), proteasome mediated ubiquitin dependent degradation (mediumorchid), protein catabolic process (orangered1), and secondary metabolite biosynthesis (blue4).

Page 21: 2019-05-20. Module preservation

Zsummary = (Zdensity + Z connectivity)/2

Density is basically the average correlation of the off-diagonal of an adjacency matrix. Higher Density means more interconnected nodes.

- Density can be estimated for a given module, just take the mean of the off diagonal of an adjencancy matrix for a module(subset of genes)

Connectivity is defined as the sum of all of the strengths in a network for a given node.

Zdensity = median (Zmeancor, Zmean.adj, Zpropvarexpl, ZmeanKmE)

- Zmeancor
  - meanCor is the mean correlation density of a given module
- Zmean.adj
  - meanAdj is the mean density within a module
- Zpropvarexpl
  - propVarExpl is the mean square of membership modules (correlation between expression and first pca loadings)
- ZmeanKME
  - meanKME asssesses the sign of the reference and test set intheir kME (module membership)

Zconnectivity = median (Zcor.kIM, Zcor.kME, Zcor.cor)

- Zcor.kIM
  - kIM is the intramodular connectivity of a node within a given module. So sum the strength of the correlations of the genes within a module.
- Zcor.kME
  - kME is module membership (correlation between expression and first pca loadings); so you can simply take the correlation in kME between reference and test data sets.
- Zcor.cor
  - cor.cor correlation in the weights of each node between reference and test adjacency matrix

---

Page 22: 2019-05-22. stats dump for R. cerasi module-phenotype analysis: updated, ecluded dead flies

Just dumping some stats so I can reference it; R. cerasi dataset

common responses: pearsons correlations between module eigengene values and proportion of adult emergence

pearsons correlations

```
signif(cor(cerph1.1$emp.mean[-1],mergedMEs2[,1:12], use="p"),2)
     MEdarkred MEgreenyellow MEdarkgreen MElightyellow MEroyalblue MEgrey60
MEmagenta MEdarkturquoise MEmidnightblue MEbrown
[1,]      -0.9         -0.12       -0.26         -0.57       -0.26    -0.56
-0.22            0.44           0.22    0.45
     MEblack MEturquoise
[1,]    0.51        0.76
> #corPvalueStudent(cor(cerph1.1$mod[-1],mergedMEs2[,1:13], use="p"), nSamples =
length(cerph1.1$mod[-1]))
> corPvalueStudent(cor(cerph1.1$emp.mean[-1],mergedMEs2[,1:12], use="p"),
nSamples = length(cerph1.1$mod[-1]))
        MEdarkred MEgreenyellow MEdarkgreen MElightyellow MEroyalblue  MEgrey60
MEmagenta MEdarkturquoise MEmidnightblue
[1,] 0.0009314735      0.761383   0.4977714     0.1095706   0.4980005 0.1132141
0.5623444       0.2338802      0.5612785
       MEbrown   MEblack MEturquoise
[1,] 0.2256807 0.1611093  0.01706775
```

regression models p value

```
 apply(mergedMEs2[,1:12],2,function(x)
{summary(lm(cerph1.1$emp.mean[-1]~x))$coefficient[2,4]})
       MEdarkred   MEgreenyellow     MEdarkgreen   MElightyellow     MEroyalblue
       MEgrey60       MEmagenta MEdarkturquoise
    0.0009314735    0.7613830007    0.4977714142    0.1095705929    0.4980004815
   0.1132141352    0.5623444452    0.2338802404
  MEmidnightblue         MEbrown         MEblack     MEturquoise
    0.5612785061    0.2256806551    0.1611092631    0.0170677526
```

regression model beta

```
apply(mergedMEs2[,1:12],2,function(x)
{summary(lm(cerph1.1$emp.mean[-1]~x))$coefficient[2,1]})
      MEdarkred   MEgreenyellow     MEdarkgreen   MElightyellow     MEroyalblue
       MEgrey60       MEmagenta MEdarkturquoise
     -0.8516187      -0.1120932      -0.2467533      -0.5385239      -0.2466265
     -0.5340631      -0.2118618       0.4178283
 MEmidnightblue         MEbrown         MEblack     MEturquoise
      0.2124249       0.4244426       0.4819980       0.7204863
```

common responses: pearsons correlations between module eigengene values and adult emergence timing

```
signif(cor(cerph1.1$ave_eclosion[-1],mergedMEs2[,1:13], use="p"),2)
     MEdarkred MEgreenyellow MEdarkgreen MElightyellow MEroyalblue MEgrey60
MEmagenta MEdarkturquoise MEmidnightblue MEbrown
[1,]      0.87         -0.04       0.097          0.46        0.31     0.47
 0.19          -0.43          -0.34   -0.35
     MEblack MEturquoise MEgrey
[1,]   -0.51       -0.81   0.19

corPvalueStudent(cor(cerph1.1$ave_eclosion[-1],mergedMEs2[,1:13], use="p"),
nSamples = length(cerph1.1$ave_eclosion[-1]))
       MEdarkred MEgreenyellow MEdarkgreen MElightyellow MEroyalblue  MEgrey60
MEmagenta MEdarkturquoise MEmidnightblue   MEbrown
[1,] 0.002600856     0.9193297   0.8036208     0.2121152   0.4138235 0.2068954
0.6237771       0.2505735      0.3636643 0.3590661
       MEblack MEturquoise    MEgrey
[1,] 0.1564674 0.007824699 0.6258017
```

regression models p value

```
apply(mergedMEs2[,1:12],2,function(x)
{summary(lm(cerph1.1$ave_eclosion[-1]~x))$coefficient[2,4]})
      MEdarkred   MEgreenyellow     MEdarkgreen   MElightyellow     MEroyalblue
       MEgrey60       MEmagenta MEdarkturquoise
    0.002600856     0.919329671     0.803620752     0.212115162     0.413823489
    0.206895351     0.623777070     0.250573531
 MEmidnightblue         MEbrown         MEblack     MEturquoise
    0.363664252     0.359066141     0.156467395     0.007824699
```

regression model beta

```
apply(mergedMEs2[,1:12],2,function(x)
{summary(lm(cerph1.1$ave_eclosion[-1]~x))$coefficient[2,1]})
      MEdarkred   MEgreenyellow     MEdarkgreen   MElightyellow     MEroyalblue
       MEgrey60       MEmagenta MEdarkturquoise
     18.4767635      -0.8469441       2.0753091       9.8388107       6.6630204
      9.9388403       4.0654657      -9.1400893
 MEmidnightblue         MEbrown         MEblack     MEturquoise
     -7.3625798      -7.4289238     -10.9888640     -17.3484209
```

divergent responses: testing interaction of module and population on proportion of adult emergence

```
apply(timpop[,1:12],2,function(module)
{summary(lm(timpop$emp.mean~timpop$Population*module))$coefficient[,4]})
MEpalevioletred2 MEthistle2 MEdarkseagreen4   MEyellow4 MEindianred3
MEblueviolet MEdarkviolet
(Intercept)                      0.01794046  0.1763799       0.3480972
0.003319187   0.01212795   0.02472698  0.006891079
timpop$PopulationLow             0.80295901  0.6232462       0.4842312
0.871145068   0.86465272   0.85080035  0.757150130
module                           0.40406413  0.3535975       0.2947903
0.088259033   0.17381326   0.81498110  0.173810218
```

```
timpop$PopulationLow:module       0.17267879  0.2396853       0.2319097
0.023518037   0.07997573   0.68653764  0.089854339
                         MEblue4 MEmediumorchid MElightpink4 MEorangered1
 MEgrey
(Intercept)                0.01905084      0.1453797   0.03395295  0.009606442
0.007299098
timpop$PopulationLow       0.92458086      0.9571990   0.88875568  0.996576298
0.576596998
module                     0.33594821      0.9998301   0.50301573  0.323735715
0.144787366
timpop$PopulationLow:module 0.42095440     0.8543616   0.33935752  0.059478710
0.622020362

apply(timpop[,1:12],2,function(module)
{summary(lm(timpop$mod~timpop$Population*module))$coefficient[,4]})
                          MEpalevioletred2 MEthistle2 MEdarkseagreen4
MEyellow4 MEindianred3 MEblueviolet MEdarkviolet
(Intercept)                         0.02547447  0.1956658       0.3877154
0.003681874   0.01495837   0.03081938  0.009832124
timpop$PopulationLow                0.85710949  0.5930438       0.4502395
0.929556286   0.89065556   0.88259190  0.715268077
module                              0.46919031  0.2964026       0.2426120
0.076035984   0.17130225   0.77763265  0.190580659
timpop$PopulationLow:module         0.22640093  0.1937862       0.1823005
0.024261302   0.09082904   0.79857712  0.114968170
                            MEblue4 MEmediumorchid MElightpink4 MEorangered1
   MEgrey
(Intercept)                 0.0190113       0.1327767   0.02991592  0.009337004
0.007236771
timpop$PopulationLow        0.8852585       0.8808033   0.82147343  0.979644295
0.538926397
module                      0.2807883       0.9295860   0.42931856  0.255812671
0.122325208
timpop$PopulationLow:module 0.3428887       0.9133783   0.27830496  0.047290044
0.531236680
```

divergent responses: testing interaction of module and population on adult emergence timing

```
apply(timpop[,1:12],2,function(module)
{summary(lm(timpop$ave_eclosion~timpop$Population*module))$coefficient[,4]})
                            MEpalevioletred2  MEthistle2 MEdarkseagreen4
 MEyellow4 MEindianred3 MEblueviolet MEdarkviolet
(Intercept)                     0.0004046353 0.001626606     0.002195613
0.0001883821  0.000404306 0.0008181747  0.001547038
timpop$PopulationLow            0.8679106813 0.574904698     0.431459474
0.7561857108  0.793005595 0.7632566195  0.959116207
module                          0.7700067953 0.361186948     0.271143469
0.1602729207  0.319923133 0.6701365949  0.378686059
timpop$PopulationLow:module     0.4010238049 0.235638654     0.194801647
0.0474923888  0.160535491 0.9348727949  0.255237920
                                 MEblue4 MEmediumorchid MElightpink4
MEorangered1       MEgrey
(Intercept)                 0.0008272252     0.01127329  0.003227091 8.193302e-
05 0.0006061964
timpop$PopulationLow        0.9952251344     0.81715304  0.879583369 8.503189e-
01 0.7163357188
module                      0.2777980482     0.75547123  0.351308548 1.831403e-
01 0.2097695265
timpop$PopulationLow:module 0.3374693036     0.92475619  0.210410781 2.446135e-
02 0.7782808525
```

Page 23: 2019-06-17. Additional and re-analysis of cerasi brain transcriptome (after meeting with gragland)

Dataset- start with cerasi

- Correlate population level effects in the modules with phenotype
- Reset baseline expression for common responses; set log2fold chagne values to each population's
- Redo population level differences and include 2 month high
- correlate average expression with phenotype and add error bars in figures

Correlate population level effects in the modules with phenotype

Phenotype data

```
#phenotype data
cerph1.10
# A tibble: 10 x 5
# Groups:   Population [2]
   Population month    mod emp.mean ave_eclosion
   <chr>      <dbl>  <dbl>    <dbl>        <dbl>
 1 High         2   0.0632    0              0
 2 High         2.5 0.153     0.182         38.9
 3 High         3.5 0.566     0.6           33.7
 4 High         4   0.778     0.768         28.6
 5 High         4.5 0.904     0.866         24.1
 6 Low          2   0.118     0.159         42.8
 7 Low          2.5 0.289     0.242         36.1
 8 Low          3.5 0.788     0.865         26.6
 9 Low          4   0.918     0.918         23.2
```

```
10 Low          4.5 0.971     0.959         21.6
```

phenotype data with modules; grey module is not a real module

```
MEturquoise             MEgrey month Population        mod  emp.mean
ave_eclosion
1    0.0000000                  0     2       High 0.06318026 0.0000000
 0.00000
2    0.3707243 -0.192962914046971   2.5       High 0.15328540 0.1818182
38.88889
3    0.4028881 -0.044042980443642   3.5       High 0.56606430 0.6000000
33.70175
4    0.3638362 0.0371880608336409     4       High 0.77786161 0.7676768
28.61842
5    0.2256386 -0.189526080361635   4.5       High 0.90384399 0.8659794
24.07143
6   -0.5971878  0.796938298452717     2        Low 0.11820542 0.1587302
42.80000
7   -0.2015581 -0.372529805564627   2.5        Low 0.28864996 0.2419355
36.13333
8   -0.1268667  0.168839674415984   3.5        Low 0.78805162 0.8653846
26.55556
9   -0.2816223  0.122032278154512     4        Low 0.91840018 0.9183673
23.24444
10  -0.1558523  -0.32593653143998   4.5        Low 0.97148489 0.9591837
21.57447
```

regressions with proportion emergence

```
propmod<-lm(mod~MEturquoise,data=mergedMEs2)
> summary(propmod)

Call:
lm(formula = mod ~ MEturquoise, data = mergedMEs2)

Residuals:
     Min       1Q   Median       3Q      Max
-0.49172 -0.32779  0.06606  0.30170  0.43732

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.5549     0.1214   4.572  0.00182 **
MEturquoise   0.1330     0.3838   0.347  0.73781
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3838 on 8 degrees of freedom
Multiple R-squared:  0.0148,    Adjusted R-squared:  -0.1084
F-statistic: 0.1201 on 1 and 8 DF,  p-value: 0.7378
```

regressions with average eclosion

```
ecmod<-lm(ave_eclosion~MEturquoise,data=mergedMEs2)
> summary(ecmod)
```

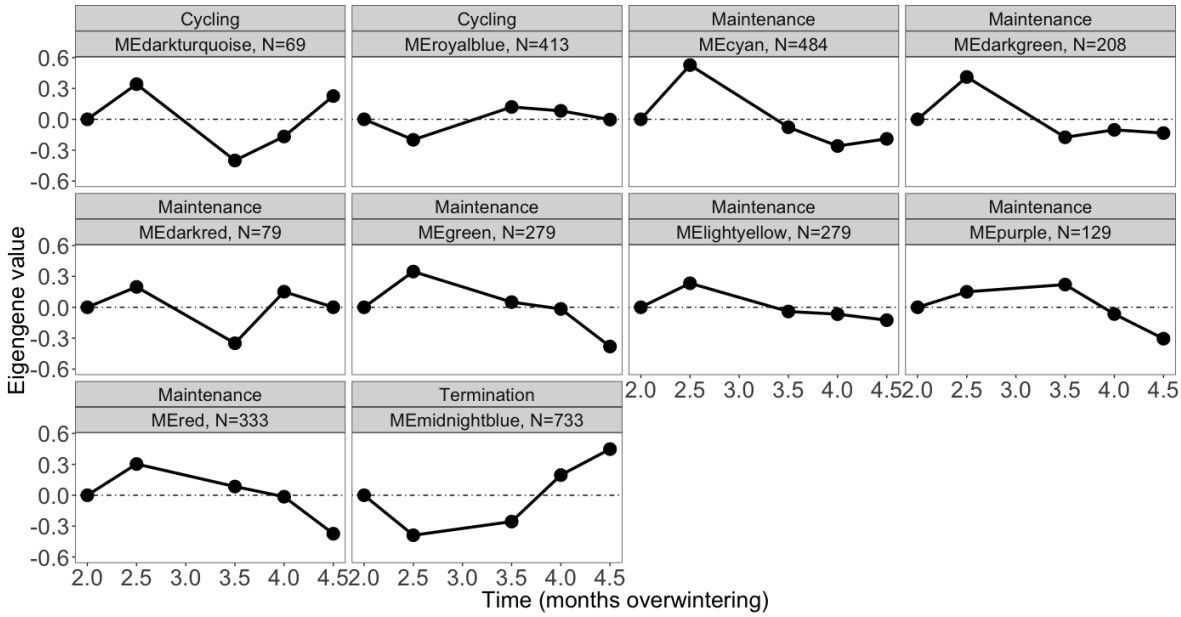
```
Call:
lm(formula = ave_eclosion ~ MEturquoise, data = mergedMEs2)

Residuals:
    Min      1Q  Median      3Q     Max
-27.5588  -4.4608   0.2985   7.8515  13.8788

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   27.559      4.023   6.850 0.000131 ***
MEturquoise   -2.281     12.723  -0.179 0.862155
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.72 on 8 degrees of freedom
Multiple R-squared:  0.004003,  Adjusted R-squared:  -0.1205
F-statistic: 0.03215 on 1 and 8 DF,  p-value: 0.8622
```

Take home; the module found for overall population level differences is not related to the phenotype

Reset baseline expression for common responses

- redid analysis to find power to estimate scale free network---power = 16 like before

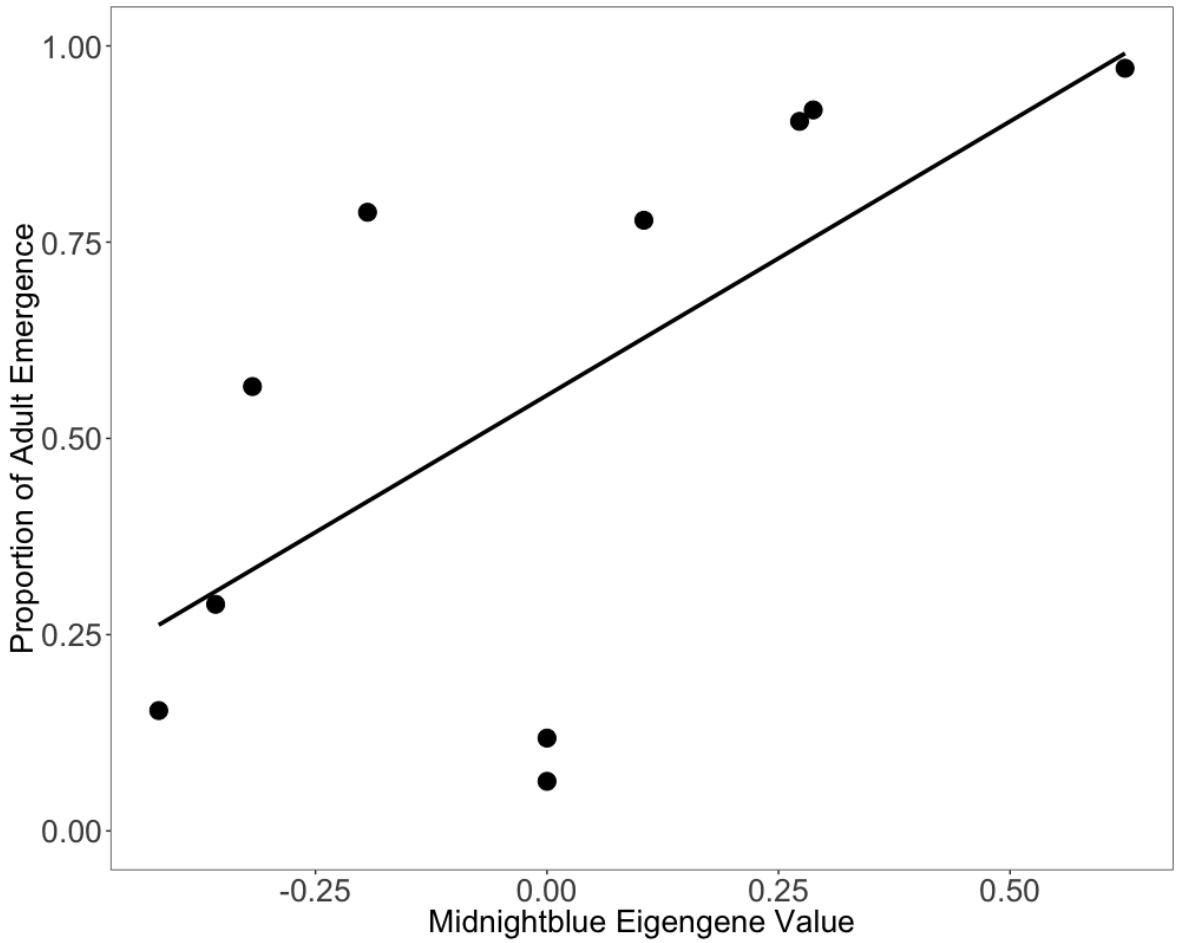
Common responses; **overall eig vs time**



Relationship with proportion emergence

```
> apply(all.dat[,1:10],2,function(x)
{summary(lm(all.dat$mod~x))$coefficient[2,4]})# p value
 MEmidnightblue      MEroyalblue          MEgreen    MElightyellow         MEpurple
         MEcyan            MEred       MEdarkred
      0.04713421       0.90765719       0.13551968       0.22736757       0.60756629
      0.08903030       0.34208676       0.99238184
    MEdarkgreen MEdarkturquoise
      0.18791211       0.44172873
> apply(all.dat[,1:10],2,function(x)
{summary(lm(all.dat$mod~x))$coefficient[2,1]}) # beta
 MEmidnightblue      MEroyalblue          MEgreen    MElightyellow         MEpurple
         MEcyan            MEred       MEdarkred
     0.697883310      0.046254388     -0.553590928     -0.458946602     -0.203075229
    -0.617576644     -0.367805222     -0.003809156
    MEdarkgreen MEdarkturquoise
    -0.496155974     -0.300894235
```

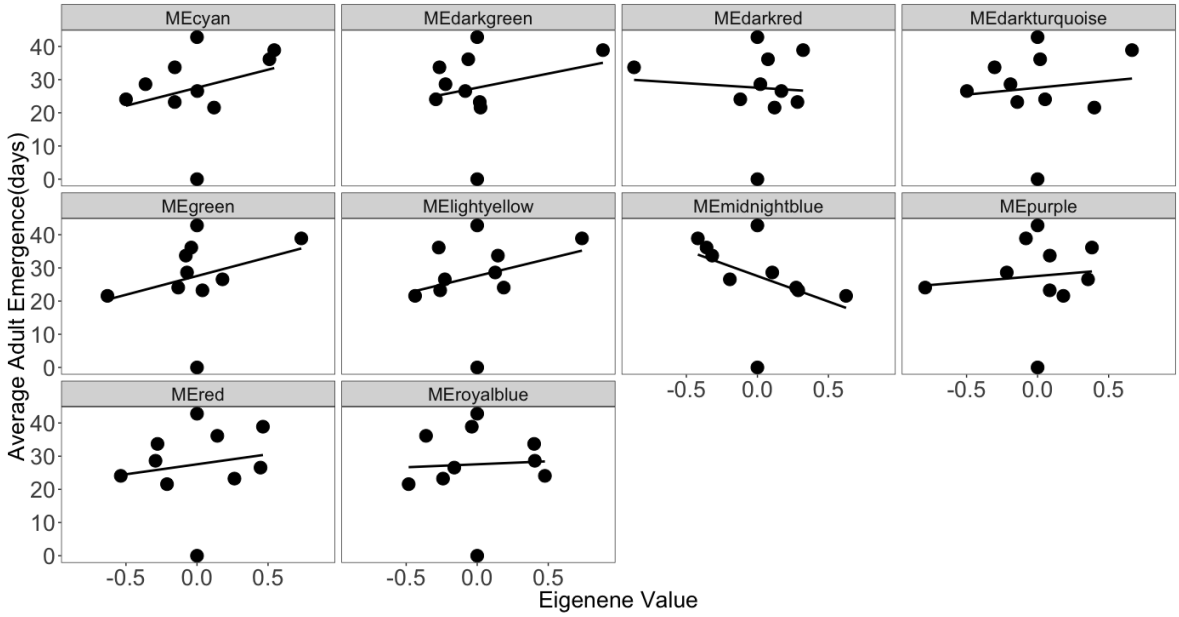
only midnightblue module is sig



Relationship with average adult emergence

```
> apply(all.dat[,1:10],2,function(x)
{summary(lm(all.dat$ave_eclosion~x))$coefficient[2,4]})# p value
 MEmidnightblue      MEroyalblue          MEgreen    MElightyellow         MEpurple
          MEcyan            MEred       MEdarkred
        0.2185414        0.8912286        0.3778584        0.4199857        0.7838553
        0.3965589        0.6450017        0.8360219
    MEdarkgreen MEdarkturquoise
        0.5120202        0.7507097
> apply(all.dat[,1:10],2,function(x)
{summary(lm(all.dat$ave_eclosion~x))$coefficient[2,1]}) # beta
 MEmidnightblue      MEroyalblue          MEgreen    MElightyellow         MEpurple
          MEcyan            MEred       MEdarkred
      -15.392859         1.797375        11.301238        10.378513         3.598409
        10.885836         6.016268        -2.718303
    MEdarkgreen MEdarkturquoise
         8.500415         4.164233
```

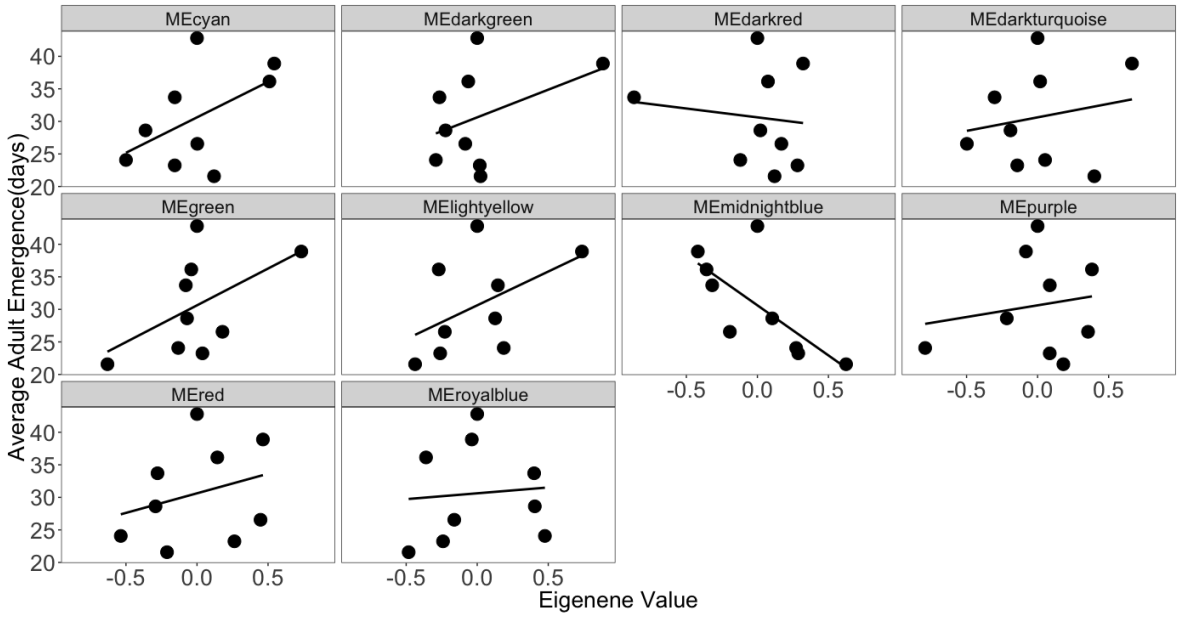
none significant



Thoughts on average eclosion data

I'm tempted to eclude the 2 month high populations because they don't eclose at all. Perhaps including 2 month high is not a good comparison because of this.

Ok, if I do this, midnight blue is the only sample that is sig for both proportion fo adult emergence and days until adult emergence



regressions without 2 month high

```
> apply(all.dat[-1,1:10],2,function(x)
{summary(lm(all.dat$ave_eclosion[-1]~x))$coefficient[2,4]})# p value
 MEmidnightblue      MEroyalblue         MEgreen    MElightyellow         MEpurple
          MEcyan            MEred       MEdarkred
       0.0285026        0.8295862        0.1430067       0.1848082        0.6648201
       0.1610619        0.4627993        0.7442192
    MEdarkgreen MEdarkturquoise
       0.2888008        0.6152071
> apply(all.dat[-1,1:10],2,function(x)
{summary(lm(all.dat$ave_eclosion[-1]~x))$coefficient[2,1]}) # beta
 MEmidnightblue      MEroyalblue         MEgreen    MElightyellow         MEpurple
          MEcyan            MEred       MEdarkred
      -15.392859         1.797375        11.301238       10.378513         3.598409
       10.885836         6.016268        -2.718303
    MEdarkgreen MEdarkturquoise
        8.500415         4.164233
```

Redo population level differences and include 2 month high

stats: taking the average eclosion vs average proportion emergence or adult emergence proportion

```
> signif(cor(m.wide.ave$mod.mean,m.wide.ave[,4:13], use="p"),2)
     MEblack MEbrown MEdarkgreen MEdarkred MEdarkturquoise MEgreenyellow MEgrey
MEgrey60 MElightyellow MEmagenta
[1,]    0.55    0.84       -0.28     -0.91            0.58          -0.3   -0.3
   -0.77         -0.62     -0.32
> corPvalueStudent(cor(m.wide.ave$mod.mean,m.wide.ave[,4:13], use="p"), nSamples
= length(m.wide.ave$mod.mean))
       MEblack     MEbrown MEdarkgreen MEdarkred MEdarkturquoise MEgreenyellow
 MEgrey   MEgrey60 MElightyellow MEmagenta
[1,] 0.3401208 0.07425528    0.6472964 0.0341849       0.3064338     0.6246071
0.6277011 0.1313196      0.263838 0.5953915
```

darkred is sig

adult emergence

```
> signif(cor(m.wide.ave$ecl,m.wide.ave[,4:13], use="p"),2)
     MEblack MEbrown MEdarkgreen MEdarkred MEdarkturquoise MEgreenyellow MEgrey
MEgrey60 MElightyellow MEmagenta
[1,]   -0.94   -0.25        0.11       0.4            0.49          0.21 -0.095
    0.81          0.78       0.9
> corPvalueStudent(cor(m.wide.ave$ecl,m.wide.ave[,4:13], use="p"), nSamples =
length(m.wide.ave$mod.mean))
        MEblack    MEbrown MEdarkgreen MEdarkred MEdarkturquoise MEgreenyellow
 MEgrey    MEgrey60 MElightyellow  MEmagenta
[1,] 0.01762315 0.6861367   0.8603313 0.5100662       0.4015382     0.7342587
0.8790918 0.09642874     0.1175061 0.03567657
```

black and magenta are significant

Redoing connectivity for

```
kruskal.test(kTotal~module1,data=tmc.long2)

    Kruskal-Wallis rank sum test

data:  kTotal by module1
Kruskal-Wallis chi-squared = 9224.6, df = 11, p-value < 2.2e-16

> pairwise.wilcox.test(tmc.long2$kTotal, tmc.long2$module1,
+                   p.adjust.method = "BH")

    Pairwise comparisons using Wilcoxon rank sum test

data:  tmc.long2$kTotal and tmc.long2$module1

              black   brown   darkgreen darkred darkturquoise greenyellow grey60
 lightyellow magenta midnightblue royalblue
brown         2.7e-14 -       -         -       -             -           -
  -           -       -             -
darkgreen     < 2e-16 < 2e-16 -         -       -             -           -
  -           -       -             -
darkred       < 2e-16 < 2e-16 7.7e-14   -       -             -           -
  -           -       -             -
darkturquoise < 2e-16 < 2e-16 0.096     < 2e-16 -             -           -
  -           -       -             -
greenyellow   < 2e-16 < 2e-16 < 2e-16   < 2e-16 2.3e-09       -           -
  -           -       -             -
grey60        0.012   < 2e-16 < 2e-16   < 2e-16 < 2e-16       < 2e-16     -
  -           -       -             -
lightyellow   < 2e-16 < 2e-16 < 2e-16   5.3e-10 < 2e-16       < 2e-16     < 2e-
16 -           -       -             -
magenta       0.349   < 2e-16 < 2e-16   < 2e-16 < 2e-16       < 2e-16     < 2e-
16 < 2e-16     -       -             -
midnightblue  1.5e-09 0.349   < 2e-16   < 2e-16 < 2e-16       < 2e-16     < 2e-
16 1.6e-08     < 2e-16 -             -
royalblue     < 2e-16 < 2e-16 < 2e-16   < 2e-16 < 2e-16       7.9e-12     < 2e-
16 < 2e-16     < 2e-16 < 2e-16       -
turquoise     < 2e-16 < 2e-16 < 2e-16   < 2e-16 < 2e-16       < 2e-16     < 2e-
16 < 2e-16     < 2e-16 < 2e-16       < 2e-16
```

```
P value adjustment method: BH
```

Page 24: 2019-06-18. more analysis ; cerasi commonr responses

On thinking about the data, I'm separating each population for a given time point, but for a time effect, I really need to take the average between them (at a given time point).

The issue with month2, is that high eclosion day is 0 and low 2 month is around 40. The average is 20ish then. So the average at time 2 is misleading

re-estimate the common response from model outputs :

```
cerph<-fread("../Data/08_cerasi_eclosion_tom_modified.csv")
#filter out month 5 , not in transcriptome dataset
cerph<-cerph%>%
  dplyr::filter(month!=5)

#eclosed
cerph9<-cerph%>%
  dplyr::filter(died==0)
#########################################################################
##analyzing proportion emerged with glm, binomial logistic regression
#########################################################################
mod1.1<-glm(eclosed~Population*month,data=cerph9,family="binomial")
summary(mod1.1)#
#anova(mod1.1,test="Chisq")
#estimate just the effects that were sig, and the effect for each month
mod1.12<-glm(eclosed~Population+factor(month),data=cerph9,family="binomial")
summary(mod1.12)

cerph9$mod.pred.prob<-predict(mod1.12,type="response")
cerph9.ave<-cerph9%>%
  group_by(month)%>%
  dplyr::summarise(ave.mod=mean(mod.pred.prob),sd.mod=sd(mod.pred.prob))

cerph9.ave<-cerph9.ave%>%
  filter(month!=3)

cerph9.ave

#########################################################################
#eclosion days, average
#########################################################################
cerph3<-cerph9%>%
  dplyr::filter(month !=2)

mod1.2<-lm(eclosion_day~Population*factor(month),data=cerph3)
summary(mod1.2)
##get estimated model effects
se.mod<-summary(mod1.2)$coefficients[-2,2]
se.mod<-se.mod[-2]
```

```
cerph9.ave$ecl.mod<-
as.vector(c(0,summary(mod1.2)$coefficients[1,1],summary(mod1.2)$coefficients[1,1
]+summary(mod1.2)$coefficients[4:6,1]))

cerph9.ave$ecl.se.mod<-as.vector(c(0,se.mod[1:4]))
cerph9.ave
# A tibble: 5 x 5
  month ave.mod sd.mod ecl.mod ecl.se.mod
  <dbl>   <dbl>  <dbl>   <dbl>      <dbl>
1   2    0.0613 0.0280     0         0
2   2.5  0.205  0.0786    38.9       0.855
3   3.5  0.694  0.0900    33.7       0.981
4   4    0.818  0.0588    28.6       0.951
5   4.5  0.897  0.0355    24.1       0.942
```

Parsing out the sign of each module

```
hl.par$n<-seq(1,length(hl.par$Name))
hl.par$high2month<-rep(0,length(hl.par$Month_FDR))
hl.par$low2month<-rep(0,length(hl.par$Month_FDR))

hl.par2<-cbind(hl.par[,1:18],hl.par[,21:22],hl.par[,19:20])
##making long format dataset
te.long<-gather(hl.par2,treatment,expression,month2_5vs2_high_logFC.x:low2month)
head(te.long)

###some data prepping
treatment<-unique(te.long$treatment)
Population<-c(rep("High",4),rep("Low",4),"High","Low")
month<-c(2.5,3.5,4,4.5,2.5,3.5,4,4.5,2,2)
li.dat<-data.frame(treatment,Population,month)
##preparing extra labels
moduleME<-paste("ME",unique(te.long$module),sep="")
module<-unique(te.long$module)
gg<-data.frame(module,moduleME)
####data merger
names(ss)[2]<-"moduleME"
gg<-inner_join(gg,ss,by="moduleME")
gg$facet<-paste(gg$module,", N=",gg$N,sep="")#facet labels
#gg$color<-gg$module1

####merging data
te.long2<-inner_join(te.long,li.dat,by="treatment")
te.long3<-inner_join(te.long2,gg,by="module")
#te.long3$Name<-toupper(te.long3$Name)
#te.long3<-inner_join(te.long3,go.dat.sub,by="Name")
###module memberhsip
datKME<-signedKME(t(hl.par[,11:18]), mergedMEs2[,1:11])
datKME$n<-seq(1,length(datKME$kMEdarkred))
datkme.long<-
gather(datKME[,-11],module,membership,kMEmidnightblue:kMEdarkturquoise)

#filter only genes with high membership
datkme.long2<-datkme.long%>%
  dplyr::filter(membership>.8 | membership< -.8 )

##ones with negative patterns
```

```
#datkme.long2.neg<-datkme.long%>%
#  dplyr::filter(membership< -0.8)

datkme.long2$module<-substr(datkme.long2$module,4,30)

#merge data set with expression dataset
te.long4<-inner_join(te.long3,datkme.long2,by=c("n","module"))
#plots
te.long4$sign<-ifelse(te.long4$membership>0,"negative","positive")#these are
opposite labels, but it matches the phenotype in this direction much better and
the loadings are arbitary
#te.long4%>%
#
filter(Name=="10001_OASESK45LO45M04LOCUS_12158_TRANSCRIPT_1_2_CONFIDENCE_0.600_L
ENGTH_596_EVGCLASS=MAIN,OKAY,MATCH:OASES")
ggplot(te.long4,aes(x=month,y=expression,group=paste(module,sign)))+geom_point()
+stat_smooth(method="loess",se=FALSE)+facet_wrap(~facet)
#take average of each gxp for a given module and sign
te.long4.ave<-te.long4%>%
  group_by(module,month,sign)%>%
  dplyr::summarise(gxp=mean(expression),gxp.med=median(expression),n=length(uniq
ue(Name)),tot=length(expression),gxp.sd=sd(expression))

ten<-te.long4%>%
  group_by(facet,module,month)%>%
  dplyr::summarise(memberSS=length(unique(Name)))

te.long4.ave<-inner_join(te.long4.ave,gg,by=c("module"))
te.long4.ave<-inner_join(te.long4.ave,ten,by=c("facet","module","month"))
#te.long4.ave$facet2<-paste(te.long4.ave$module,",
N=",te.long4.ave$memberSS,sep="")
te.long4.ave$facet2<-paste(te.long4.ave$module,", ",te.long4.ave$memberSS," high
member genes out of ",te.long4.ave$N,sep="")
te.long4.ave$facet2<-paste(te.long4.ave$memberSS," high member genes out of
",te.long4.ave$N,sep="")
#te.long4.ave$ss<-paste(module,paste(te.long4.ave$n,"high member genes out
of",te.long4.ave$N,"genes"),sep="
#                        ")

ggplot(te.long4.ave,aes(x=month,y=gxp,group=paste(module,sign),color=sign))+geom
_hline(yintercept=0,lty="dotdash")+geom_errorbar(aes(ymin=gxp-
gxp.sd,ymax=gxp+gxp.sd),width=.1)+geom_line(size=1.25)+geom_point(size=5)+facet_
wrap(module~facet2,scale="free")+ylab("Log Fold
Change")+xlab(expression(paste("Time (months at
4",degree,"C)")))+scale_color_manual(values=c("grey50","black"),name="Relationsh
ip with\n Module Eigengene Values")+T+theme(legend.position = c(.85,
0.05),legend.justification = c("right", "bottom"))
```

Correlate modules with each sign with proportion emergence

```
apply(te.wide.merge[,2:21],2,function(x)
{summary(lm(te.wide.merge$ave.mod~x))$coefficient[2,4]})# p value
```

```
        cyan-negative          cyan-positive     darkgreen-negative
darkgreen-positive       darkred-negative
          0.029290645            0.023009508            0.421080026
 0.123814195            0.966714239
     darkred-positive darkturquoise-negative darkturquoise-positive
green-negative         green-positive
          0.757028524            0.453278253            0.341437081
 0.805302818            0.217500531
 lightyellow-negative   lightyellow-positive  midnightblue-negative
 midnightblue-positive        purple-negative
          0.772511632            0.559212171            0.138425347
 0.161230687            0.128796577
      purple-positive           red-negative           red-positive
royalblue-negative     royalblue-positive
          0.169394531            0.742381367            0.143725946
 0.035747364            0.009214083
> ###cyan and royal blue correlate
> apply(te.wide.merge[,2:21],2,function(x)
{summary(lm(te.wide.merge$ave.mod~x))$coefficient[2,1]}) # beta
        cyan-negative          cyan-positive     darkgreen-negative
darkgreen-positive       darkred-negative
           -1.1197001             0.9097996            -0.7203947
 0.7917246             0.0240686
     darkred-positive darkturquoise-negative darkturquoise-positive
green-negative         green-positive
            0.2139435            -0.5044448             0.6502168
-0.1603223             0.6931761
 lightyellow-negative   lightyellow-positive  midnightblue-negative
 midnightblue-positive        purple-negative
            0.1804440             0.6558825             0.5484777
-0.6720739            -0.8212709
      purple-positive           red-negative           red-positive
royalblue-negative     royalblue-positive
            0.6439203            -0.2084917             0.9514867
 0.9259735            -1.6130741
```

Page 25: 2019-06-24. stats dump, stemminer cerasi common responses only

Stemminer revisit for common responses

```
#stemres<-fread("../Data/STEMminer/2019-06-
19_cerasi_rebaseline_commonresponses_STEMmineroutput.csv") # 50 profiles
examined
#significant Profiles include;
#14,46,38,6,41,49,37,44,19,9,43 for 50 profiles examined
#lots of redundant profiles
stemres<-fread("../Data/STEMminer/2019-06-
19_cerasi_rebaseline_commonresponses_STEMmineroutput_30profilesexamined.csv")
#significant profiles for 30 profiles examiend
#25,23,27,3,28,5,22
```

```
#chose this cut off because there is no redudancy while identifying as many
profiles as possible

stem.long<-gather(stemres,treatment,gxp,tm2:tm4_5)
treatment<-unique(stem.long$treatment)
month<-c(2,2.5,3.5,4,4.5)
linker<-data.frame(treatment,month)
stem.long<-inner_join(stem.long,linker,by="treatment")
#stem.long2<-stem.long%>%
#  filter(Profile==14|
Profile==46|Profile==38|Profile==6|Profile==41|Profile==49|Profile==37|Profile==
44|Profile==19|Profile==9|Profile==43)
stem.long2<-stem.long%>%
  filter(Profile==25|
Profile==23|Profile==27|Profile==3|Profile==28|Profile==5|Profile==22)
dim(stem.long2)

stem.long3<-stem.long2%>%
  group_by(Profile,month)%>%
  dplyr::summarise(exp=mean(gxp),gsd=sd(gxp),N=length(gxp))
stem.long3$facet<-paste("Profile ",stem.long3$Profile,",
N=",stem.long3$N,sep="")

ggplot(stem.long3,aes(x=month,y=exp,group=Profile))+geom_hline(yintercept =
0,lty="dotdash")+geom_errorbar(aes(ymin=exp-
gsd,ymax=exp+gsd),width=.05)+geom_line(size=1.15,color="grey50")+facet_wrap(~fac
et)+geom_point(size=3)+T+xlab(expression(paste("Time (months at
4",degree,"C)")))+ylab("Log Fold Change")
#+stat_smooth(method="loess",colour="grey50")
```

Ok, correlate with phenotype; stemm miner data

```
#cerph9.ave
#
stem.long3<-stem.long3%>%
  arrange(Profile)

stem.wide<-spread(stem.long3[,-4:-6],Profile,exp)
stem.wide1<-inner_join(stem.wide,cerph9.ave,by="month")
#stats
###proportion
apply(stem.wide1[,2:8],2,function(x)
{summary(lm(stem.wide1$ave.mod~x))$coefficient[2,4]})# p value
#sig- 5,25
apply(stem.wide1[,2:8],2,function(x)
{summary(lm(stem.wide1$ave.mod~x))$coefficient[2,1]}) # beta

####eclosion
apply(stem.wide1[-1,2:8],2,function(x)
{summary(lm(stem.wide1$ecl.mod[-1]~x))$coefficient[2,4]})# p value
###sig; 6, 9,14,41,44 for 50 profile
###30 profile --- 3, 5, 25, 28
apply(stem.wide1[-1,2:8],2,function(x)
{summary(lm(stem.wide1$ecl.mod[-1]~x))$coefficient[2,1]}) # beta

###if we include 0
```

```
apply(stem.wide1[,2:8],2,function(x)
{summary(lm(stem.wide1$ecl.mod~x))$coefficient[2,4]})# p value
###23
apply(stem.wide1[,2:8],2,function(x)
{summary(lm(stem.wide1$ecl.mod~x))$coefficient[2,1]}) # beta
```

Create some plots for common responses STEM MINER

```
stem.comb<-inner_join(stem.long3,cerph9.ave,by="month")

#stem.comb

##parsing for proportion
stem.comb.prop<-stem.comb%>%
  filter(Profile ==25|Profile==5)
stem.comb.prop$facet<-factor(stem.comb.prop$facet,levels=c("Profile 5, N=94"
,"Profile 25, N=158" ))
#ggplot(stem.comb.prop,aes(x=exp,y=ave.mod,group=Profile))+stat_smooth(method="l
m",se=FALSE,size=1.15,colour="grey50")+geom_point(size=5)+facet_wrap(~facet)+xla
b("Log Fold Change")+ylab("Proportion of Adult Emergence")+T
#with error bars
pro1<-
ggplot(stem.comb.prop,aes(x=exp,y=ave.mod,group=Profile))+geom_errorbar(aes(ymin
=ave.mod-sd.mod,ymax=ave.mod+sd.mod),width=.1)+geom_errorbarh(aes(xmin=exp-
gsd,xmax=exp+gsd),height=.04)+stat_smooth(method="lm",se=FALSE,size=1.15,colour=
"grey50")+geom_point(size=5)+facet_wrap(~facet,scale="free")+xlab("Log Fold
Change")+ylab("Proportion of Adult Emergence")+T


###parsing for elcosions
step.comb.ecl<-stem.comb%>%
  filter(month!=2)%>%
  filter(Profile==3|Profile==5|Profile==25|Profile==28)#3, 5, 25, 28
#step.comb.ecl
step.comb.ecl$facet<-factor(step.comb.ecl$facet,levels=c("Profile 3, N=265"
,"Profile 5, N=94"  ,"Profile 25, N=158" ,"Profile 28, N=109"))
ecl1<-
ggplot(step.comb.ecl,aes(x=exp,y=ecl.mod,group=Profile))+geom_errorbar(aes(ymin=
ecl.mod-
ecl.se.mod,ymax=ecl.mod+ecl.se.mod),width=.1)+geom_errorbarh(aes(xmin=exp-
gsd,xmax=exp+gsd),height=.4)+geom_point(size=5)+stat_smooth(method="lm",se=FALSE
,size=1.15,colour="grey50")+facet_wrap(~facet,nrow=1,scale="free")+xlab("Log
Fold Change")+ylab("Adult Emergence Timing (days)")+T

#####grabbing only sig profiles
stem.comb.pro<-stem.comb%>%
  filter(Profile==3|Profile==5|Profile==25|Profile==28)
stem.comb.pro$facet<-factor(stem.comb.pro$facet,levels=c("Profile 3, N=265"
,"Profile 5, N=94"  ,"Profile 25, N=158" ,"Profile 28, N=109"))
#factor(stem.comb$facet,levels=c("Profile 3, N=265" ,"Profile 5, N=94"  ,
"Profile 22, N=79",  "Profile 23, N=155" ,"Profile 25, N=158", "Profile 27,
N=50" ,"Profile 28, N=109"))
```

```
profiles<-
ggplot(stem.comb.pro,aes(x=month,y=exp,group=Profile))+geom_hline(yintercept =
0,lty="dotdash")+geom_errorbar(aes(ymin=exp-
gsd,ymax=exp+gsd),width=.15,size=1.1)+geom_line(size=1.15,color="grey50")+facet_
wrap(~facet,nrow=1)+geom_point(size=5)+T+xlab(expression(paste("Time (months at
4",degree,"C)")))+ylab("Log Fold Change")
profiles

profiles/((pro1|ecl1)+plot_layout(widths=c(.35,.65)))


#### color the tiem points
pro1.col<-
ggplot(stem.comb.prop,aes(x=exp,y=ave.mod,group=Profile,colour=month))+geom_erro
rbar(aes(ymin=ave.mod-
sd.mod,ymax=ave.mod+sd.mod),width=.1)+geom_errorbarh(aes(xmin=exp-
gsd,xmax=exp+gsd),height=.04)+stat_smooth(method="lm",se=FALSE,size=1.15,colour=
"grey50")+geom_point(size=5)+facet_wrap(~facet,scale="free")+xlab("Log Fold
Change")+ylab("Proportion of Adult Emergence")+T
pro1.col

ecl1.col<-
ggplot(step.comb.ecl,aes(x=exp,y=ecl.mod,group=Profile,colour=month))+geom_error
bar(aes(ymin=ecl.mod-
ecl.se.mod,ymax=ecl.mod+ecl.se.mod),width=.1)+geom_errorbarh(aes(xmin=exp-
gsd,xmax=exp+gsd),height=.4)+geom_point(size=5)+stat_smooth(method="lm",se=FALSE
,size=1.15,colour="grey50")+facet_wrap(~facet,nrow=1,scale="free")+xlab("Log
Fold Change")+ylab("Adult Emergence Timing (days)")+T
ecl1.col

profiles.col<-
ggplot(stem.comb.pro,aes(x=month,y=exp,group=Profile,colour=month))+geom_hline(y
intercept = 0,lty="dotdash")+geom_errorbar(aes(ymin=exp-
gsd,ymax=exp+gsd),width=.15,size=1.1)+stat_smooth(size=1.15,colour="grey50")+fac
et_wrap(~facet,nrow=1)+geom_point(size=5)+T+xlab(expression(paste("Time (months
at 4",degree,"C)")))+ylab("Log Fold
Change")#+geom_line(size=1.15,color="grey50")
profiles.col


##put it all together
profiles.col/((pro1.col|ecl1.col)+plot_layout(widths=c(.35,.65)))
```

Page 26: 2019-07-01. stats dump ; cerasi divergent responses

testing interaction between population and module on phenotypes

```
##proportion
> apply(cp.wide2[,6:11],2,function(module)
{summary(lm(cp.wide2$ave.mod~cp.wide2$Population*module))$coefficient[,4]})
                              darkseagreen4 negative darkseagreen4 positive
mediumorchid negative mediumorchid positive
```

```
(Intercept)                                0.87204442             0.92992974
        0.8721410             0.9645869
cp.wide2$PopulationLow                     0.13109332             0.08519676
        0.1236924             0.1264185
module                                     0.10768148             0.06661807
        0.1272397             0.1465161
cp.wide2$PopulationLow:module              0.08046733             0.04670370
        0.1087025             0.1229579
                                thistle2 negative thistle2 positive
(Intercept)                            0.03120859        0.01471711
cp.wide2$PopulationLow                 0.41663150        0.60524339
module                                 0.68563707        0.40015538
cp.wide2$PopulationLow:module          0.25442229        0.21021745

#eclosion
> apply(cp.wide3[,6:11],2,function(module)
{summary(lm(cp.wide3$ecl.mean~cp.wide3$Population*module))$coefficient[,4]})
                                darkseagreen4 negative darkseagreen4 positive
mediumorchid negative mediumorchid positive
(Intercept)                                0.01262539             0.00461915
      0.008491811             0.03371633
cp.wide3$PopulationLow                     0.22713699             0.10825594
      0.126471427             0.21678810
module                                     0.30813178             0.15040630
      0.202998181             0.31790011
cp.wide3$PopulationLow:module              0.25980594             0.11798476
      0.173879993             0.28417671
                                thistle2 negative thistle2 positive
(Intercept)                           0.002255349      0.0008511635
cp.wide3$PopulationLow                0.542888803      0.8150731706
module                                0.504044912      0.3887091650
cp.wide3$PopulationLow:module         0.238691854      0.2370570948
```

darkseagreen4 positive had sig interaction

Page 27: 2019-07-05. numbers dump: the number of genes per module and module+sign

cerasi:

population level response

number of genes per module

Var1	Freq
blue	269
brown	182
grey	1

total number of high member genes

Var1	Freq
blue	134
brown	79
grey	1

total number of high member genes per module per sign

module	sign	n
blue	negative	59
blue	positive	75
brown	negative	16
brown	positive	63
grey	positive	1

total numbers of high member genes per module per sign

module	sign	n
blue	negative	118
blue	positive	150
brown	negative	32
brown	positive	126

common

total number of high member genes (both signs counted)

module	memberSS
cyan	396
green	279
midnightblue	570
royalblue	276

total number by sign

module	sign	memberSS
cyan	negative	311
cyan	positive	85
green	negative	229
green	positive	50
midnightblue	negative	490
midnightblue	positive	80
royalblue	negative	168
royalblue	positive	108

divergent

color	sign	mod.num
blueviolet	negative	547
blueviolet	positive	3511
darkseagreen4	negative	787
darkseagreen4	positive	1198
mediumorchid	negative	1167
mediumorchid	positive	1628
thistle2	negative	41
thistle2	positive	164

pomonella:

need to redo some stats; using tom's dataset, but to be consistent, I'll use greg's

proportional eclosion data, we found a population by time interaction from 2-6 months

```
fm<-glm(ecl ~ factor(treatment)*factor(host),family=binomial,data=pom.dat.ecl)
> summary(fm)

Call:
glm(formula = ecl ~ factor(treatment) * factor(host), family = binomial,
    data = pom.dat.ecl)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1899   0.4366   0.5430   0.7011   1.2846

Coefficients:
```

```
                                 Estimate Std. Error z value Pr(>|z|)
(Intercept)                        0.9808     0.1596   6.147 7.91e-10 ***
factor(treatment)3                -1.2293     0.2197  -5.594 2.21e-08 ***
factor(treatment)4                -0.8165     0.2179  -3.748 0.000178 ***
factor(treatment)5                 0.8591     0.2615   3.285 0.001018 **
factor(treatment)6                 0.2973     0.2413   1.232 0.218079
factor(host)haw                    0.6232     0.2407   2.589 0.009616 **
factor(treatment)3:factor(host)haw 1.1587     0.3323   3.487 0.000488 ***
factor(treatment)4:factor(host)haw 1.5151     0.3755   4.035 5.47e-05 ***
factor(treatment)5:factor(host)haw -0.5699     0.3610  -1.578 0.114457
factor(treatment)6:factor(host)haw -1.0582     0.3343  -3.166 0.001548 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2326.4  on 2108  degrees of freedom
Residual deviance: 2117.7  on 2099  degrees of freedom
AIC: 2137.7

Number of Fisher Scoring iterations: 4
```

eclosion day data ; time and pop main effects

```
summary(ecl.mod)

Call:
lm(formula = dayToEclosion ~ factor(host) * factor(treatment),
   data = pom.dat.ecl)

Residuals:
   Min      1Q  Median      3Q     Max
-46.326  -9.014  -0.014   8.967  55.876

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                        66.090      1.150  57.458  < 2e-16 ***
factor(host)haw                    12.236      1.536   7.967 3.06e-15 ***
factor(treatment)3                 11.076      1.940   5.708 1.36e-08 ***
factor(treatment)4                  4.475      1.802   2.483   0.0131 *
factor(treatment)5                  1.033      1.563   0.661   0.5087
factor(treatment)6                -13.076      1.638  -7.982 2.74e-15 ***
factor(host)haw:factor(treatment)3  3.866      2.409   1.605   0.1087
factor(host)haw:factor(treatment)4  1.910      2.311   0.826   0.4088
factor(host)haw:factor(treatment)5 -1.039      2.053  -0.506   0.6129
factor(host)haw:factor(treatment)6  4.218      2.219   1.901   0.0575 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 13.8 on 1592 degrees of freedom
  (507 observations deleted due to missingness)
Multiple R-squared:  0.3751,    Adjusted R-squared:  0.3716
F-statistic: 106.2 on 9 and 1592 DF,  p-value: < 2.2e-16
```

common

all Genes

color	total_num
darkred	755
green	563
greenyellow	997
grey	2
pink	275

high member genes by module only

color	n
darkred	409
green	366
greenyellow	422
grey	2
pink	128

high mbmer genes by module and sign

color	sign	n
darkred	negative	83
darkred	positive	326
green	negative	148
green	positive	218
greenyellow	negative	128
greenyellow	positive	294
grey	negative	1
grey	positive	1
pink	negative	38
pink	positive	90

inserting some stats with correlating ave gxp with newly anlayzed phenotypes (no modules gxp sig):

proportion

```
apply(hmc.wide2[,2:9],2,function(x)
{summary(lm(hmc.wide2$common.mean~x))$coefficient[2,4]})# p value
    darkred negative     darkred positive       green negative       green
positive greenyellow negative
           0.6454439            0.7384548            0.6558284
 0.7729242            0.6559972
greenyellow positive        pink negative        pink positive
           0.5608325            0.8651901            0.8915960
```

eclosion

```
apply(hmc.wide2[,2:9],2,function(x)
{summary(lm(hmc.wide2$ecl~x))$coefficient[2,4]})# p value
    darkred negative     darkred positive       green negative       green
positive greenyellow negative
           0.2971868            0.4202125            0.2767820
 0.4228912            0.3044809
greenyellow positive        pink negative        pink positive
           0.2228193            0.7834458            0.8709504
```

again, not sig

divergent

2019-07-08 contd

total number per module; and extra column with number of high member genes

color	total_num	n
black	557	194
blue	590	300
grey	2	2
grey60	542	209
red	1017	651
turquoise	194	152

total number per module and sign

color	sign	n
black	negative	21
black	positive	173
blue	negative	116
blue	positive	184
grey	negative	1
grey	positive	1
grey60	negative	29
grey60	positive	180
red	negative	179
red	positive	472
turquoise	negative	28
turquoise	positive	124

stats on divergent responses

tested main effects of time and host (time by host interaction was not sig, so trying a simpler model)

proportion data

```
mo.out1<-apply(ana.dat[,5:14],2,function(module)
{summary(lm(ana.dat$mean.prop~ana.dat$Host+module))$coefficient[,4]})
> round(data.frame(mo.out1),4)
               black.negative black.positive blue.negative blue.positive
grey60.negative grey60.positive red.negative
(Intercept)             0.0002         0.0001        0.0007        0.0002
   0.0000          0.0000       0.0000
ana.dat$HostHaw         0.1233         0.1473        0.1193        0.2115
   0.0623          0.0794       0.2171
module                  0.7115         0.9602        0.3909        0.6712
   0.2345          0.3174       0.9762
               red.positive turquoise.negative turquoise.positive
(Intercept)           0.0000             0.0002             0.0005
ana.dat$HostHaw       0.1806             0.4693             0.5080
module                0.9084             0.5572             0.3668
```

no factors are significant

adult emergence data

```
mo.out2<-apply(ana.dat[,5:14],2,function(module)
{summary(lm(ana.dat$ecl~ana.dat$Host+module))$coefficient[,4]})
> #mo.out2
> round(data.frame(mo.out2),4)
               black.negative black.positive blue.negative blue.positive
grey60.negative grey60.positive red.negative
(Intercept)             0.0000         0.0000        0.0000         0.000
   0.0000          0.0000       0.0000
ana.dat$HostHaw         0.0658         0.0756        0.1543         0.154
   0.0420          0.0404       0.0120
module                  0.1241         0.2505        0.7709         0.810
   0.5524          0.5301       0.1123
               red.positive turquoise.negative turquoise.positive
(Intercept)           0.0000             0.0000             0.0001
ana.dat$HostHaw       0.0157             0.2898             0.2110
module                0.1542             0.4677             0.5510
```

module gxp differ between hosts for grey60 (positive and negative) and red (postiive and negative)

Summary of Enrichment analysis

There are two levels in which we can do a functional enrichment analysis :

1. at the module level with high member Genes
2. at the module and sign level with high member genes

Cerasi:

population level response

by module -

- blue - no GO or KEGG enrichment
- brown - no GO or KEGG enrichment

by module and sign

- blue positive - none
- blue negative - none
- brown positive - none
- brown negative - none

common responses

by module-

- cyan - no GO or KEGG enrichment
- green - no GO or KEGG enrichment
- midnightblue - GO = neurogenesis
- royal blue - no GO or KEGG enrichment

by module and sign-

- cyan positive - none
- cyan negative - none
- green positive - none
- green negative -none

- midnightblue positive - none (downregulated over time)
- midnigthblue negative - GO = neurogenesis (up-regatuled over time)
- royal blue positive - none
- royal blue negative - none

divergent responses

by module and sign

- blueviolet negative -none
- blueviolet positive KEGG: dme04391:Hippo signaling pathway - fly, dme04320:Dorso-ventral axis formation, dme04931:Insulin resistance; and
    - GO:0007476~imaginal disc-derived wing morphogenesis
    - GO:0007411~axon guidance
    - GO:0048813~dendrite morphogenesis
    - GO:0006468~protein phosphorylation
    - GO:0007015~actin filament organization
    - GO:0007298~border follicle cell migration
    - GO:0048749~compound eye development
    - GO:0006351~transcription, DNA-templated
    - GO:0006355~regulation of transcription, DNA-templated
    - GO:0008045~motor neuron axon guidance
    - GO:0007391~dorsal closure
    - GO:0007616~long-term memory
    - GO:0008586~imaginal disc-derived wing vein morphogenesis
    - GO:0007424~open tracheal system development
    - GO:0030707~ovarian follicle cell development
    - GO:0048477~oogenesis
    - GO:0007480~imaginal disc-derived leg morphogenesis
    - GO:0007422~peripheral nervous system development
    - GO:0007614~short-term memory
    - GO:0000122~negative regulation of transcription from RNA polymerase II promoter
    - GO:0008355~olfactory learning
    - GO:0007155~cell adhesion
    - GO:0046580~negative regulation of Ras protein signal transduction
    - GO:0045944~positive regulation of transcription from RNA polymerase II promoter
    - GO:0008360~regulation of cell shape
    - GO:0016318~ommatidial rotation
    - GO:0016477~cell migration
- darkseagreen4 negative -none
- darkseagreen4 positive -none
- mediumorchid negative -none
- mediumorchid positive -none
- thistle2 negative -none
- thistle2 positive -none

Pomonella

Population level responses

- none

Common Responses

by module -

- darkred - GO = GO:0007411~axon guidance and GO:0008045 motor neuron axon guidance; no kegg
- green - none
- greenyellow -none
- pink - none

by module and sign

- darkred positive (increase in expression over time) - GO:0007411 axon guidance; GO:0008045 motor neuron axon guidance
- darkred negative - none
- green positive - none
- green negative - none
- greenyellow positive (decrease in expression over time) - GO:0006032 chitin catabolic process; GO:0007498 mesoderm development
- greenyellow negative - none
- pink postiive - none
- pink negative -none

Divergent responses

by module and sign

- black negative - none
- black positive - none
- blue negative - none
- blue positive - none
- grey60 negative - none
- grey60 positive - KEGG: Oxphosph; GO:0032543~mitochondrial translation, GO:0006120~mitochondrial electron transport, NADH to ubiquinone
- red negative -none
- red positive - GO:0007411~axon guidance ,GO:0007476~imaginal disc-derived wing morphogenesis,GO:0006351~transcription, DNA-templated, GO:0045944~positive regulation of transcription from RNA polymerase II promoter
- turquoise negative -none
- turqoise positive - KEGG: dme03010:Ribosome; GO:0002181~cytoplasmic translation, GO:0006412~translation, GO:0051298~centrosome duplication, GO:0000027~ribosomal large subunit assembly, GO:0000028~ribosomal small subunit assembly, GO:0051297~centrosome organization

---

Page 28: 2019-07-16. Helping Jbrown with data analysis and figure preparation

From DHahn meeting, he wants:

1. The relationship between wet mass, lean masss, lipid mass on day 1 by strain and photoperiod
    - If NO- report but no figures
    - If YES- report then add figuers in a multi panel
    - Use day 1 as a covariate in the analysis to determine effect of photoperiod and strain on different metrics of mass on wandering day
2. Do the diff metrics of mass differ by strain and photoperiod on wandering day.

Day 1

Wet mass

```
mixedL1.2=lmer(wet_mass ~ photoperiod*strain+lean_mass + (1|rep/cohort)
,data=dat.an, REML = TRUE)
lmerTest::step(mixedL1.2)
Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                  Eliminated    Sum Sq   Mean Sq NumDF  DenDF F value    Pr(>F)

lean_mass                  1 0.0000293 0.0000293     1 290.19  0.1634    0.6863

photoperiod:strain         2 0.0004404 0.0004404     1 326.92  2.4659    0.1173

photoperiod                0 0.0099830 0.0099830     1 383.92 55.4637 6.339e-13
***
strain                     0 0.0162875 0.0162875     1  25.89 90.4901 6.190e-10
***
---
```

Lean MASS

```
mixedL1.3=lmer(lean_mass ~ photoperiod*strain+(1|rep/cohort) ,data=data, REML =
TRUE)
lmerTest::step(mixedL1.3)


Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                  Eliminated   Sum Sq  Mean Sq NumDF  DenDF F value  Pr(>F)
photoperiod:strain         1 0.210443 0.210443     1 375.92  3.3150 0.06945 .
strain                     2 0.026950 0.026950     1  26.77  0.4195 0.52270
photoperiod                3 0.033505 0.033505     1 374.79  0.5215 0.47063
---
```

Lipid Mass

```
mixedL1=lmer(lipid_mass ~ photoperiod*strain+lean_mass * (1|rep/cohort)
,data=dataL1, REML = TRUE)
```

```
Fixed effects:
                          Estimate Std. Error         df t value Pr(>|t|)
(Intercept)              1.855e-02  1.859e-02  7.000e+01   0.998    0.322
photoperiod16           -3.702e-03  1.111e-02  7.000e+01  -0.333    0.740
strainUZ                 1.113e-05  1.022e-02  7.000e+01   0.001    0.999
lean_mass               -1.738e+00  1.786e+00  7.000e+01  -0.973    0.334
photoperiod16:strainUZ   1.672e-02  1.461e-02  7.000e+01   1.144    0.256



T<-theme_bw()+theme(text=element_text(size=20),axis.text=element_text(size=20),
panel.grid.major=element_blank(), panel.grid.minor.x = element_blank(),
panel.grid = element_blank(), legend.key = element_blank())+
theme(legend.position="none")

d1_lm<-
ggplot(dataL1,aes(x=photoperiod,y=lipid_mass,fill=strain))+geom_boxplot()+scale_
fill_manual(labels=c("Long Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+T+ylab(
"Lipid Mass (g)")+xlab("Photoperiod")
d1_lm
```

sig effect of photoperiod; used lean mass as a covariate

Wandering day

Wet mass

```
mixedLW3=lmer(wet_mass ~ photoperiod * strain +lean_mass +(1|rep/cohort)
,data=dataLW, REML = TRUE)

Fixed effects:
                          Estimate Std. Error         df t value Pr(>|t|)
(Intercept)               0.083280   0.002537 190.000000  32.822  < 2e-16 ***
photoperiod16            -0.021455   0.003673 190.000000  -5.841 2.21e-08 ***
strainUZ                  0.014469   0.003625 190.000000   3.991 9.38e-05 ***
lean_mass                 0.001161   0.001377 190.000000   0.843    0.400
photoperiod16:strainUZ    0.006668   0.005194 190.000000   1.284    0.201
```

Lean MASS

```
mixedLW2=lmer(lean_mass ~ photoperiod * strain +(1|rep/cohort) ,data=dataLW,
REML = TRUE)
lmerTest::step(mixedLW2)

Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                  Eliminated   Sum Sq  Mean Sq NumDF   DenDF F value Pr(>F)
photoperiod:strain          1 0.141911 0.141911     1 184.054  1.3256 0.2511
photoperiod                 2 0.006671 0.006671     1 183.596  0.0618 0.8040
strain                      3 0.011131 0.011131     1  14.491  0.1035 0.7522
```

Lipid Mass

```
mixedLW=lmer(lipid_mass ~ photoperiod * strain +lean_mass +(1|rep/cohort)
,data=dataLW, REML = TRUE)
lmerTest::step(mixedLW)


Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                   Eliminated     Sum Sq    Mean Sq NumDF   DenDF  F value
 Pr(>F)
lean_mass                   1 0.00000007 0.00000007     1  16.247   0.0086
 0.9272
photoperiod:strain          2 0.00001164 0.00001164     1 185.644   1.4634
 0.2279
photoperiod                 0 0.00083453 0.00083453     1 191.042 104.7444 <
2.2e-16 ***
strain                      0 0.00013262 0.00013262     1 186.252  16.6458
6.674e-05 ***
---
```

sig effect of photoperiod and strain

figure all in one :

structure of data

```
glimpse(data)
Observations: 411
Variables: 23
$ cohort                              <fct> 20180412, 20180412, 20180412,
20180412, 20180412, 20180412, 20180412, 2018…
$ rep                                 <dbl> 29, 29, 29, 29, 29, 29, 29, 29, 29,
29, 29, 29, 29, 29, 29, 15, 15, 15, 15…
$ sample_day                          <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,…
$ sample_id                           <fct> 0506-01, 0506-02, 0506-03, 0506-04,
0506-05, 0506-06, 0506-07, 0506-08, 05…
$ tag_id                              <chr> "0506-01", "0506-02", "0506-03",
"0506-04", "0506-05", "0506-06", "0506-07…
$ batch                               <chr> "13", "13", "13", "13", "13", "13",
"13", "13", "13", "13", "13", "13", "1…
$ FAME                                <chr> "F", "F", NA, "F", "F", NA, NA, NA,
"F", "F", NA, "F", "F", NA, NA, "F", "…
$ `5th_date`                          <dbl> 20180506, 20180506, 20180506,
20180506, 20180506, 20180506, 20180506, 2018…
$ treat                               <fct> UZ16, UZ16, UZ16, UZ12, UZ12, UZ12,
UZ12, UZ12, BE16, BE16, BE16, BE12, BE…
$ `Microtube wt + Beads`              <dbl> 1.9374, 2.0227, 2.0014, 2.0463,
1.9731, 2.0759, 2.1160, 2.0542, 2.2007, 2.…
$ `Microtube wt + Beads + WET Larvae` <dbl> 1.9986, 2.0782, 2.0402, 2.1012,
2.0273, 2.1288, 2.1714, 2.1023, 2.2179, 2.…
$ `Microtube + Beads + DRY Larvae`    <dbl> 1.9500, 2.0326, 2.0076, 2.0571,
1.9831, 2.0856, 2.1259, 2.0630, 2.2038, 2.…
$ `Microtube + Beads + LEAN Larvae`   <dbl> 1.9517, 2.0352, 2.0115, 2.0593,
1.9857, 2.0885, 2.1282, 2.0659, 2.2075, 2.…
```

```
$ `TV wt`                          <dbl> 14.3068, 14.5217, 13.9372, 15.3506,
15.3435, 15.0740, 15.4462, 14.3725, 15…
$ `TV wt + DRY Lipids`             <dbl> 14.3098, 14.5242, 13.9376, 15.3521,
15.3435, 15.0752, 15.4470, 14.3757, 15…
$ wet_mass                         <dbl> 0.0612, 0.0555, 0.0388, 0.0549,
0.0542, 0.0529, 0.0554, 0.0481, 0.0172, 0.…
$ dry_mass                         <dbl> 0.0126, 0.0099, 0.0062, 0.0108,
0.0100, 0.0097, 0.0099, 0.0088, 0.0031, 0.…
$ lean_mass                        <dbl> 0.0143, 0.0125, 0.0101, 0.0130,
0.0126, 0.0126, 0.0122, 0.0117, 0.0068, 0.…
$ lipid_mass                       <dbl> 0.0030, 0.0025, 0.0004, 0.0015,
0.0000, 0.0012, 0.0008, 0.0032, -0.0002, -…
$ tag_mass                         <dbl> 0.0008408, 0.0007848, 0.0007288,
0.0008068, 0.0008096, 0.0008056, 0.000833…
$ calc_lean_mass                   <dbl> -0.0017, -0.0026, -0.0039, -0.0022,
-0.0026, -0.0029, -0.0023, -0.0029, -0…
$ photoperiod                      <fct> 16, 16, 16, 12, 12, 12, 12, 12, 16,
16, 16, 12, 12, 12, 12, 16, 16, 16, 16…
$ strain                           <fct> UZ, UZ, UZ, UZ, UZ, UZ, UZ, UZ, BE,
BE, BE, BE, BE, BE, BE, UZ, UZ, UZ, UZ…


###exploring data points that look like outliers, but dont see a reason to
exclude them

data%>%
  filter(lean_mass>.099)%>%
  dplyr::select(dry_mass,sample_id,lean_mass,lipid_mass)

  dry_mass sample_id lean_mass lipid_mass
      <dbl> <fct>         <dbl>      <dbl>
 1    0.101 0411-08       0.100    0.00720
 2    0.134 0531-05       0.127    0.0135
 3    0.117 0602-07       0.107    0.0175
```

converting to long to make multi panel plots

```
d.long<-gather(data,weights,grams,wet_mass:lipid_mass)
d.long2<-d.long%>%
  dplyr::filter(sample_day=="1" | sample_day=="W")%>%
  dplyr::filter(weights!="dry_mass")


d.long2$photoperiod<-substr(d.long2$treat,3,4)
d.long2$strain<-substr(d.long2$treat,1,2)

d.long2$facet<-ifelse(d.long2$sample_day=="1","Day 1","Wandering Day")

d.long2<-d.long2%>%
  filter(grams>0)



d.long2%>%
  filter(sample_day==1,weights=="lipid_mass")%>%
```

```
  dplyr::select(grams,sample_id)%>%
  filter(grams>.2)

##excluding this outlier


d.long3<-d.long2%>%
  filter(sample_id!="0323-06" & grams< 0.264)
```

Making the plot

```
d.long<-gather(data,weights,grams,wet_mass:lipid_mass)
d.long2<-d.long%>%
  dplyr::filter(sample_day=="1" | sample_day=="W")%>%
  dplyr::filter(weights!="dry_mass")


d.long2$photoperiod<-substr(d.long2$treat,3,4)
d.long2$strain<-substr(d.long2$treat,1,2)

d.long2$facet<-ifelse(d.long2$sample_day=="1","Day 1","Wandering Day")

d.long2<-d.long2%>%
  filter(grams>0)



d.long2%>%
  filter(sample_day==1,weights=="lipid_mass")%>%
  dplyr::select(grams,sample_id)%>%
  filter(grams>.2)


d.long3<-d.long2%>%
  filter(sample_id!="0323-06" & grams< 0.264)

d.long3$mg<-round(d.long3$grams,5)*1000

ggplot(d.long3,aes(x=photoperiod,y=mg,fill=strain))+facet_wrap(facet~weights,sca
les="free")+geom_boxplot()+T+ylab("Mass (mg)")+xlab("Photoperiod (hours)")


d.long3$mass_type<-ifelse(d.long3$weights=="lean_mass","Lean
Mass",ifelse(d.long3$weights=="lipid_mass","Lipid Mass","Wet Mass"))

d.long3$mt.fac<-factor(d.long3$mass_type,levels=c("Wet Mass","Lean Mass","Lipid
Mass"))


ggplot(d.long3,aes(x=photoperiod,y=mg,fill=strain))+
facet_wrap(mt.fac~facet,scales="free",ncol=2)+geom_boxplot()+
T+ylab("Mass (mg)")+xlab("Photoperiod (hours)")+scale_fill_manual(labels=c("Long
Diapause","Short
Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))
```

Wandering day redo ; boxplots

The days that needed to be compared:

1. UZ -16 hours day 5
2. UZ - 12 hours day 10
3. BE - 16 hours day 3
4. BE - 12 hours day 8

parsed out that data

```
d1<-data%>%
  filter(strain=="BE" & treat=="1" & day==8)

d2<-data%>%
  filter(strain=="BE" & treat=="2" & day==3)

d3<-data%>%
  filter(strain=="UZ" & treat=="1" & day==10)

d4<-data%>%
  filter(strain=="UZ" & treat=="2" & day==5)

dat<-data.frame(rbind(d1,d2,d3,d4))


dat$Photoperiod<-ifelse(dat$treat=="1","12","16")
dat$mass.mg<-dat$mass*1000


dat2<-dat%>%
  filter(Tray!="12,23 1201-01" & Tray!="12,23 1201-02" & Tray!="16,23 1201"&
Tray!="16,23 1202")
```

lets do the stats

```
dat2%>%
  group_by(Photoperiod,strain,Tray)%>%
  dplyr::summarise(n=length(Tray))


  summary(comp)

  Error: Photoperiod
              Df Sum Sq Mean Sq
  Photoperiod  1  682.9   682.9

  Error: Photoperiod:Tray
                    Df Sum Sq Mean Sq F value Pr(>F)
  strain             1  27190   27190  18.497 0.0126 *
  Photoperiod:strain  1    332     332   0.226 0.6595
  Residuals          4   5880    1470
  ---
  Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Error: Within
           Df Sum Sq Mean Sq F value Pr(>F)
Residuals 113  29944     265
```

lets plot out the data in a nice way

```
ggplot(dat2,aes(x=Photoperiod,y=mass.mg,fill=strain))+geom_boxplot()+ylab("Wet Mass (mg)")+xlab("Photoperiod (hours)")+scale_fill_manual(labels=c("Long Diapause","Short Diapause"),values=c("lightsalmon1","mediumpurple1"),breaks=c("UZ","BE"))+T+scale_y_continuous(breaks=seq(0,160,20),labels=seq(0,160,20),limits = c(0,160))
```

Page 29: 2019-07-17. Jbrown stats revisit

Meeting with DHahn:

- Take out lean mass in the statistical models for wet mass and lipid mass.

Day 1

wet mass

```
mixedL1.2=lmer(wet_mass ~ photoperiod*strain + (1|rep/cohort) ,data=dataL1, REML = TRUE)
lmerTest::step(mixedL1.2)
Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                  Eliminated     Sum Sq    Mean Sq NumDF  DenDF F value  Pr(>F)

photoperiod:strain          0 0.00012639 0.00012639     1 63.402  4.7147 0.03365
*
```

lipid mass

```
mixedL1=lmer(lipid_mass ~ photoperiod*strain + (1|rep/cohort) ,data=dataL1, REML
= TRUE,control=lmerControl(check.conv.singular = .makeCC(action = "ignore",  tol
= 1e-4)))
summary(mixedL1)

Fixed effects:
                          Estimate Std. Error         df t value Pr(>|t|)
(Intercept)              0.0021600  0.0078731 71.0000000   0.274    0.785
photoperiod16           -0.0006711  0.0106602 71.0000000  -0.063    0.950
strainUZ                -0.0002191  0.0102102 71.0000000  -0.021    0.983
photoperiod16:strainUZ   0.0135002  0.0142264 71.0000000   0.949    0.346
```

Wandering stage

wet mass

```
mixedLW3=lmer(wet_mass ~ photoperiod * strain +(1|rep/cohort) ,data=dataLW, REML
= TRUE,control=lmerControl(check.conv.singular = .makeCC(action = "ignore",  tol
= 1e-4)))
Fixed effects:
                          Estimate Std. Error         df t value Pr(>|t|)
(Intercept)               0.084779   0.002784 141.000000  30.456  < 2e-16 ***
photoperiod16            -0.021908   0.004272 141.000000  -5.129 9.47e-07 ***
strainUZ                  0.013863   0.004162 141.000000   3.331  0.00111 **
photoperiod16:strainUZ    0.006003   0.006032 141.000000   0.995  0.32132
```

lipid mass

```
mixedLW=lmer(lipid_mass ~ photoperiod * strain+(1|rep/cohort) ,data=dataLW, REML
= TRUE)
lmerTest::step(mixedLW)

Backward reduced fixed-effect table:
Degrees of freedom method: Satterthwaite

                  Eliminated    Sum Sq   Mean Sq NumDF  DenDF  F value
 Pr(>F)
photoperiod:strain          1 1.503e-05 1.503e-05     1 131.40   3.6229
0.0591757 .
photoperiod                 0 6.545e-04 6.545e-04     1 134.61 155.4339 < 2.2e-
16 ***
strain                      0 6.116e-05 6.116e-05     1 133.30  14.5241
0.0002105 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Page 30: 2019-09-02. data revisit with Dan

I'm not a post doc anymore, so I'll be adding entries infrequently, when I work on the projects. Dhahn asked me to revisit the diapause exit paper and split the PCA's of the metabolic rate trajectories by host. I sent him results, but its a similar story.

Page 31:

Page 32:

Page 33:

Page 34:

Page 35:

Page 36:

Page 37:

Page 38:

Page 39:

Page 40:

Page 41:

Page 42:

Page 43:

Page 44:

Page 45:

Page 46:

Page 47:

Page 48:

Page 49:

Page 50: