Frontloading_RMD

Samuel Gurr

9/23/2021

Call modules representing higher expression by naive clams throughout the subseqent exposures

- 'Naive_modules' == all mods of interest

```
day7.brown    <- day7.ModMem %>% dplyr::filter(moduleColor %in% 'brown')     %>% dplyr::mutate(day ='Da
day14.brown   <- day14.ModMem %>% dplyr::filter(moduleColor %in% 'brown')    %>% dplyr::mutate(day ='Da
day21.blue    <- day21.ModMem %>% dplyr::filter(moduleColor %in% 'blue')     %>% dplyr::mutate(day ='Da
day21.magenta <- day21.ModMem %>% dplyr::filter(moduleColor %in% 'magenta')  %>% dplyr::mutate(day ='Da

Naive_modules <- rbind(day7.brown, day14.brown, day21.blue, day21.magenta)
```

- 'NaiveResponse_genes_data' call all genes that occured THREE times in 'Naive_modules'
- Why? These genes are represent those with persistant high expression relative to the pre-exposed (primed) clams!
- there are 315 total genes in this category

```
NaiveResponse_genes <- Naive_modules %>%
                       dplyr::group_by(Pgen_ID) %>%
                       dplyr::summarise(count = n()) %>%
                       dplyr::filter(count == 3)
# View(NaiveResponse_genes)
NaiveResponse_genes        = NaiveResponse_genes$Pgen_ID
NaiveResponse_genes_ANNOT  = match(NaiveResponse_genes, annot$V1)
NaiveResponse_genes_data   <- data.frame(geneSymbol = annot$V1[NaiveResponse_genes_ANNOT],
                                         Annotation = annot$V7[NaiveResponse_genes_ANNOT])
# nrow(NaiveResponse_genes_data) # 315 genes present on all sampling days
```

Calculate the contorl ratio (Y axis) and the foldchangie ratio (x axis)

```
# Day 7 gene expression
Day_WGCNA_genes <- Day7_exp_data %>%
  dplyr::filter(Gene %in% day7.brown$Pgen_ID ) # call all genes in the module
  #dplyr::filter(Gene %in% NaiveResponse_genes)

Day7_WGCNA_genes_melted     <- Day_WGCNA_genes %>%
```

```
  reshape2::melt(id.var = 'Gene') %>%
  dplyr::rename(Sample.Name = variable)
Day7_WGCNA_genes_Merge <- merge(Day7_WGCNA_genes_melted, Day7_Master.Exp.Metadata, by = 'Sample.Name') %
  dplyr::group_by(Gene, All_Treatment) %>%
  dplyr::select(!'Sample.Name') %>%
  dplyr::summarise(meanExp = mean(value))
```

```
## `summarise()` has grouped output by 'Gene'. You can override using the `.groups` argument.
```

```
Day7_READY <- dcast(Day7_WGCNA_genes_Merge, Gene ~ All_Treatment)
```

```
## Using meanExp as value column: use value.var to override.
```

```
for (i in 1:nrow(Day7_READY)) {
          # Moderate - higher expression AM > AA
        if (Day7_READY$AM[i] > Day7_READY$AA[i]) {
          Day7_READY$wgcna.xall_mod[i] <- ( (Day7_READY$MM[i] / Day7_READY$MA[i]) / (Day7_READY$AM[i] /
          Day7_READY$wgcna.yall_mod[i] <- (Day7_READY$MA[i] / Day7_READY$AA[i]) # Y Axis - this is simp
              } else {
                Day7_READY$wgcna.xall_mod[i] <- NA  # X axis  - call NA
                Day7_READY$wgcna.yall_mod[i] <- NA # Y Axis  - call NA
              }
            # Severe - higher expression AS > AA
          if (Day7_READY$AS[i] > Day7_READY$AA[i]) {
            Day7_READY$wgcna.xall_sev[i] <- ( (Day7_READY$MS[i] / Day7_READY$MA[i]) / (Day7_READY$AS[i]
            Day7_READY$wgcna.yall_sev[i] <- (Day7_READY$MA[i] / Day7_READY$AA[i])            # Y Axis -
                } else {
                  Day7_READY$wgcna.xall_sev[i] <- NA  # X axis  - call NA
                  Day7_READY$wgcna.yall_sev[i] <- NA # Y Axis  - call NA
                }
}
# Day7_READY
x = data.frame(Gene = annot$V1[(match(Day7_READY$Gene, annot$V1))],
               Gene_Description = annot$V7[(match(Day7_READY$Gene, annot$V1))])
Day7_Frontload   <- merge(x, Day7_READY, by = 'Gene')
Day7_Frontload_2 <- Day7_Frontload %>%
                      dplyr::mutate(baseMeanNAIVE_control = AA) %>%
                      dplyr::mutate(baseMeanPRIMED_control = MA) %>%
                      dplyr::mutate(baseMeanNAIVE_foldChangeModerate = ((AM) / (AA))  ) %>%
                      dplyr::mutate(baseMeanPRIMED_foldChangeModerate = ((MM) / (MA)) ) %>%
                      dplyr::mutate(baseMeanNAIVE_foldChangeSevere = ((AS) / (AA))  ) %>%
                      dplyr::mutate(baseMeanPRIMED_foldChangeSevere = ((MS) / (MA)) ) %>%
                      dplyr::rename(ControlRatioModerate = wgcna.yall_mod) %>%
                      dplyr::rename(foldChangeRatioModerate = wgcna.xall_mod) %>%
                      dplyr::rename(ControlRatioSevere = wgcna.yall_sev) %>%
                      dplyr::rename(foldChangeRatioSevere = wgcna.xall_sev)
write.csv(Day7_Frontload_2,paste("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabol
```

```
D7Frontloaded <- Day7_Frontload_2 %>% filter(ControlRatioModerate > 1) %>% filter(foldChangeRatioModera
for (i in 1:nrow(D7Frontloaded)) {
 x <-  D7Frontloaded[i,]
```

```
  x2 <- x  %>%
    dplyr::select(!c("foldChangeRatioModerate","ControlRatioModerate","foldChangeRatioSevere","ControlRa
    gather(variable, value, -c(Gene,Gene_Description)) %>%
    dplyr::filter(variable %in% c('AA','AM','MA','MM')) %>%
    dplyr::mutate(Primary_Treatment = substr(variable, 1,1)) %>%
    dplyr::mutate(Second_Treatment = substr(variable, 2,2)) %>%
    # dplyr::group_by(Primary_Treatment, Third_Treatment) %>%
    # dplyr::summarise(meanEXP = mean(value), sdExp = sd(value)) %>%
      ggplot(aes(x=Second_Treatment , y=value, fill=Primary_Treatment)) +  # , colour=supp, group=supp),
        theme_classic() +
        #geom_errorbar(aes(ymin=meanEXP-sdExp, ymax=meanEXP+sdExp), colour="black", width=.1, position=p
        # geom_point(position=pd, size = 4, shape=21) +
        geom_bar(position=position_dodge(), aes(y=value), stat="identity", width=0.5) +
        xlab("Second pCO2 treatment") +
        ylab('Gene Expression (averaged raw data)') +                  # note the mean was first by samp
        scale_fill_manual(values=c("grey85","grey50")) +
        #scale_color_manual(values=c("#56B4E9","#E69F00")) +
        # ggtitle(paste("Day 7:",apriori_DESeq2_condenced[i,1],":", apriori_DESeq2_condenced[i,3],sep='
        ggtitle(paste(  D7Frontloaded[i,1], gsub(" .*", "\\1", D7Frontloaded[i,2]), sep = '_') ) +
        # expand_limits(y=0) +                                                  # Expand y range
        # scale_y_continuous(limits=c((min_p1), (max_p1))) +
        # scale_y_continuous(limits = c((max(sdExp)+0.5),(min(sdExp)-0.5)))  +
        # ylim(ExpMin, ExpMin) +
        theme(axis.text.x = element_text(size = 20),
              axis.text.y = element_text(size = 20),
              axis.ticks.length=unit(.25, "cm"))+
        theme(legend.position = "none")

  gene <- D7Frontloaded[i,1]
  title <- gsub(" .*", "\\1", (sub(" ", "_",  D7Frontloaded[i,2])) )

pdf(paste0("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabolomics/TagSeq/Analysis/
print(x2)
dev.off()

  }









# Day 14 gene expression
Day_WGCNA_genes <- Day14_exp_data %>%
  dplyr::filter(Gene %in% day14.brown$Pgen_ID ) # call all genes in the module
```

```
  # dplyr::filter(Gene %in% NaiveResponse_genes)
Day14_WGCNA_genes_melted     <- Day_WGCNA_genes %>%
  reshape2::melt(id.var = 'Gene') %>%
  dplyr::rename(Sample.Name = variable)
Day14_WGCNA_genes_Merge <- merge(Day14_WGCNA_genes_melted, Day14_Master.Exp.Metadata, by = 'Sample.Name
  dplyr::group_by(Gene, All_Treatment) %>%
  dplyr::select(!'Sample.Name') %>%
  dplyr::summarise(meanExp = mean(value))
```

```
## `summarise()` has grouped output by 'Gene'. You can override using the `.groups` argument.
```

```
Day14_READY <- dcast(Day14_WGCNA_genes_Merge, Gene ~ All_Treatment)
```

```
## Using meanExp as value column: use value.var to override.
```

```
for (i in 1:nrow(Day14_READY)) {
          # Moderate - higher expression AM > AA
        if (Day14_READY$AM[i] > Day14_READY$AA[i]) {
          Day14_READY$wgcna.xall_mod[i] <- ( (Day14_READY$MM[i] / Day14_READY$MA[i]) / (Day14_READY$AM[i
          Day14_READY$wgcna.yall_mod[i] <- (Day14_READY$MA[i] / Day14_READY$AA[i]) # Y Axis - this is s
              } else {
                Day14_READY$wgcna.xall_mod[i] <- NA  # X axis  - call NA
                Day14_READY$wgcna.yall_mod[i] <- NA # Y Axis  - call NA
              }
            # Severe - higher expression AS > AA
          if (Day14_READY$AS[i] > Day14_READY$AA[i]) {
            Day14_READY$wgcna.xall_sev[i] <- ( (Day14_READY$MS[i] / Day14_READY$MA[i]) / (Day14_READY$AS
            Day14_READY$wgcna.yall_sev[i] <- (Day14_READY$MA[i] / Day14_READY$AA[i])            # Y Axi
                } else {
                  Day14_READY$wgcna.xall_sev[i] <- NA  # X axis  - call NA
                  Day14_READY$wgcna.yall_sev[i] <- NA # Y Axis  - call NA
                }
}
# Day14_READY
x = data.frame(Gene = annot$V1[(match(Day14_READY$Gene, annot$V1))],
               Gene_Description = annot$V7[(match(Day14_READY$Gene, annot$V1))])
Day14_Frontload   <- merge(x, Day14_READY, by = 'Gene')
Day14_Frontload_2 <- Day14_Frontload %>%
                     dplyr::mutate(baseMeanNAIVE_control = AA) %>%
                     dplyr::mutate(baseMeanPRIMED_control = MA) %>%
                     dplyr::mutate(baseMeanNAIVE_foldChangeModerate = ((AM) / (AA))  ) %>%
                     dplyr::mutate(baseMeanPRIMED_foldChangeModerate = ((MM) / (MA)) ) %>%
                     dplyr::mutate(baseMeanNAIVE_foldChangeSevere = ((AS) / (AA))  ) %>%
                     dplyr::mutate(baseMeanPRIMED_foldChangeSevere = ((MS) / (MA)) ) %>%
                     dplyr::rename(ControlRatioModerate = wgcna.yall_mod) %>%
                     dplyr::rename(foldChangeRatioModerate = wgcna.xall_mod) %>%
                     dplyr::rename(ControlRatioSevere = wgcna.yall_sev) %>%
                     dplyr::rename(foldChangeRatioSevere = wgcna.xall_sev)
write.csv(Day14_Frontload_2,paste("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabol
```

```
D14Frontloaded <- Day14_Frontload_2 %>% filter(ControlRatioModerate > 1) %>% filter(foldChangeRatioModer
```

```
for (i in 1:nrow(D14Frontloaded)) {
 x <-  D14Frontloaded[i,]
  x2 <- x  %>%
    dplyr::select(!c("foldChangeRatioModerate","ControlRatioModerate","foldChangeRatioSevere","ControlRa
    gather(variable, value, -c(Gene,Gene_Description)) %>%
    dplyr::filter(variable %in% c('AA','AM','MA','MM')) %>%
    dplyr::mutate(Primary_Treatment = substr(variable, 1,1)) %>%
    dplyr::mutate(Second_Treatment = substr(variable, 2,2)) %>%
    # dplyr::group_by(Primary_Treatment, Third_Treatment) %>%
    # dplyr::summarise(meanEXP = mean(value), sdExp = sd(value)) %>%
      ggplot(aes(x=Second_Treatment , y=value, fill=Primary_Treatment)) +  # , colour=supp, group=supp),
        theme_classic() +
        #geom_errorbar(aes(ymin=meanEXP-sdExp, ymax=meanEXP+sdExp), colour="black", width=.1, position=
        # geom_point(position=pd, size = 4, shape=21) +
        geom_bar(position=position_dodge(), aes(y=value), stat="identity", width=0.5) +
        xlab("Second pCO2 treatment") +
        ylab('Gene Expression (averaged raw data)') +                 # note the mean was first by samp
        scale_fill_manual(values=c("grey85","grey50")) +
        #scale_color_manual(values=c("#56B4E9","#E69F00")) +
        # ggtitle(paste("Day 7:",apriori_DESeq2_condenced[i,1],":", apriori_DESeq2_condenced[i,3],sep='
        ggtitle(paste(  D14Frontloaded[i,1], gsub(" .*", "\\1", D14Frontloaded[i,2]), sep = '_') ) +
        # expand_limits(y=0) +                                                    # Expand y range
        # scale_y_continuous(limits=c((min_p1), (max_p1))) +
        # scale_y_continuous(limits = c((max(sdExp)+0.5),(min(sdExp)-0.5)))  +
        # ylim(ExpMin, ExpMin) +
        theme(axis.text.x = element_text(size = 20),
              axis.text.y = element_text(size = 20),
              axis.ticks.length=unit(.25, "cm"))+
        theme(legend.position = "none")

  gene <- D14Frontloaded[i,1]
  title <- gsub(" .*", "\\1", (sub(" ", "_",  D14Frontloaded[i,2])) )

pdf(paste0("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabolomics/TagSeq/Analysis/
print(x2)
dev.off()

  }









# Day 21 gene expression
day21.blue_magenta <- rbind(day21.blue, day21.magenta)
Day_WGCNA_genes <- Day21_exp_data %>%
  dplyr::filter(Gene %in% day21.blue_magenta$Pgen_ID) # call all genes in the module
```

```
  #dplyr::filter(Gene %in% NaiveResponse_genes)
Day21_WGCNA_genes_melted     <- Day_WGCNA_genes %>%
  reshape2::melt(id.var = 'Gene') %>%
  dplyr::rename(Sample.Name = variable)
Day21_WGCNA_genes_Merge <- merge(Day21_WGCNA_genes_melted, Day21_Master.Exp.Metadata, by = 'Sample.Name
  dplyr::group_by(Gene, All_Treatment) %>%
  dplyr::select(!'Sample.Name') %>%
  dplyr::summarise(meanExp = mean(value))
```

```
## `summarise()` has grouped output by 'Gene'. You can override using the `.groups` argument.
```

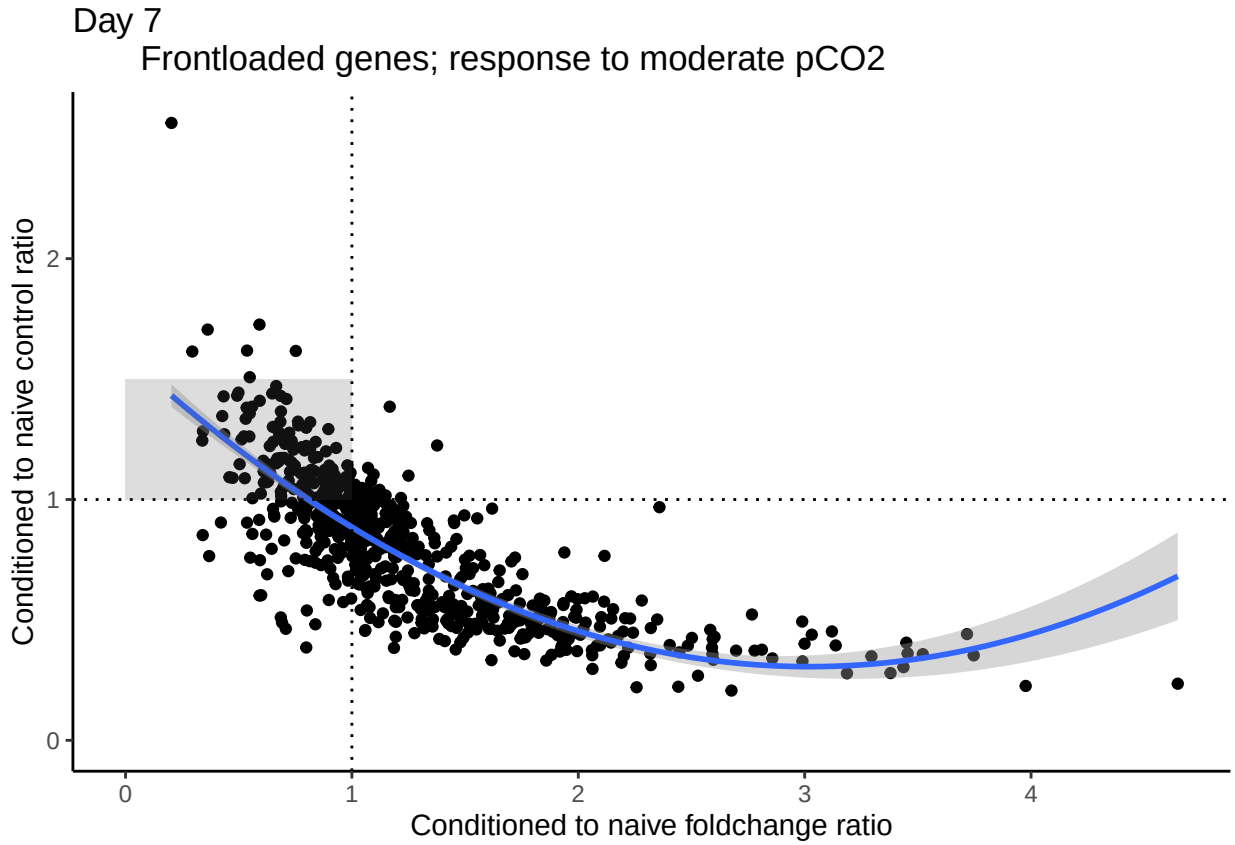
```
Day21_READY <- dcast(Day21_WGCNA_genes_Merge, Gene ~ All_Treatment)
```

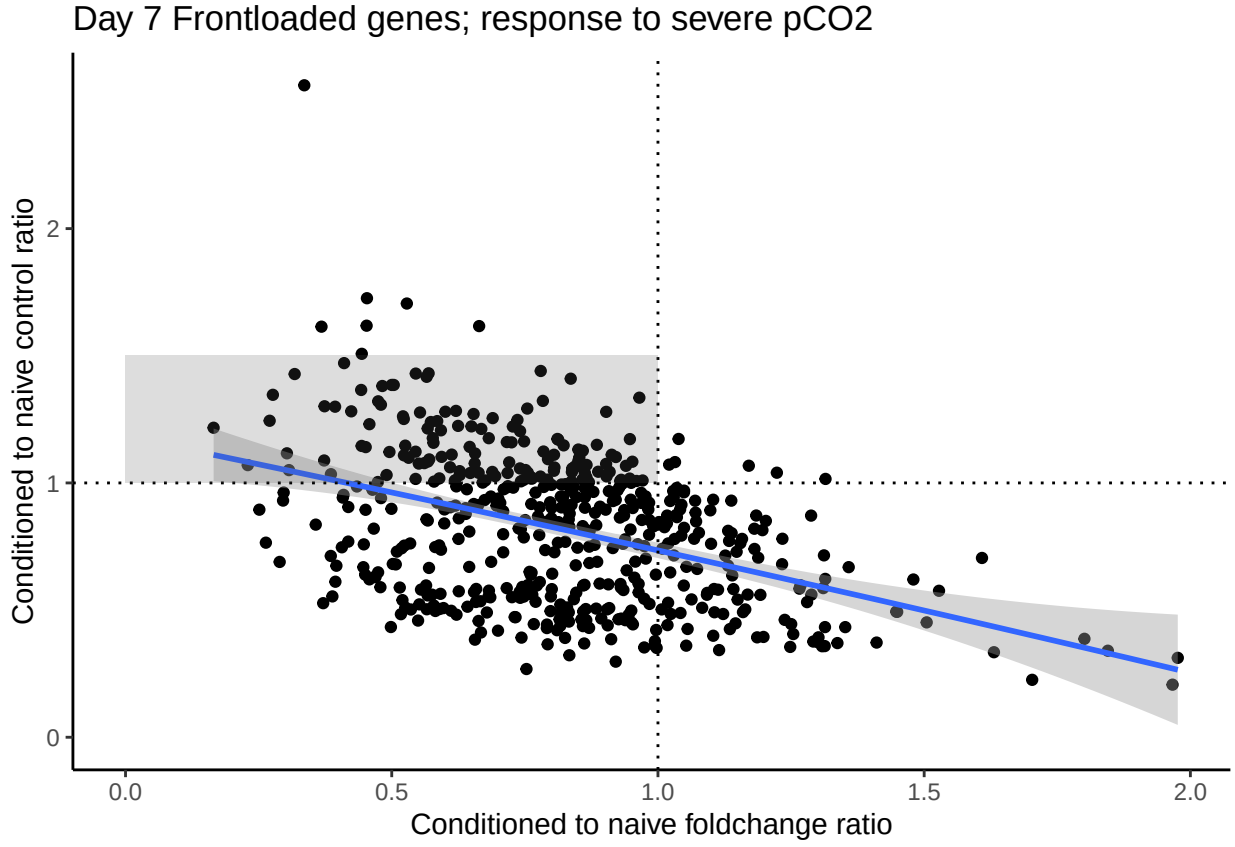
```
## Using meanExp as value column: use value.var to override.
```

```
for (i in 1:nrow(Day21_READY)) {
          # Moderate - higher expression AM > AA
        if (Day21_READY$AAM[i] > Day21_READY$AAA[i]) {
          Day21_READY$wgcna.xall_mod[i] <- ( (Day21_READY$MAM[i] / Day21_READY$MAA[i]) / (Day21_READY$AA
          Day21_READY$wgcna.yall_mod[i] <- (Day21_READY$MAA[i] / Day21_READY$AAA[i]) # Y Axis - this is
              } else {
                Day21_READY$wgcna.xall_mod[i] <- NA  # X axis  - call NA
                Day21_READY$wgcna.yall_mod[i] <- NA # Y Axis  - call NA
              }
}
# Day21_READY


x = data.frame(Gene = annot$V1[(match(Day21_READY$Gene, annot$V1))],
               Gene_Description = annot$V7[(match(Day21_READY$Gene, annot$V1))])
Day21_Frontload   <- merge(x, Day21_READY, by = 'Gene')
Day21_Frontload_2 <- Day21_Frontload %>%
                     dplyr::mutate(baseMeanNAIVE_control = AAA) %>%
                     dplyr::mutate(baseMeanPRIMED_control = MAA) %>%
                     dplyr::mutate(baseMeanNAIVE_foldChangeModerate = ((AAM) / (AAA))  ) %>%
                     dplyr::mutate(baseMeanPRIMED_foldChangeModerate = ((MAM) / (MAA)) ) %>%
                     dplyr::rename(ControlRatio = wgcna.yall_mod) %>%
                     dplyr::rename(foldChangeRatio = wgcna.xall_mod)
write.csv(Day21_Frontload_2,paste("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabol


library(tidyverse)
D21Frontloaded <- Day21_Frontload_2 %>% filter(ControlRatio > 1) %>% filter(foldChangeRatio < 1)
for (i in 1:nrow(D21Frontloaded)) {
 x <-  D21Frontloaded[i,]
  x2 <- x  %>%
    dplyr::select(!c('foldChangeRatio','ControlRatio','baseMeanNAIVE_control','baseMeanPRIMED_control',
    gather(variable, value, -c(Gene,Gene_Description)) %>%
    dplyr::filter(variable %in% c('AAA','AAM','MAA','MAM')) %>%
    dplyr::mutate(Primary_Treatment = substr(variable, 1,1)) %>%
    dplyr::mutate(Second_Treatment = substr(variable, 2,2)) %>%
    dplyr::mutate(Third_Treatment = substr(variable, 3,3)) %>%
```

```
    # dplyr::group_by(Primary_Treatment, Third_Treatment) %>%
    # dplyr::summarise(meanEXP = mean(value), sdExp = sd(value)) %>%
      ggplot(aes(x=Third_Treatment , y=value, fill=Primary_Treatment)) +  # , colour=supp, group=supp))
        theme_classic() +
        #geom_errorbar(aes(ymin=meanEXP-sdExp, ymax=meanEXP+sdExp), colour="black", width=.1, position=
        # geom_point(position=pd, size = 4, shape=21) +
        geom_bar(position=position_dodge(), aes(y=value), stat="identity", width=0.5) +
        xlab("Third pCO2 treatment") +
        ylab('Gene Expression (averaged raw data)') +                  # note the mean was first by samp
        scale_fill_manual(values=c("grey85","grey50")) +
        #scale_color_manual(values=c("#56B4E9","#E69F00")) +
        # ggtitle(paste("Day 7:",apriori_DESeq2_condenced[i,1],":", apriori_DESeq2_condenced[i,3],sep='
        ggtitle(paste(  D21Frontloaded[i,1], gsub(" .*", "\\1", D21Frontloaded[i,2]), sep = '_') ) +
        # expand_limits(y=0) +                                                  # Expand y range
        # scale_y_continuous(limits=c((min_p1), (max_p1))) +
        # scale_y_continuous(limits = c((max(sdExp)+0.5),(min(sdExp)-0.5)))  +
        # ylim(ExpMin, ExpMin) +
        theme(axis.text.x = element_text(size = 20),
              axis.text.y = element_text(size = 20),
              axis.ticks.length=unit(.25, "cm"))+
        theme(legend.position = "none")

  gene <- D21Frontloaded[i,1]
  title <- gsub(" .*", "\\1", (sub(" ", "_",  D21Frontloaded[i,2])) )

pdf(paste0("C:/Users/samjg/Documents/Github_repositories/Pgenerosa_TagSeq_Metabolomics/TagSeq/Analysis/
print(x2)
dev.off()

  }
```

Day 7 - Frontloaded genes (Plot)





Day 7 - Frontloaded genes (Table)

```
# MODERATE second treatment
Day7.wgcna.frontloaded_genes_mod <- Day7_READY %>%
                                dplyr::filter(wgcna.xall_mod < 1) %>%
                                dplyr::filter(wgcna.yall_mod > 1) %>%
                                dplyr::select('Gene')

Day7.wgcna.frontloadprobes_mod        = Day7.wgcna.frontloaded_genes_mod$Gene
probes2annot_mod                      = match(Day7.wgcna.frontloadprobes_mod, annot$V1)
Day7.wgcna.frontloaded_mod_ANNOT      = data.frame(geneSymbol = annot$V1[probes2annot_mod],
                                                   Genes = annot$V7[probes2annot_mod])
kable(head(Day7.wgcna.frontloaded_mod_ANNOT))
```

geneSymbol	Genes
PGEN_.00g006150	GPI mannosyltransferase 3 (EC 2.4.1.-) (GPI mannosyltransferase III) (GPI-MT-III) (Phosphatidyli
PGEN_.00g015380	Sarcoplasmic reticulum histidine-rich calcium-binding protein
PGEN_.00g019140	NA
PGEN_.00g019670	Dr1-associated corepressor (Dr1-associated protein 1) (Negative cofactor 2-alpha) (NC2-alpha)
PGEN_.00g021170	Kinesin light chain (KLC)
PGEN_.00g021940	Metastasis-associated protein MTA1

```
perc_mod7_mod <- 100 -( ( ( (nrow(day7.brown)) - (nrow(Day7.wgcna.frontloaded_mod_ANNOT)) ) / (nrow(day
print(paste("Total frontloaded genes = ", nrow(Day7.wgcna.frontloaded_mod_ANNOT), "; Percent of naive m
```

```
## [1] "Total frontloaded genes = 140; Percent of naive module = 16.2412993039443"
```

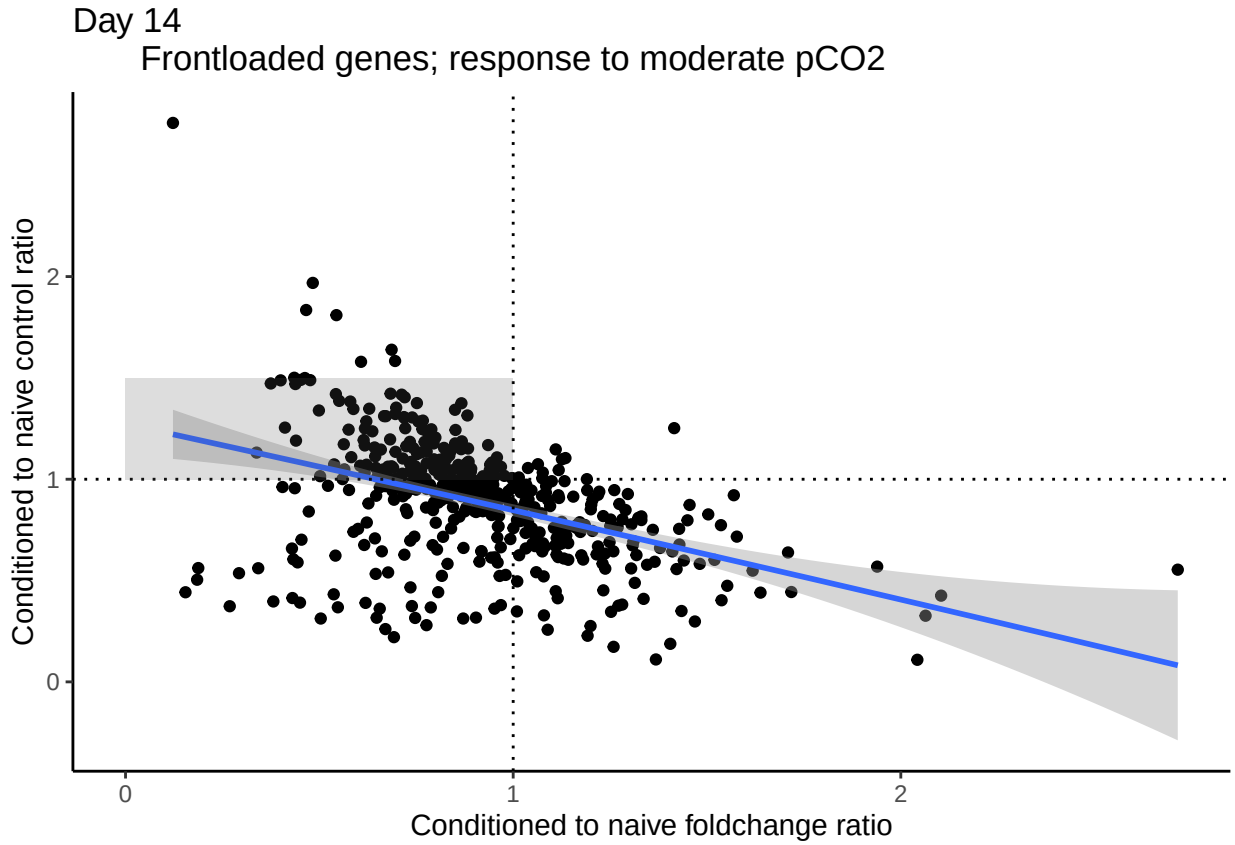
```
# SEVERE second treatment
Day7.wgcna.frontloaded_genes_sev <- Day7_READY %>%
                                dplyr::filter(wgcna.xall_sev < 1) %>%
                                dplyr::filter(wgcna.yall_sev > 1) %>%
                                dplyr::select('Gene')

Day7.wgcna.frontloadprobes_sev        = Day7.wgcna.frontloaded_genes_sev$Gene
probes2annot_sev                      = match(Day7.wgcna.frontloadprobes_sev, annot$V1)
Day7.wgcna.frontloaded_sev_ANNOT      = data.frame(geneSymbol = annot$V1[probes2annot_sev],
                                                   Genes = annot$V7[probes2annot_sev])

kable(head(Day7.wgcna.frontloaded_sev_ANNOT))
```

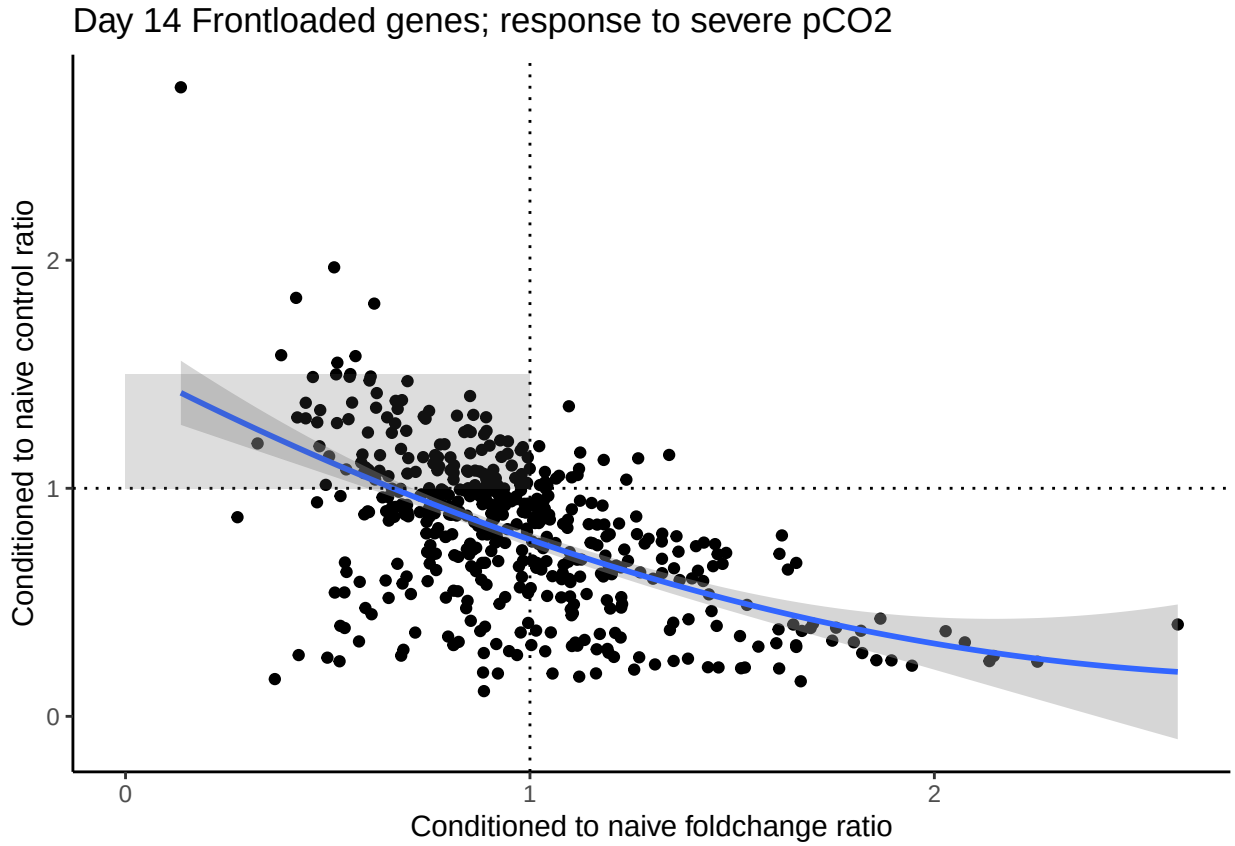
geneSymbol	Genes
PGEN_.00g002440	Cytochrome P450 10 (EC 1.14.-.-) (CYPX)
PGEN_.00g006150	GPI mannosyltransferase 3 (EC 2.4.1.-) (GPI mannosyltransferase III) (GPI-MT-III) (Phosphatidyli
PGEN_.00g019140	NA
PGEN_.00g021170	Kinesin light chain (KLC)
PGEN_.00g022390	G patch domain-containing protein 2
PGEN_.00g026480	Ras-related protein Rab-32B

```
perc_mod7_sev <- 100 -( ( ( (nrow(day7.brown)) - (nrow(Day7.wgcna.frontloaded_sev_ANNOT)) ) / (nrow(day
print(paste("Total frontloaded genes = ", nrow(Day7.wgcna.frontloaded_sev_ANNOT), "; Percent of naive m
```

```
## [1] "Total frontloaded genes = 141; Percent of naive module = 16.3573085846868"
```

Day 14 - Frontloaded genes (Plot)





Day 14 - Frontloaded genes (Table)

```
# MODERATE second treatment
Day14.wgcna.frontloaded_genes_mod <- Day14_READY %>%
                                dplyr::filter(wgcna.xall_mod < 1) %>%
                                dplyr::filter(wgcna.yall_mod > 1) %>%
                                dplyr::select('Gene') # %>%

Day14.wgcna.frontloadprobes_mod        = Day14.wgcna.frontloaded_genes_mod$Gene
probes2annot_mod                       = match(Day14.wgcna.frontloadprobes_mod, annot$V1)
Day14.wgcna.frontloaded_mod_ANNOT      = data.frame(geneSymbol = annot$V1[probes2annot_mod],
                                         Genes = annot$V7[probes2annot_mod])
kable(head(Day14.wgcna.frontloaded_mod_ANNOT))
```

geneSymbol	Genes
PGEN_.00g012030	NA
PGEN_.00g012200	Cytochrome P450 1A1 (CYP1A1) (EC 1.14.14.1) (CYPIA1) (Cytochrome P450 form 6) (Cytochrom
PGEN_.00g014140	Tectonin beta-propeller repeat-containing protein 2 (WD repeat-containing protein KIAA0329/KIA
PGEN_.00g014830	Bifunctional protein GlmU [Includes: UDP-N-acetylglucosamine pyrophosphorylase (EC 2.7.7.23) (N
PGEN_.00g016340	Peroxisome assembly factor 2 (PAF-2) (Peroxin-6) (Peroxisomal biogenesis factor 6) (Peroxisomal-ty
PGEN_.00g019210	Proteasome subunit alpha type-6 (EC 3.4.25.1) (Macropain iota chain) (Multicatalytic endopeptidas

```
perc_mod14_mod <- 100 -( ( ( (nrow(day14.brown)) - (nrow(Day14.wgcna.frontloaded_mod_ANNOT)) ) / (nrow(
print(paste("Total frontloaded genes = ", nrow(Day14.wgcna.frontloaded_mod_ANNOT), "; Percent of naive m
```

```
## [1] "Total frontloaded genes = 140; Percent of naive module = 12.0274914089347"
```

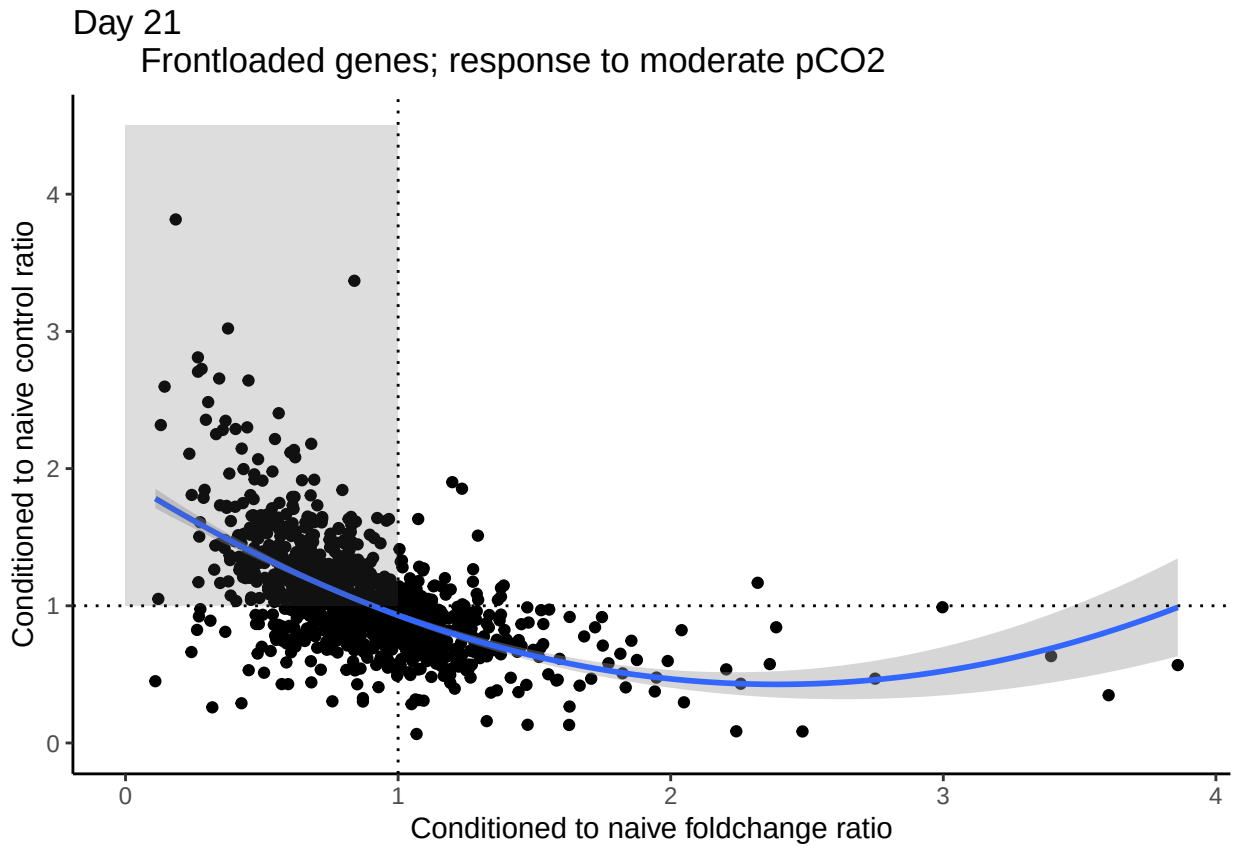
```
# SEVERE second treatment
Day14.wgcna.frontloaded_genes_sev <- Day14_READY %>%
                                dplyr::filter(wgcna.xall_sev < 1) %>%
                                dplyr::filter(wgcna.yall_sev > 1) %>%
                                dplyr::select('Gene') # %>%

Day14.wgcna.frontloadprobes_sev        = Day14.wgcna.frontloaded_genes_sev$Gene
probes2annot_sev                       = match(Day14.wgcna.frontloadprobes_sev, annot$V1)
Day14.wgcna.frontloaded_sev_ANNOT      = data.frame(geneSymbol = annot$V1[probes2annot_sev],
                                               Genes = annot$V7[probes2annot_sev])
kable(head(Day14.wgcna.frontloaded_sev_ANNOT))
```

geneSymbol	Genes
PGEN_.00g012030	NA
PGEN_.00g012200	Cytochrome P450 1A1 (CYP1A1) (EC 1.14.14.1) (CYPIA1) (Cytochrome P450 form 6) (Cytochrom
PGEN_.00g014140	Tectonin beta-propeller repeat-containing protein 2 (WD repeat-containing protein KIAA0329/KIA
PGEN_.00g014830	Bifunctional protein GlmU [Includes: UDP-N-acetylglucosamine pyrophosphorylase (EC 2.7.7.23) (N
PGEN_.00g016340	Peroxisome assembly factor 2 (PAF-2) (Peroxin-6) (Peroxisomal biogenesis factor 6) (Peroxisomal-ty
PGEN_.00g019210	Proteasome subunit alpha type-6 (EC 3.4.25.1) (Macropain iota chain) (Multicatalytic endopeptidas

```
perc_mod14_sev <- 100 -( ( ( (nrow(day14.brown)) - (nrow(Day14.wgcna.frontloaded_sev_ANNOT)) ) / (nrow(
print(paste("Total frontloaded genes = ", nrow(Day14.wgcna.frontloaded_sev_ANNOT), "; Percent of naive m
```

```
## [1] "Total frontloaded genes = 113; Percent of naive module = 9.70790378006873"
```

Day 21 - Frontloaded genes (Plot)



Day 21 - Frontloaded genes (Table)

```
# MODERATE second treatment
Day21.wgcna.frontloaded_genes_mod <- Day21_READY %>%
                                dplyr::filter(wgcna.xall_mod < 1) %>%
                                dplyr::filter(wgcna.yall_mod > 1) %>%
                                dplyr::select('Gene') # %>%

Day21.wgcna.frontloadprobes_mod        = Day21.wgcna.frontloaded_genes_mod$Gene
probes2annot_mod                       = match(Day21.wgcna.frontloadprobes_mod, annot$V1)
Day21.wgcna.frontloaded_mod_ANNOT      = data.frame(geneSymbol = annot$V1[probes2annot_mod],
                                         Genes = annot$V7[probes2annot_mod])
kable(head(Day21.wgcna.frontloaded_mod_ANNOT))
```

geneSymbol	Genes
PGEN_.00g000280	Uncharacterized protein PAE1111
PGEN_.00g000320	NA
PGEN_.00g000970	ATP-binding cassette sub-family D member 3 (68 kDa peroxisomal membrane protein) (PMP68) (70
PGEN_.00g002130	Dedicator of cytokinesis protein 7
PGEN_.00g002370	NA
PGEN_.00g002400	E3 ubiquitin-protein ligase HECTD3 (EC 2.3.2.26) (HECT domain-containing protein 3) (HECT-typ

```
perc_mod21 <- ( ( ( (nrow(day21.blue_magenta)) - (nrow(Day21.wgcna.frontloaded_mod_ANNOT)) ) / (nrow(da
print(paste("Total frontloaded genes = ", nrow(Day21.wgcna.frontloaded_mod_ANNOT), "; Percent of naive
```

```
## [1] "Total frontloaded genes = 452; Percent of naive module(s) = 74.578177727784"
```

```
View(rbind(Day7.wgcna.frontloaded_mod_ANNOT, Day14.wgcna.frontloaded_mod_ANNOT, Day21.wgcna.frontloaded
  dplyr::count(geneSymbol))
```