Heat Maps After Quantile Normalization

Abigail Seeger

November 23, 2020

Contents

Load Packages

```
library(dplyr)
library(tidyverse)
library(ggplot2)
### Lemon is used in ggplot2 -
### facet_rep_grid modification
library(lemon)
library(data.table)
library(ggthemes)
library(extrafont)
### Routliers is used for outliersmad to
### find outliers
library(Routliers)
library(stringi)
library(wesanderson)
library(viridis)
library(reshape2)
library(sm)
library(lme4)
library(lmerTest)
library(lsmeans)
library(car)
library(ggcorrplot)
library(preprocessCore)
library(grid)
```

Create Functions

```
remove_outliers <- function(df) {
    out <- df %>% group_by(genotype, measurement,
        time_point) %>% mutate(measured_value = replace(measured_value,
        outliers_mad(measured_value, b = 1.4826,
            threshold = 3.5, na.rm = TRUE)$outliers_pos,
        NA)) %>% arrange(genotype, measurement,
        time_point)
    return(out)
}
```

```
add_number <- function(data_frame) {
    ### First, if the genotype is Col0 (only
    ### genotype with length 4), assign 0 as
    ### number Else, assign number as genotype
    ### with 'mpk' removed Example: mpk1 will
    ### be 1, mpk1-17 will be 1-17
    data_frame <- data_frame %>% mutate(number = ifelse(genotype !=
        "Col0", (stri_sub(genotype, 4, length(genotype))),
        0))
    ### Next, for all double mutants, replace
    ### '-' with '0' Example: 1-17 becomes 1017
    data_frame$number <- as.numeric(gsub("-",
        "0", data_frame$number))
```

```
    ### Almost there! There's a problem with
    ### two single digit double mutants We need
    ### a four digit number to sort correctly
    ### Example: mpk1-3 -> 1-3 -> 103, but we
    ### need it to be 1003 to sort correctly
    data_frame$number[data_frame$number ==
        "103"] <- "1003"
    data_frame$number[data_frame$number ==
        "506"] <- "5006"
    data_frame$number[data_frame$number ==
        "608"] <- "6008"
    data_frame$number[data_frame$number ==
        "609"] <- "6009"
    ### Convert number to a numberic in order
    ### to sort
    data_frame$number <- as.numeric(data_frame$number)
    data_frame <- data_frame %>% arrange(number)
    data_frame <- data_frame %>% mutate(number_2 = number)
    data_frame$number_2[nchar(data_frame$number_2) ==
        4] <- 0
    data_frame$number_2[nchar(data_frame$number_2) ==
        5] <- 0
    return(data_frame)
}
```

Note that this function has been modified to use the normalized_value instead of the measured_value!!

```
cell_371_data <- function(data_frame) {

    npq_phi2 <- data_frame %>% filter(measurement %in%
        c("npq", "phi2")) %>% group_by(time_point,
        measurement) %>% mutate_each(funs(./median(.[genotype ==
        "Col0"])), normalized_value) %>%
        group_by(time_point, measurement,
            genotype) %>% mutate(log2_fold = log2(median(normalized_value)))


    start_end <- unique((data_frame %>% group_by(day) %>%
        filter(time_point %in% c(min(time_point),
            max(time_point))))$time_point)

    leaf_area <- data_frame %>% filter(measurement ==
        "leafarea") %>% filter(time_point %in%
        start_end) %>% group_by(time_point,
        measurement) %>% mutate_each(funs(./median(.[genotype ==
        "Col0"])), measured_value) %>% group_by(time_point,
        measurement, genotype) %>% mutate(log2_fold = log2(median(measured_value)))

    out <- rbind(npq_phi2, leaf_area) %>%
        group_by(genotype, time_point, measurement)

    return(as.data.frame(out))
```

```
}
```

```
# Function to add vector as column
addToDF <- function(df, v) {
    nRow <- nrow(df)
    lngth <- length(v)
    if (nRow > lngth) {
        length(v) <- nRow
    } else if (nRow < lngth) {
        df[(nRow + 1):lngth, ] <- NA
    }
    cbind(df, v)
}
```

Load Data

```
### Read in the data
depi_jan <- read.table("C:/Users/Owner/Documents/Research/Shiu_Lab/Shiu_Lab_R/Data/Correct_January_Crea
    sep = ",", header = TRUE, stringsAsFactors = FALSE)
depi_dec_feb <- read.table("C:/Users/Owner/Documents/Research/Shiu_Lab/Shiu_Lab_R/Data/DEPI_analysis_Se
    sep = ",", header = FALSE)
depi_jan_growth <- read.table("C:/Users/Owner/Documents/Research/Shiu_Lab/Shiu_Lab_R/Data/Jan_Growth_Da
    sep = ",", header = TRUE, stringsAsFactors = FALSE)
### Add column names
names(depi_jan) <- c("measurement_ID", "plant_ID",
    "DEPI_ID", "time_point", "measured_value",
    "light_regimen", "measurement", "individual_plant_metadata",
    "genotype", "line", "subline", "full_subline_information",
    "experiment_number", "flat_number", "cell_number",
    "row_number", "column_number", "border",
    "treatment")
names(depi_dec_feb) <- c("individual_plant_metadata",
    "genotype", "line", "subline", "border",
    "flat_number", "measurement_ID", "plant_ID",
    "measurement", "time_point", "measured_value")
names(depi_jan_growth) <- c("individual_plant_metadata",
    "genotype", "line", "subline", "full_subline_information",
    "experiment_number", "flat_number", "cell_number",
    "row_number", "column_number", "border",
    "treatment", "measurement_ID", "plant_ID",
    "DEPI_ID", "time_point", "measured_value",
    "light_regimen", "measurement")

depi_jan <- rbind(depi_jan_growth, depi_jan)

indiv_plant_metadata <- read.table("C:/Users/Owner/Documents/Research/Shiu_Lab/Shiu_Lab_R/Data/Individu
    sep = ",", header = FALSE, stringsAsFactors = FALSE)


### Rename columns
indiv_plant_metadata <- indiv_plant_metadata %>%
```

```
    select(V1, V5) %>% rename(plant_ID = V1,
    full_subline_information = V5)
### Merge the individual plant metadata
### with the depi_dec_feb data
depi_dec_feb <- merge(depi_dec_feb, indiv_plant_metadata,
    by = c("plant_ID"))

### For each month: Select only the columns
### that we need and add a column with the
### month
dec_data <- depi_dec_feb %>% filter(!genotype %in%
    c("b1", "b3", "b1b3", "ftsz2-1", "ftsz2-2",
        "ftsz-dbl", "Col0") | (genotype ==
    "Col0" & full_subline_information %in%
    c("Col1-1", "Col1-3", "Col1-4", "Col1-2"))) %>%
    filter(substr(plant_ID, 1, 4) == "1217") %>%
    filter(border == FALSE) %>% select(individual_plant_metadata,
    genotype, flat_number, measurement, time_point,
    measured_value, border, subline, full_subline_information) %>%
    mutate(month = "Dec")
jan_data <- depi_jan %>% filter(!genotype %in%
    c("b1", "b3", "b1b3", "ftsz2-1", "ftsz2-2",
        "ftsz-dbl", "Col0") | (genotype ==
    "Col0" & full_subline_information %in%
    c("Col1_1", "Col1_3", "Col1_4", "Col1_2"))) %>%
    filter(border == FALSE) %>% select(individual_plant_metadata,
    genotype, flat_number, measurement, time_point,
    measured_value, border, subline, full_subline_information) %>%
    mutate(month = "Jan")
feb_data <- depi_dec_feb %>% filter(!genotype %in%
    c("b1", "b3", "b1b3", "ftsz2-1", "ftsz2-2",
        "ftsz-dbl", "Col0") | (genotype ==
    "Col0" & full_subline_information %in%
    c("Col1-1", "Col1-3", "Col1-4", "Col1-2"))) %>%
    filter(border == FALSE) %>% filter(substr(plant_ID,
    1, 4) == "0218") %>% select(individual_plant_metadata,
    genotype, flat_number, measurement, time_point,
    measured_value, border, subline, full_subline_information) %>%
    mutate(month = "Feb")

### Remove the 'X' in front of some time
### points
dec_data$time_point <- as.numeric(gsub("X",
    "", dec_data$time_point))
feb_data$time_point <- as.numeric(gsub("X",
    "", feb_data$time_point))

### Add a column with the day
dec_data <- add_day_col(dec_data)
jan_data <- add_day_col(jan_data)
feb_data <- add_day_col(feb_data)

### Shift all NPQ values so the minimum is
```

```
### 0
dec_data$measured_value[dec_data$measurement ==
    "npq"] <- (dec_data$measured_value[dec_data$measurement ==
    "npq"]) + abs(min((filter(dec_data, measurement ==
    "npq"))$measured_value))

jan_data$measured_value[jan_data$measurement ==
    "npq"] <- (jan_data$measured_value[jan_data$measurement ==
    "npq"]) + abs(min((filter(jan_data, measurement ==
    "npq"))$measured_value))

feb_data$measured_value[feb_data$measurement ==
    "npq"] <- (feb_data$measured_value[feb_data$measurement ==
    "npq"]) + abs(min((filter(feb_data, measurement ==
    "npq"))$measured_value))

### Remove the 2 NA values for phi2, and
### shift the phi2 measured values by the
### minimum values to ensure the minimum
### value is 0
dec_data <- na.omit(dec_data)

dec_data$measured_value[dec_data$measurement ==
    "phi2"] <- (dec_data$measured_value[dec_data$measurement ==
    "phi2"]) + abs(min((filter(dec_data,
    measurement == "phi2"))$measured_value))

### Merge the two data frames, only using
### the columns they have in common
depi_all <- rbind(dec_data, jan_data, feb_data) %>%
    arrange(genotype, time_point, month)

### Filter the time points by the max time
### point of the shortest experiment -
### can't make comparisons on days that the
### three experiments don't share
depi_all <- as.data.frame(depi_all %>% filter(time_point <=
    min(c(max(jan_data$time_point), max(dec_data$time_point),
        max(feb_data$time_point)))))

### Rename size and growth to be leafarea
depi_all$measurement[depi_all$measurement ==
    "size"] <- "leafarea"
depi_all$measurement[depi_all$measurement ==
    "growth"] <- "leafarea"
```

Remove Outliers

```
remove_out <- remove_outliers(depi_all)
```

Quantile Normalization

```
### Create a loop for each measurement and
### experiment
for (i in c("npq", "phi2", "leafarea")) {
    for (j in c("Dec", "Jan", "Feb")) {
        ### Filter to select each specific month
        ### and measured value

        ### Use the data frame with the outliers
        ### removed!
        temp_vector <- filter(remove_out,
            month == j, measurement == i)
        ### Initialize an empty data frame
        temp_df <- data.frame()
        ### Loop through each flat
        for (k in 1:length(unique(temp_vector$flat_number))) {
            ### Create a temporary data frame - each
            ### column is the measured values for each
            ### flat
            temp <- filter(temp_vector, flat_number ==
                k)$measured_value
            temp_df <- addToDF(temp_df, temp)
        }

        ### Normalize across the flats
        temp_normalize <- as.data.frame(normalize.quantiles(as.matrix(temp_df))) %>%
            ### Add columns with the measurement and
        ### experiment to be certain there hasn't
        ### been any mix-ups
        mutate(measurement = i, experiment = j)
        ### Create a name to give the normalized
        ### data based on the measurement and
        ### experiment
        temp_name <- paste(tolower(j), "_",
            i, "_normalize", sep = "")
        ### Rename the columns
        if (ncol(temp_normalize) == 6) {
            temp_normalize <- temp_normalize %>%
                rename(flat_1 = V1, flat_2 = V2,
                  flat_3 = V3, flat_4 = V4)
        } else {
            temp_normalize <- temp_normalize %>%
                rename(flat_1 = V1, flat_2 = V2,
                  flat_3 = V3, flat_4 = V4,
                  flat_5 = V5)
        }
        ### Assign the name to the data frame
        assign(temp_name, temp_normalize)
    }
}
```

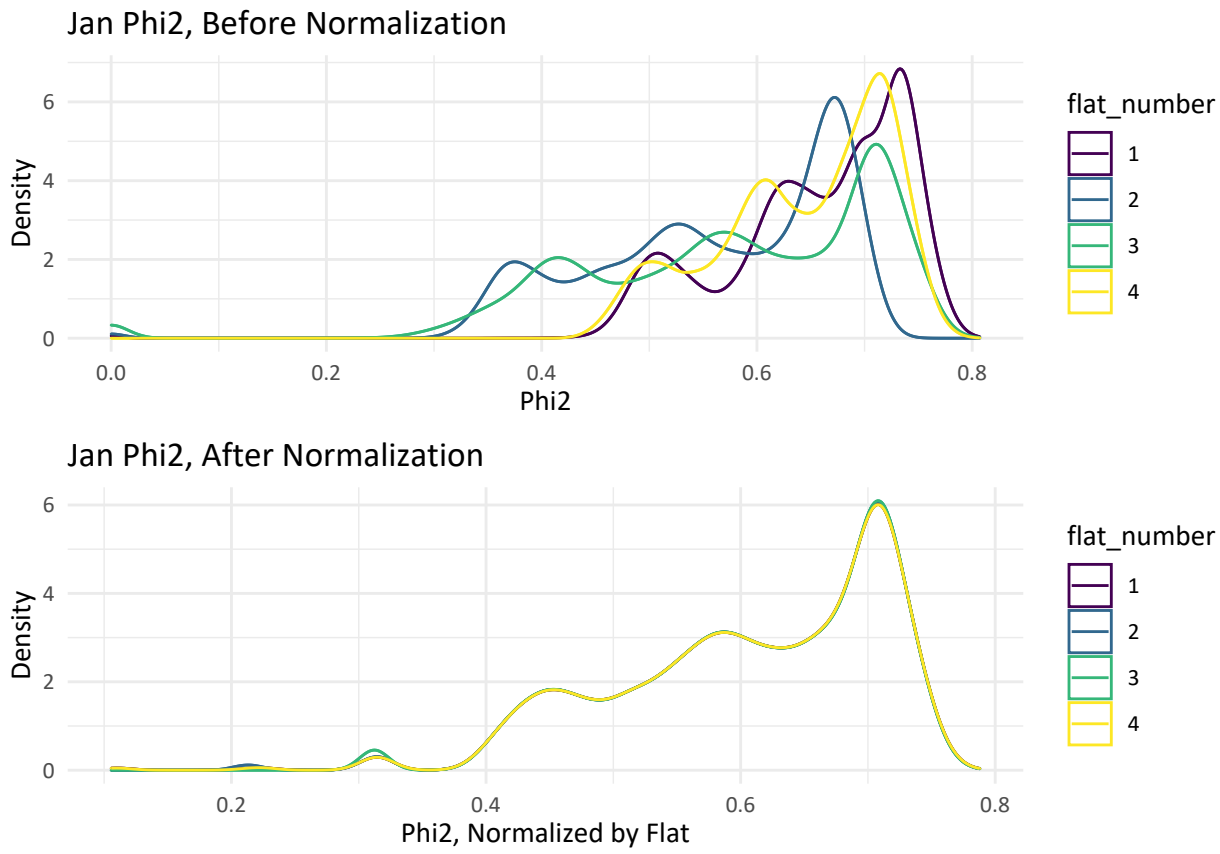
```
### Create a loop for each measurement and
### experiment
for (i in c("npq", "phi2", "leafarea")) {
    for (j in c("Dec", "Jan", "Feb")) {
        ### Filter to select each specific month
        ### and measured value
        temp_vector <- filter(depi_all, month ==
            j, measurement == i)
        ### Initialize an empty data frame
        temp_df <- data.frame()
        ### Loop through each flat
        for (k in 1:length(unique(temp_vector$flat_number))) {
            ### Create a temporary data frame - each
            ### column is the measured values for each
            ### flat
            temp <- filter(temp_vector, flat_number ==
                k)$measured_value
            temp_df <- addToDF(temp_df, temp)
        }
        ### Normalize across the flats
        temp_normalize <- as.data.frame(normalize.quantiles(as.matrix(temp_df))) %>%
            ### Add columns with the measurement and
        ### experiment to be certain there hasn't
        ### been any mix-ups
        mutate(measurement_verify = i, experiment_verify = j)
        ### Create a name to give the normalized
        ### data based on the measurement and
        ### experiment
        temp_name <- paste(tolower(j), "_",
            i, "_normalize", sep = "")
        ### Rename the columns
        if (ncol(temp_normalize) == 6) {
            temp_normalize <- temp_normalize %>%
                rename(flat_1 = V1, flat_2 = V2,
                  flat_3 = V3, flat_4 = V4)
        } else {
            temp_normalize <- temp_normalize %>%
                rename(flat_1 = V1, flat_2 = V2,
                  flat_3 = V3, flat_4 = V4,
                  flat_5 = 5)
        }
        ### Assign the name to the data frame
        assign(temp_name, temp_normalize)
        ### Loop through each of the columns that
        ### have measured values for each flat
        for (num in 1:(ncol(temp_normalize) -
        2)) {
            ### Filter the relevant matching rows from
            ### the depi_all dataframe
            temp_depi_information <- depi_all %>%
                filter(measurement == unique(temp_normalize$measurement_verify),
                  month == unique(temp_normalize$experiment_verify),
                  flat_number == num)
```

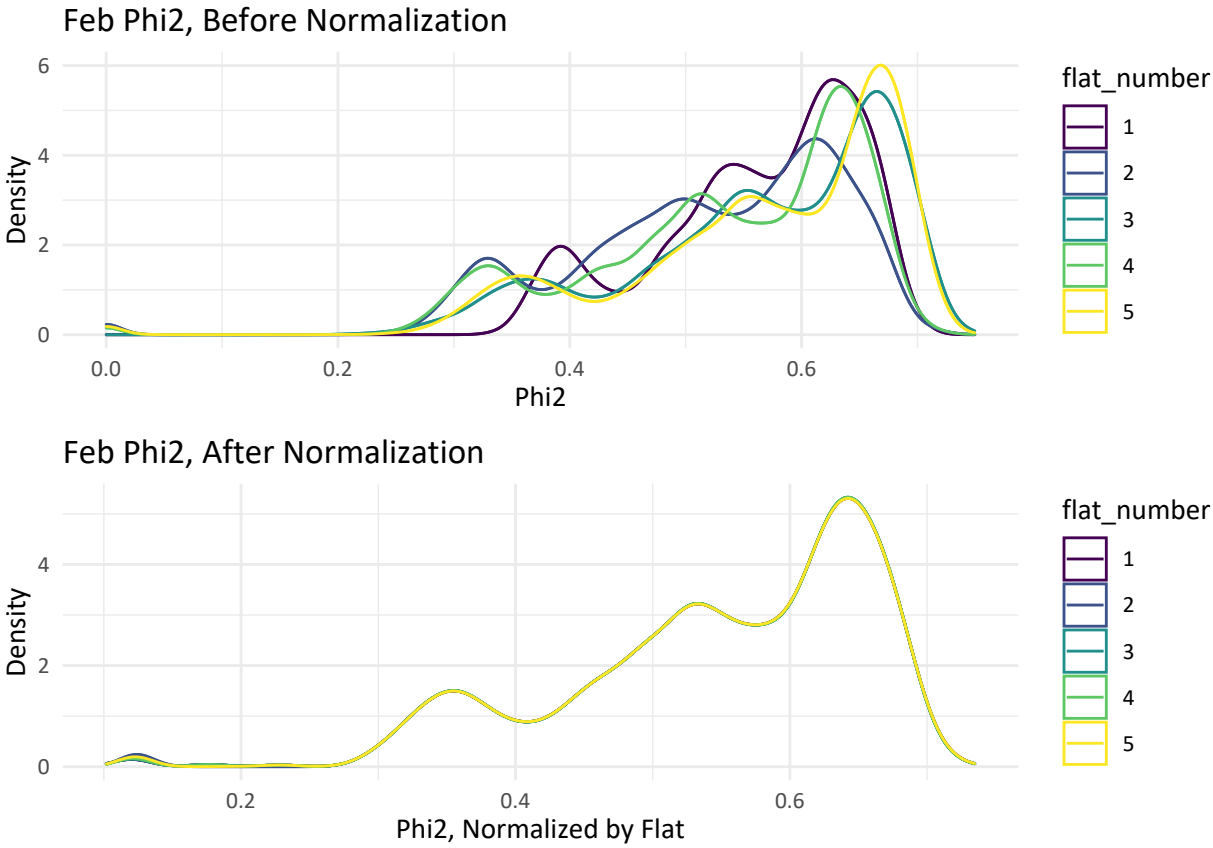
```
            ### Add the column with the normalized data
            ### to the depi subset
            temp_merged <- cbind(temp_depi_information,
                na.omit(temp_normalize[num]))
            ### Create a name for this data frame -
            ### based on the measurement, month, and
            ### flat
            temp_name_final <- paste(temp_name,
                "_flat_", num, sep = "")

            names(temp_merged)[length(names(temp_merged))] <- "normalized_value"
            assign(temp_name_final, temp_merged)

        }
    }
}
### Combine all seperate data frames
quantile_normalize_all <- rbind(dec_leafarea_normalize_flat_1,
    dec_leafarea_normalize_flat_2, dec_leafarea_normalize_flat_3,
    dec_leafarea_normalize_flat_4, jan_leafarea_normalize_flat_1,
    jan_leafarea_normalize_flat_2, jan_leafarea_normalize_flat_3,
    jan_leafarea_normalize_flat_4, feb_leafarea_normalize_flat_1,
    feb_leafarea_normalize_flat_2, feb_leafarea_normalize_flat_3,
    feb_leafarea_normalize_flat_4, feb_leafarea_normalize_flat_5,
    dec_npq_normalize_flat_1, dec_npq_normalize_flat_2,
    dec_npq_normalize_flat_3, dec_npq_normalize_flat_4,
    jan_npq_normalize_flat_1, jan_npq_normalize_flat_2,
    jan_npq_normalize_flat_3, jan_npq_normalize_flat_4,
    feb_npq_normalize_flat_1, feb_npq_normalize_flat_2,
    feb_npq_normalize_flat_3, feb_npq_normalize_flat_4,
    feb_npq_normalize_flat_5, dec_phi2_normalize_flat_1,
    dec_phi2_normalize_flat_2, dec_phi2_normalize_flat_3,
    dec_phi2_normalize_flat_4, jan_phi2_normalize_flat_1,
    jan_phi2_normalize_flat_2, jan_phi2_normalize_flat_3,
    jan_phi2_normalize_flat_4, feb_phi2_normalize_flat_1,
    feb_phi2_normalize_flat_2, feb_phi2_normalize_flat_3,
    feb_phi2_normalize_flat_4, feb_phi2_normalize_flat_5)
```

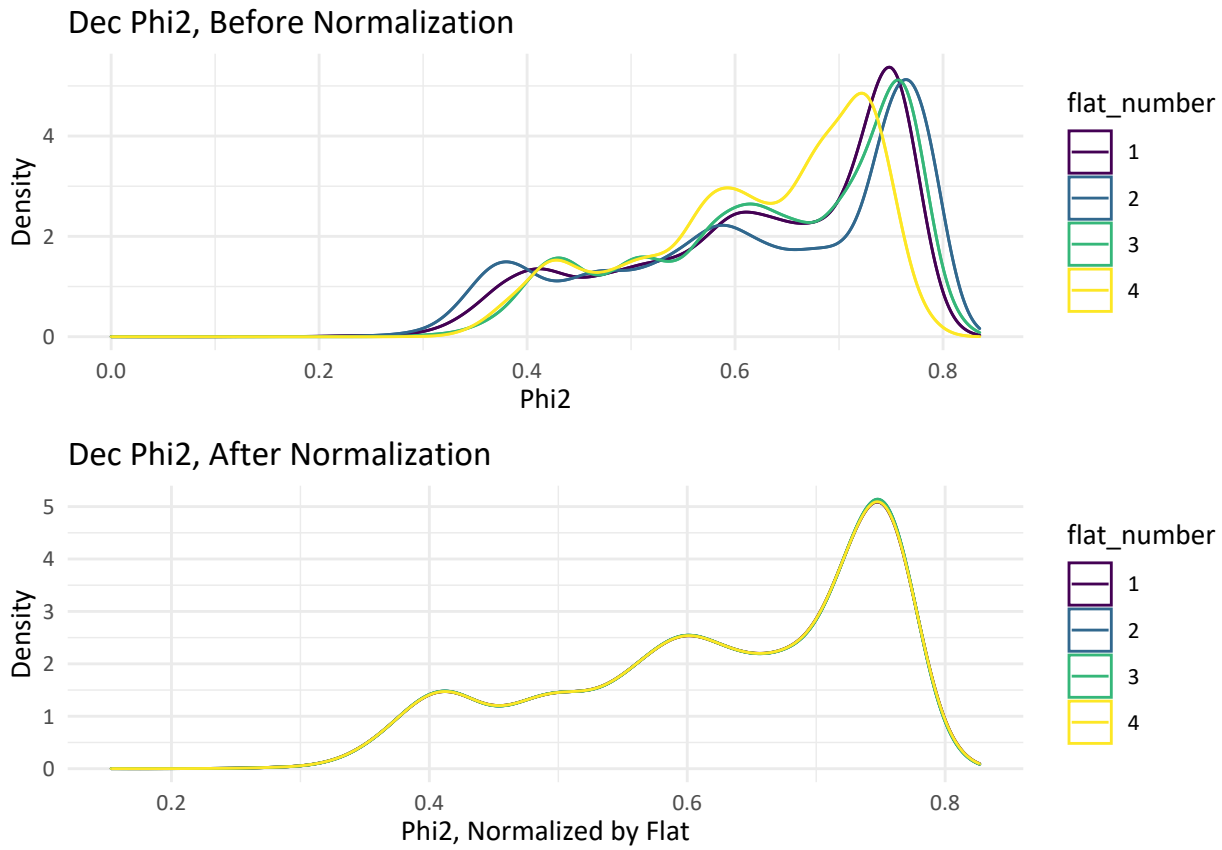
Verify that the quantile normalization accomplished what we wanted it to:

```
library(gridExtra)
for (i in c("Jan", "Feb", "Dec")) {
    tmpTitle <- paste(i, "Phi2, Before Normalization",
        sep = " ")
    plotData <- filter(quantile_normalize_all,
        month == i, measurement == "phi2")
    plotData$flat_number <- as.factor(plotData$flat_number)

    plot1 <- ggplot(data = plotData, aes(x = measured_value,
        color = flat_number)) + geom_density() +
        stat_density(geom = "line", position = "identity") +
        theme_minimal(base_family = "Calibri") +
        labs(x = "Phi2", y = "Density", title = tmpTitle) +
```

```
        scale_color_viridis(discrete = TRUE)

    tmpTitle2 <- paste(i, "Phi2, After Normalization",
        sep = " ")
    plot2 <- ggplot(data = plotData, aes(x = normalized_value,
        color = flat_number)) + geom_density() +
        stat_density(geom = "line", position = "identity") +
        theme_minimal(base_family = "Calibri") +
        labs(x = "Phi2, Normalized by Flat",
            y = "Density", title = tmpTitle2) +
        scale_color_viridis(discrete = TRUE)
    grid.arrange(plot1, plot2)

}
```



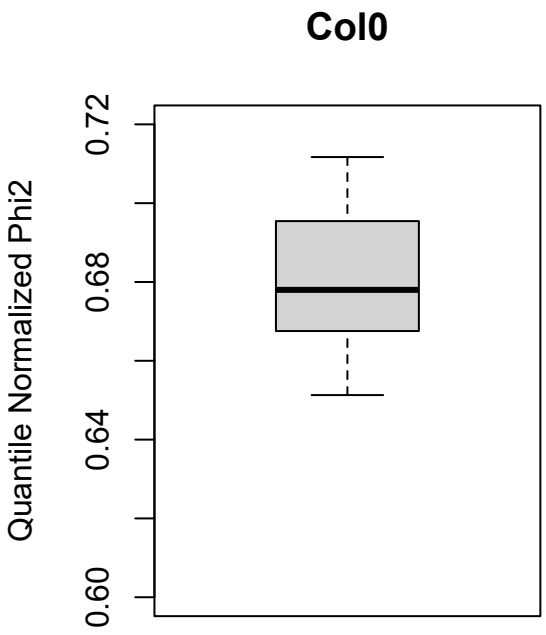

Feb Phi2, Before Normalization
Density
Phi2
flat_number
1
2
3
4
5
Feb Phi2, After Normalization
Density
Phi2, Normalized by Flat
flat_number
1
2
3
4
5

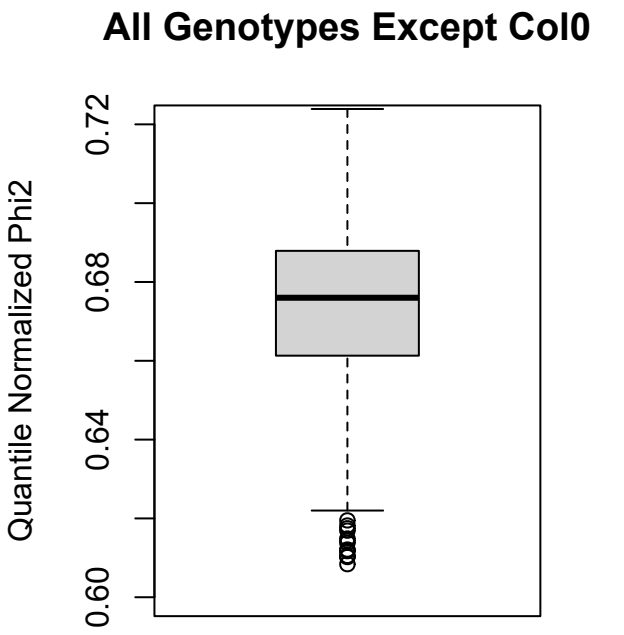


Investigate February Col0

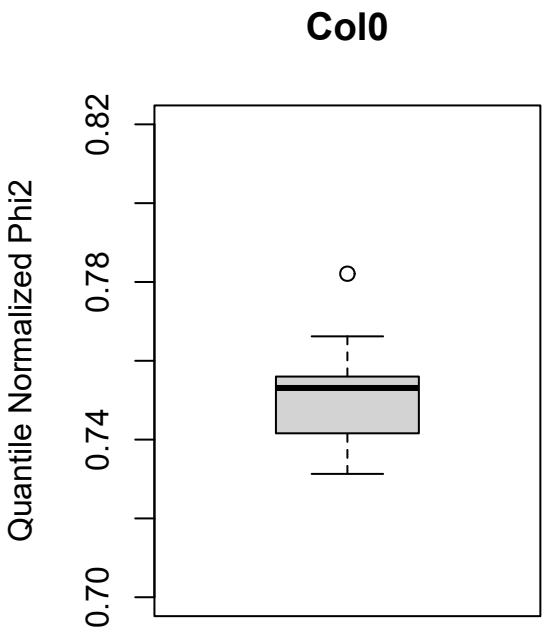
Melissa noted that the February plots have the opposite direction of log fold change than the January and December experiments. Investigate this by creating box plots of phi2:

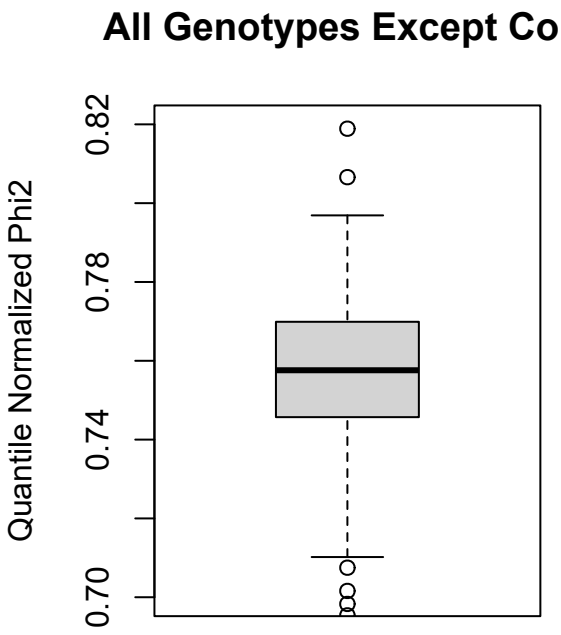
```
plotDataCol0 <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Feb",
        measurement == "phi2", genotype ==
            "Col0")
plotDataOther <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Feb",
        measurement == "phi2", genotype !=
            "Col0")
par(mfrow = c(1, 2))
boxplot(plotDataCol0$measured_value, ylab = "Quantile Normalized Phi2",
    main = "Col0", sub = "February Phi2 Measurements at Time 0",
    ylim = c(0.6, 0.72))

boxplot(plotDataOther$measured_value, ylab = "Quantile Normalized Phi2",
    main = "All Genotypes Except Col0", sub = "February Phi2 Measurements at Time 0",
    ylim = c(0.6, 0.72))
```



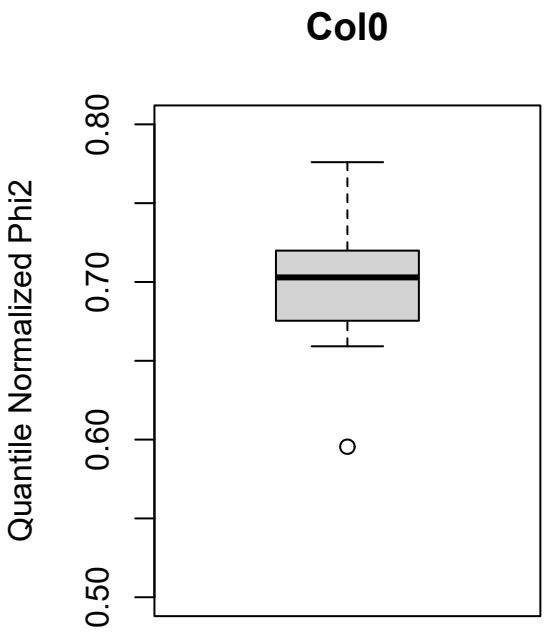


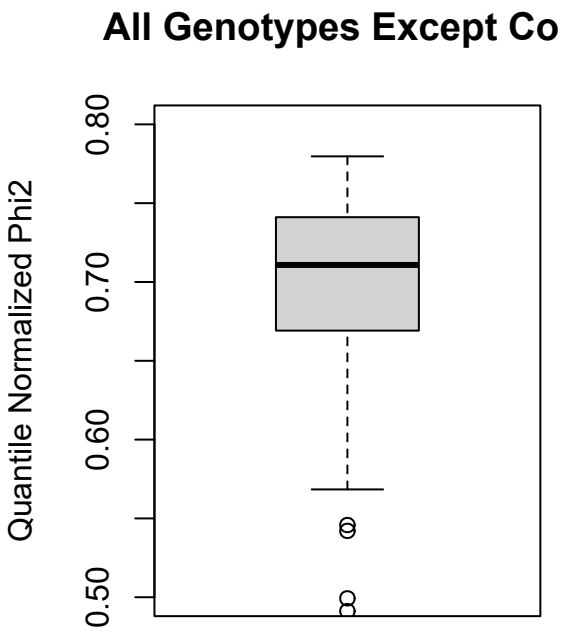

```
plotDataCol0 <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Dec",
        measurement == "phi2", genotype ==
            "Col0")
plotDataOther <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Dec",
        measurement == "phi2", genotype !=
            "Col0")
par(mfrow = c(1, 2))
boxplot(plotDataCol0$measured_value, ylab = "Quantile Normalized Phi2",
    main = "Col0", sub = "December Phi2 Measurements at Time 0",
    ylim = c(0.7, 0.82))
boxplot(plotDataOther$measured_value, ylab = "Quantile Normalized Phi2",
    main = "All Genotypes Except Col0", sub = "December Phi2 Measurements at Time 0",
    ylim = c(0.7, 0.82))
```






```
plotDataCol0 <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Jan",
        measurement == "phi2", genotype ==
            "Col0")
plotDataOther <- quantile_normalize_all %>%
    filter(time_point == 0, month == "Jan",
        measurement == "phi2", genotype !=
            "Col0")
par(mfrow = c(1, 2))
boxplot(plotDataCol0$measured_value, ylab = "Quantile Normalized Phi2",
    main = "Col0", sub = "January Phi2 Measurements at Time 0",
    ylim = c(0.5, 0.8))
boxplot(plotDataOther$measured_value, ylab = "Quantile Normalized Phi2",
    main = "All Genotypes Except Col0", sub = "January Phi2 Measurements at Time 0",
    ylim = c(0.5, 0.8))
```



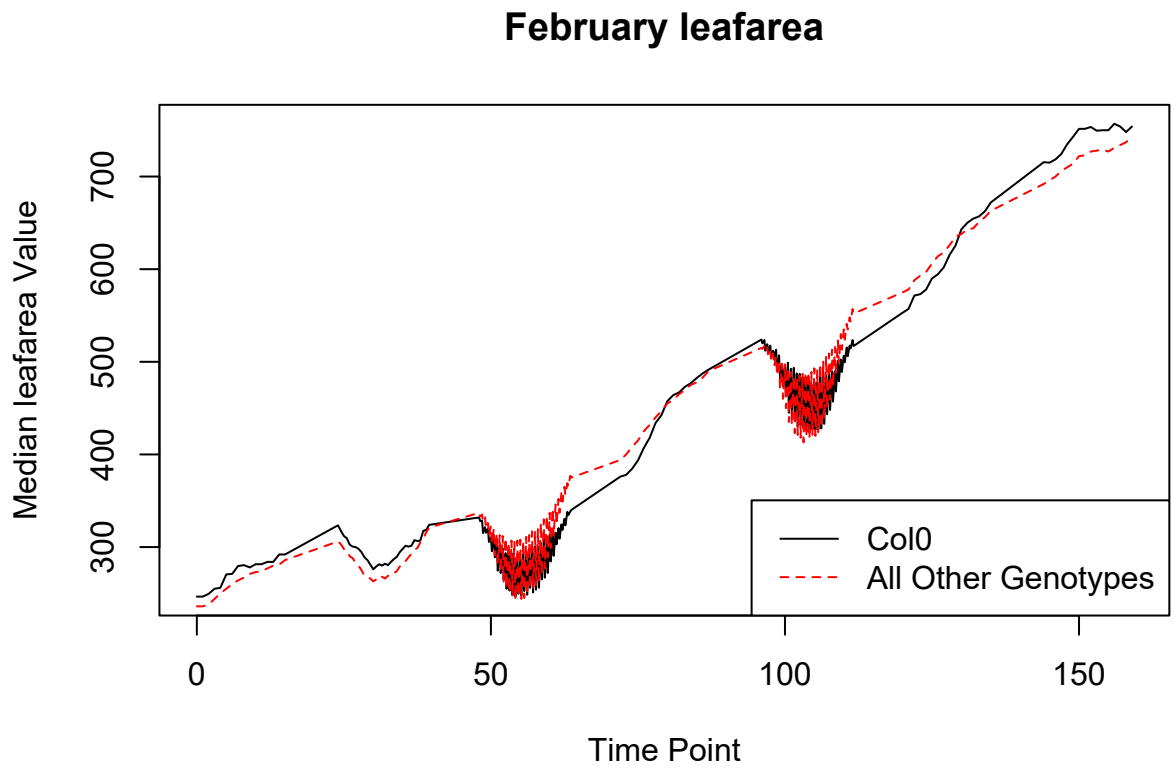


January Phi2 Measurements at Time 0

January Phi2 Measurements at Time 0

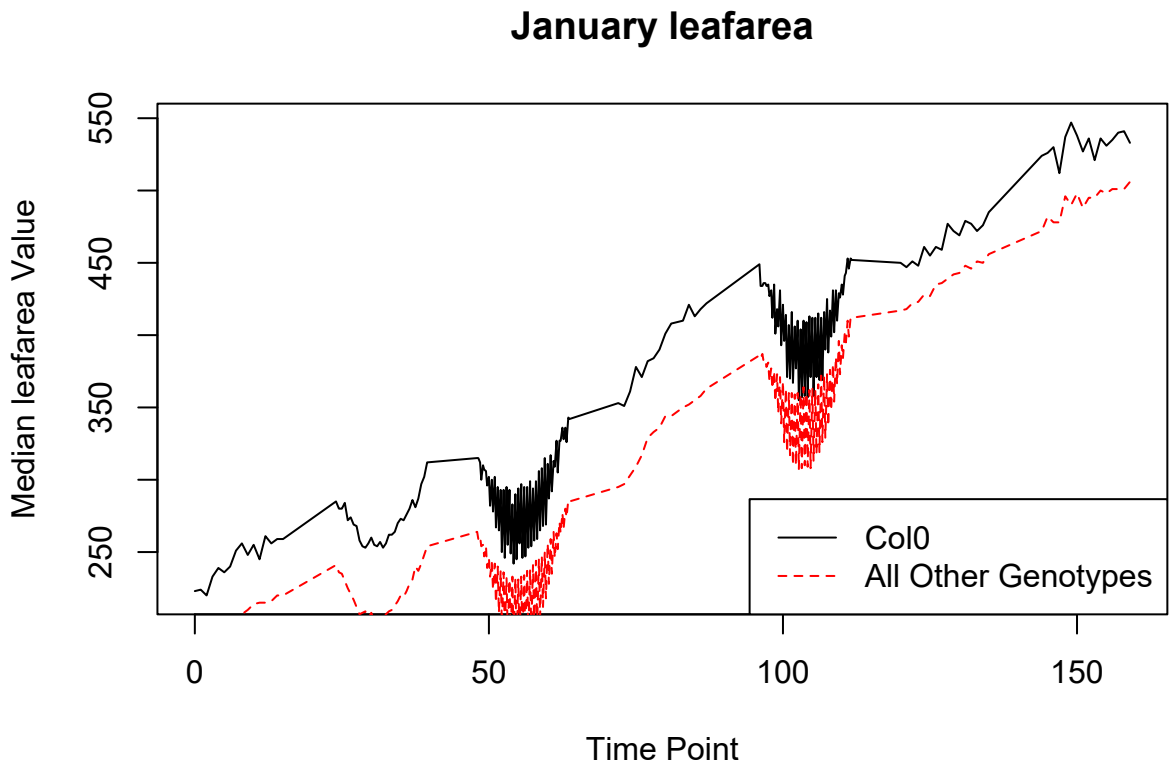
```
plotDataCol0Feb <- as.data.frame(quantile_normalize_all %>%
    filter(genotype == "Col0", measurement ==
        "leafarea", month == "Feb") %>% group_by(time_point) %>%
    summarize(Col0Med = median(measured_value)))

plotDataOtherFeb <- as.data.frame(quantile_normalize_all %>%
    filter(genotype != "Col0", measurement ==
        "leafarea", month == "Feb") %>% group_by(time_point) %>%
    summarize(OtherMed = median(measured_value)))

plotDataCol0Jan <- as.data.frame(quantile_normalize_all %>%
    filter(genotype == "Col0", measurement ==
        "leafarea", month == "Jan") %>% group_by(time_point) %>%
    summarize(Col0Med = median(measured_value)))

plotDataOtherJan <- as.data.frame(quantile_normalize_all %>%
    filter(genotype != "Col0", measurement ==
        "leafarea", month == "Jan") %>% group_by(time_point) %>%
    summarize(OtherMed = median(measured_value)))

plotDataCol0Dec <- as.data.frame(quantile_normalize_all %>%
    filter(genotype == "Col0", measurement ==
        "leafarea", month == "Dec") %>% group_by(time_point) %>%
    summarize(Col0Med = median(measured_value)))

plotDataOtherDec <- as.data.frame(quantile_normalize_all %>%
```

```
    filter(genotype != "Col0", measurement ==
        "leafarea", month == "Dec") %>% group_by(time_point) %>%
    summarize(OtherMed = median(measured_value)))




allFeb <- cbind(plotDataCol0Feb, plotDataOtherFeb)
allJan <- cbind(plotDataCol0Jan, plotDataOtherJan)
allDec <- cbind(plotDataCol0Dec, plotDataOtherDec)


plot(allFeb$Col0Med ~ allFeb$time_point,
    type = "l", xlab = "Time Point", ylab = "Median leafarea Value",
    main = "February leafarea")
lines(allFeb$OtherMed ~ allFeb$time_point,
    type = "l", col = "red", lty = 2)
legend("bottomright", legend = c("Col0",
    "All Other Genotypes"), lty = c(1, 2),
    col = c("black", "red"))
```

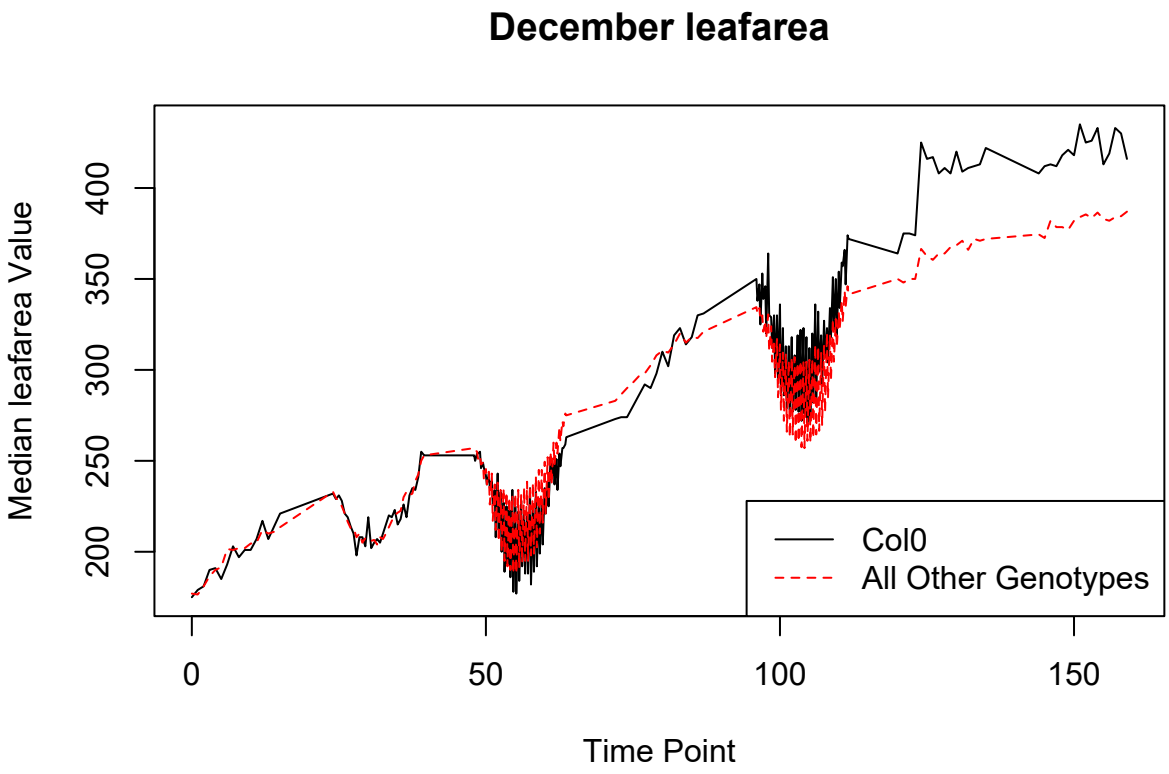



```
plot(allJan$Col0Med ~ allJan$time_point,
    type = "l", xlab = "Time Point", ylab = "Median leafarea Value",
    main = "January leafarea")
lines(allJan$OtherMed ~ allJan$time_point,
```

```
    type = "l", col = "red", lty = 2)
legend("bottomright", legend = c("Col0",
    "All Other Genotypes"), lty = c(1, 2),
    col = c("black", "red"))
```




```
plot(allDec$Col0Med ~ allDec$time_point,
    type = "l", xlab = "Time Point", ylab = "Median leafarea Value",
    main = "December leafarea")
lines(allDec$OtherMed ~ allDec$time_point,
    type = "l", col = "red", lty = 2)
legend("bottomright", legend = c("Col0",
    "All Other Genotypes"), lty = c(1, 2),
    col = c("black", "red"))
```



Selection Coefficient Calculations

```
selectionCoef <- data.frame(genotype = rep(NA,
    0), SelectionCoefficient = rep(NA, 0),
    Experiment = rep(NA, 0), Measurement = rep(NA,
        0))

for (e in c("Dec", "Jan", "Feb")) {
    for (m in c("phi2")) {
        ### Create an empty data frame to fill with
        ### the information and calcuations:
        selectionCoefTmp <- data.frame(genotype = rep(NA,
            38), SelectionCoefficient = rep(NA,
            38), Experiment = rep(NA, 38),
            Measurement = rep(NA, 38))
        count <- 1
        for (g in unique(quantile_normalize_all$genotype)) {
            fm <- mean(filter(quantile_normalize_all,
                genotype == g, month == e,
                measurement == m, day ==
                  7)$normalized_value)
            fwt <- mean(filter(quantile_normalize_all,
                genotype == "Col0", month ==
```

```
                  e, measurement == m, day ==
                  7)$normalized_value)
            TempSelectionCoef <- (fm - fwt)/fwt
            selectionCoefTmp[count, 1] <- g
            selectionCoefTmp[count, 2] <- TempSelectionCoef
            selectionCoefTmp[count, 3] <- e
            selectionCoefTmp[count, 4] <- m

            count <- count + 1
        }
        selectionCoef <- rbind(selectionCoef,
            selectionCoefTmp)
    }
}

selectionCoef <- selectionCoef %>% arrange(Measurement,
    genotype)
```

Epistasis Calculations

```
all_double_mutants = list()
for (gen in unique(quantile_normalize_all$genotype)) {
    if (str_detect(gen, "-") == T) {
        all_double_mutants = c(all_double_mutants,
            gen)
    }
}

### Epistasis Calculations: Initialize an
### empty data frame to populate with
### information:
geneticInteractions <- data.frame(genotype = rep(NA,
    0), MutantA = rep(NA, 0), MutantB = rep(NA,
    0), AdditiveEpistasis = rep(NA, 0), ProportionalEpistatis = rep(NA,
    0), Experiment = rep(NA, 0), Measurement = rep(NA,
    0))

### Loop through each experiment and
### measurement:
for (e in c("Dec", "Jan", "Feb")) {
    for (m in c("phi2")) {
        ### Filter to each specific experiment and
        ### measurement
        tempData <- filter(quantile_normalize_all,
            month == e, measurement == m,
            day == 7)
        ### Create an empty data frame to fill with
        ### the information and calcuations:
        geneticInteractionsTmp <- data.frame(genotype = rep(NA,
            25), MutantA = rep(NA, 25), MutantB = rep(NA,
            25), AdditiveEpistasis = rep(NA,
```

```
            25), ProportionalEpistatis = rep(NA,
            25), Experiment = rep(NA, 25),
            Measurement = rep(NA, 25))
        ### Initialize a row count to use to
        ### populate the data frame
        rowCount <- 1
        ### For each of the double mutants:
        for (dm in unlist(all_double_mutants)) {
            ### Extract the single mutants from the
            ### double mutant
            ma <- unlist(strsplit(dm, "-"))[1]
            mb <- paste("mpk", unlist(strsplit(dm,
                "-"))[2], sep = "")
            ### Calculate the fitness of the dm, ma,
            ### mb, and wt
            fdm <- mean(filter(tempData,
                genotype == dm)$normalized_value)
            fwt <- mean(filter(tempData,
                genotype == "Col0")$normalized_value)
            fma <- mean(filter(tempData,
                genotype == ma)$normalized_value)
            fmb <- mean(filter(tempData,
                genotype == mb)$normalized_value)
            ### Calculate Additive and Proportional
            ### Epistasis
            AddEp <- fdm + fwt - (fma + fmb)
            PropEp <- log((fdm * fwt)/(fma *
                fmb))
            ### Populate the data frame with this
            ### information:
            geneticInteractionsTmp[rowCount,
                1] <- dm
            geneticInteractionsTmp[rowCount,
                2] <- ma
            geneticInteractionsTmp[rowCount,
                3] <- mb
            geneticInteractionsTmp[rowCount,
                4] <- AddEp
            geneticInteractionsTmp[rowCount,
                5] <- PropEp
            geneticInteractionsTmp[rowCount,
                6] <- e
            geneticInteractionsTmp[rowCount,
                7] <- m
            rowCount <- rowCount + 1
        }
        ### Add the rows of the temporary genetic
        ### interaction information to the main
        ### data frame
        geneticInteractions <- rbind(geneticInteractions,
            geneticInteractionsTmp)
    }
}
```

Create Epistasis Plots

```
selectionCoef <- add_number(selectionCoef)
selectionCoef$genotype <- reorder(selectionCoef$genotype,
    desc(selectionCoef$number))

geneticInteractions <- add_number(geneticInteractions)
geneticInteractions$genotype <- reorder(geneticInteractions$genotype,
    desc(geneticInteractions$number))
```

```
tiff("DEPI_AdditiveEpistasis_Day7.tiff",
    units = "in", width = 7, height = 7,
    res = 500)
plot1 <- ggplot(data = filter(geneticInteractions),
    aes(x = (factor(Experiment, levels = c("Dec",
        "Jan", "Feb"))), y = genotype, fill = AdditiveEpistasis)) +
    geom_tile() + labs(x = "", y = "Double Mutant",
    title = "Additive Epistasis") + theme_tufte(base_family = "Calibri") +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.05,
            0.06), breaks = c(-0.05, 0, 0.06),
        labels = c("-0.05", "0", "0.06"))
print(plot1)
dev.off()
```

```
## cairo_pdf
##         2
```

```
tiff("DEPI_ProportionalEpistasis_Day7.tiff",
    units = "in", width = 7, height = 7,
    res = 500)
plot2 <- ggplot(data = filter(geneticInteractions),
    aes(x = (factor(Experiment, levels = c("Dec",
        "Jan", "Feb"))), y = genotype, fill = ProportionalEpistatis)) +
    geom_tile() + labs(x = "", y = "Double Mutant",
    title = "Proportional Epistasis") + theme_tufte(base_family = "Calibri") +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.05,
            0.09), breaks = c(-0.05, 0, 0.09),
        labels = c("-0.05", "0", "0.09"))
print(plot2)
dev.off()
```

```
## cairo_pdf
##         2
```

```
tiff("DEPI_SelectionCoefficients_Day7.tiff",
    units = "in", width = 7, height = 7,
    res = 500)
plot3 <- ggplot(data = filter(selectionCoef),
    aes(x = (factor(Experiment, levels = c("Dec",
```

```
        "Jan", "Feb"))), y = genotype, fill = SelectionCoefficient)) +
    geom_tile() + labs(x = "", y = "Genotype",
    title = "Selection Coefficients") + theme_tufte(base_family = "Calibri") +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.12,
            0.05), breaks = c(-0.12, 0, 0.05),
        labels = c("-0.12", "0", "0.05"))
print(plot3)
dev.off()
```

```
## cairo_pdf
##         2
```

Create Plots

We will just create plots for the phi2 measurement, because the npq and leaf area measurements still need to be rescaled.

Note that I modified the cell_371_data function to use normalized_value instead of measured_value!

For each month:

1. Filter to each month and to only include phi2
2. Apply functions to the data frame: add a column with the day, find the log2 fold value for the quantile normalized values, and add a column with a number which will be used to sort the plots
3. Reorder the genotype based on the number

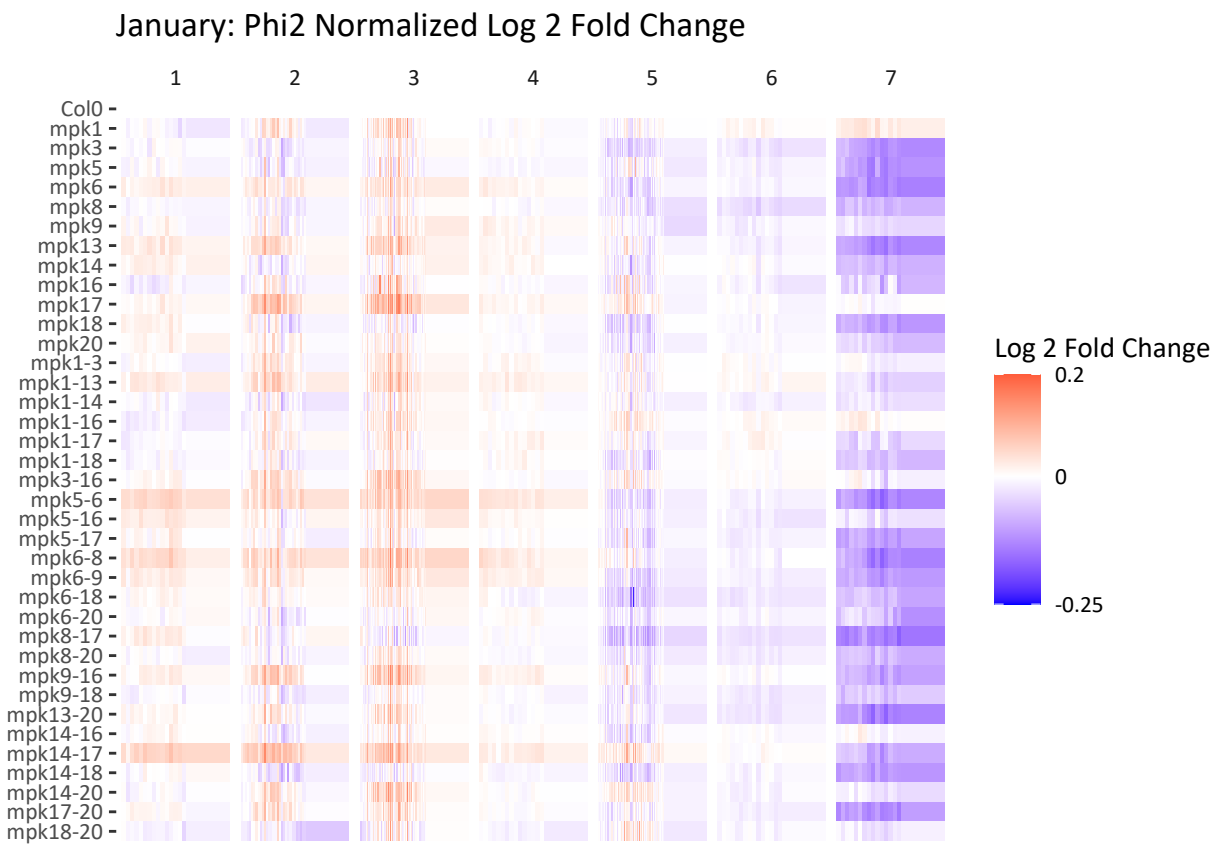
```
### January Data
jan_data <- filter(quantile_normalize_all,
    month == "Jan" & measurement == "phi2")
cell_371_jan <- add_number(cell_371_data(add_day_col(jan_data)))
cell_371_jan$genotype <- reorder(cell_371_jan$genotype,
    cell_371_jan$number)
### February Data
feb_data <- filter(quantile_normalize_all,
    month == "Feb" & measurement == "phi2")
cell_371_feb <- add_number(cell_371_data(add_day_col(feb_data)))
cell_371_feb$genotype <- reorder(cell_371_feb$genotype,
    cell_371_feb$number)
### December Data
dec_data <- filter(quantile_normalize_all,
    month == "Dec" & measurement == "phi2")
cell_371_dec <- add_number(cell_371_data(add_day_col(dec_data)))
cell_371_dec$genotype <- reorder(cell_371_dec$genotype,
    cell_371_dec$number)
```

January

```
summary(cell_371_jan$log2_fold)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.233986 -0.017805  0.000000 -0.001215  0.017245  0.172590
```

```
ggplot(data = cell_371_jan, aes(x = time_point,
    y = genotype, fill = log2_fold)) + labs(fill = "Log 2 Fold Change",
    x = "Hours", y = NULL, title = "January: Phi2 Normalized Log 2 Fold Change") +
    geom_tile(width = 10, height = 10) +
    facet_grid(genotype ~ day, scales = "free",
        switch = "y") + theme_tufte(base_family = "Calibri") +
    theme(strip.background.y = element_blank(),
        strip.text.y = element_blank(), axis.title.x = element_blank(),
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        panel.spacing = unit(0, "lines")) +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.25,
            0.2), breaks = c(-0.25, 0, 0.2),
        labels = c("-0.25", "0", "0.2"))
```

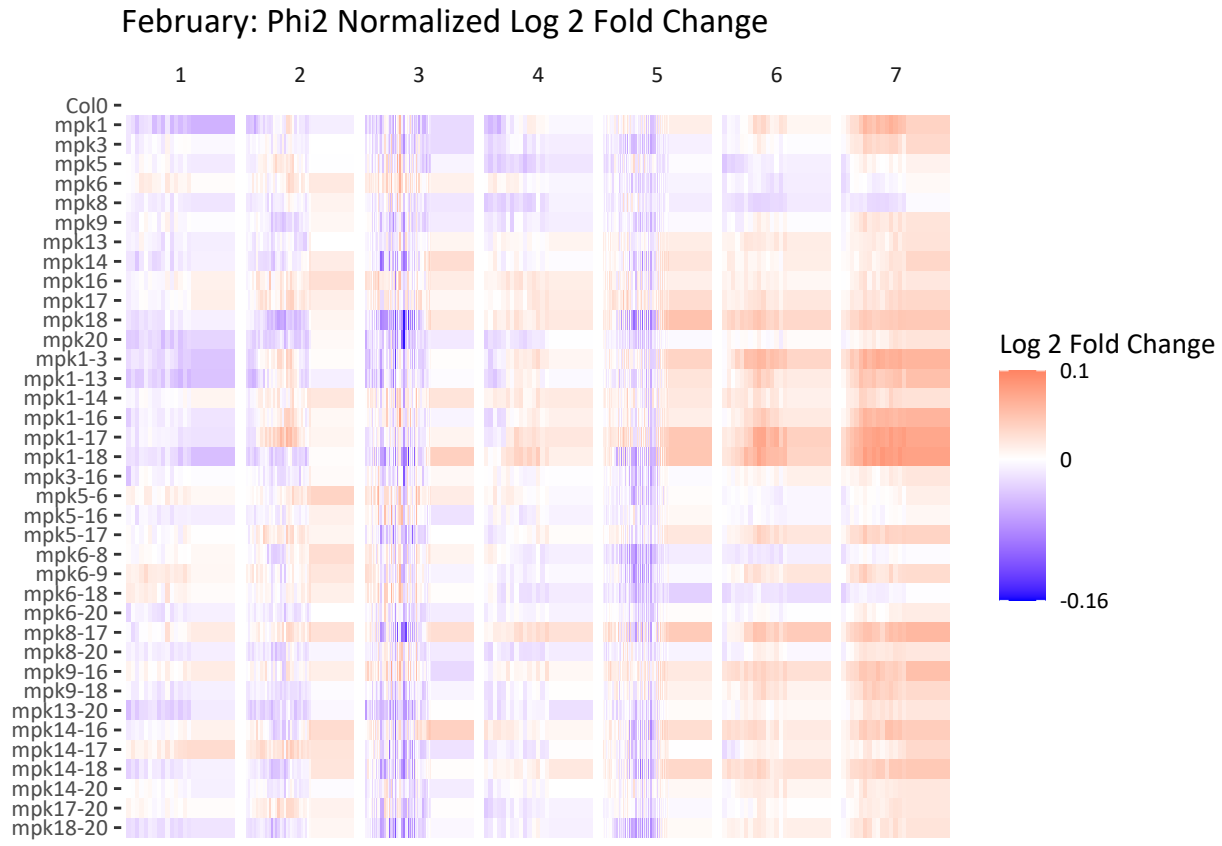


February

```
summary(cell_371_feb$log2_fold)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.154367 -0.016146 -0.001688 -0.002985  0.011761  0.085283
```

```
ggplot(data = cell_371_feb, aes(x = time_point,
    y = genotype, fill = log2_fold)) + labs(fill = "Log 2 Fold Change",
    x = "Hours", y = NULL, title = "February: Phi2 Normalized Log 2 Fold Change") +
    geom_tile(width = 10, height = 10) +
    facet_grid(genotype ~ day, scales = "free",
        switch = "y") + theme_tufte(base_family = "Calibri") +
    theme(strip.background.y = element_blank(),
        strip.text.y = element_blank(), axis.title.x = element_blank(),
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        panel.spacing = unit(0, "lines")) +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.16,
            0.1), breaks = c(-0.16, 0, 0.1),
        labels = c("-0.16", "0", "0.1"))
```

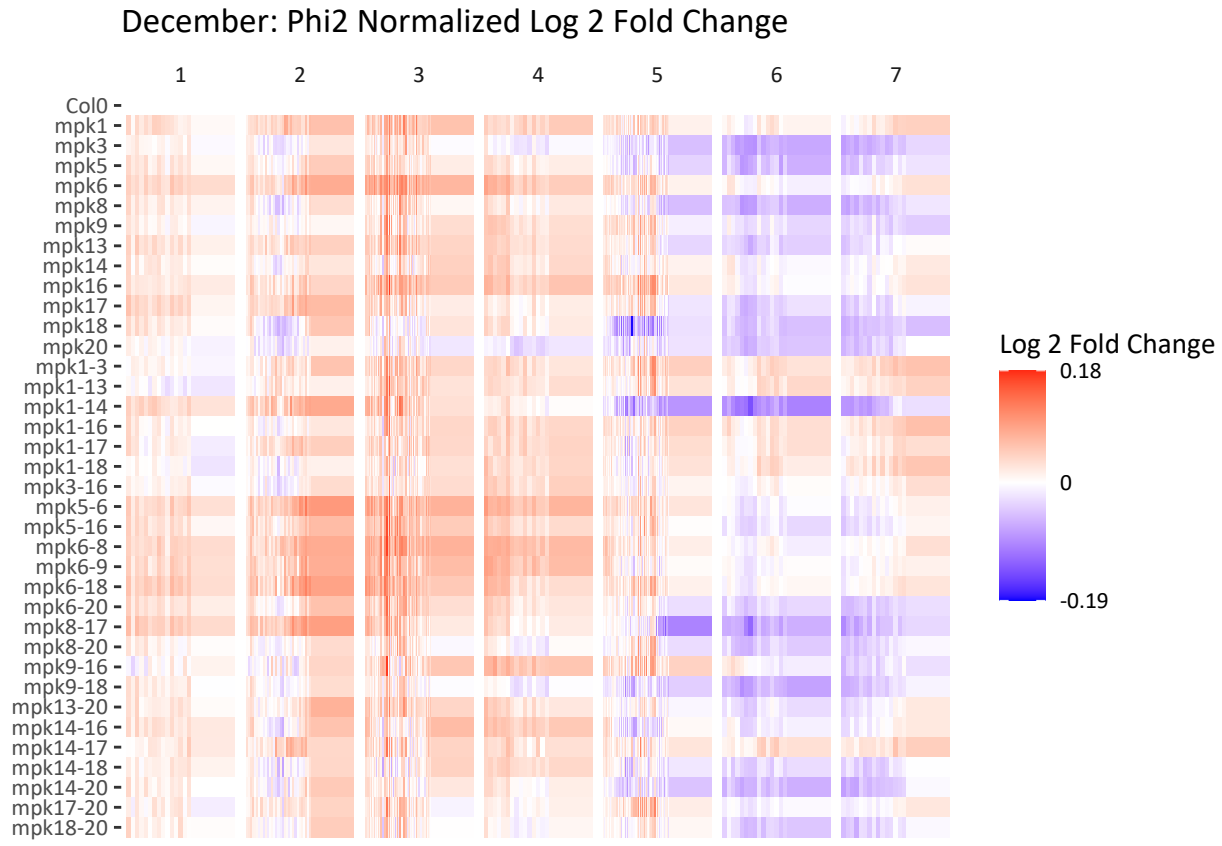


December

```
summary(cell_371_dec$log2_fold)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.187245 -0.005241  0.015746  0.015178  0.037712  0.172322
```

```
ggplot(data = cell_371_dec, aes(x = time_point,
    y = genotype, fill = log2_fold)) + labs(fill = "Log 2 Fold Change",
    x = "Hours", y = NULL, title = "December: Phi2 Normalized Log 2 Fold Change") +
    geom_tile(width = 10, height = 10) +
    facet_grid(genotype ~ day, scales = "free",
        switch = "y") + theme_tufte(base_family = "Calibri") +
    theme(strip.background.y = element_blank(),
        strip.text.y = element_blank(), axis.title.x = element_blank(),
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        panel.spacing = unit(0, "lines")) +
    scale_fill_gradient2(low = "blue", high = "red",
        mid = "white", midpoint = 0, limits = c(-0.19,
            0.18), breaks = c(-0.19, 0, 0.18),
        labels = c("-0.19", "0", "0.18"))
```



Create "triplicate" plots

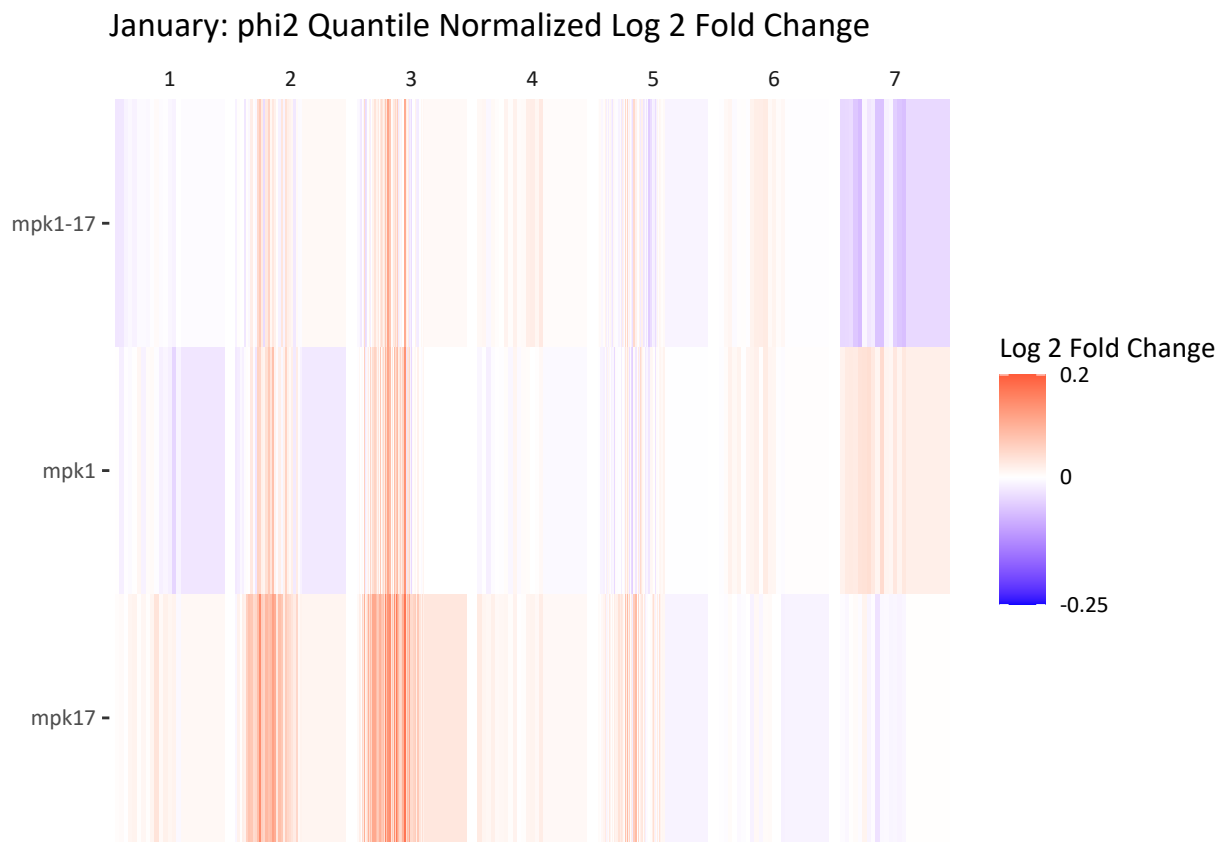
For each month, create plots for each combination of single, single, double mutants.

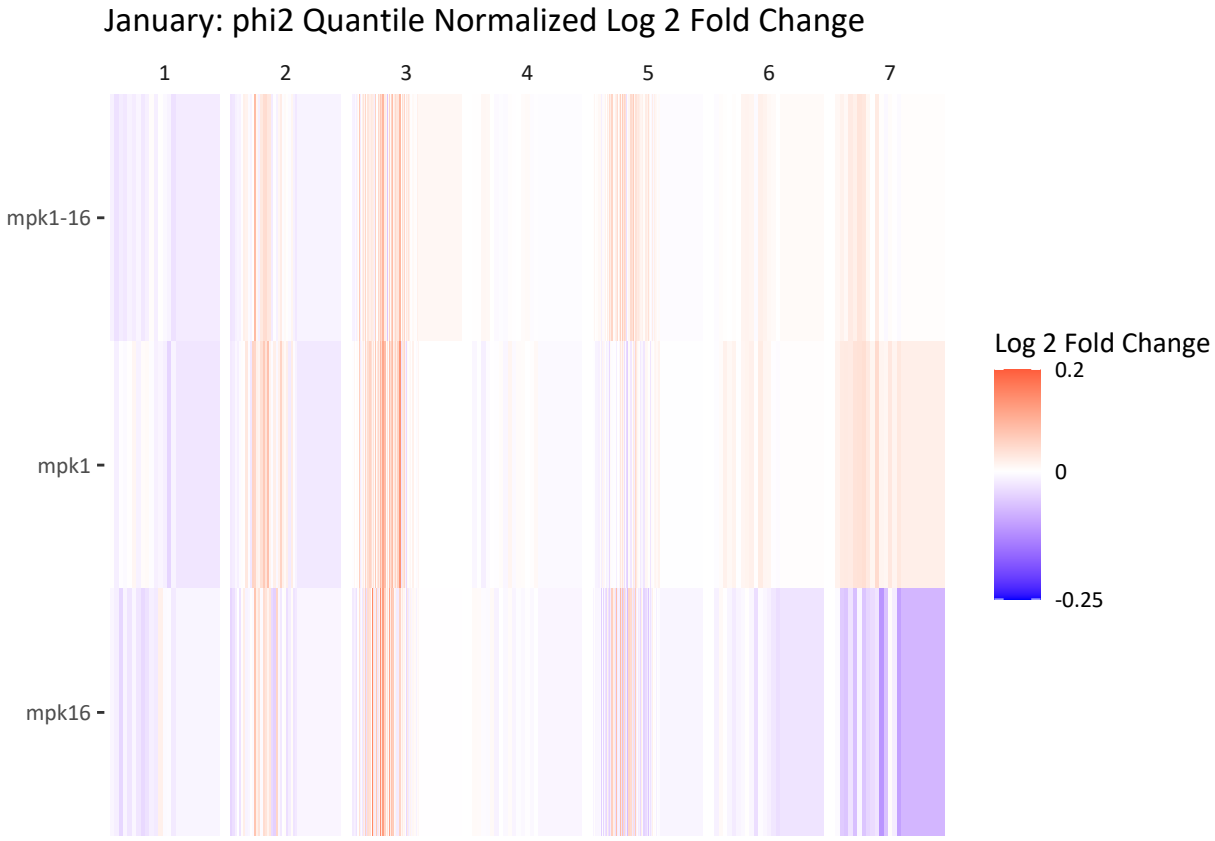
```
genotype_combinations <- list(c("mpk1-17",
    "mpk1", "mpk17"), c("mpk1-16", "mpk1",
    "mpk16"), c("mpk6-9", "mpk6", "mpk9"),
    c("mpk17-20", "mpk17", "mpk20"), c("mpk14-17",
        "mpk14", "mpk17"), c("mpk8-17", "mpk8",
        "mpk17"), c("mpk8-20", "mpk8", "mpk20"),
    c("mpk6-18", "mpk6", "mpk18"), c("mpk1-13",
        "mpk1", "mpk13"), c("mpk17-20", "mpk17",
        "mpk20"), c("mpk13-20", "mpk13",
        "mpk20"), c("mpk6-8", "mpk6", "mpk8"),
    c("mpk9-18", "mpk9", "mpk18"), c("mpk6-20",
        "mpk6", "mpk20"), c("mpk14-16", "mpk14",
        "mpk16"), c("mpk18-20", "mpk18",
        "mpk20"), c("mpk5-6", "mpk5", "mpk6"),
    c("mpk14-18", "mpk14", "mpk18"), c("mpk5-6",
        "mpk5", "mpk6"), c("mpk14-18", "mpk14",
        "mpk18"), c("mpk5-17", "mpk5", "mpk17"),
    c("mpk1-3", "mpk1", "mpk3"), c("mpk1-17",
        "mpk1", "mpk17"), c("mpk3-16", "mpk3",
        "mpk16"), c("mpk9-16", "mpk9", "mpk16"),
    c("mpk14-20", "mpk14", "mpk20"))
```

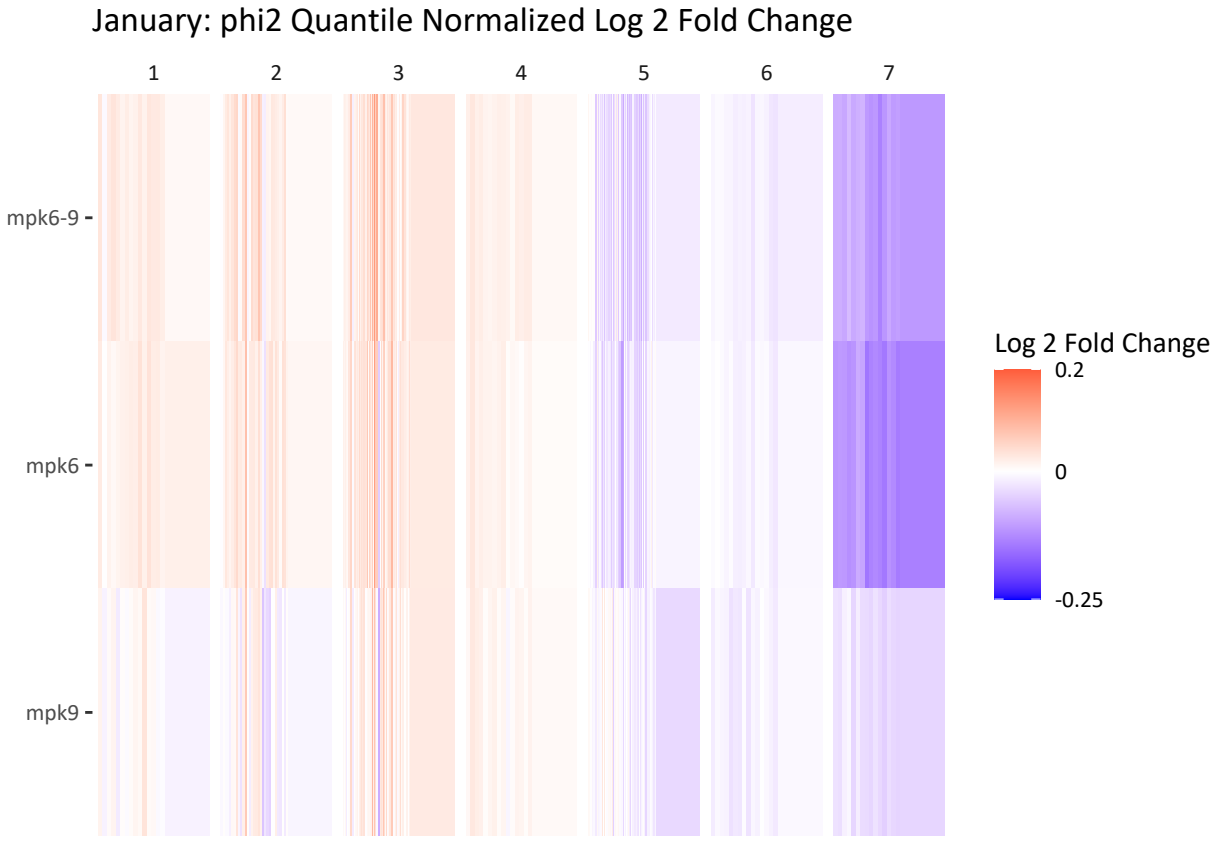
January Loop

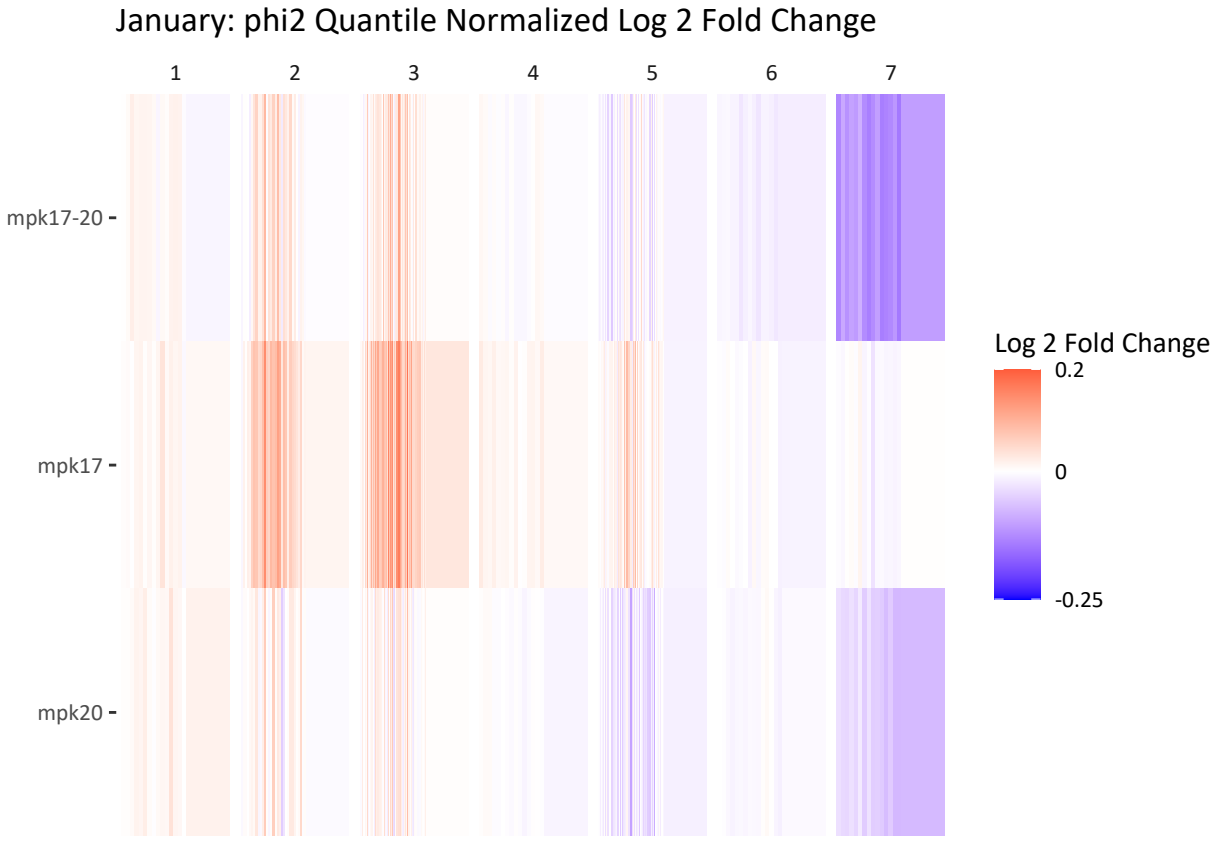
```
for (element in genotype_combinations) {
    m <- "phi2"
    data <- filter(cell_371_jan, genotype %in%
        element, measurement == m)
    data$genotype <- reorder(data$genotype,
        data$number_2)
    plot <- ggplot(data = data, aes(x = time_point,
        y = genotype, fill = log2_fold)) +
        labs(fill = "Log 2 Fold Change",
            x = "Hours", y = NULL, title = paste("January:",
                m, "Quantile Normalized Log 2 Fold Change")) +
        geom_tile(width = ifelse(m == "leafarea",
            16, 10), height = 30) + facet_grid(genotype ~
        day, scales = "free", switch = "y") +
        theme_tufte(base_family = "Calibri") +
        theme(strip.background.y = element_blank(),
            strip.text.y = element_blank(),
            axis.title.x = element_blank(),
            axis.text.x = element_blank(),
            axis.ticks.x = element_blank(),
            panel.spacing = unit(0, "lines")) +
        scale_fill_gradient2(low = "blue",
            high = "red", mid = "white",
```

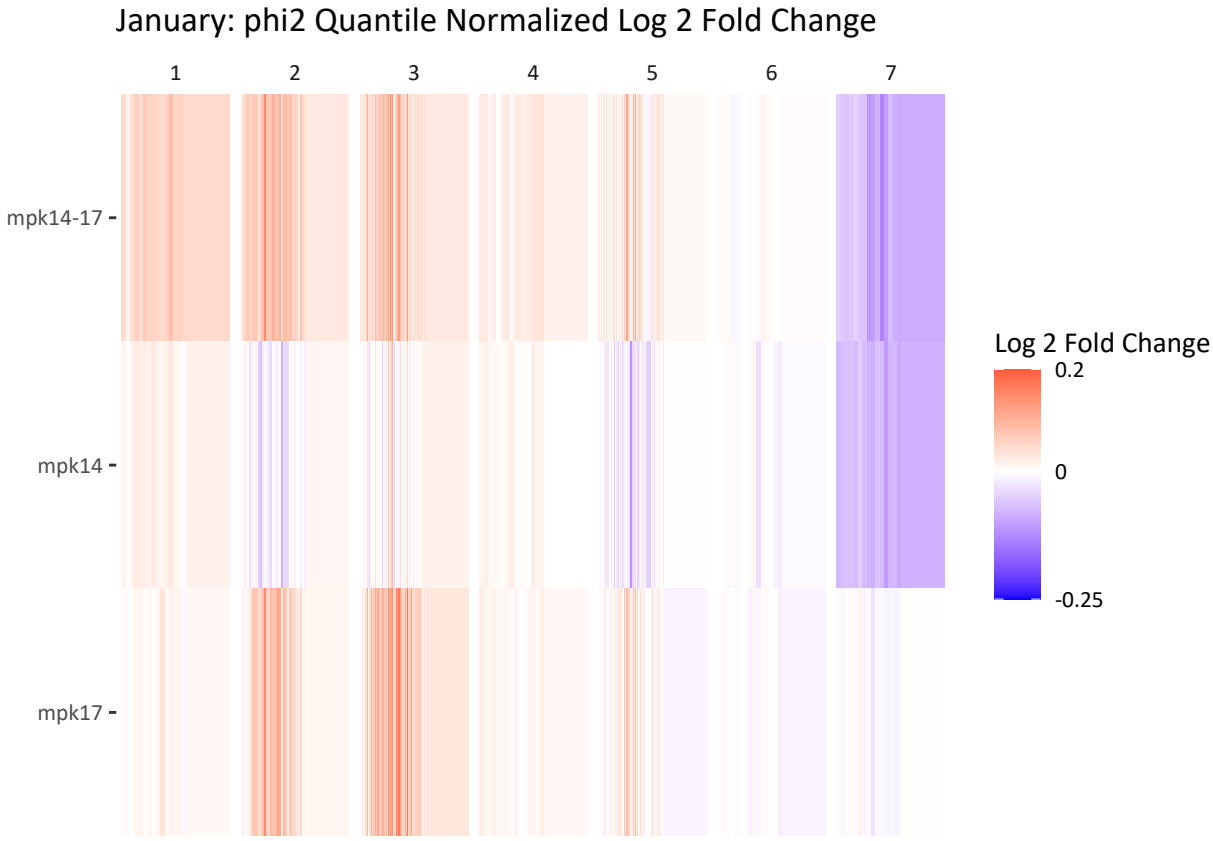
```
            midpoint = 0, limits = c(-0.25,
                0.2), breaks = c(-0.25, 0,
                0.2), labels = c("-0.25",
                "0", "0.2"))
    print(plot)
}
```

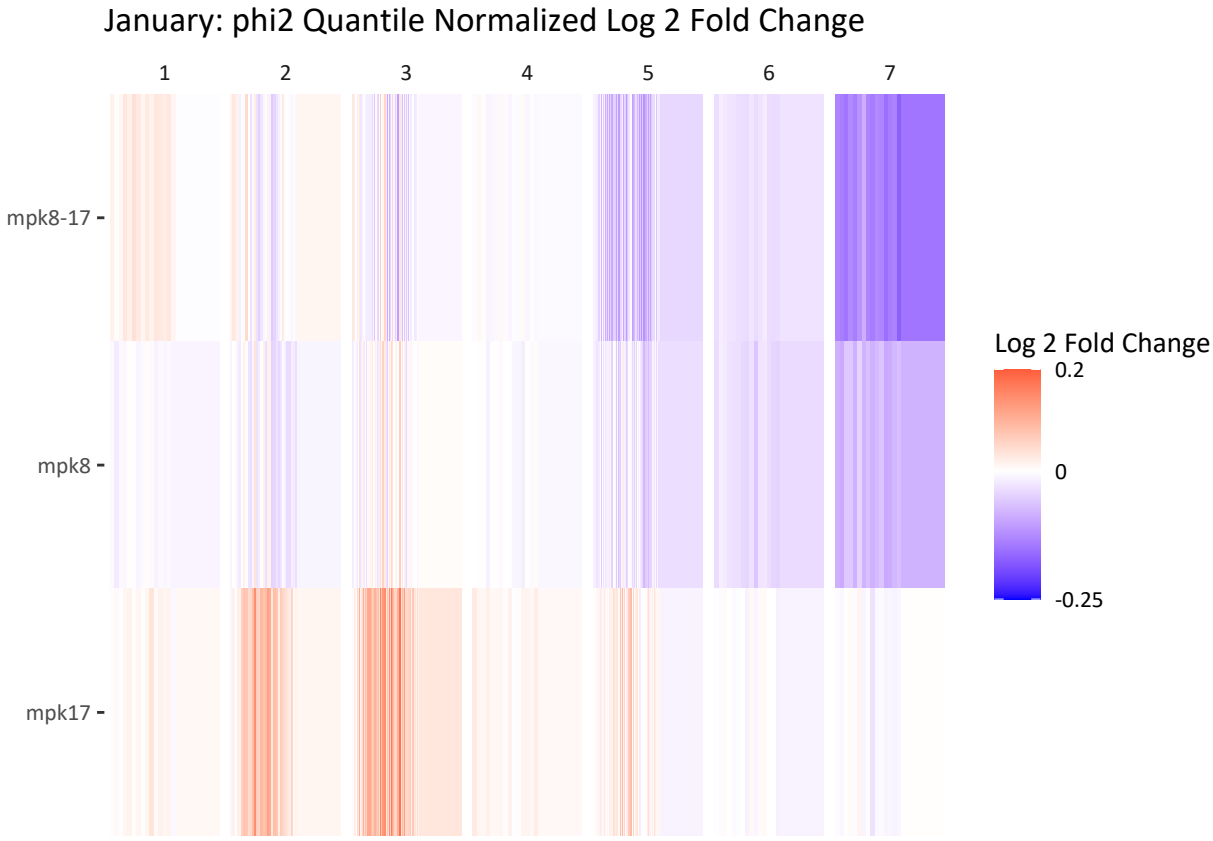


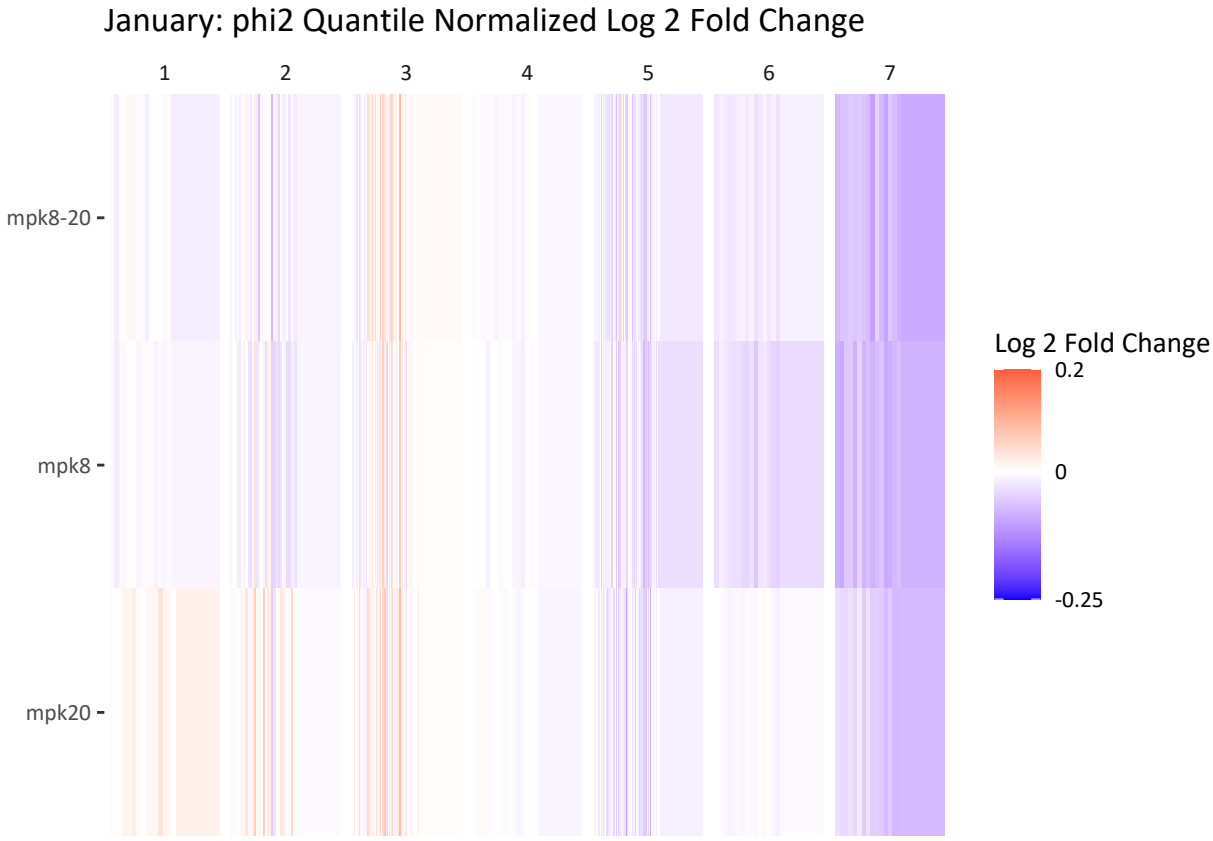

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-16
mpk1
mpk16
Log 2 Fold Change
0.2
0
-0.25

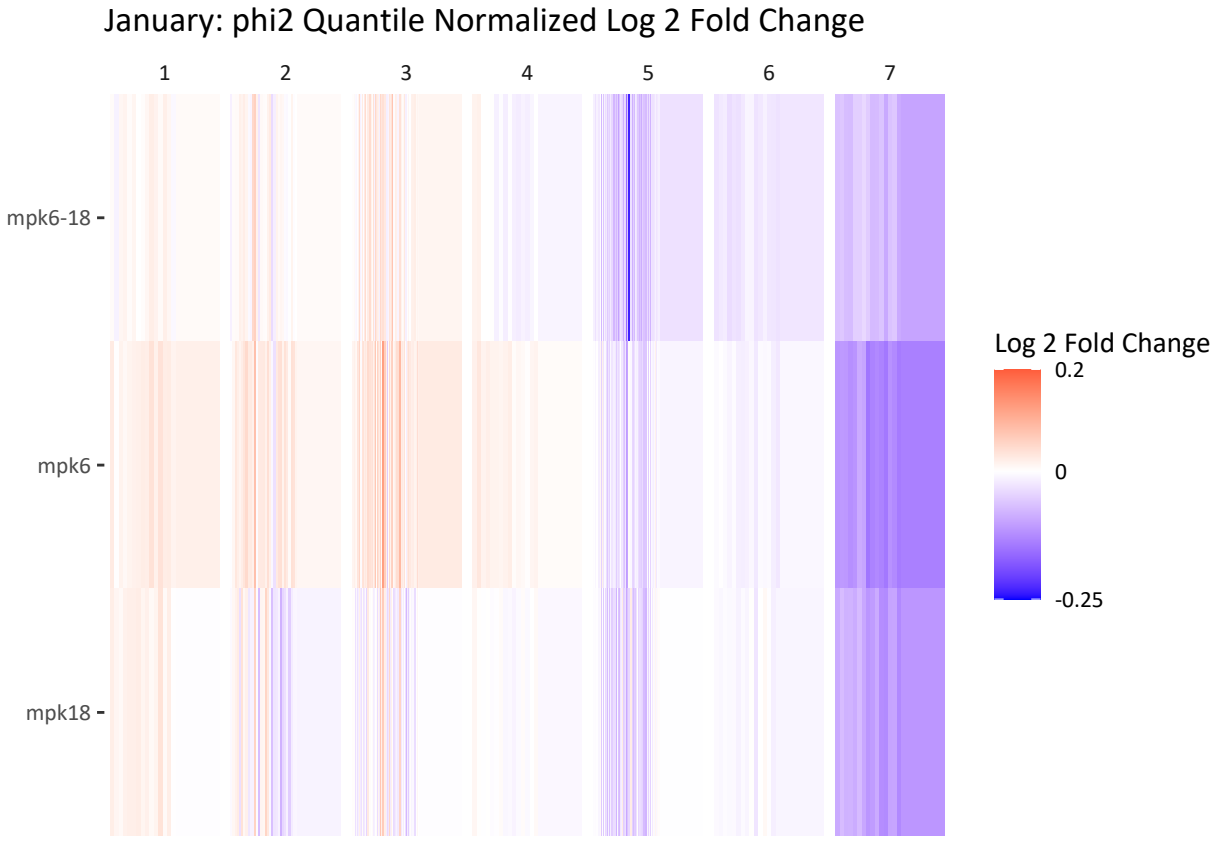

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-9
mpk6
mpk9
Log 2 Fold Change
0.2
0
-0.25

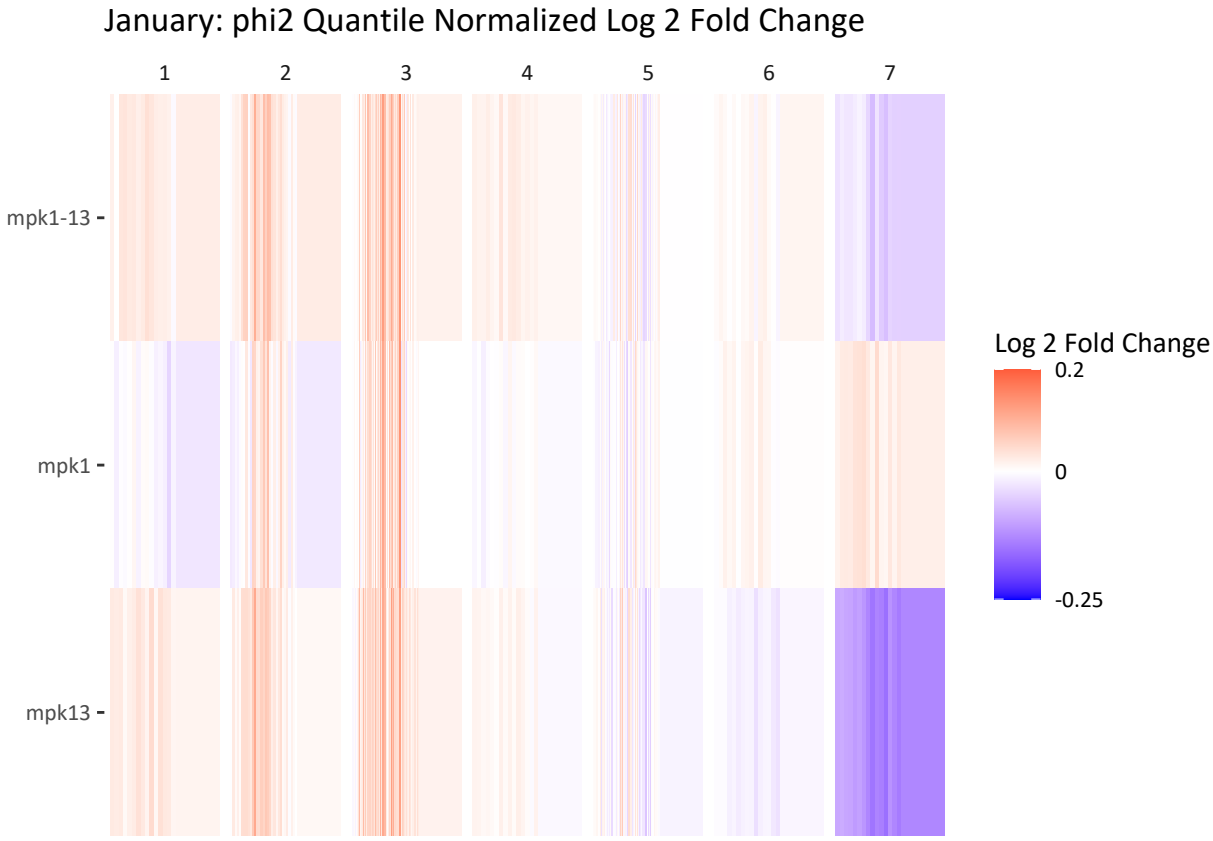

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.2
0
-0.25

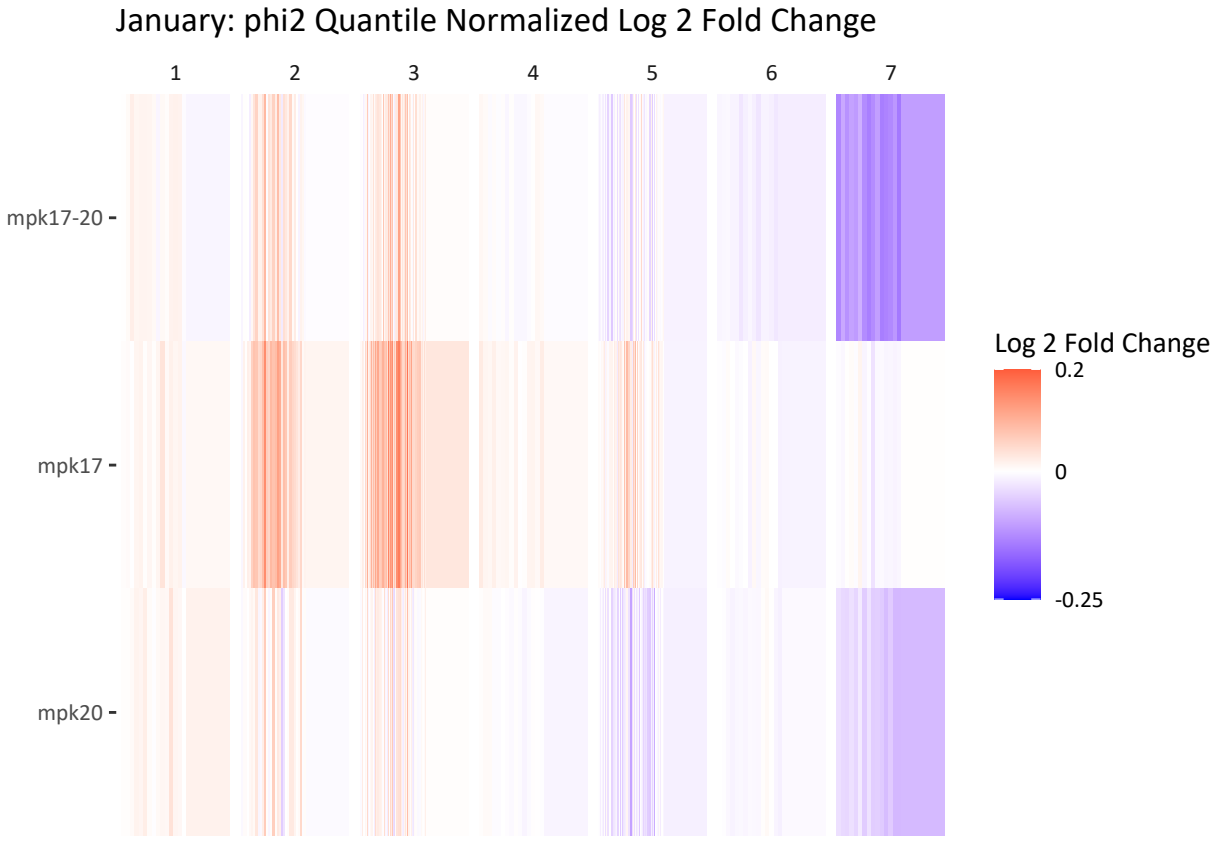

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-17
mpk14
mpk17
Log 2 Fold Change
0.2
0
-0.25

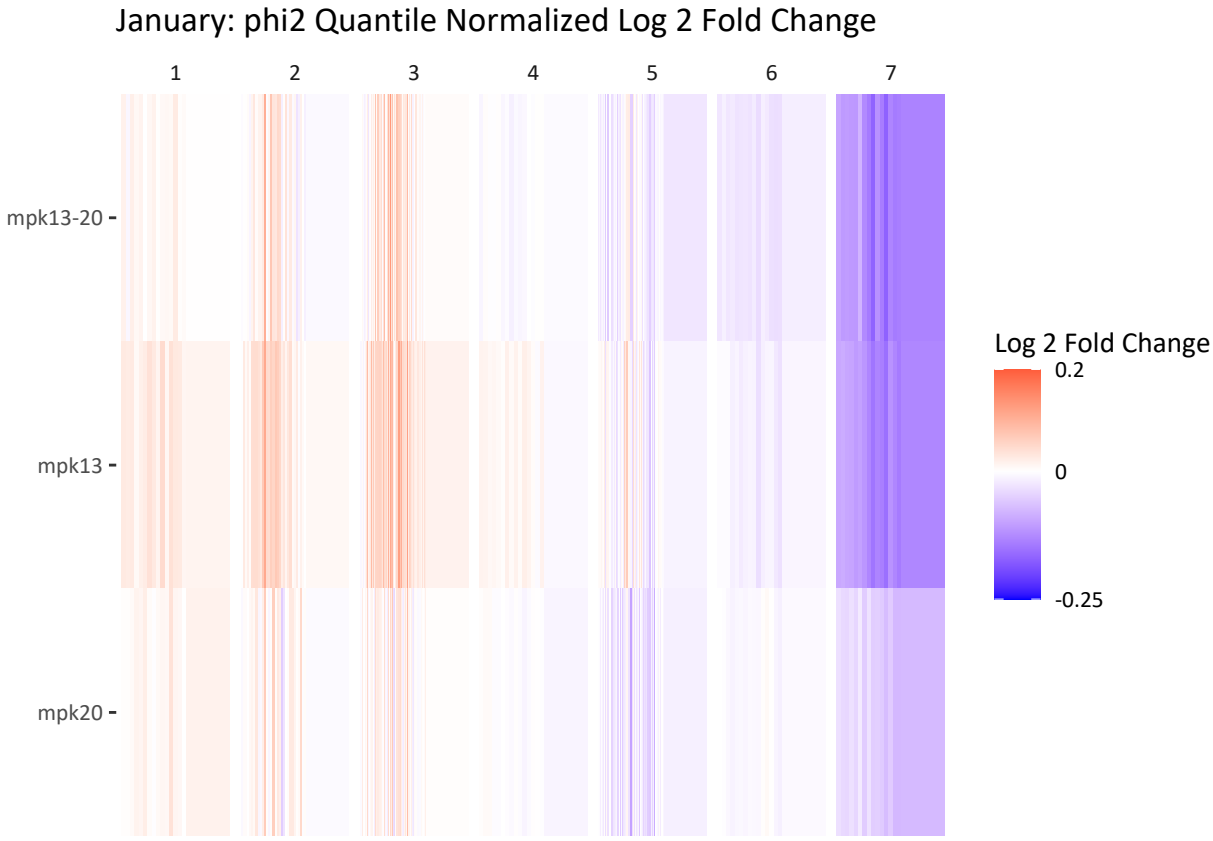

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-17
mpk8
mpk17
Log 2 Fold Change
0.2
0
-0.25

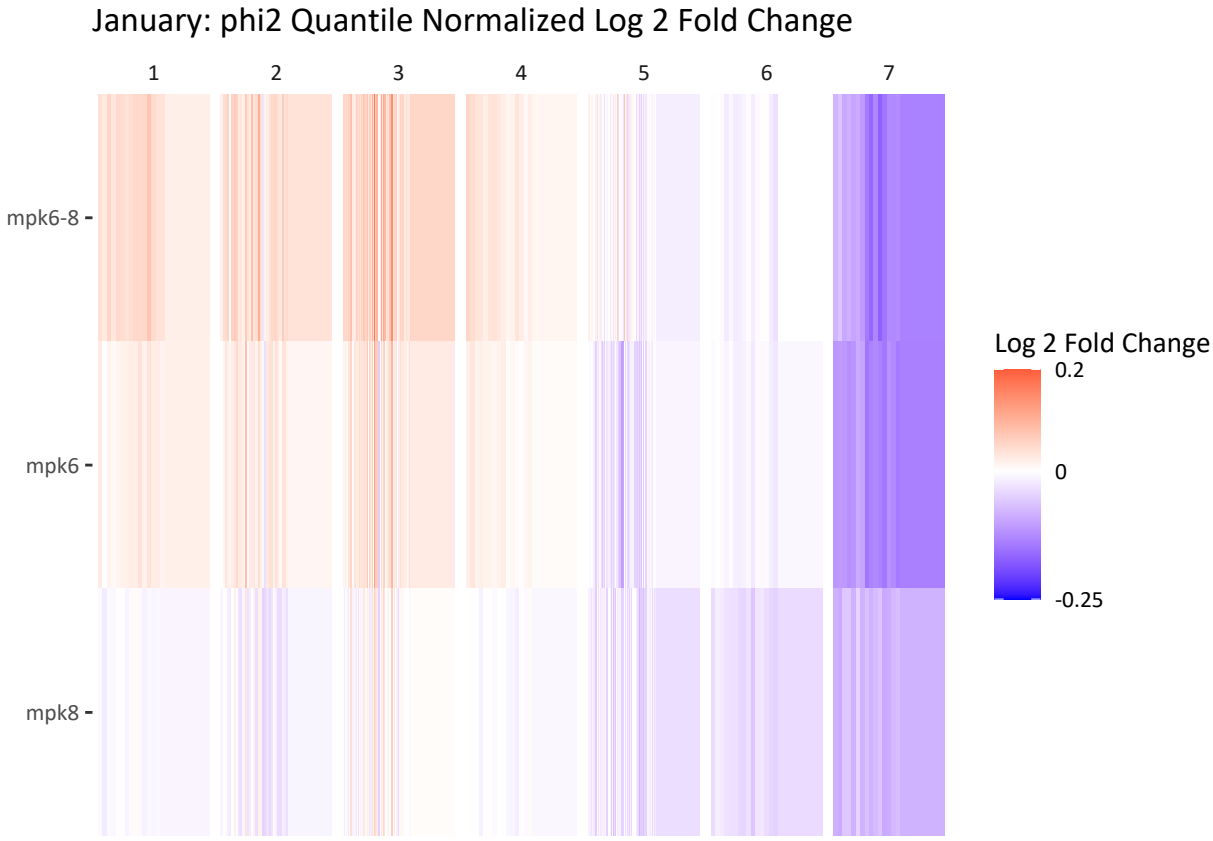

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-20
mpk8
mpk20
Log 2 Fold Change
0.2
0
-0.25

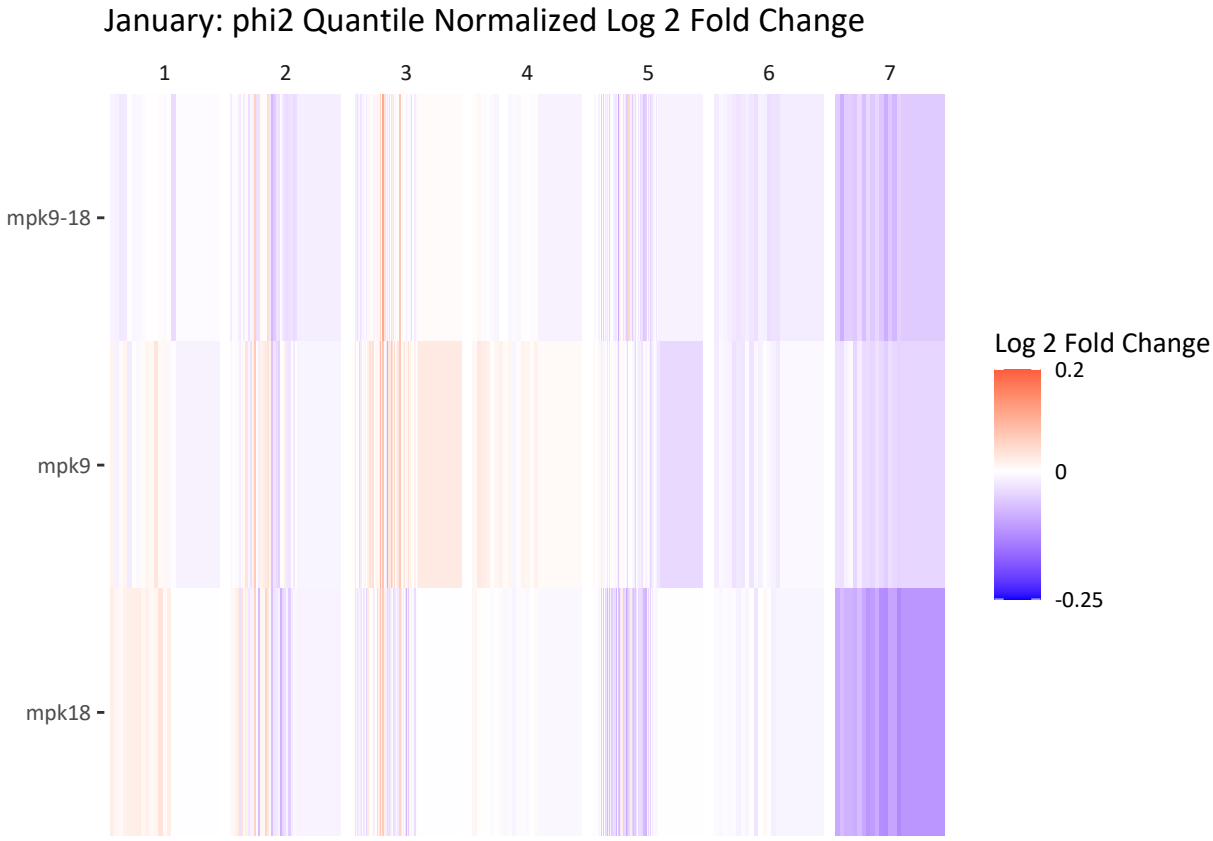

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-18
mpk6
mpk18
Log 2 Fold Change
0.2
0
-0.25

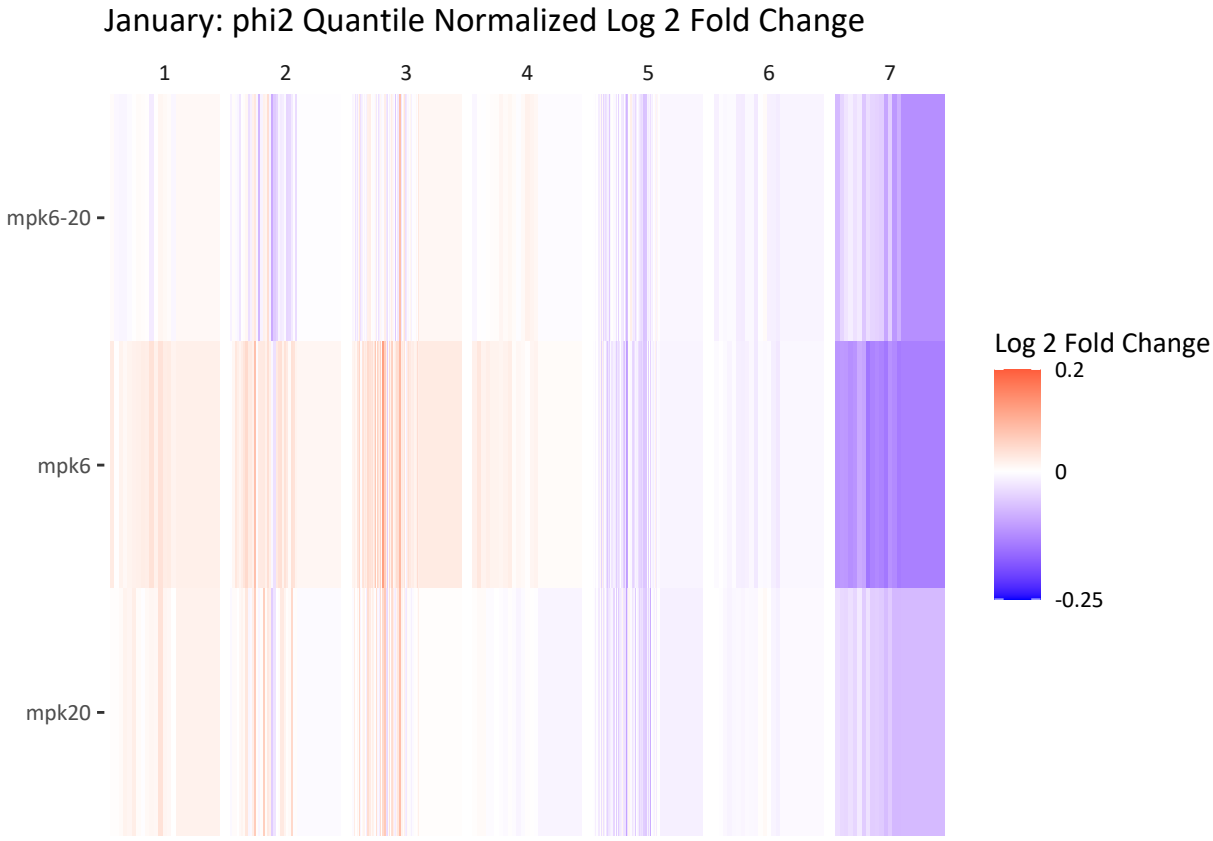

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-13
mpk1
mpk13
Log 2 Fold Change
0.2
0
-0.25

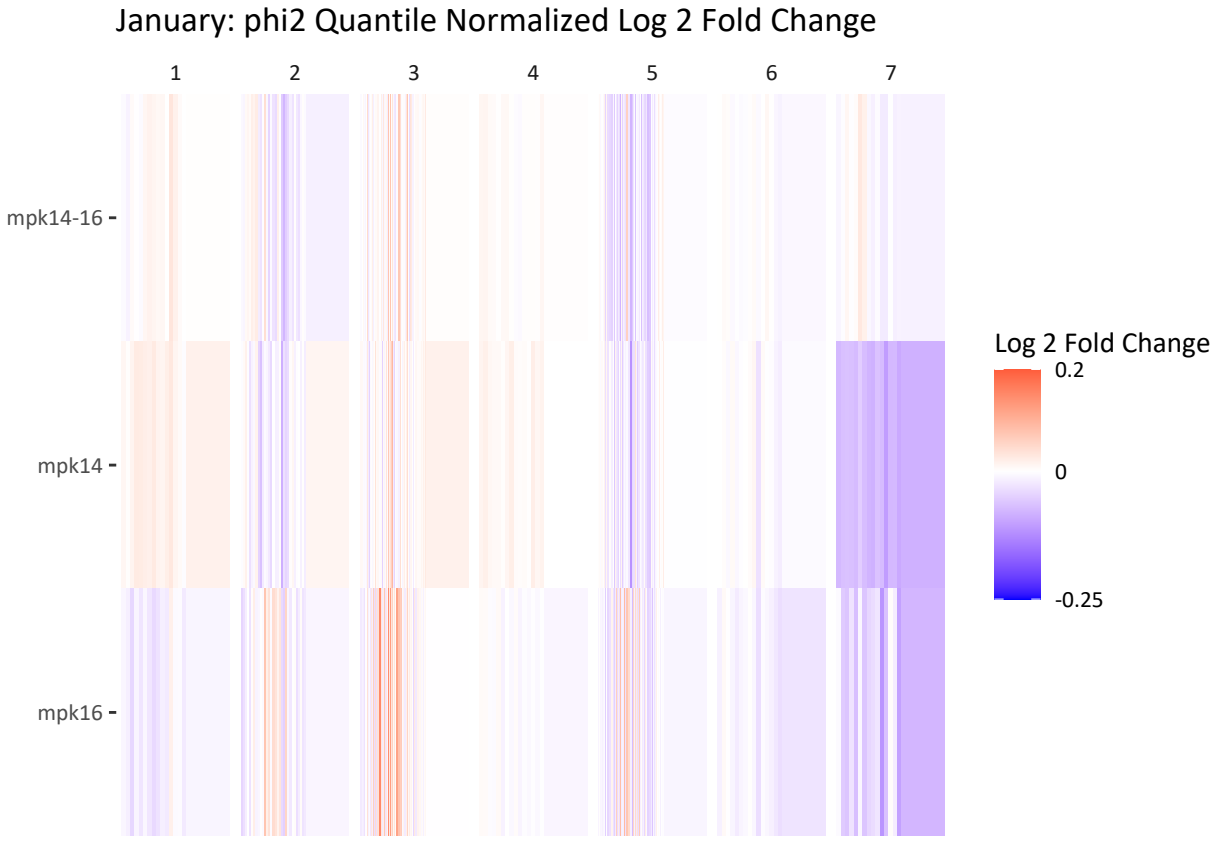

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.2
0
-0.25

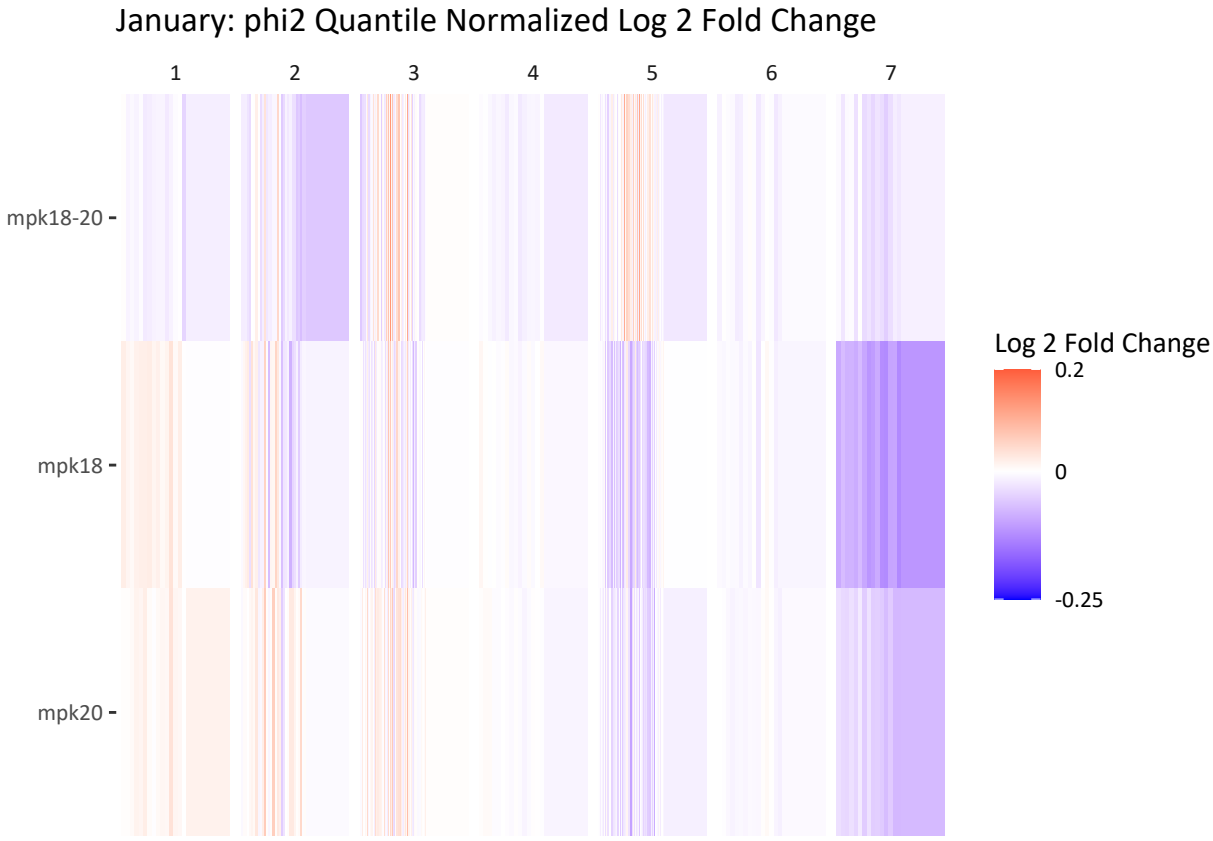

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk13-20
mpk13
mpk20
Log 2 Fold Change
0.2
0
-0.25

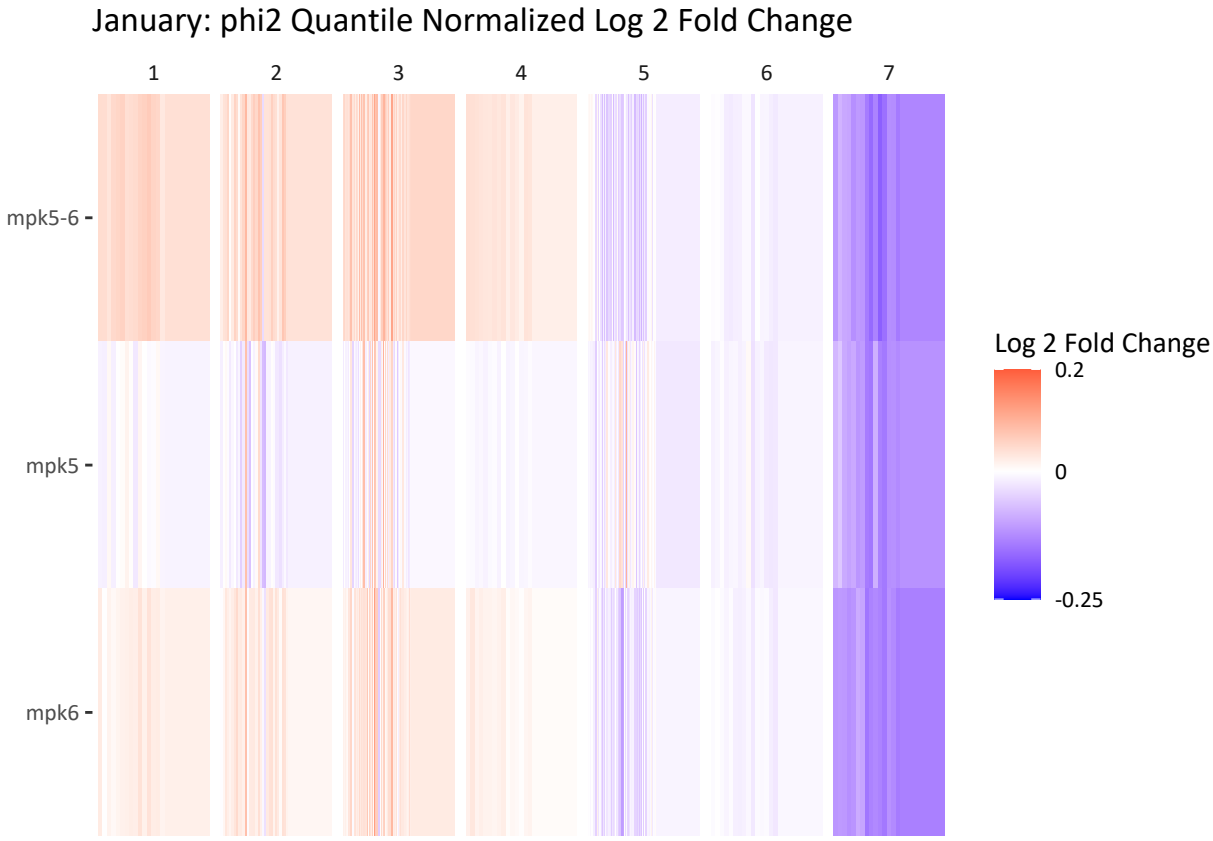

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-8
mpk6
mpk8
Log 2 Fold Change
0.2
0
-0.25

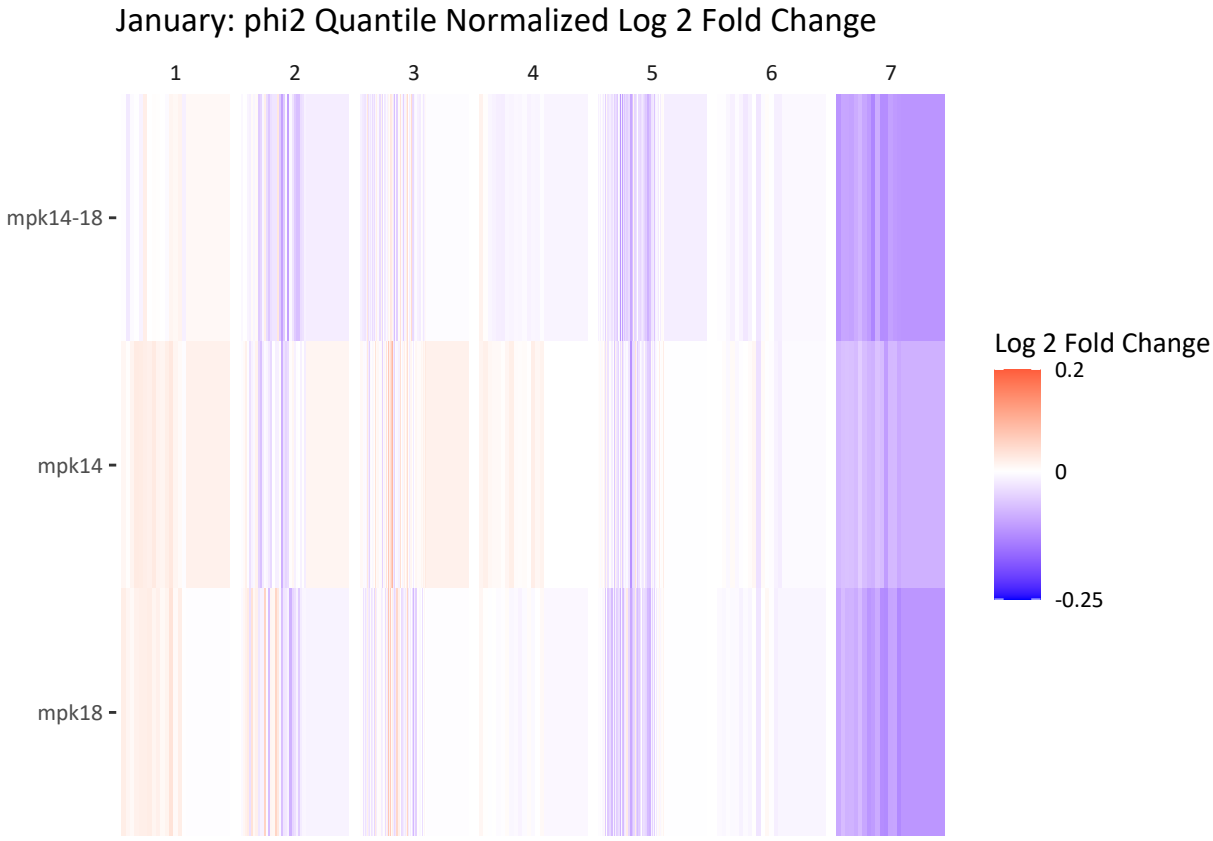

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-18
mpk9
mpk18
Log 2 Fold Change
0.2
0
-0.25

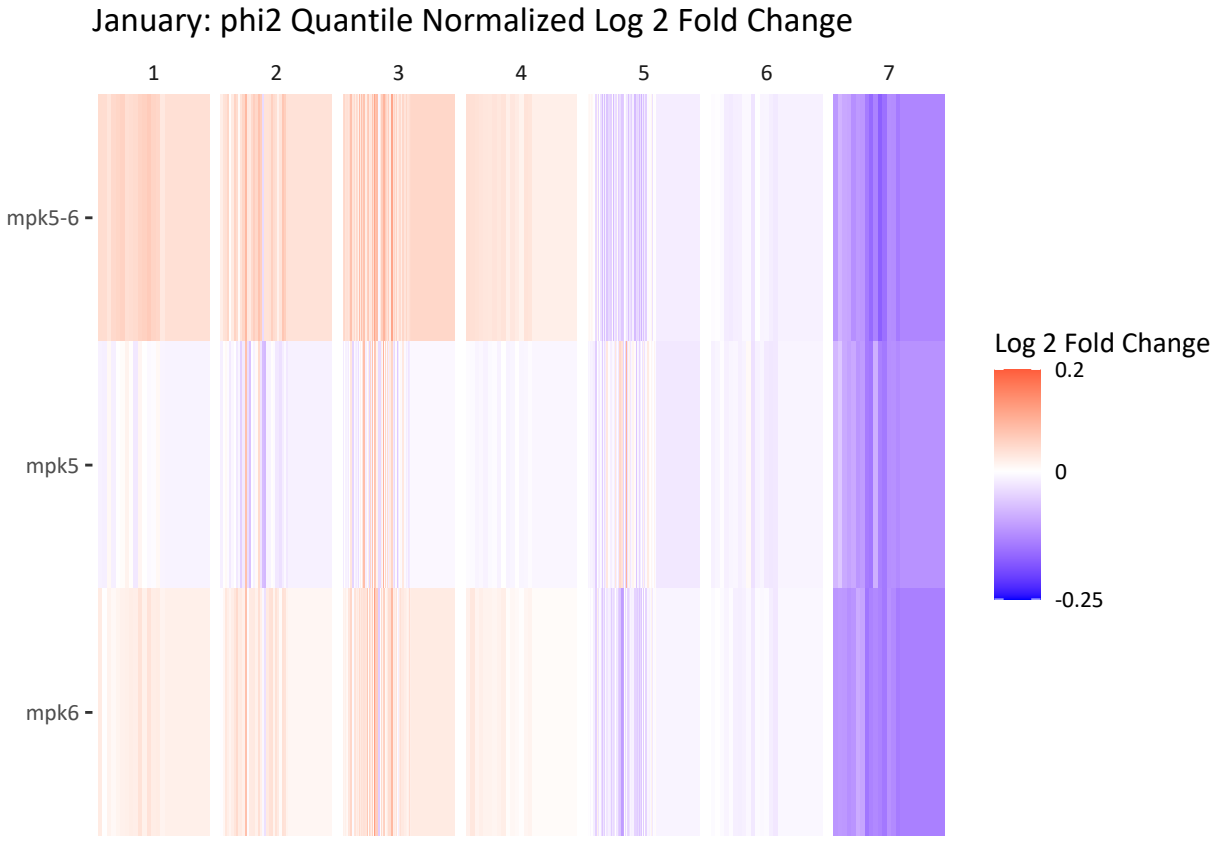

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-20
mpk6
mpk20
Log 2 Fold Change
0.2
0
-0.25

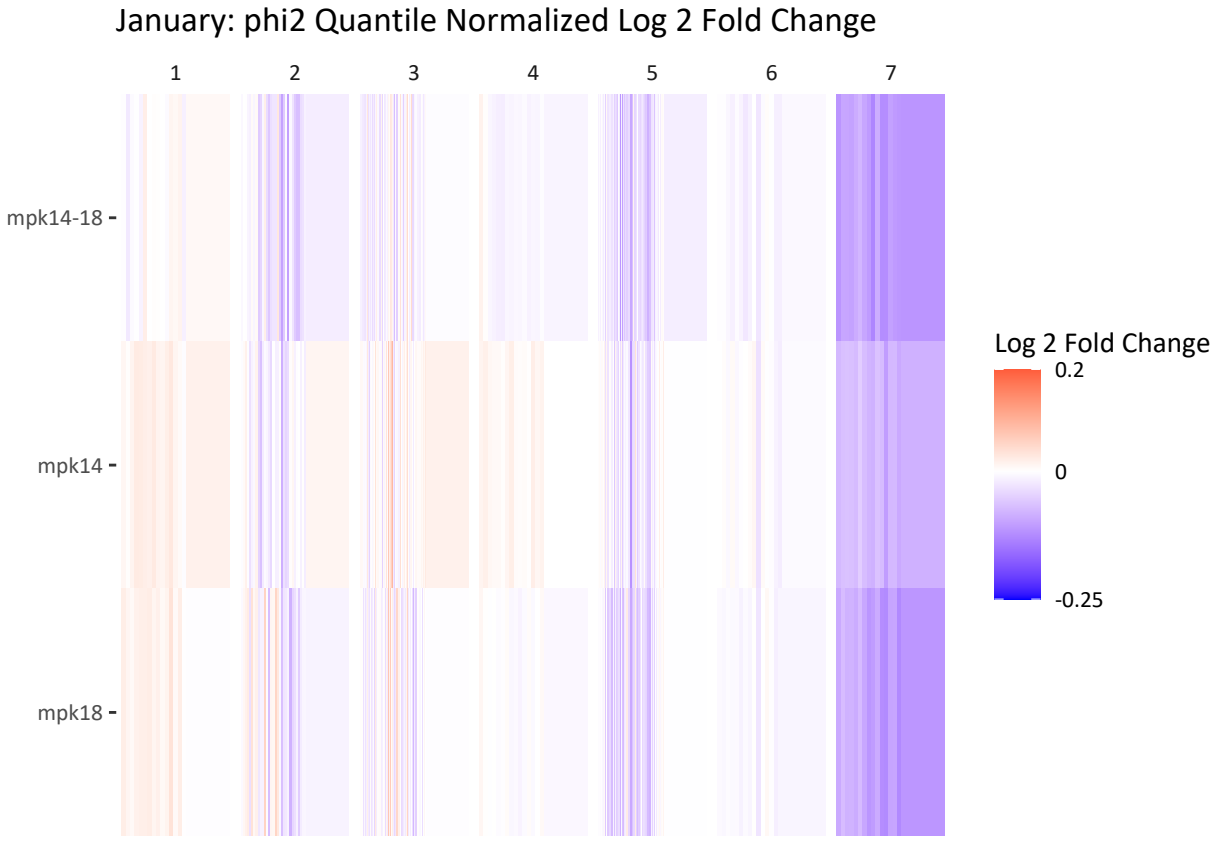

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-16
mpk14
mpk16
Log 2 Fold Change
0.2
0
-0.25

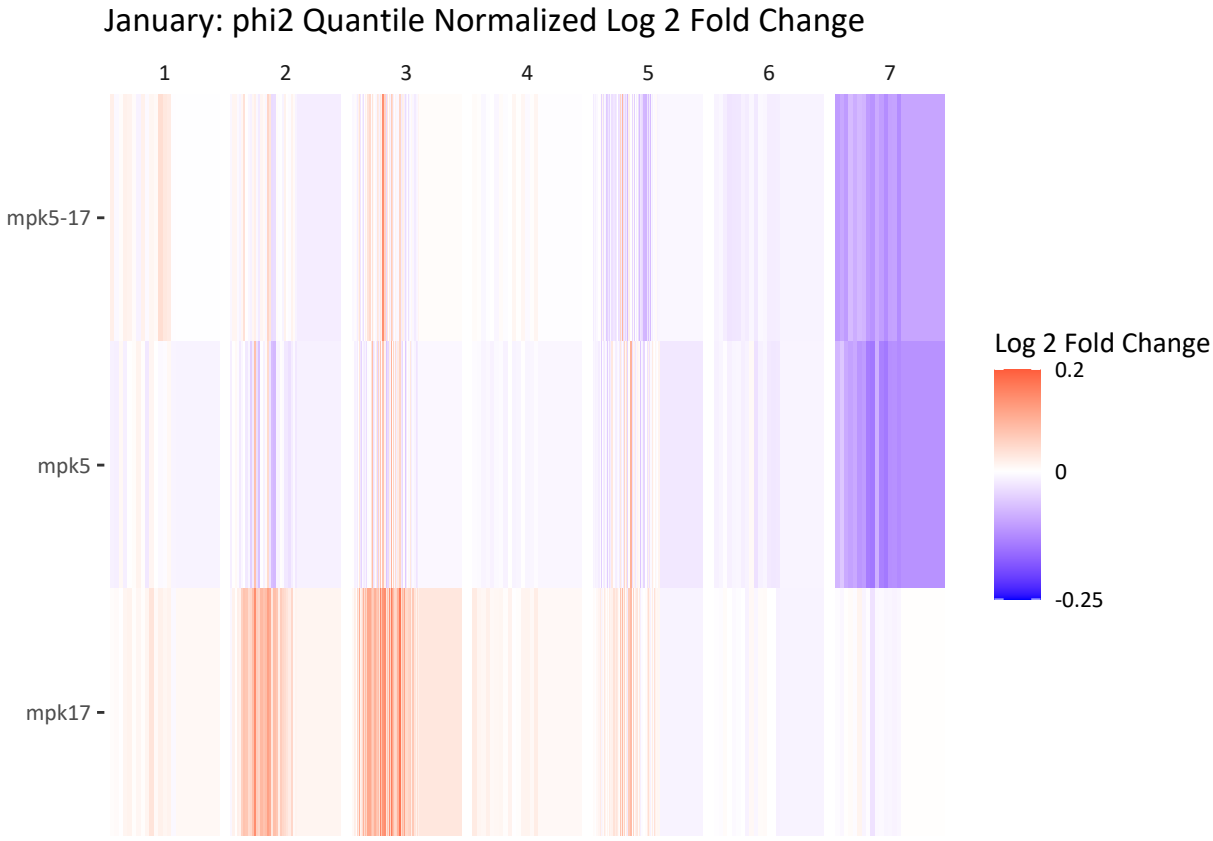

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk18-20
mpk18
mpk20
Log 2 Fold Change
0.2
0
-0.25

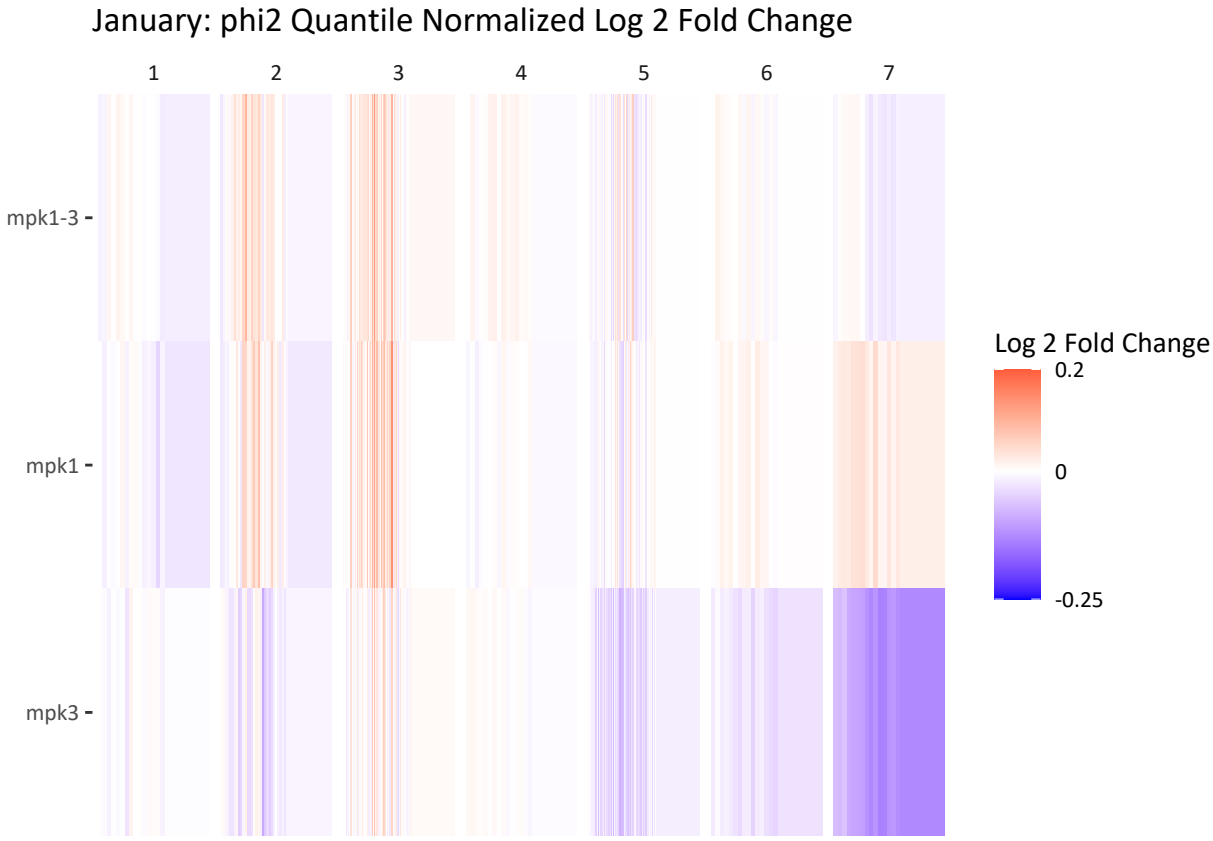

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.2
0
-0.25

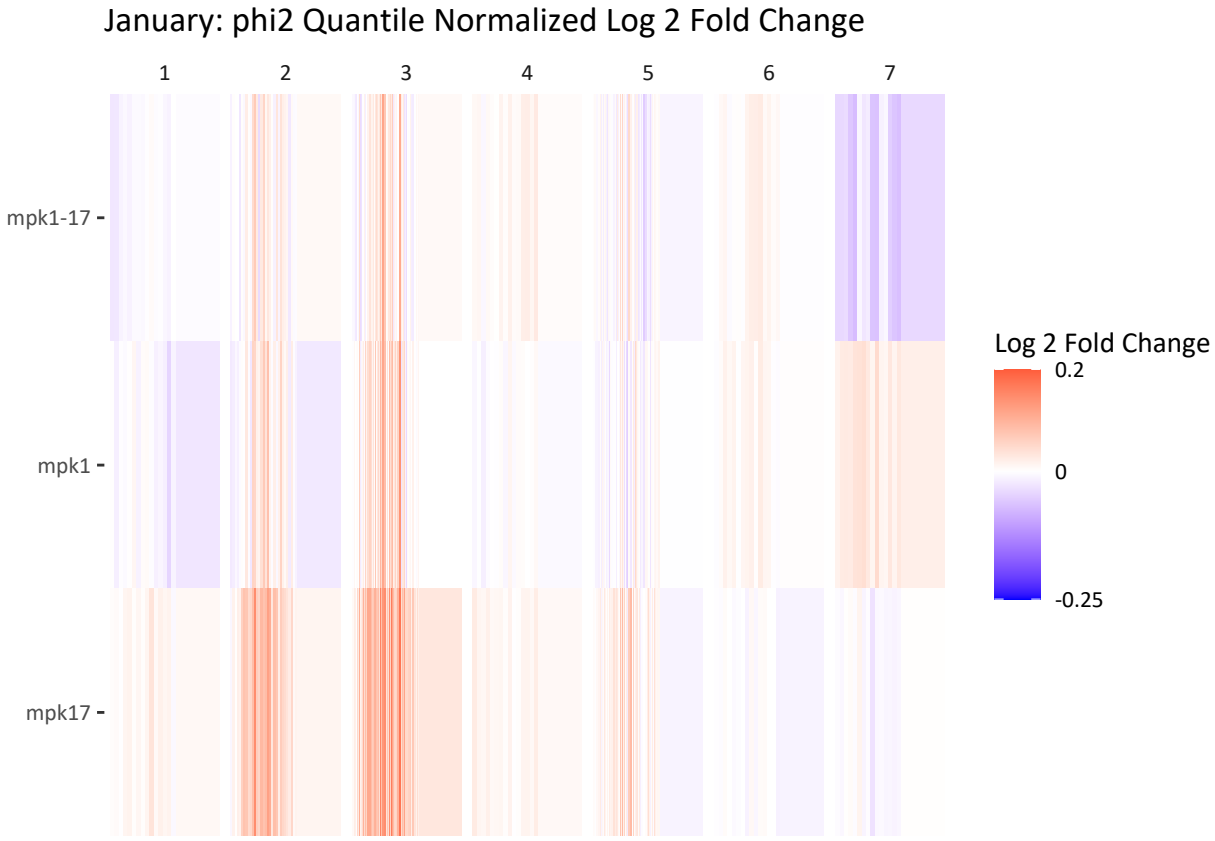

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.2
0
-0.25

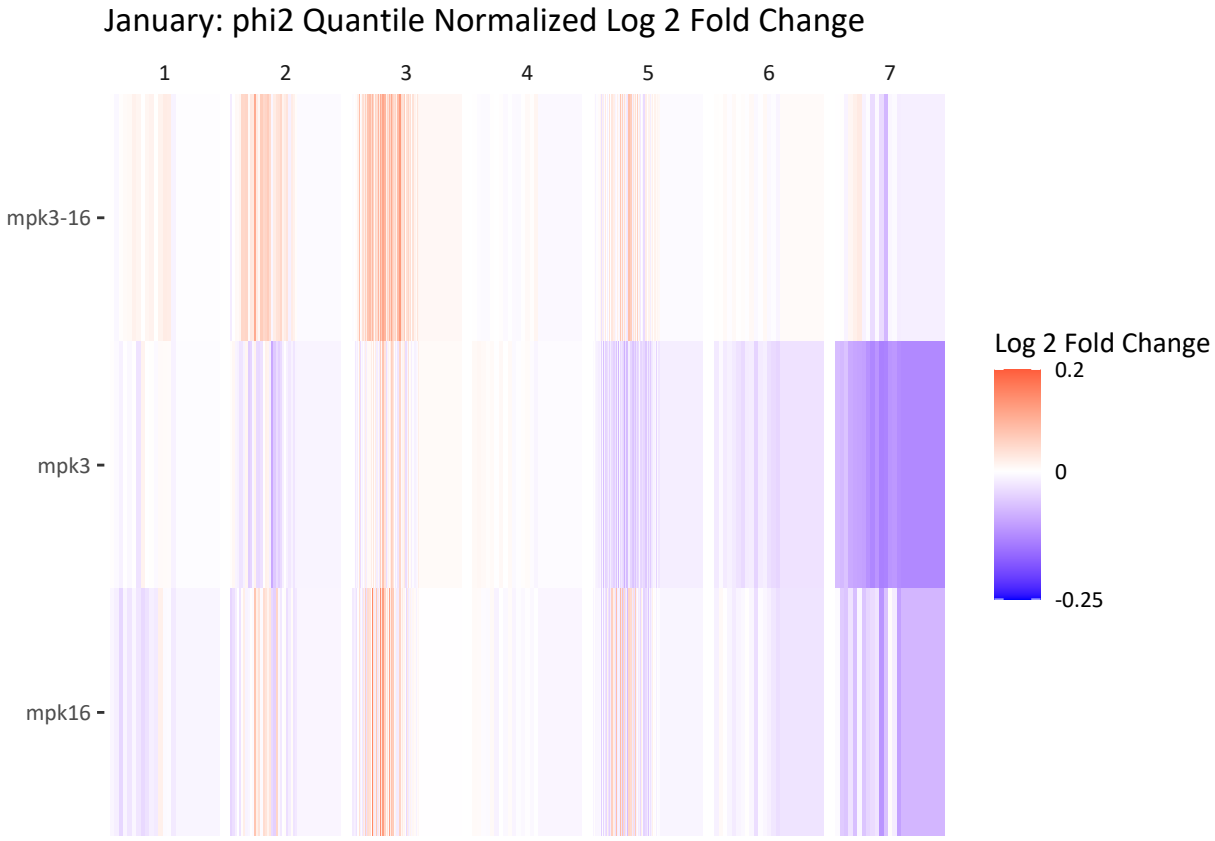

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.2
0
-0.25

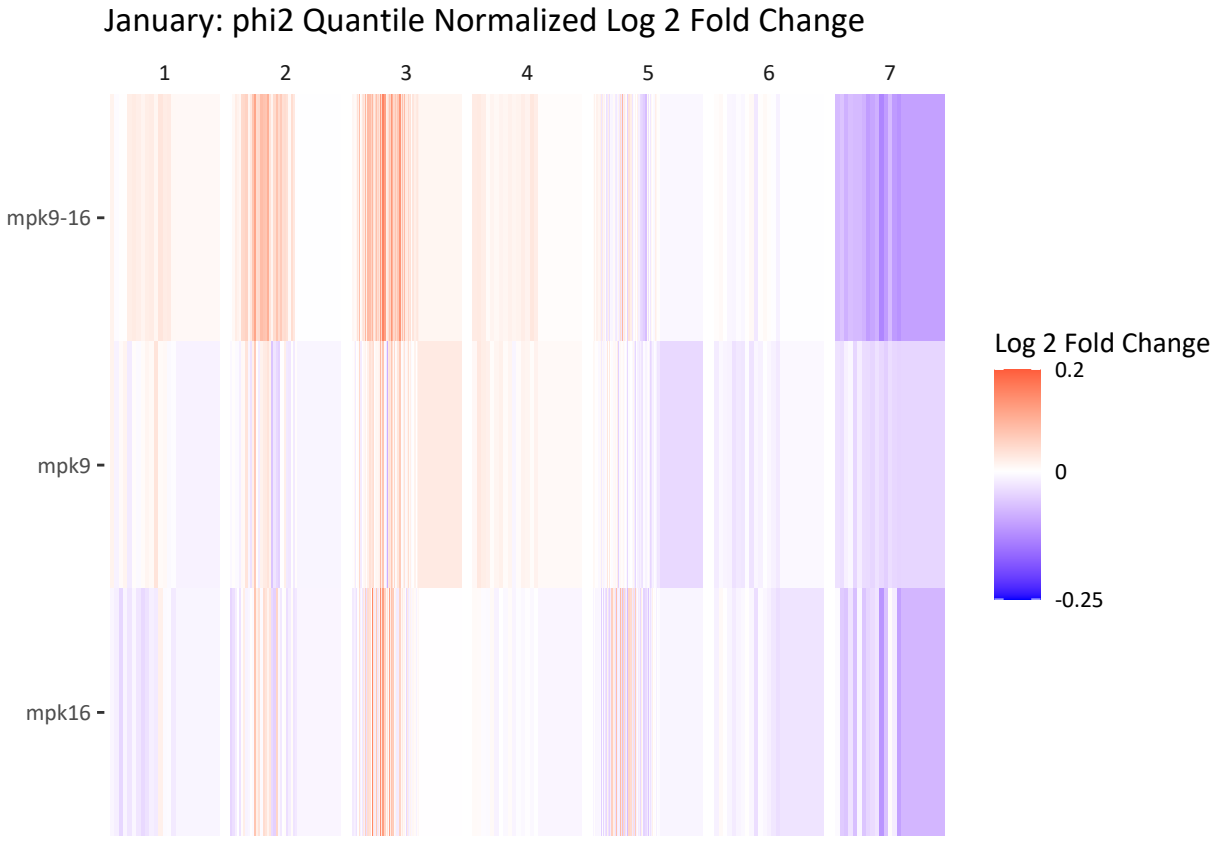

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.2
0
-0.25

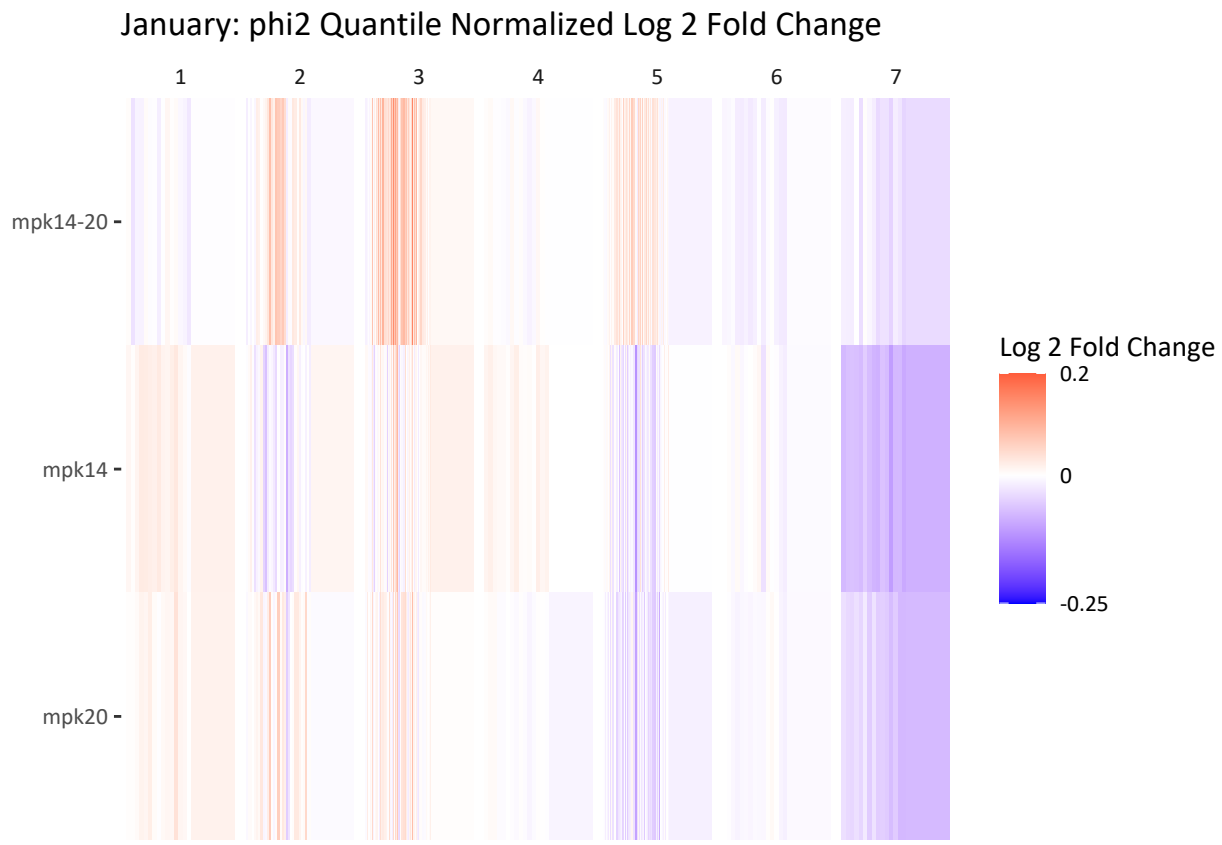

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-17
mpk5
mpk17
Log 2 Fold Change
0.2
0
-0.25


January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-3
mpk1
mpk3
Log 2 Fold Change
0.2
0
-0.25


January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-17
mpk1
mpk17
Log 2 Fold Change
0.2
0
-0.25


January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk3-16
mpk3
mpk16
Log 2 Fold Change
0.2
0
-0.25

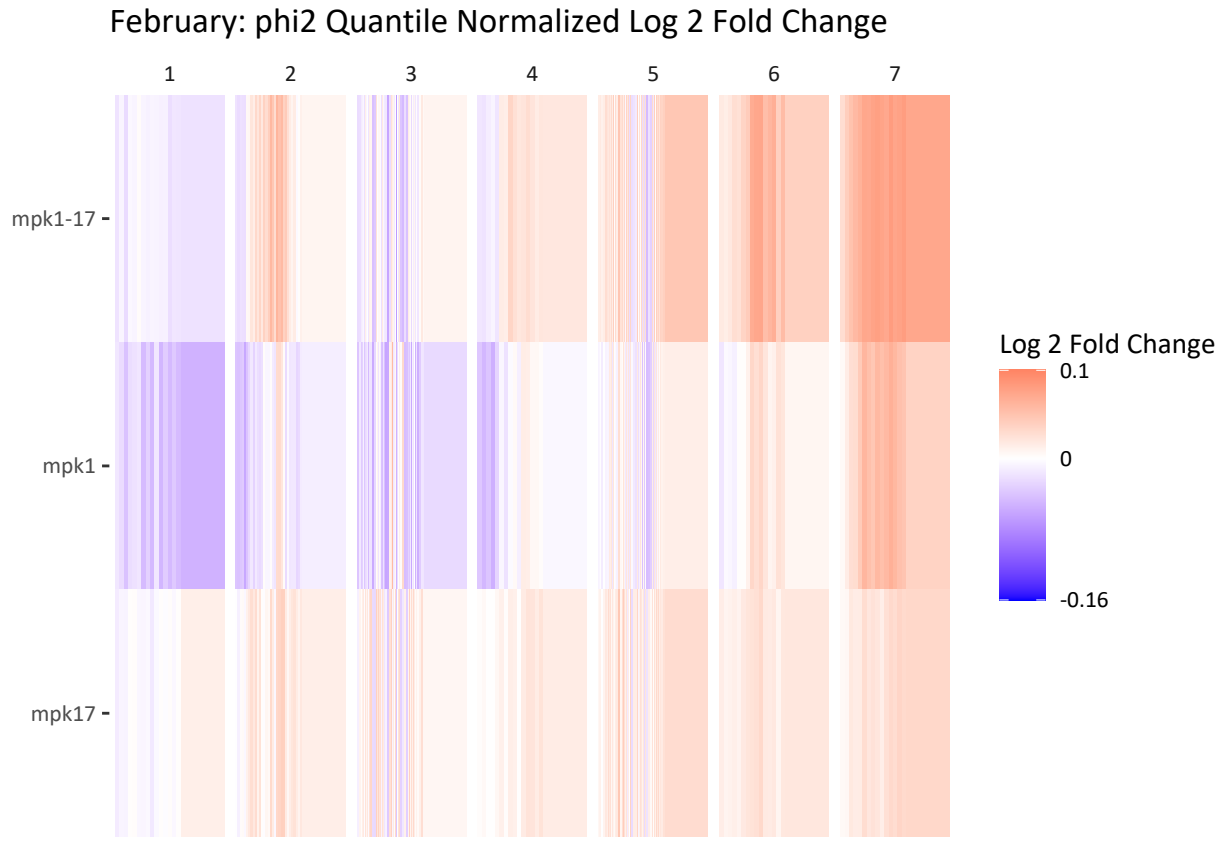

January: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-16
mpk9
mpk16
Log 2 Fold Change
0.2
0
-0.25

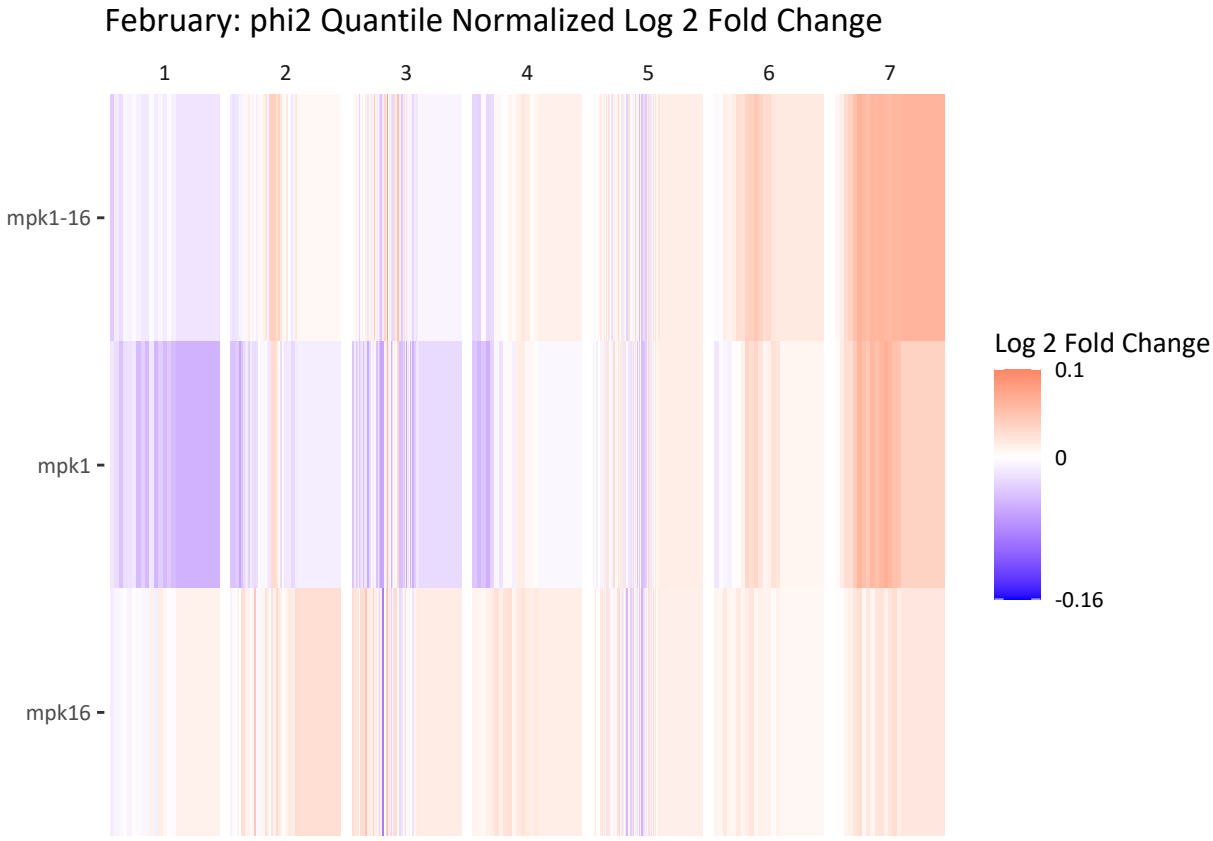


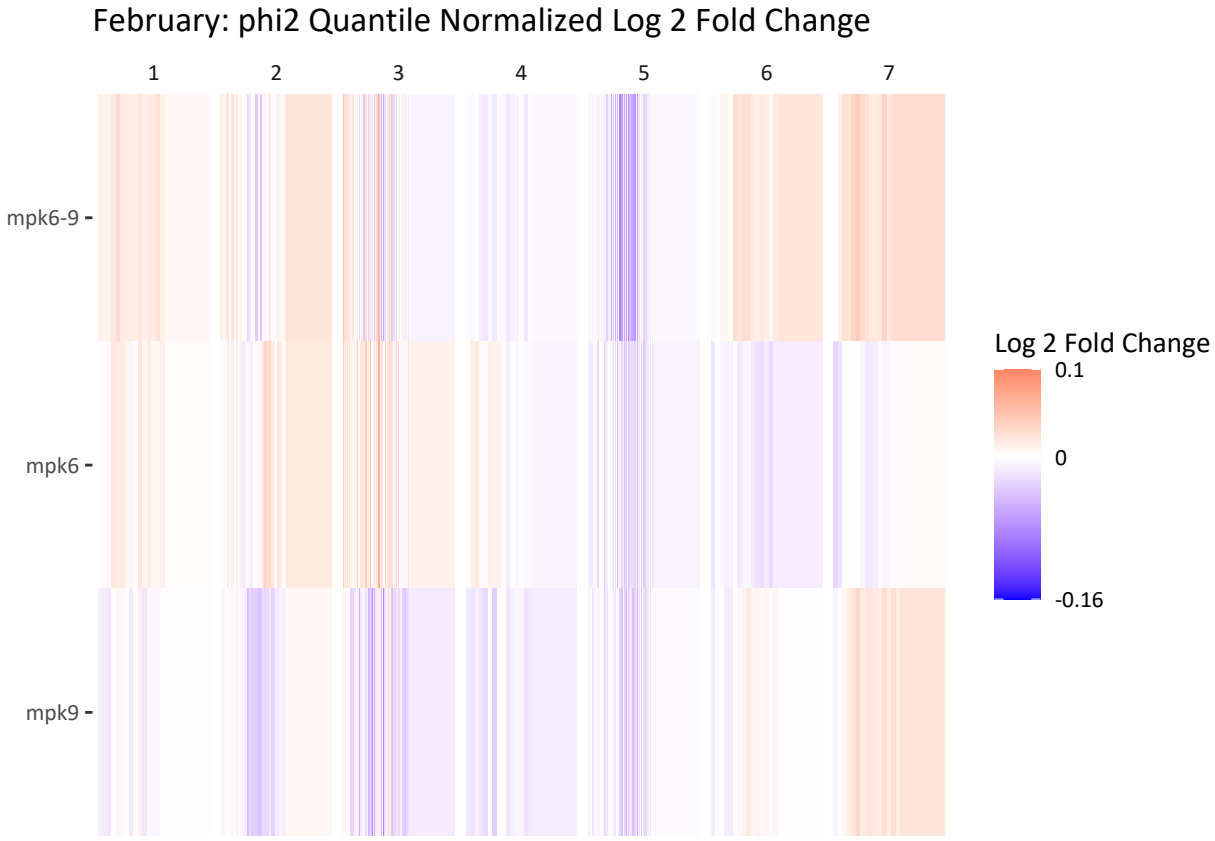
February Loop

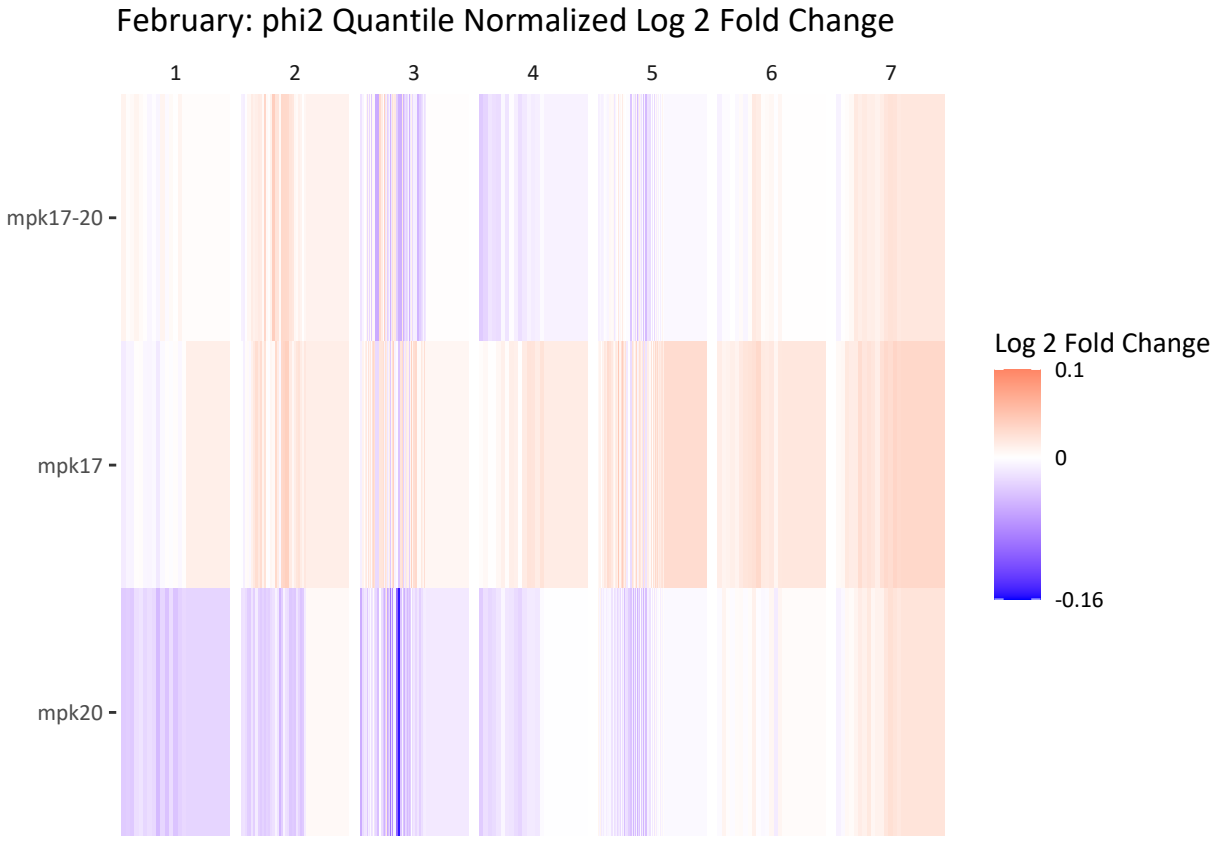
```
for (element in genotype_combinations) {
    m <- "phi2"
    data <- filter(cell_371_feb, genotype %in%
        element, measurement == m)
    data$genotype <- reorder(data$genotype,
        data$number_2)
    plot <- ggplot(data = data, aes(x = time_point,
        y = genotype, fill = log2_fold)) +
        labs(fill = "Log 2 Fold Change",
            x = "Hours", y = NULL, title = paste("February:",
                m, "Quantile Normalized Log 2 Fold Change")) +
        geom_tile(width = ifelse(m == "leafarea",
            16, 10), height = 30) + facet_grid(genotype ~
        day, scales = "free", switch = "y") +
        theme_tufte(base_family = "Calibri") +
        theme(strip.background.y = element_blank(),
            strip.text.y = element_blank(),
            axis.title.x = element_blank(),
            axis.text.x = element_blank(),
            axis.ticks.x = element_blank(),
            panel.spacing = unit(0, "lines")) +
        scale_fill_gradient2(low = "blue",
```

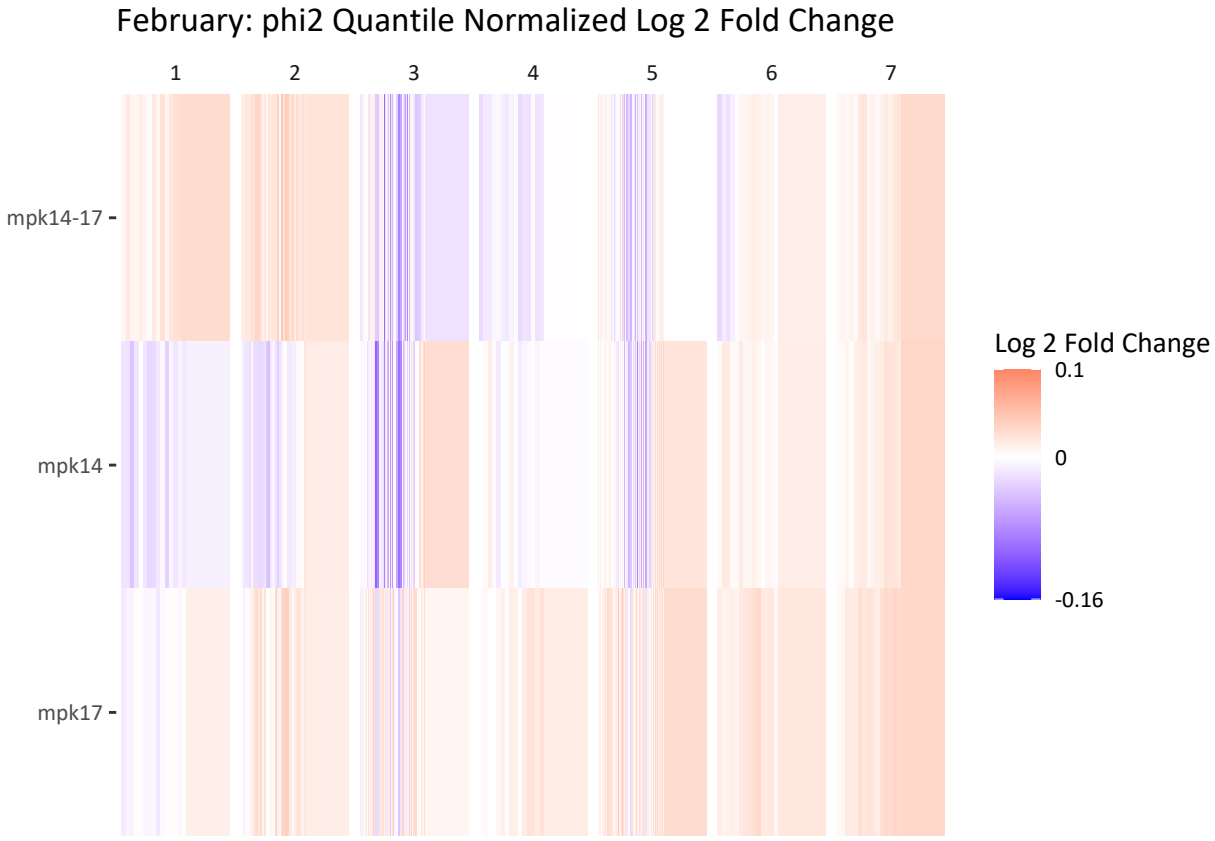
```
            high = "red", mid = "white",
            midpoint = 0, limits = c(-0.16,
                0.1), breaks = c(-0.16, 0,
                0.1), labels = c("-0.16",
                "0", "0.1"))
    print(plot)
}
```

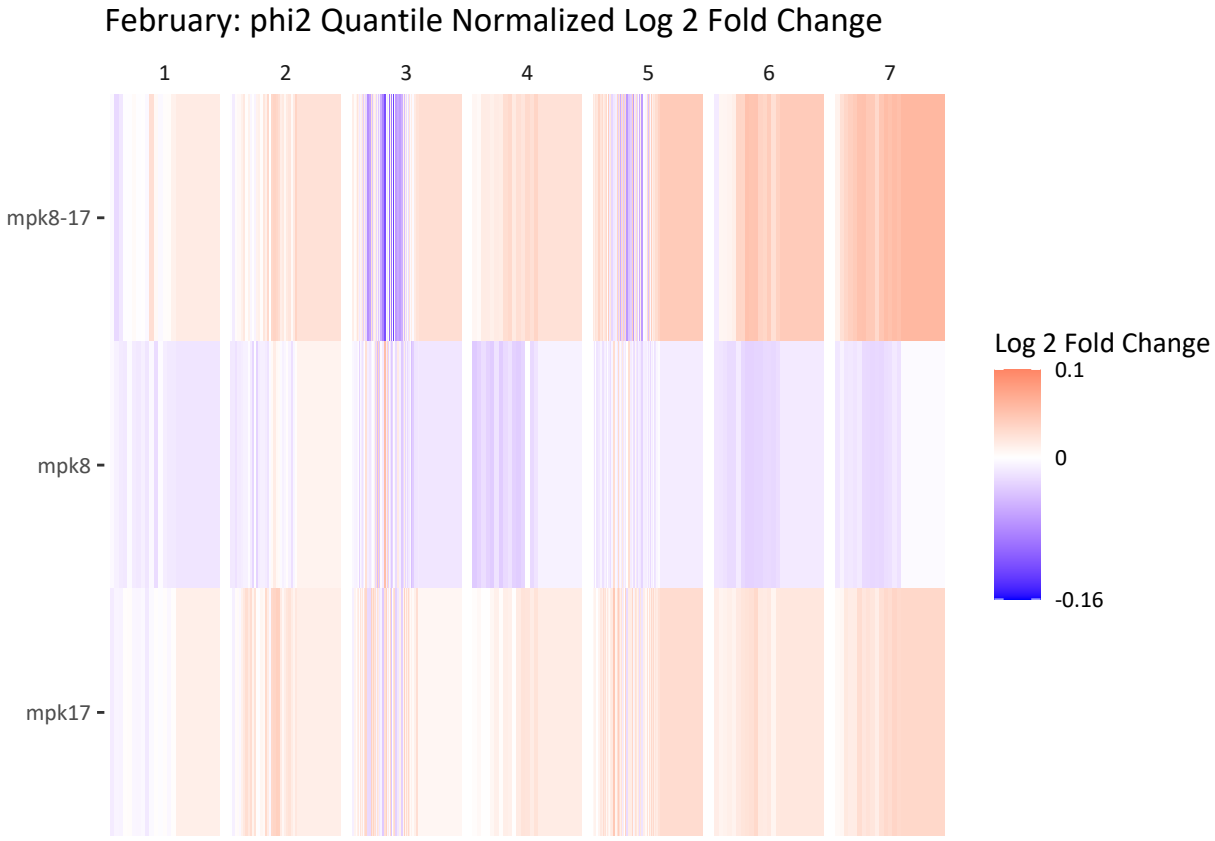


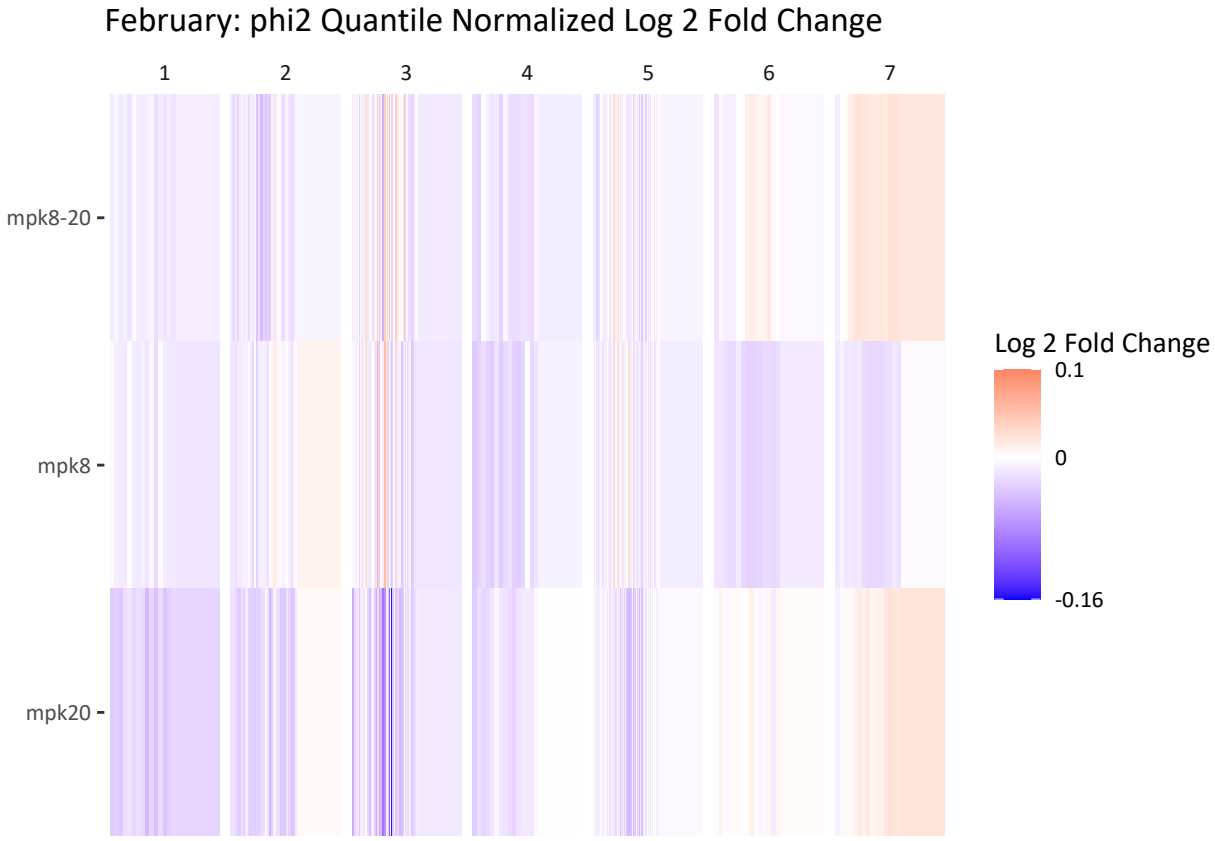

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-16
mpk1
mpk16
Log 2 Fold Change
0.1
0
-0.16

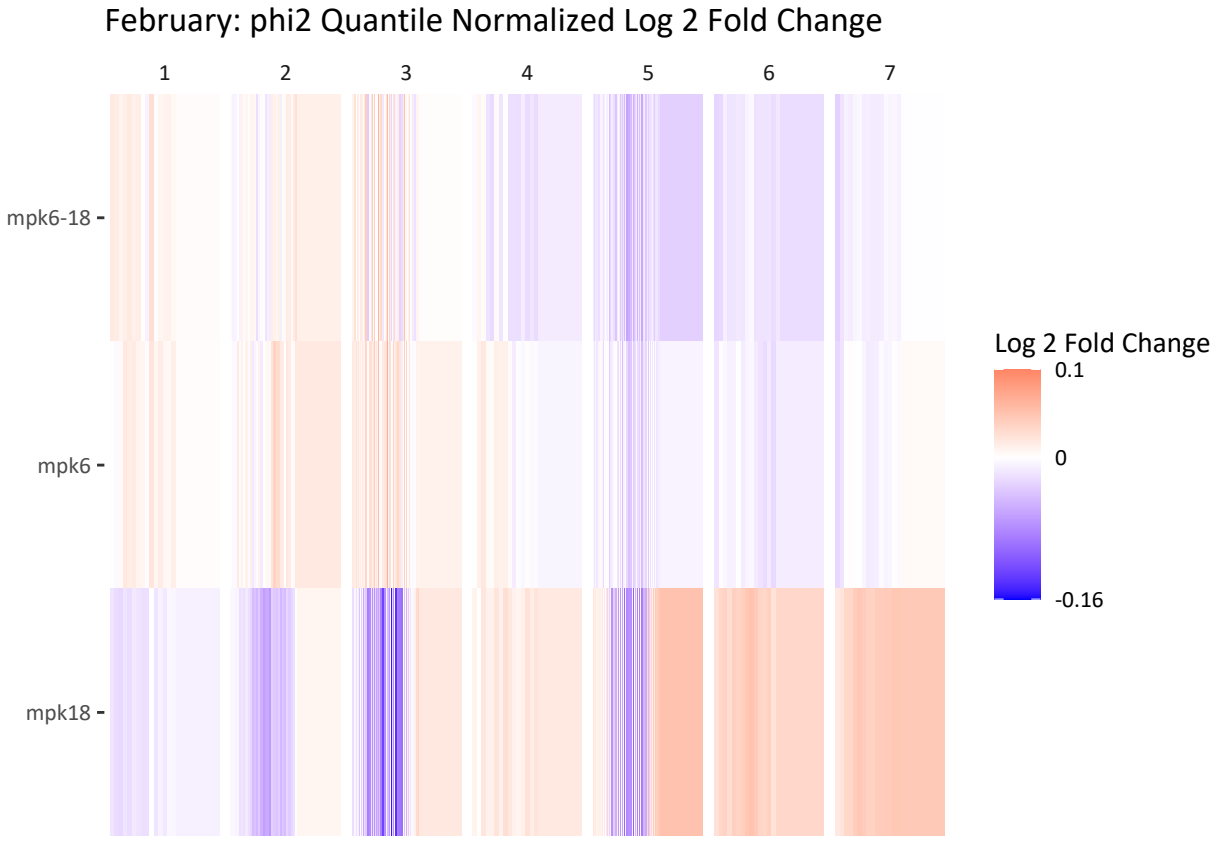

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-9
mpk6
mpk9
Log 2 Fold Change
0.1
0
-0.16

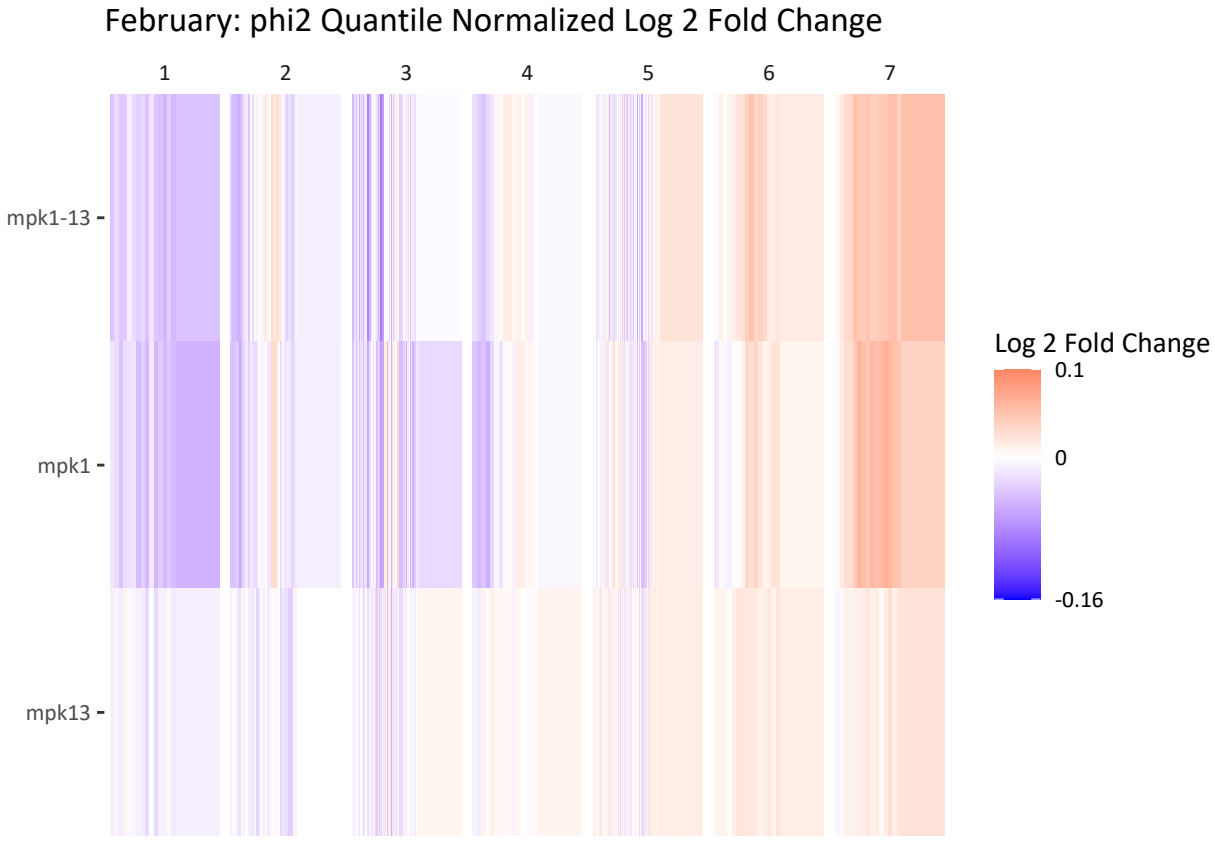

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.1
0
-0.16

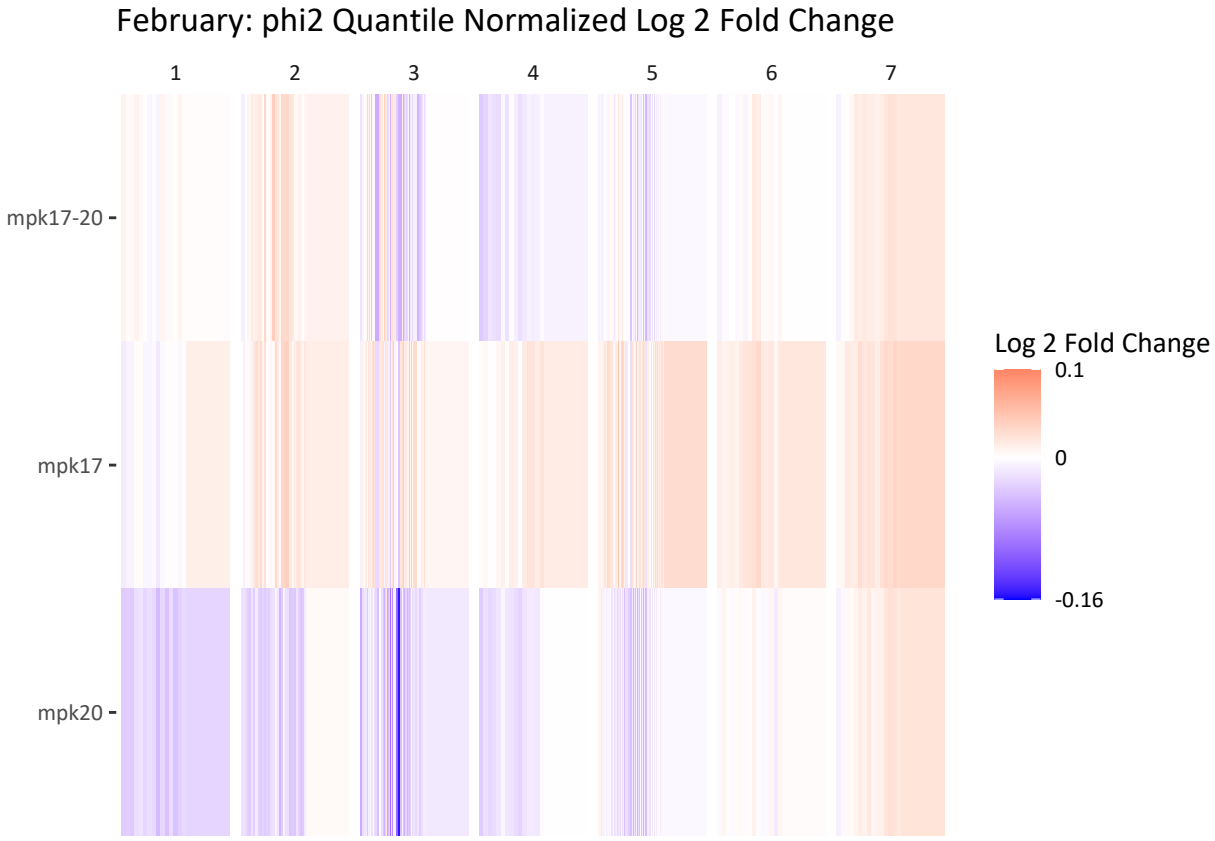

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-17
mpk14
mpk17
Log 2 Fold Change
0.1
0
-0.16

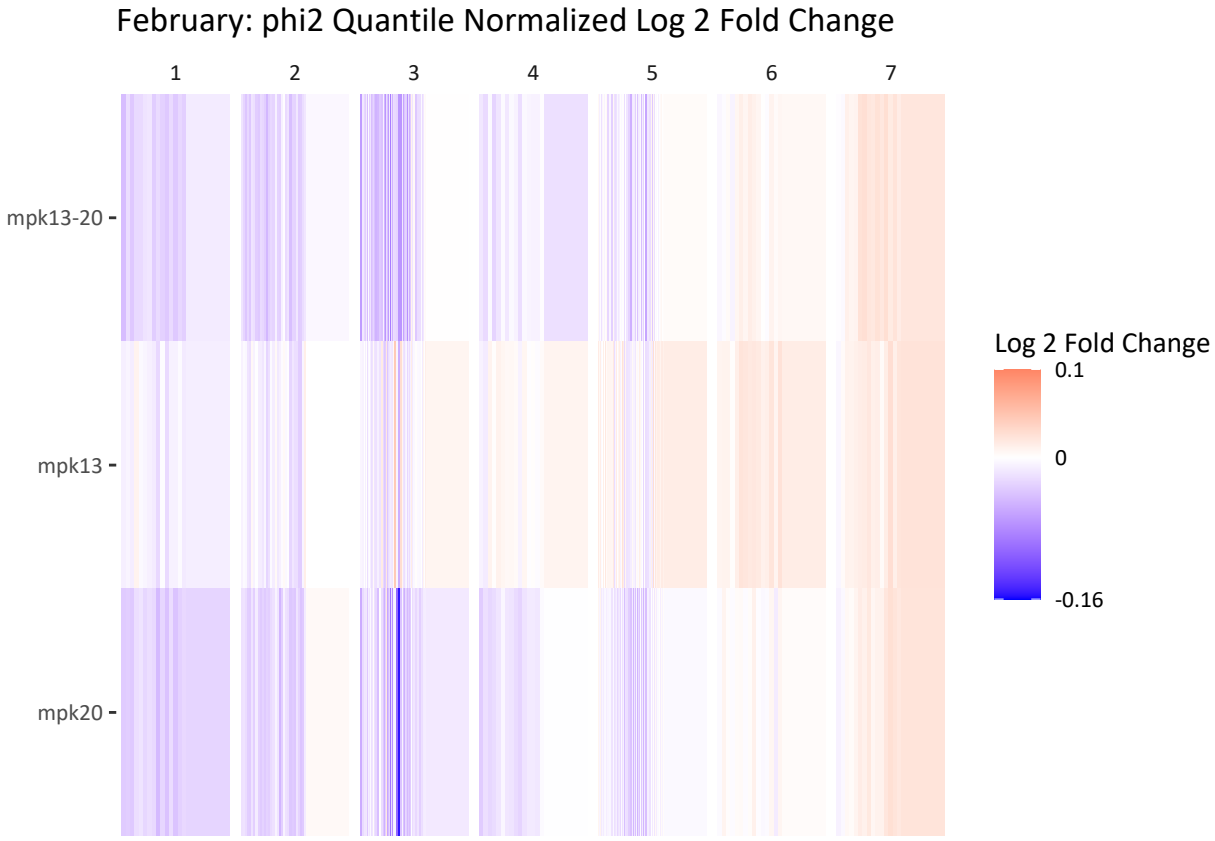

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-17
mpk8
mpk17
Log 2 Fold Change
0.1
0
-0.16

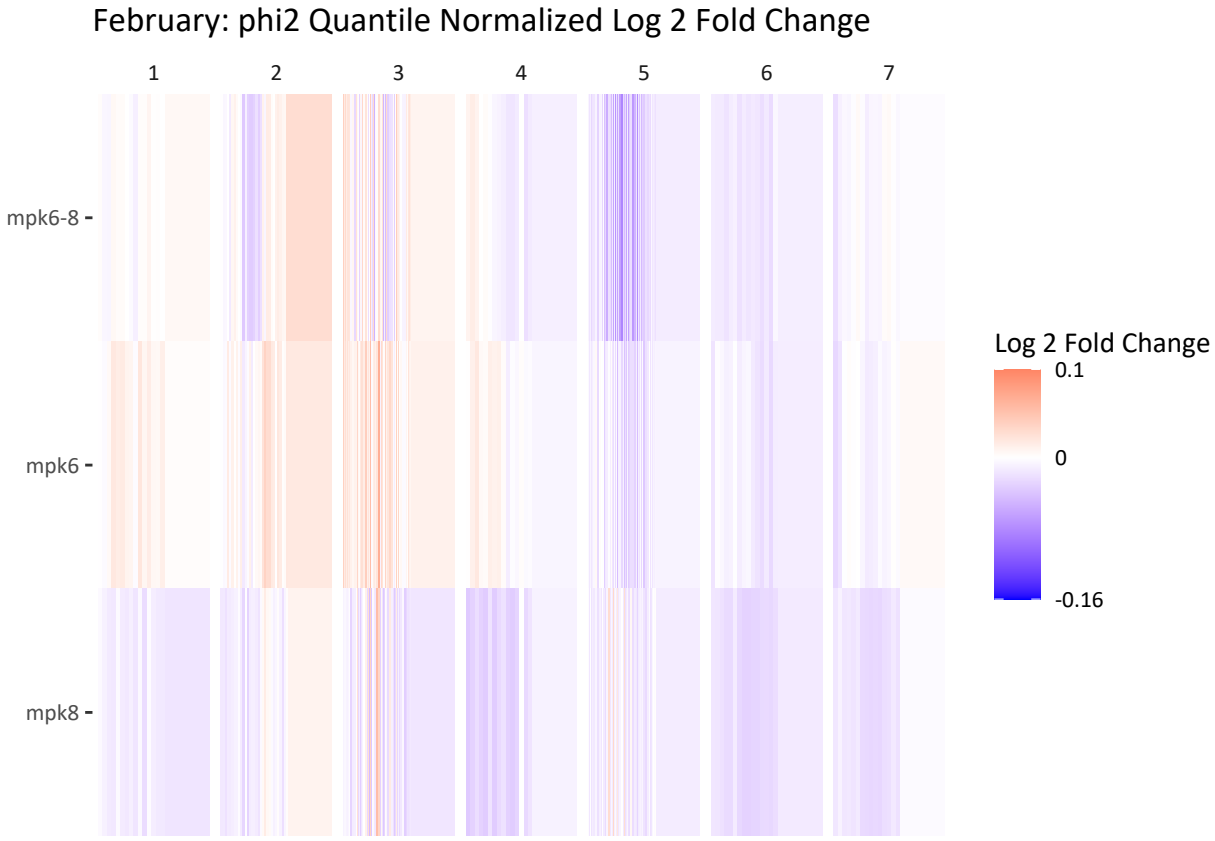

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-20
mpk8
mpk20
Log 2 Fold Change
0.1
0
-0.16

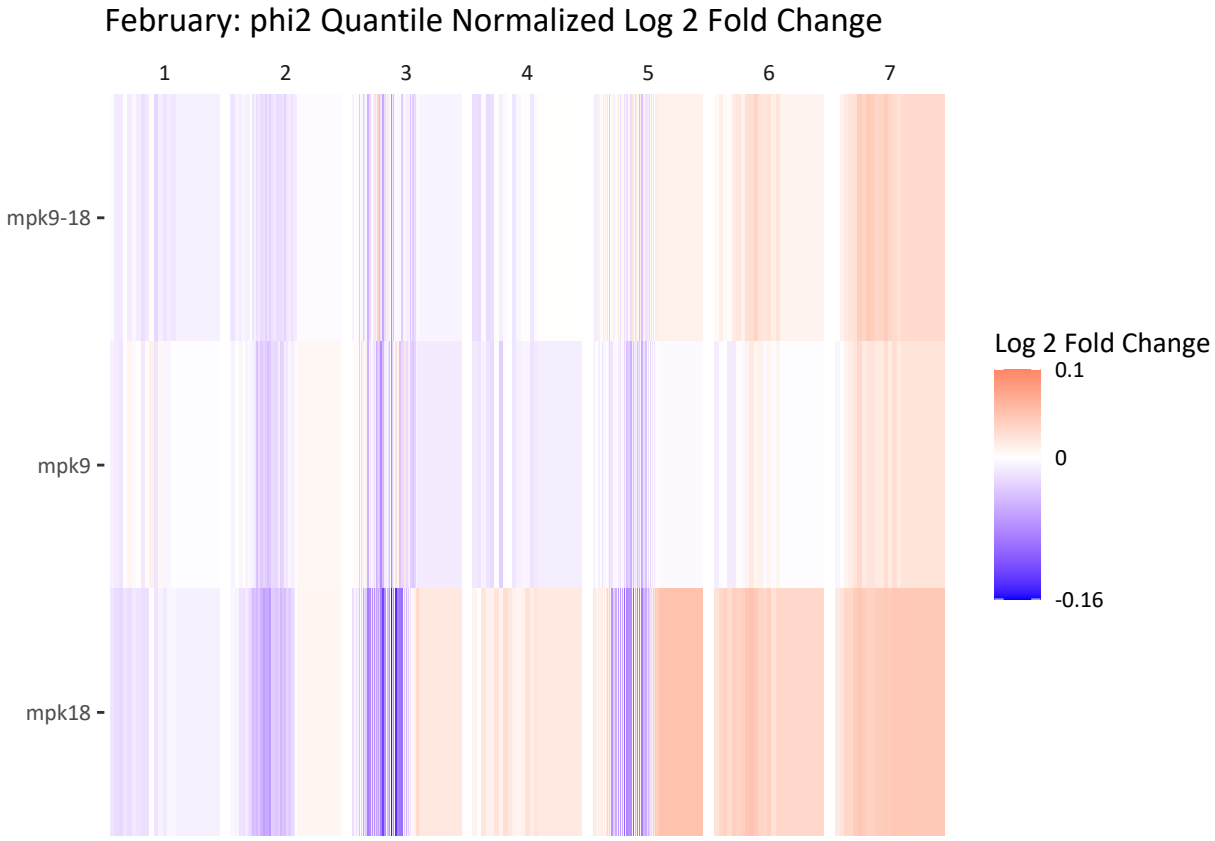

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-18
mpk6
mpk18
Log 2 Fold Change
0.1
0
-0.16

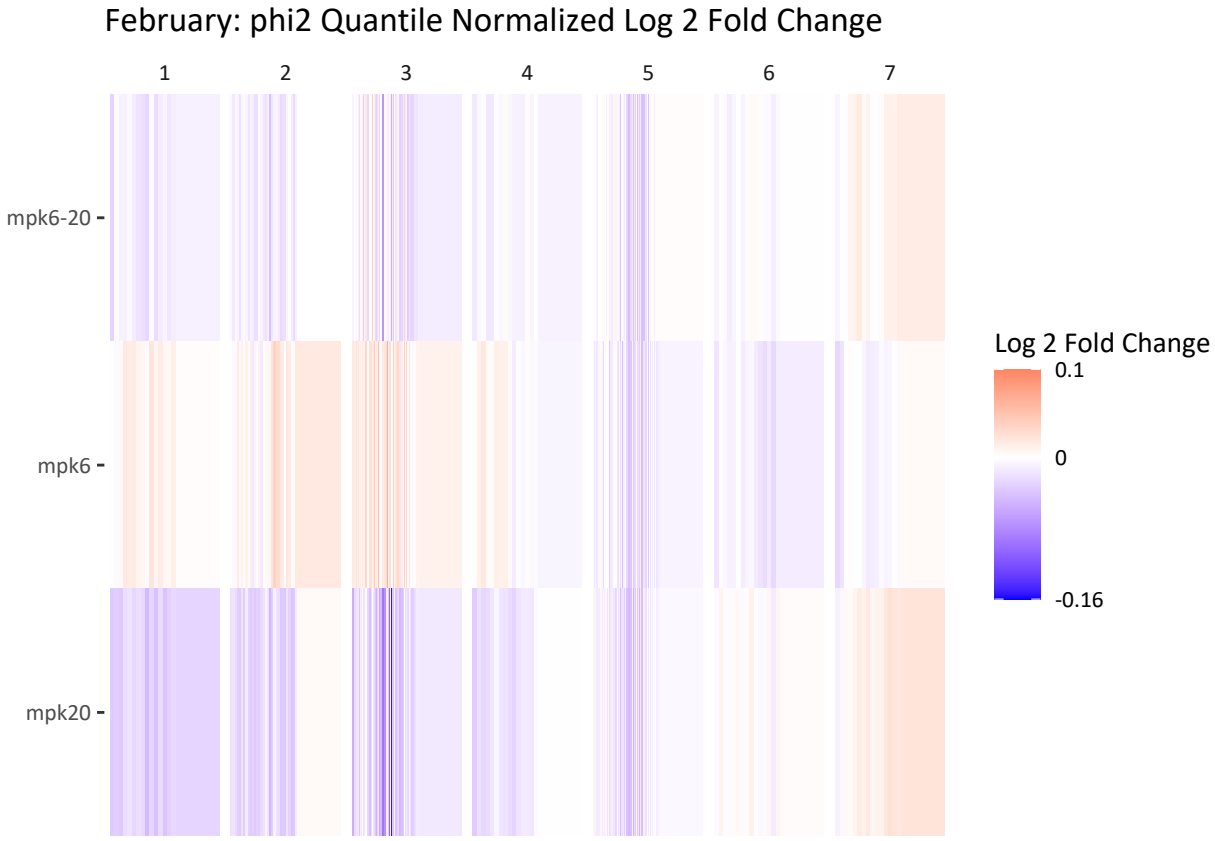

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-13
mpk1
mpk13
Log 2 Fold Change
0.1
0
-0.16

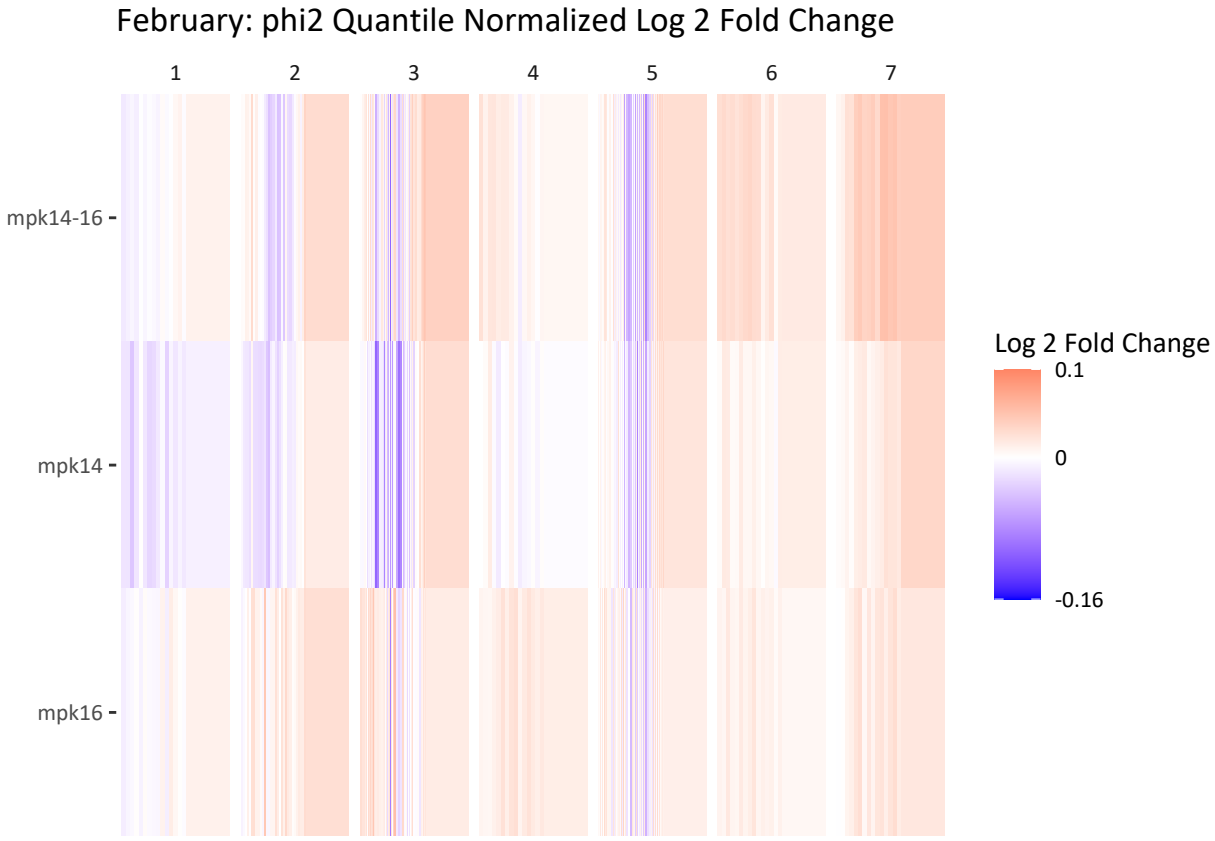

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.1
0
-0.16

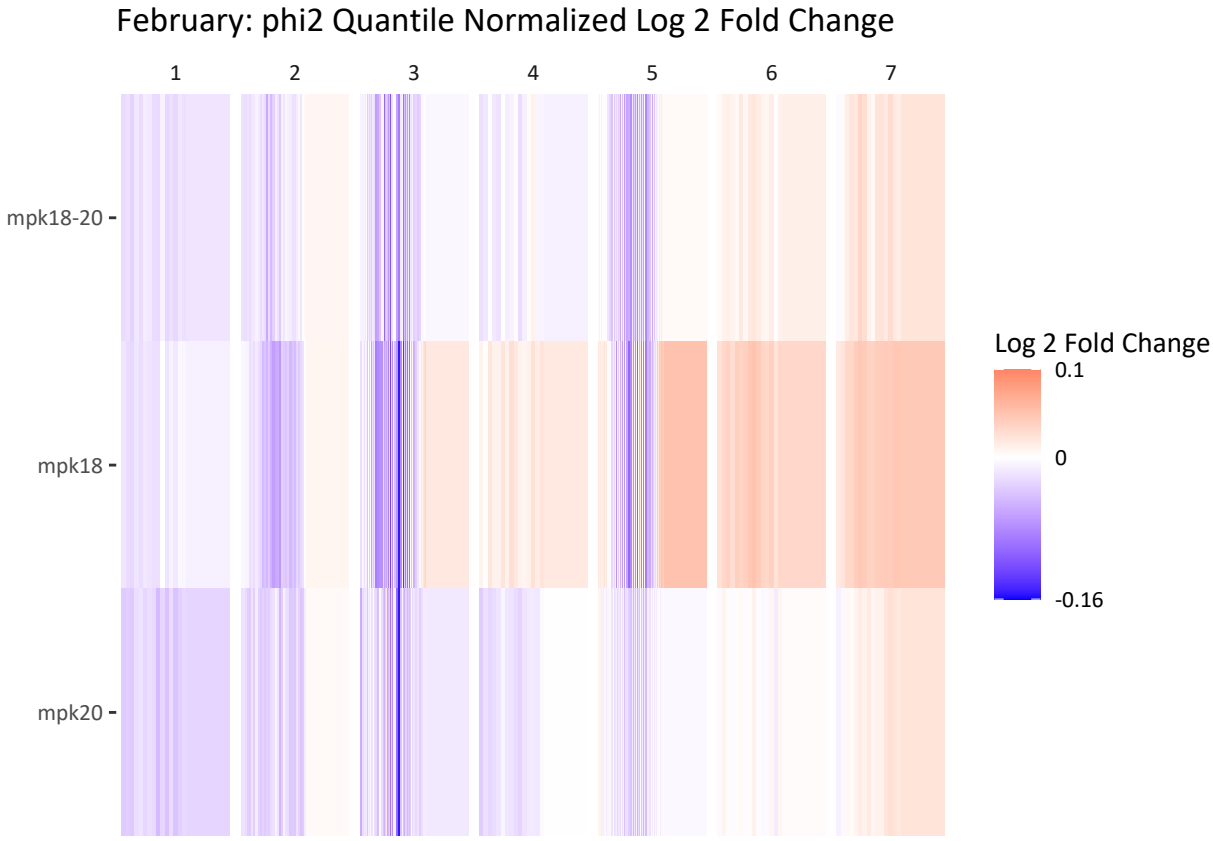

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk13-20
mpk13
mpk20
Log 2 Fold Change
0.1
0
-0.16

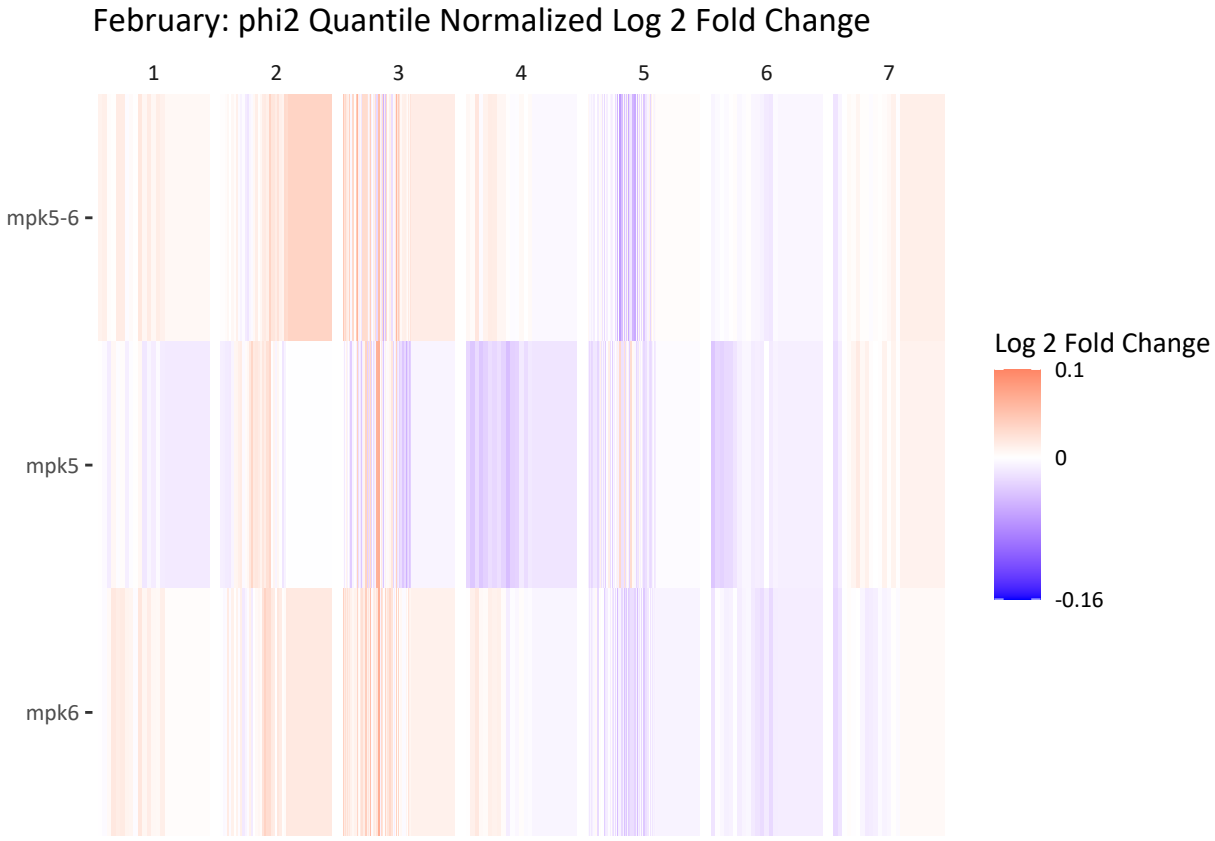

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-8
mpk6
mpk8
Log 2 Fold Change
0.1
0
-0.16

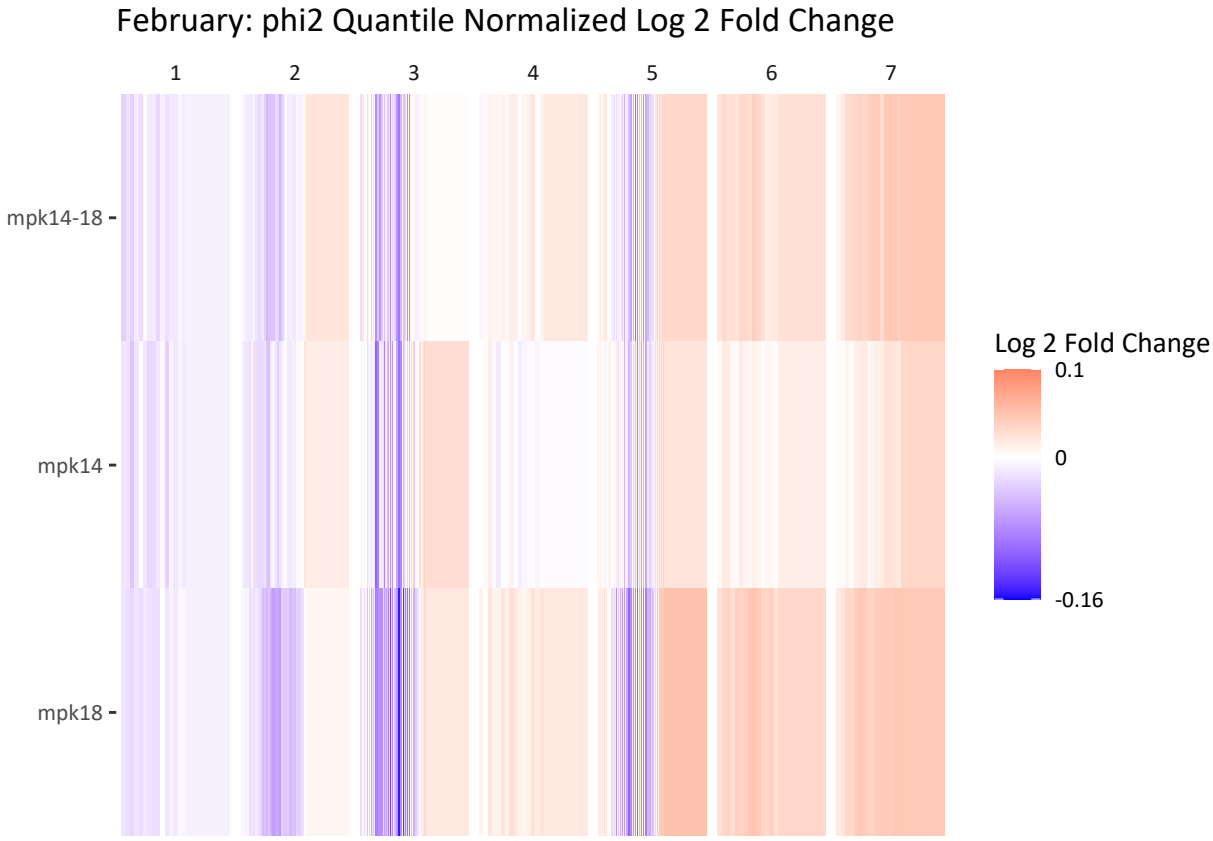

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-18
mpk9
mpk18
Log 2 Fold Change
0.1
0
-0.16

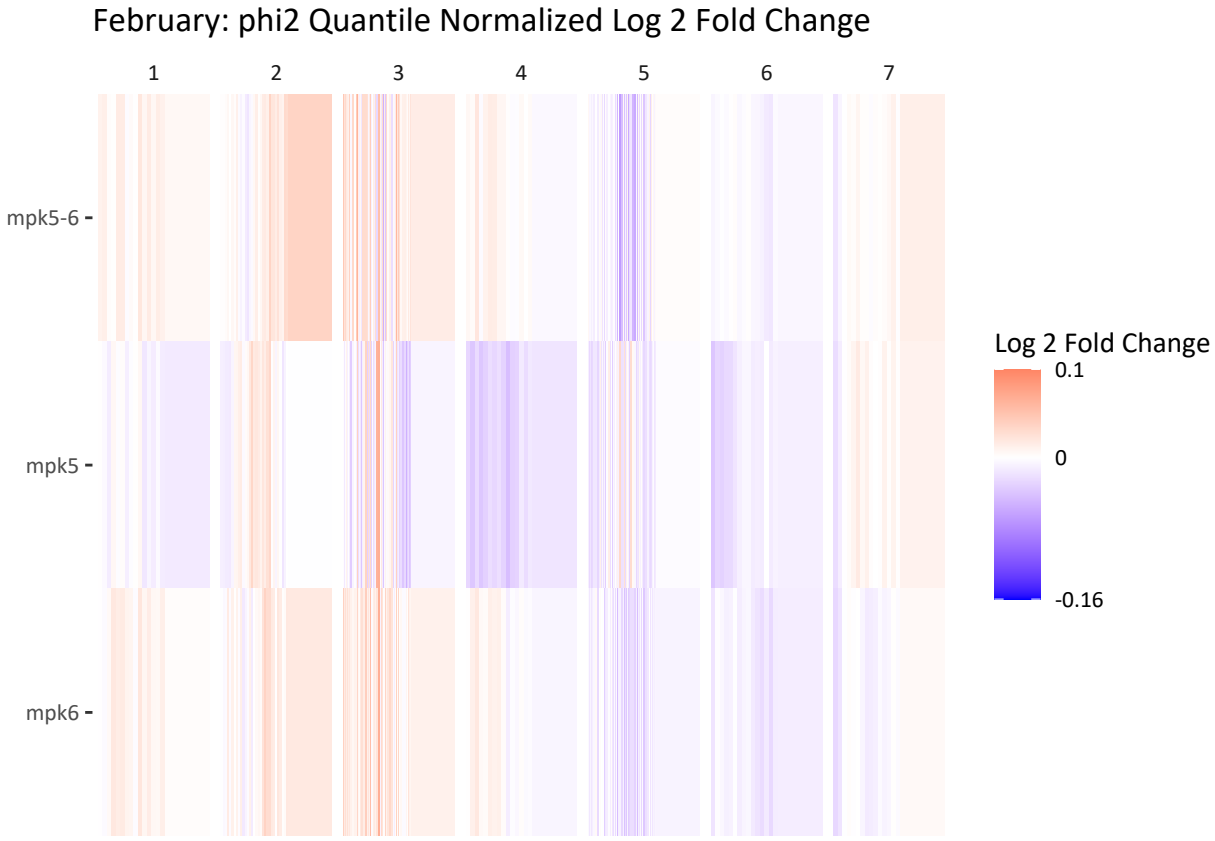

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-20
mpk6
mpk20
Log 2 Fold Change
0.1
0
-0.16

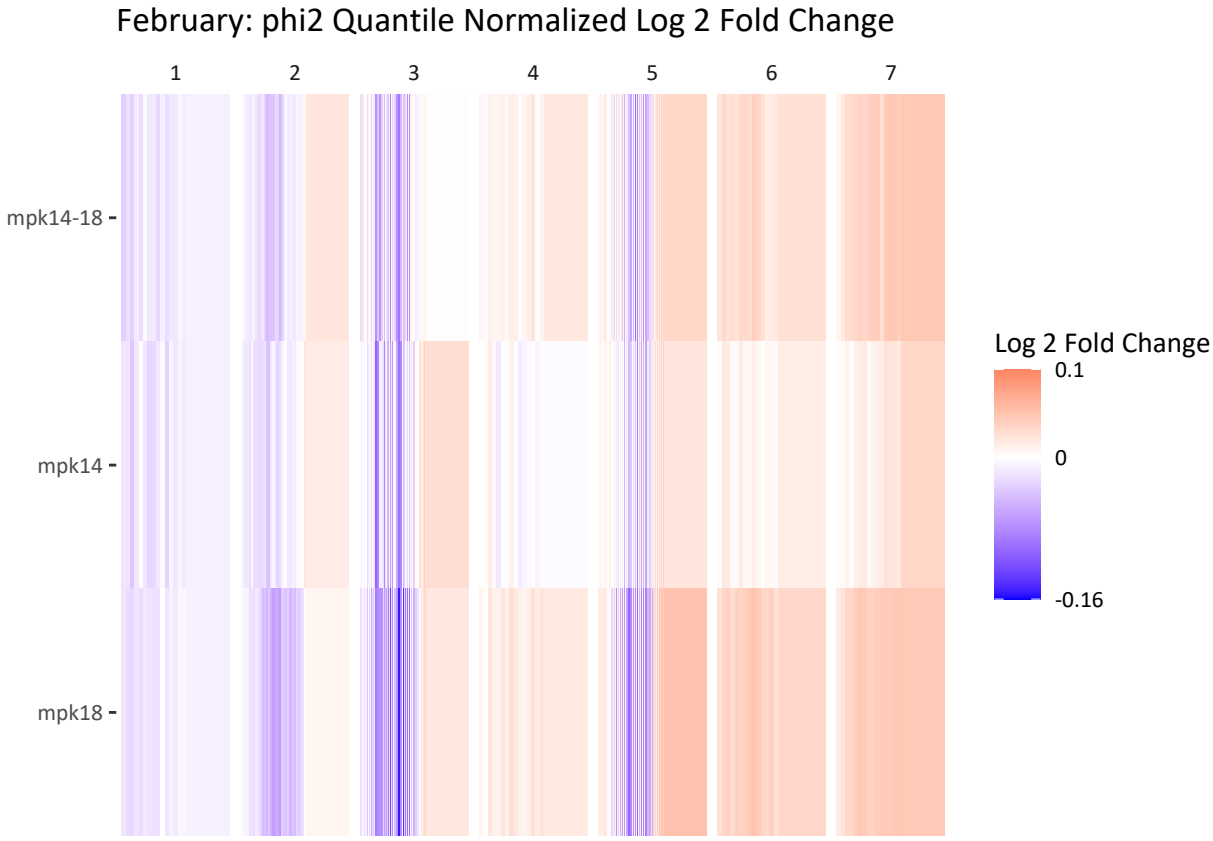

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-16
mpk14
mpk16
Log 2 Fold Change
0.1
0
-0.16

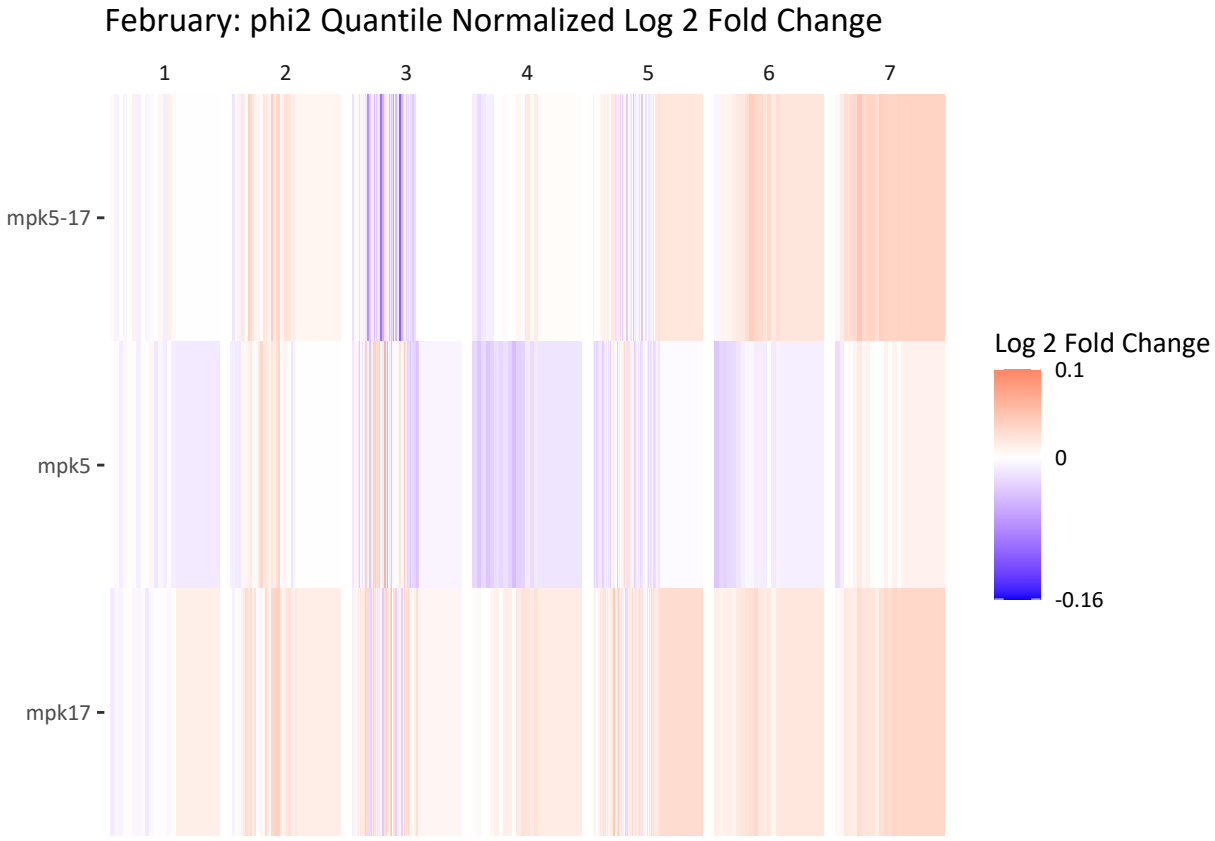

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk18-20
mpk18
mpk20
Log 2 Fold Change
0.1
0
-0.16

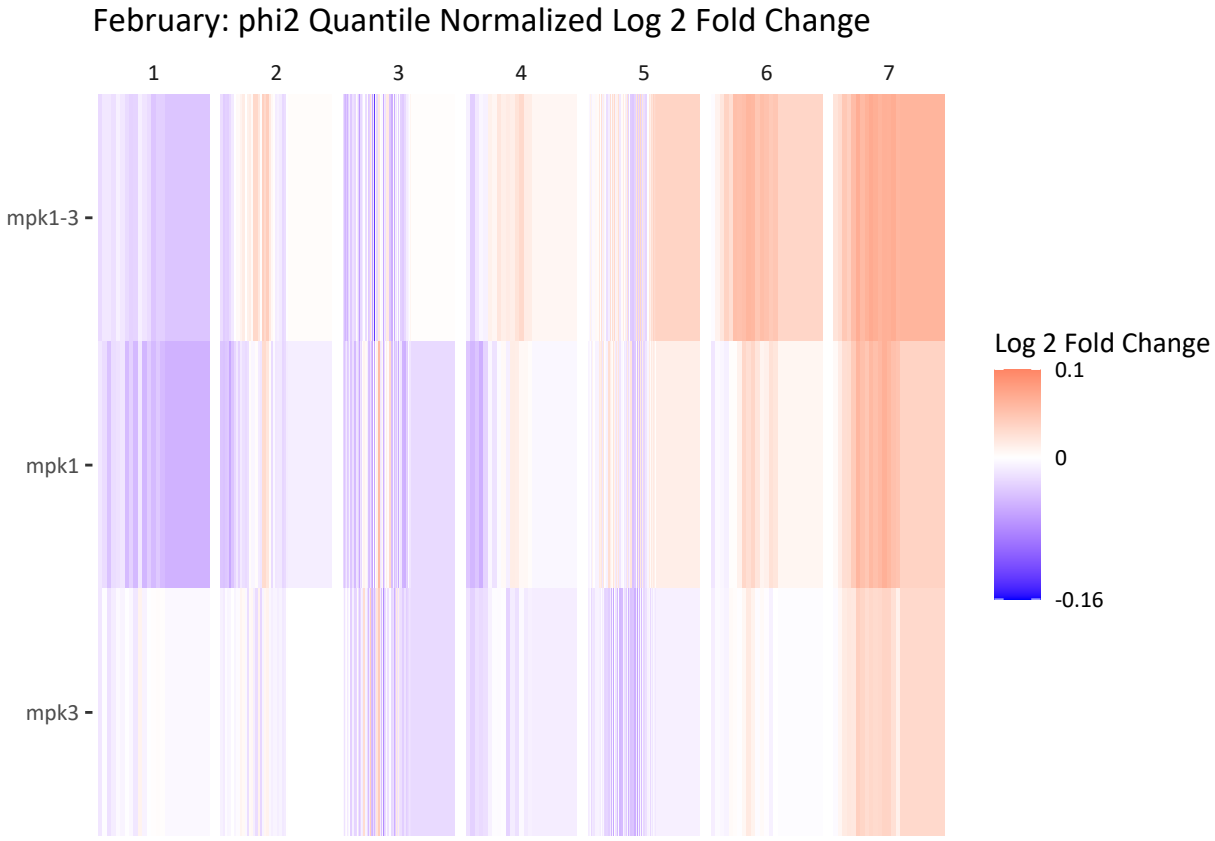

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.1
0
-0.16

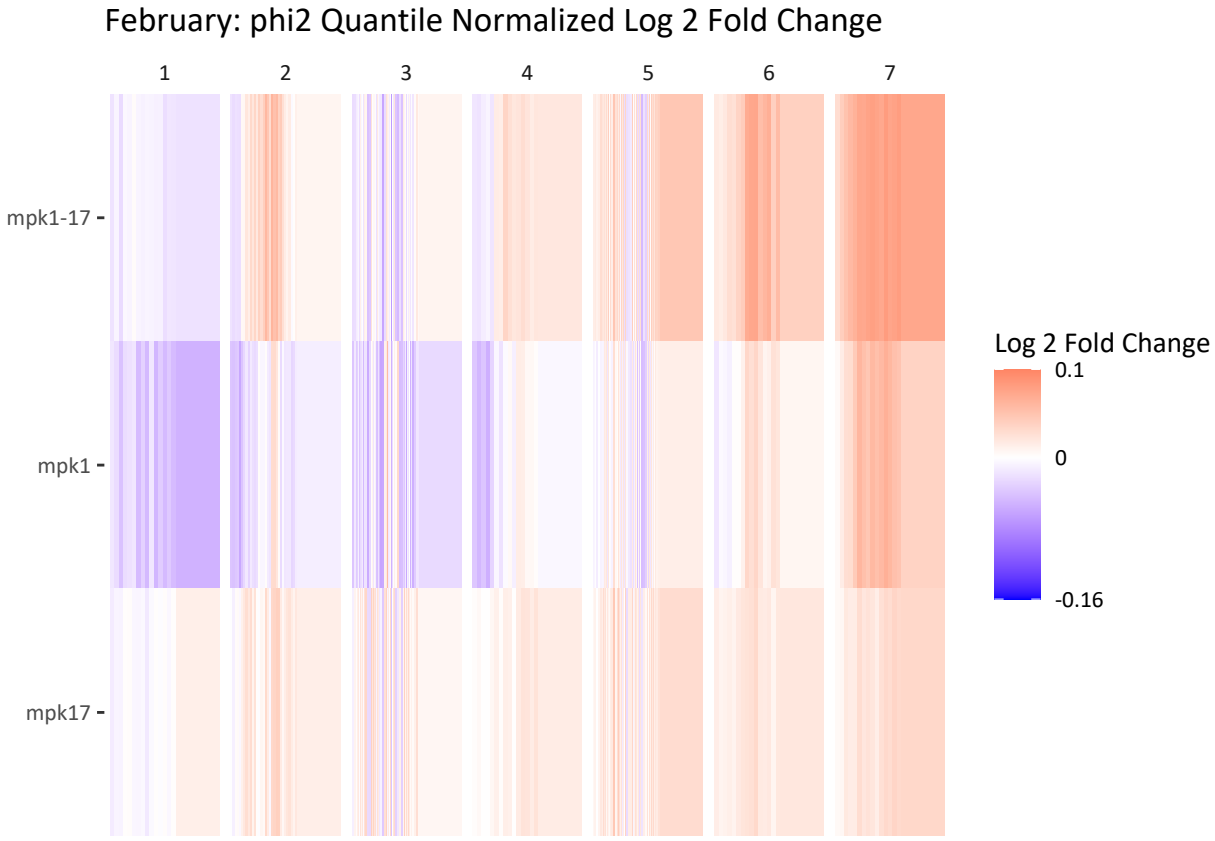

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.1
0
-0.16

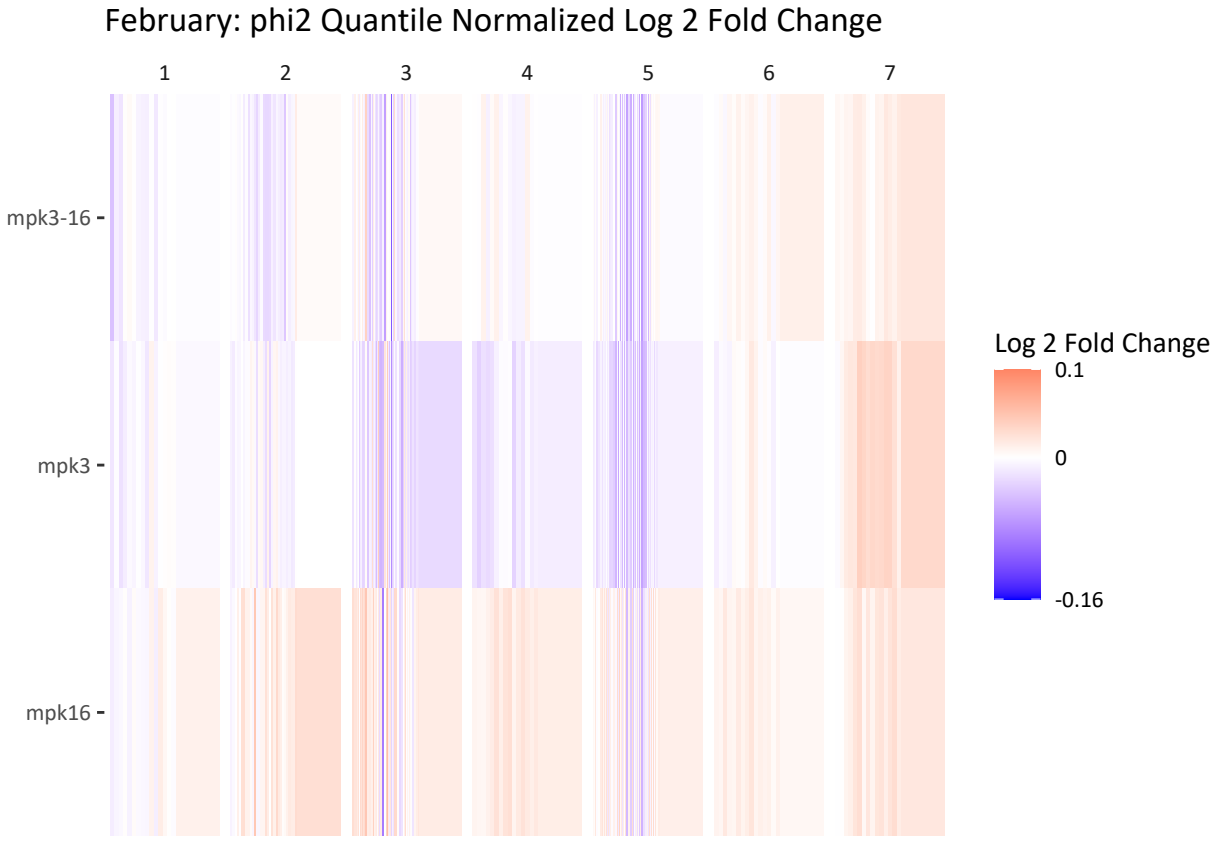

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.1
0
-0.16

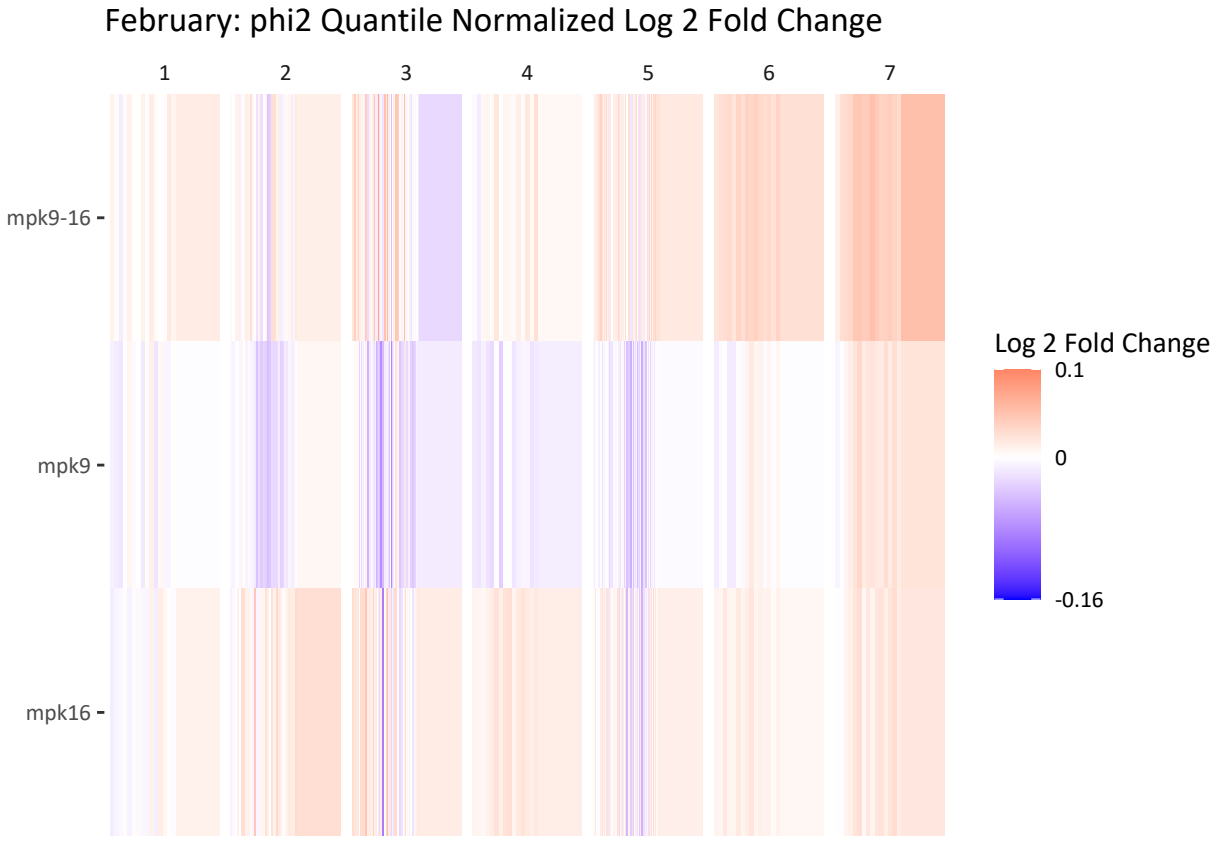

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.1
0
-0.16

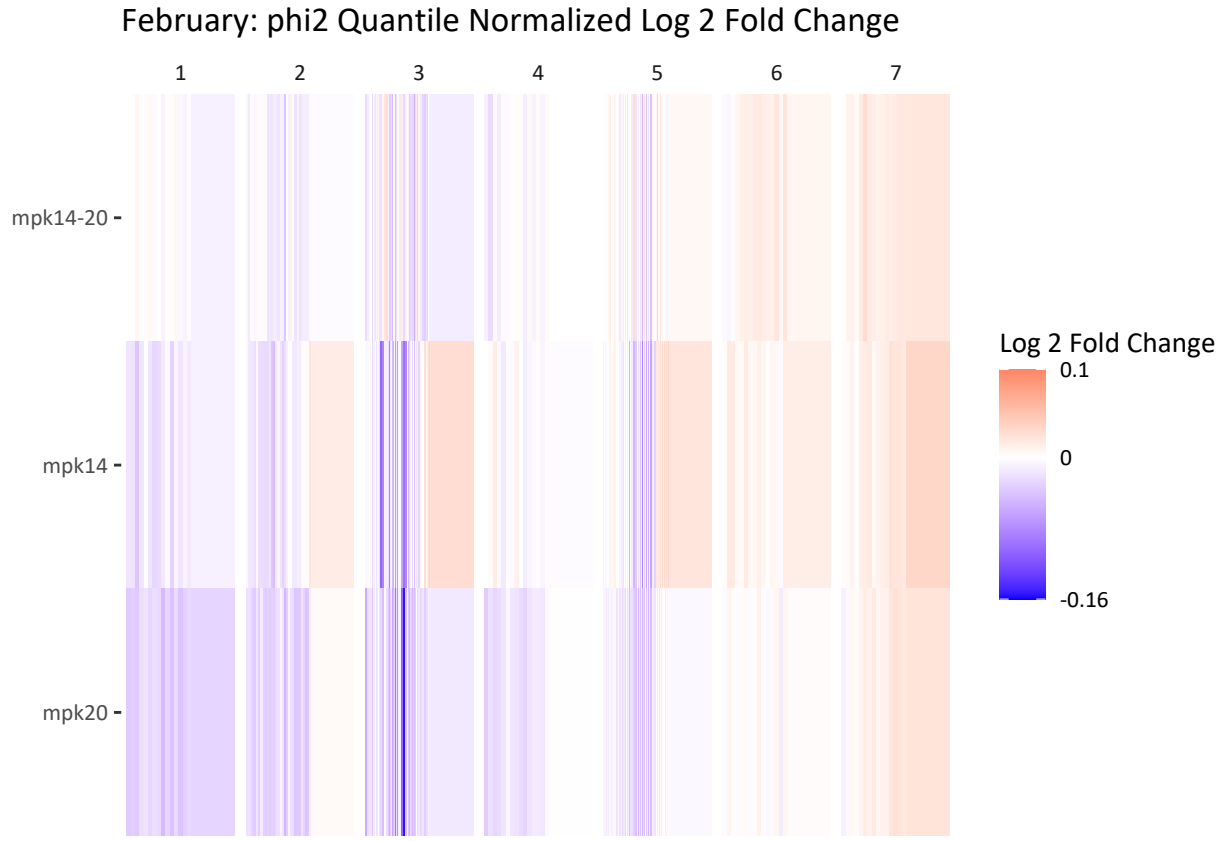

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-17
mpk5
mpk17
Log 2 Fold Change
0.1
0
-0.16


February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-3
mpk1
mpk3
Log 2 Fold Change
0.1
0
-0.16


February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-17
mpk1
mpk17
Log 2 Fold Change
0.1
0
-0.16


February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk3-16
mpk3
mpk16
Log 2 Fold Change
0.1
0
-0.16

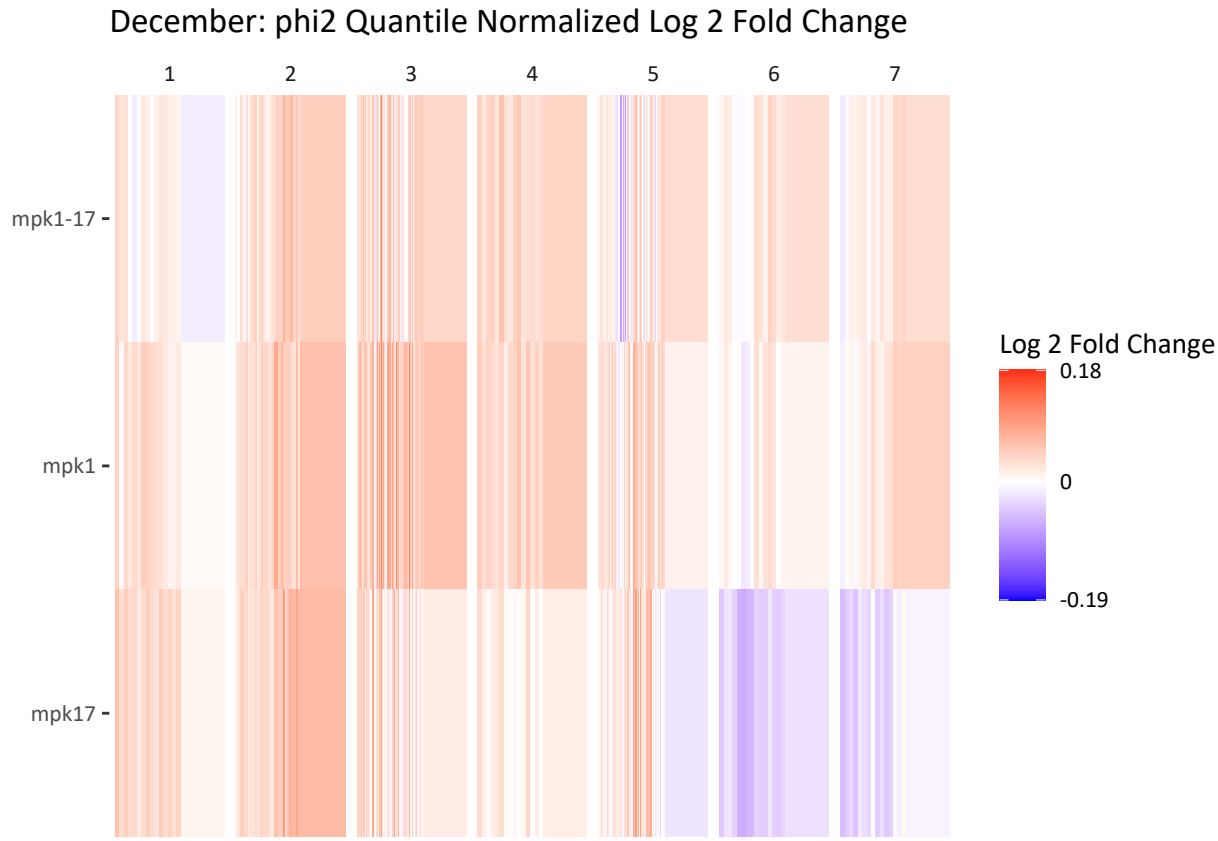

February: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-16
mpk9
mpk16
Log 2 Fold Change
0.1
0
-0.16

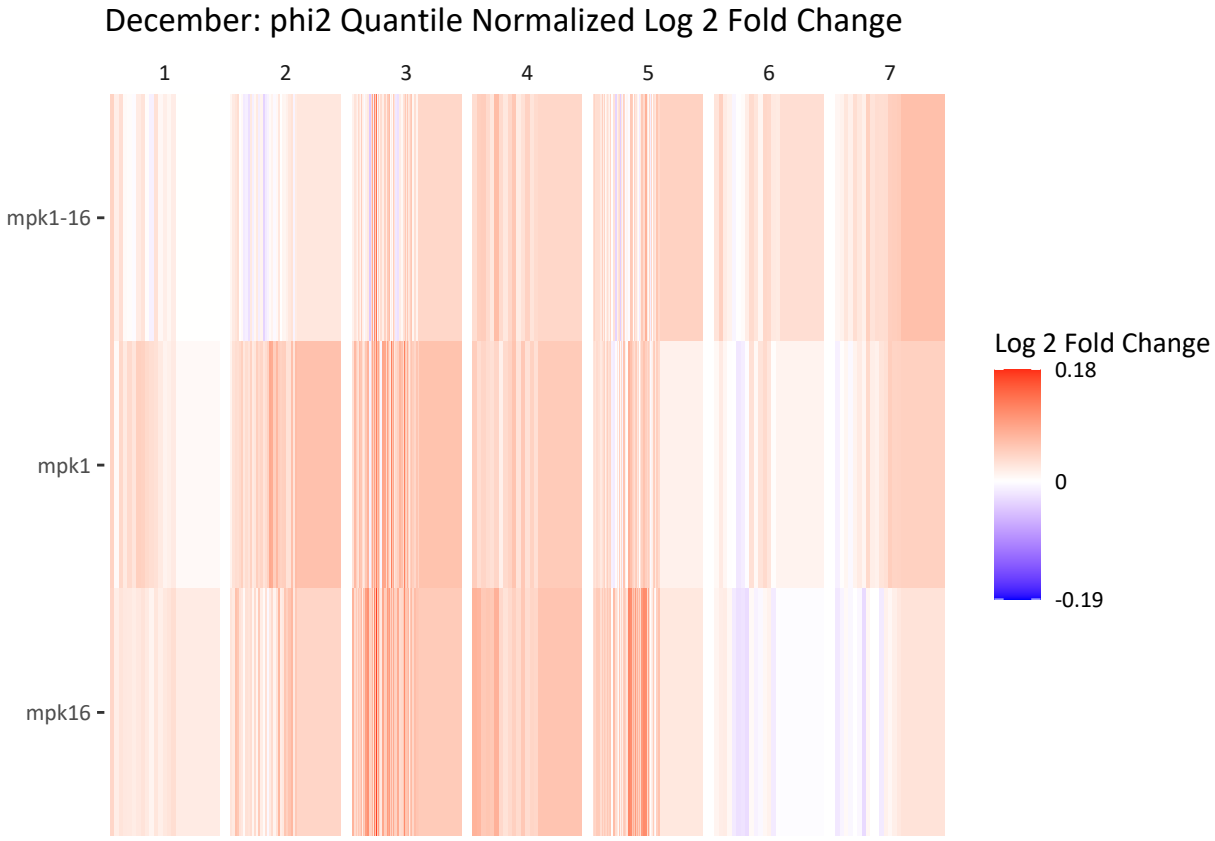


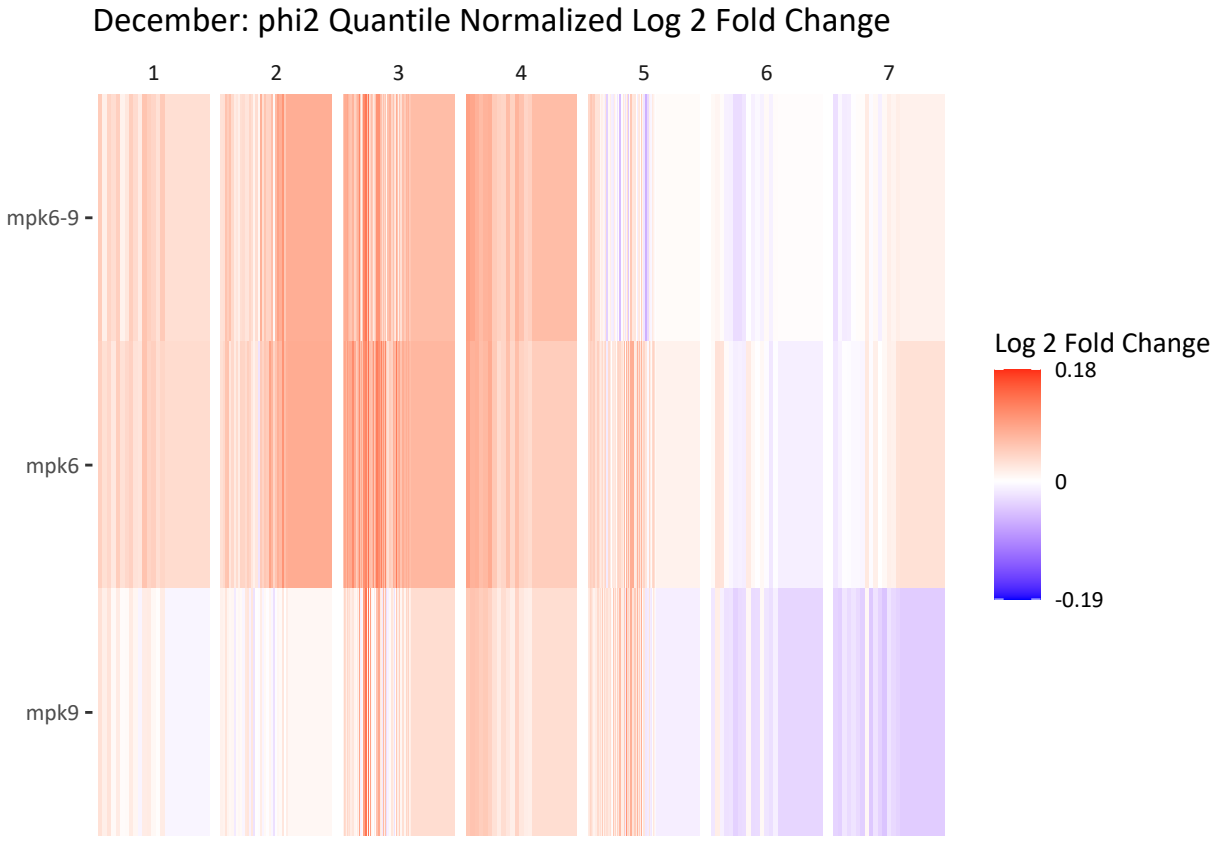
December Loop

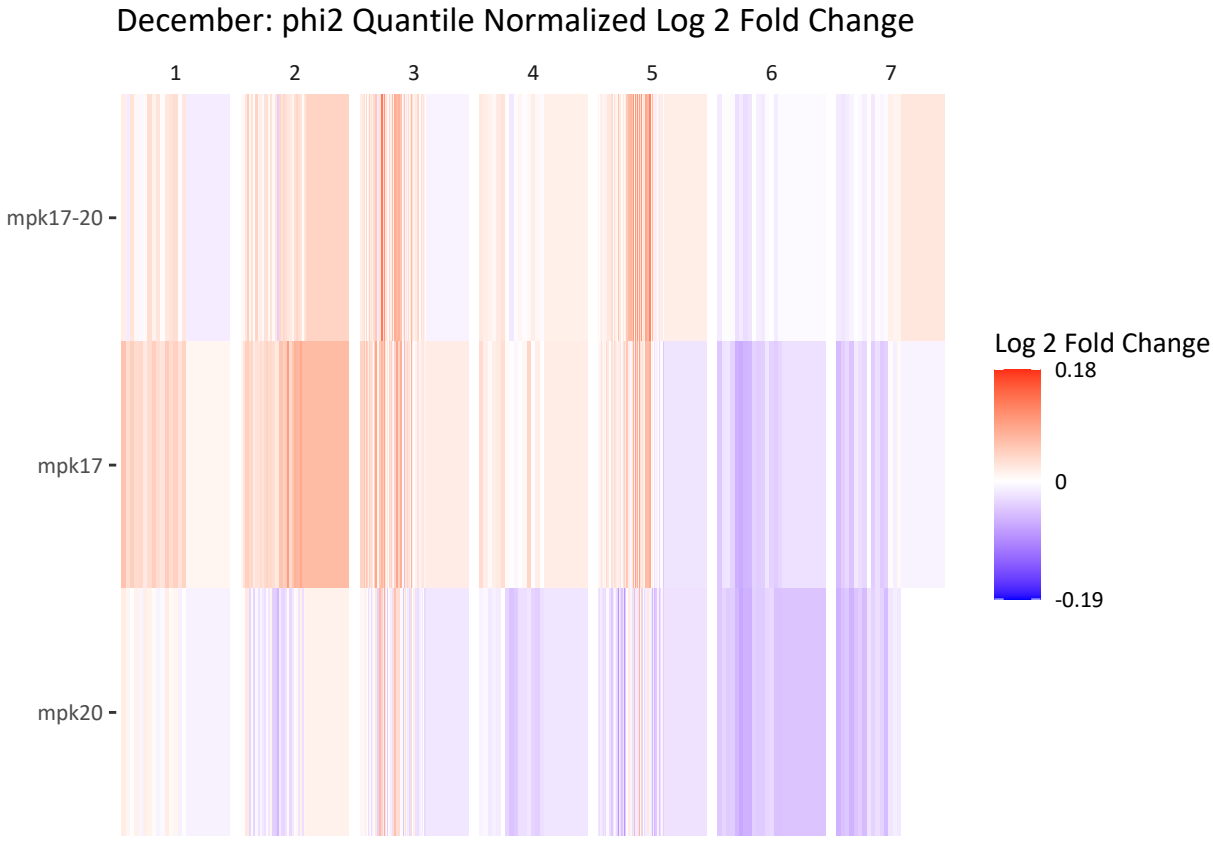
```
for (element in genotype_combinations) {
    m <- "phi2"
    data <- filter(cell_371_dec, genotype %in%
        element, measurement == m)
    data$genotype <- reorder(data$genotype,
        data$number_2)
    plot <- ggplot(data = data, aes(x = time_point,
        y = genotype, fill = log2_fold)) +
        labs(fill = "Log 2 Fold Change",
            x = "Hours", y = NULL, title = paste("December:",
                m, "Quantile Normalized Log 2 Fold Change")) +
        geom_tile(width = ifelse(m == "leafarea",
            16, 10), height = 30) + facet_grid(genotype ~
        day, scales = "free", switch = "y") +
        theme_tufte(base_family = "Calibri") +
        theme(strip.background.y = element_blank(),
            strip.text.y = element_blank(),
            axis.title.x = element_blank(),
            axis.text.x = element_blank(),
            axis.ticks.x = element_blank(),
            panel.spacing = unit(0, "lines")) +
        scale_fill_gradient2(low = "blue",
```

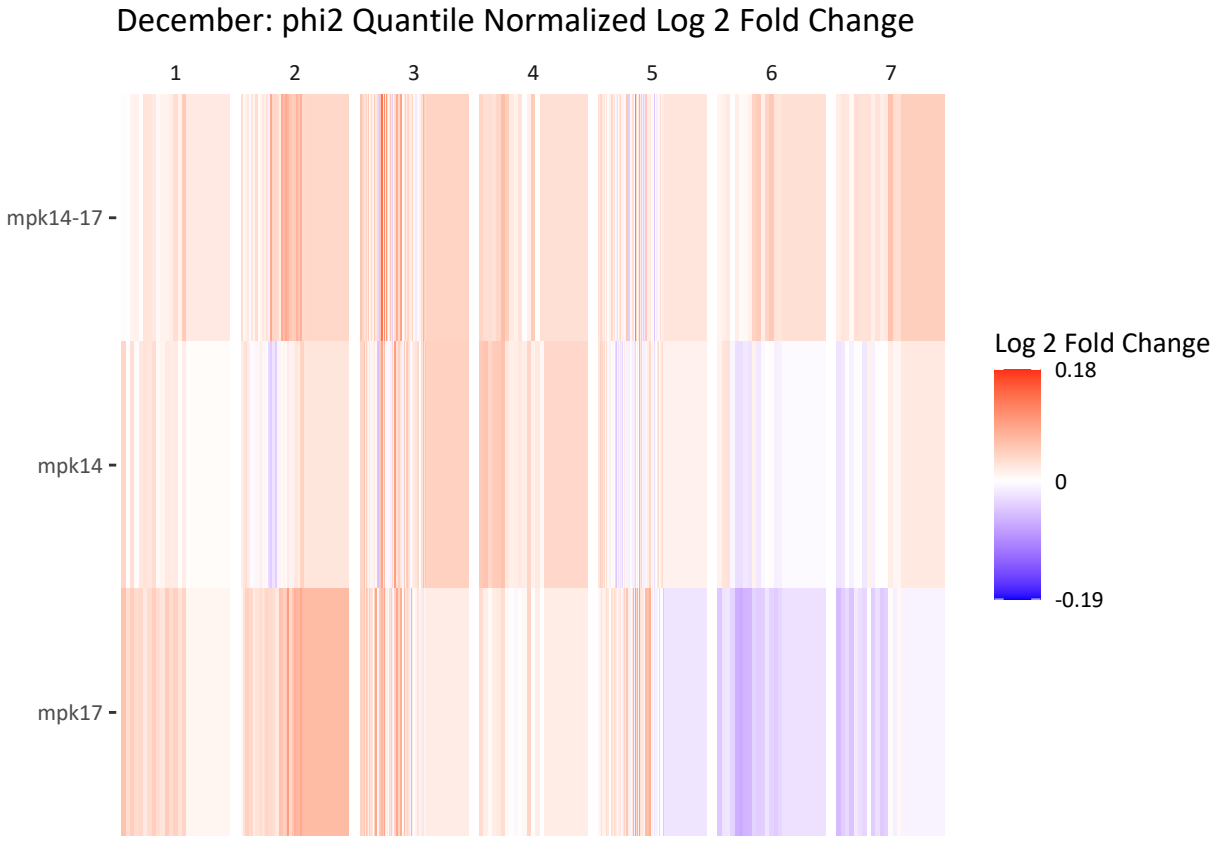
```
            high = "red", mid = "white",
            midpoint = 0, limits = c(-0.19,
                0.18), breaks = c(-0.19,
                0, 0.18), labels = c("-0.19",
                "0", "0.18"))
    print(plot)
}
```

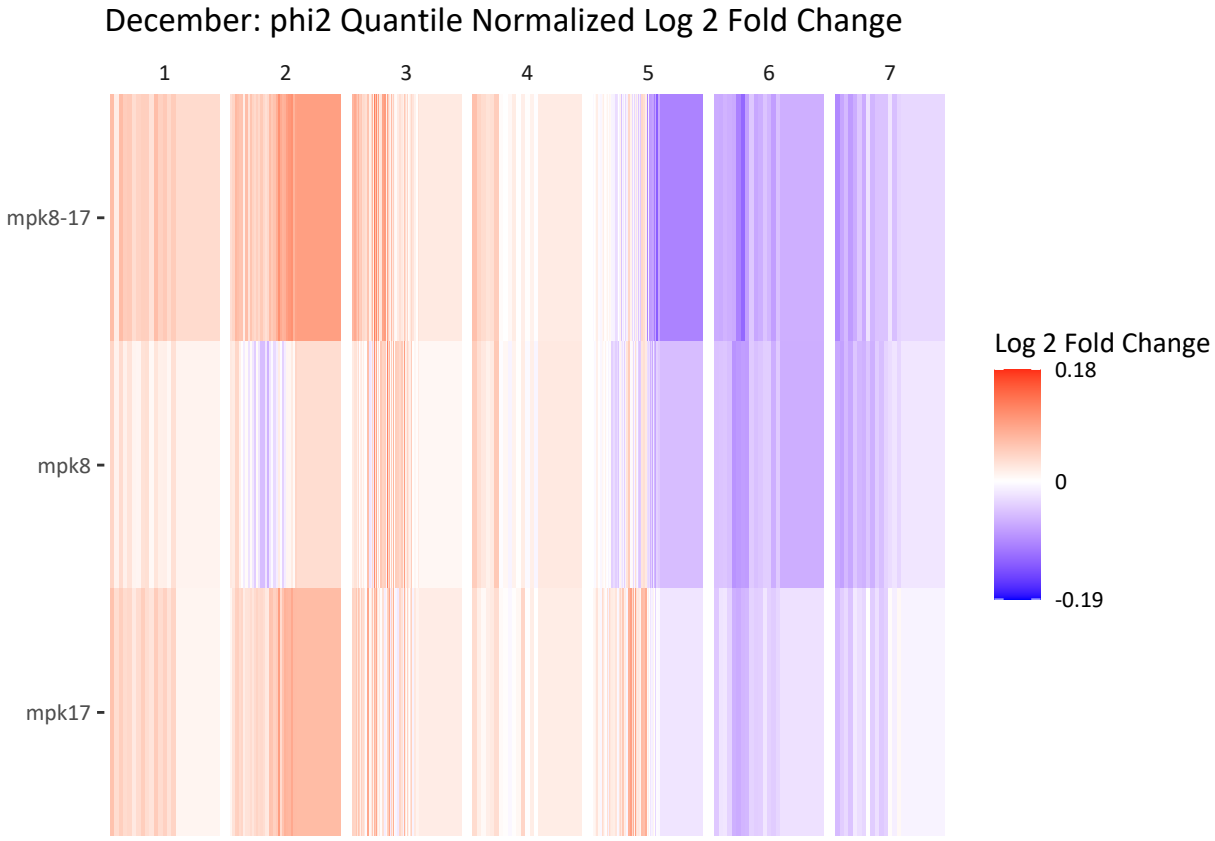


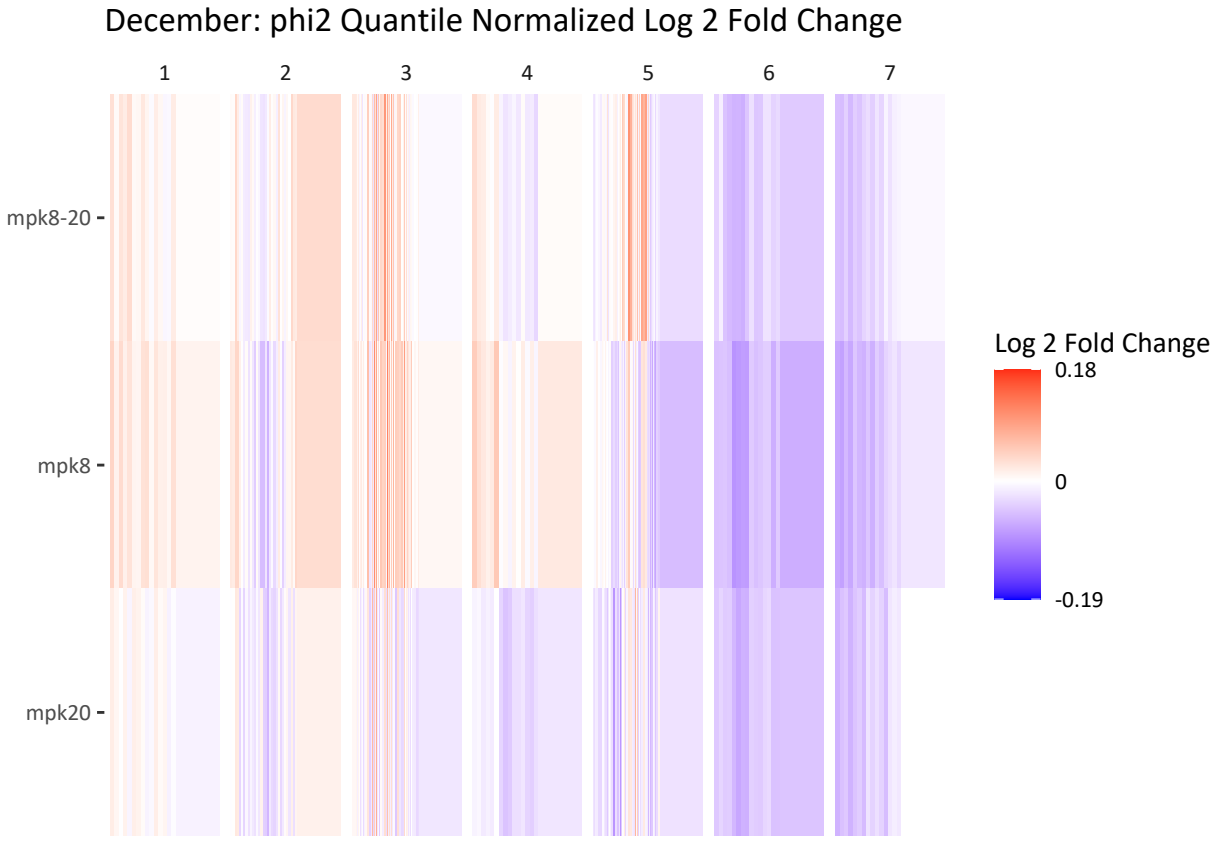

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-16
mpk1
mpk16
Log 2 Fold Change
0.18
0
-0.19

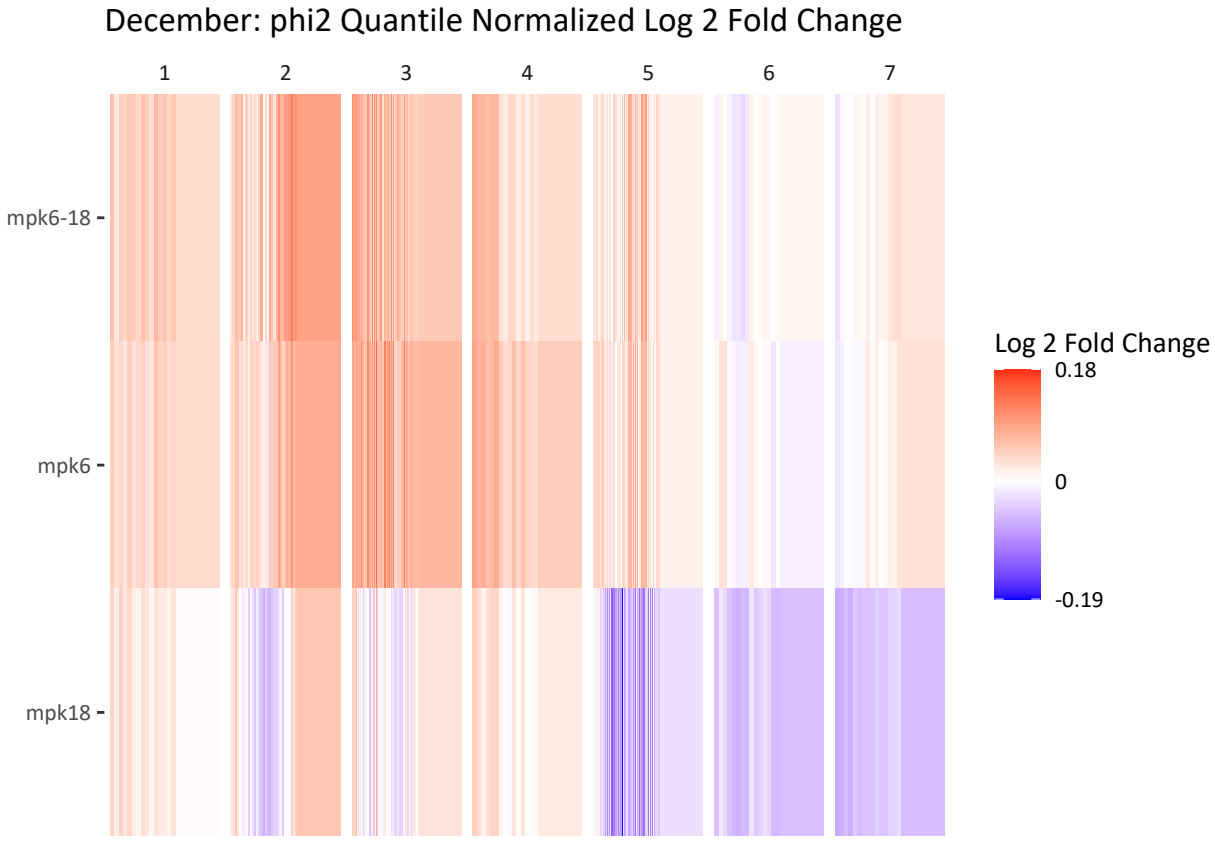

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-9
mpk6
mpk9
Log 2 Fold Change
0.18
0
-0.19

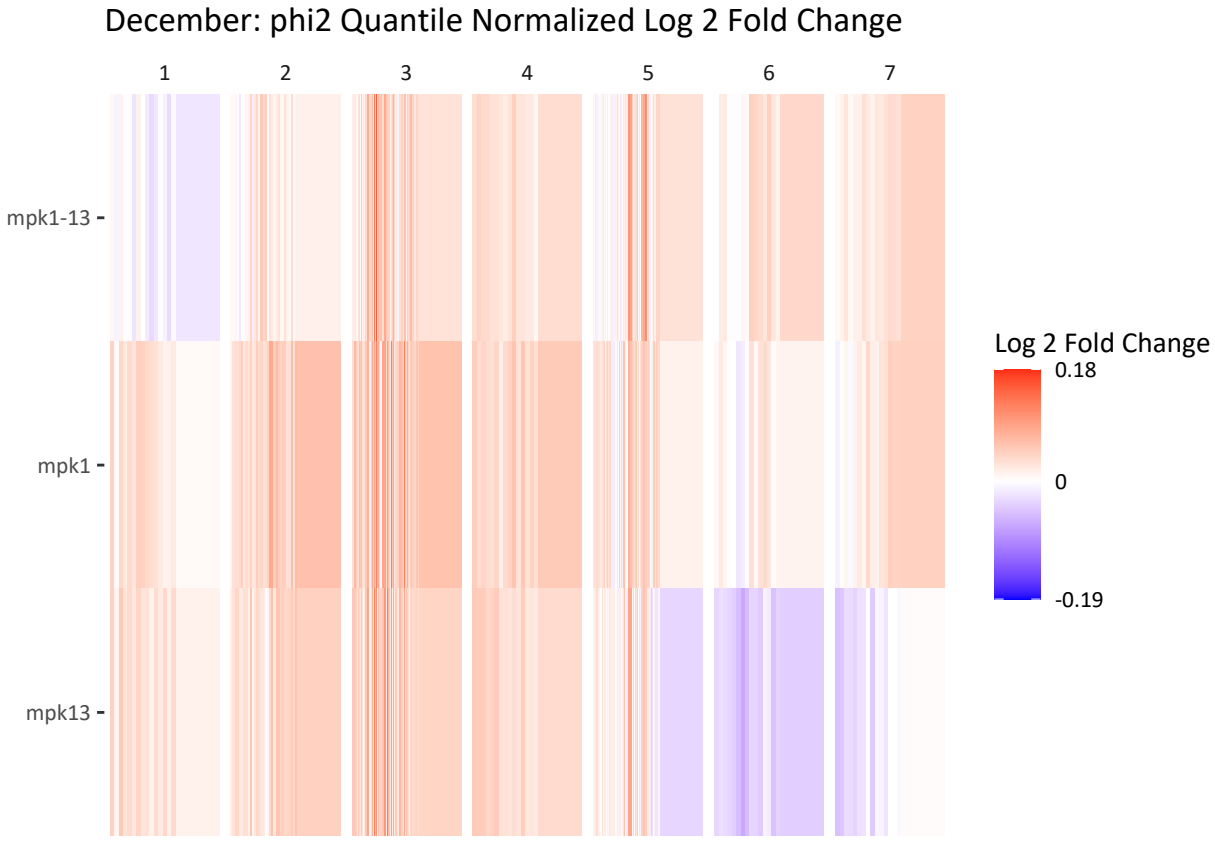

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.18
0
-0.19

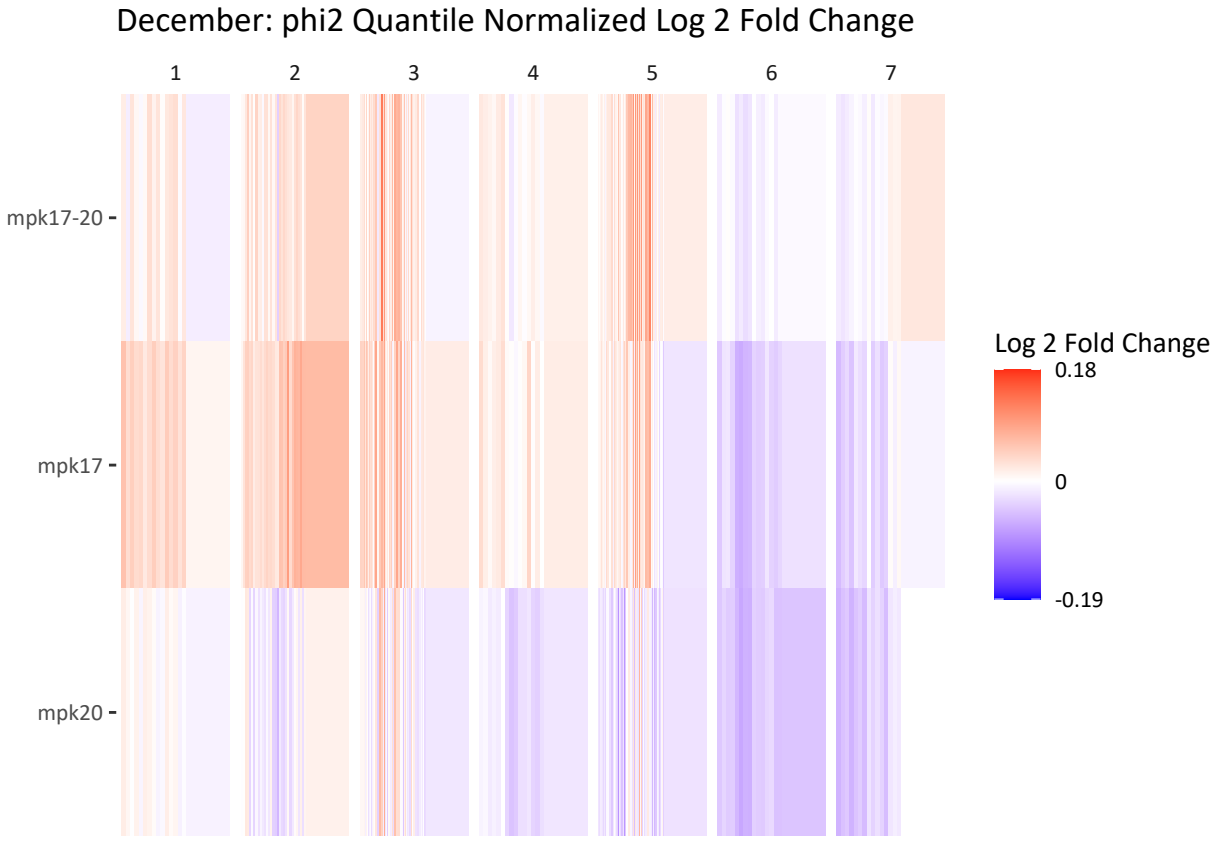

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-17
mpk14
mpk17
Log 2 Fold Change
0.18
0
-0.19

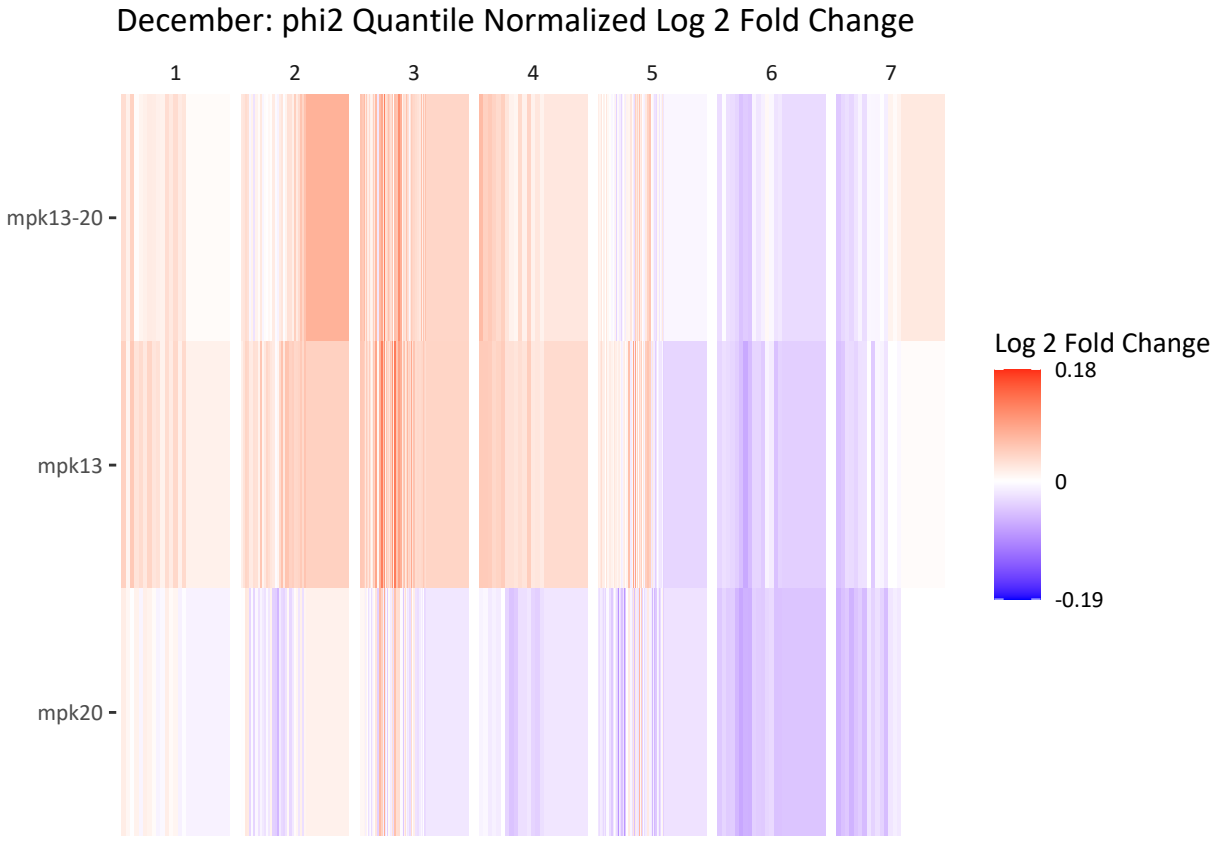

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-17
mpk8
mpk17
Log 2 Fold Change
0.18
0
-0.19

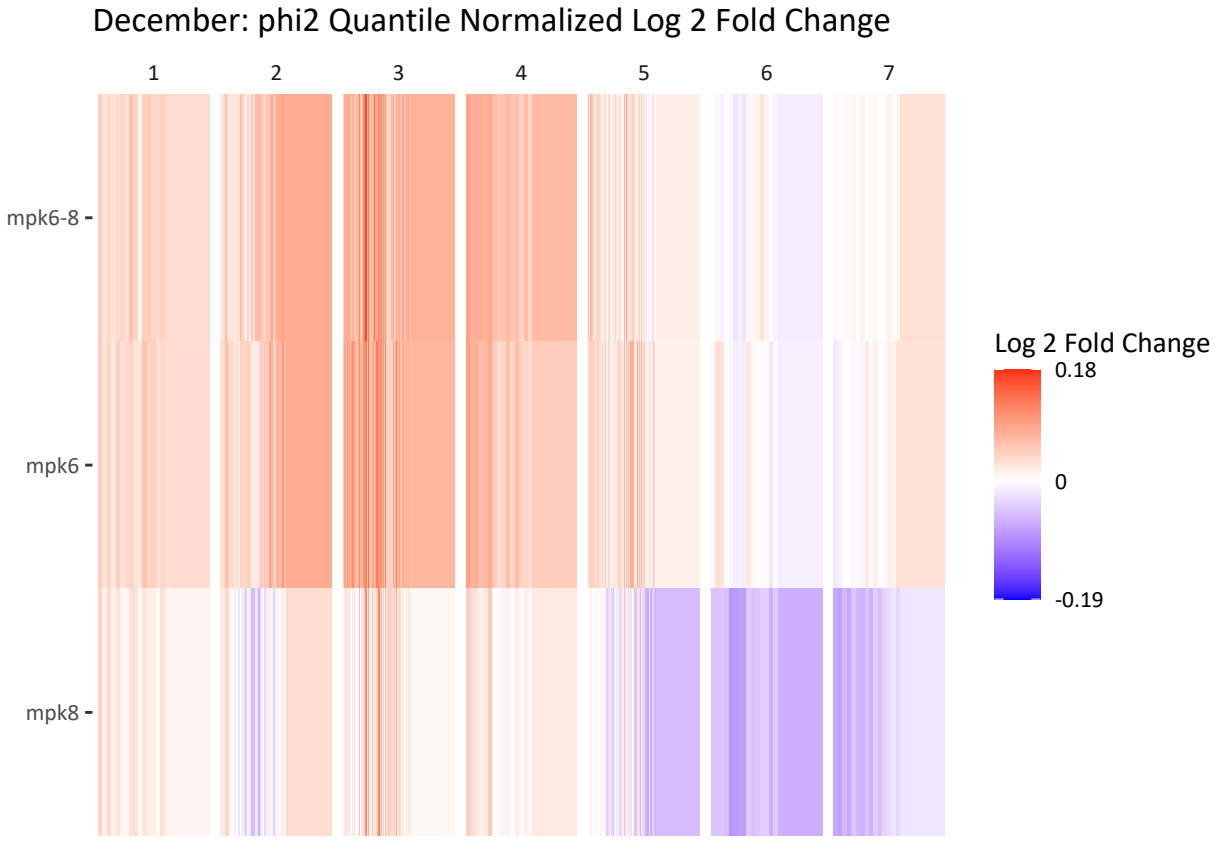

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk8-20
mpk8
mpk20
Log 2 Fold Change
0.18
0
-0.19

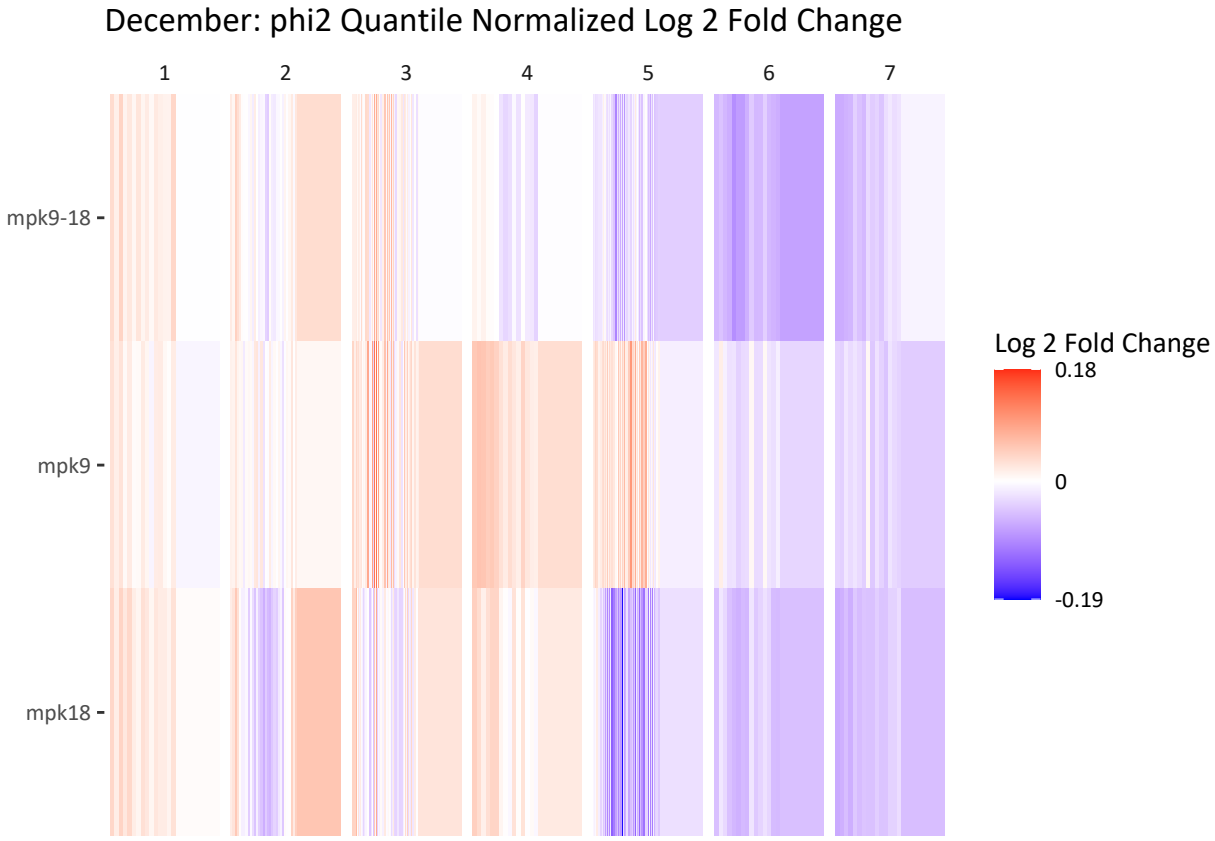

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-18
mpk6
mpk18
Log 2 Fold Change
0.18
0
-0.19

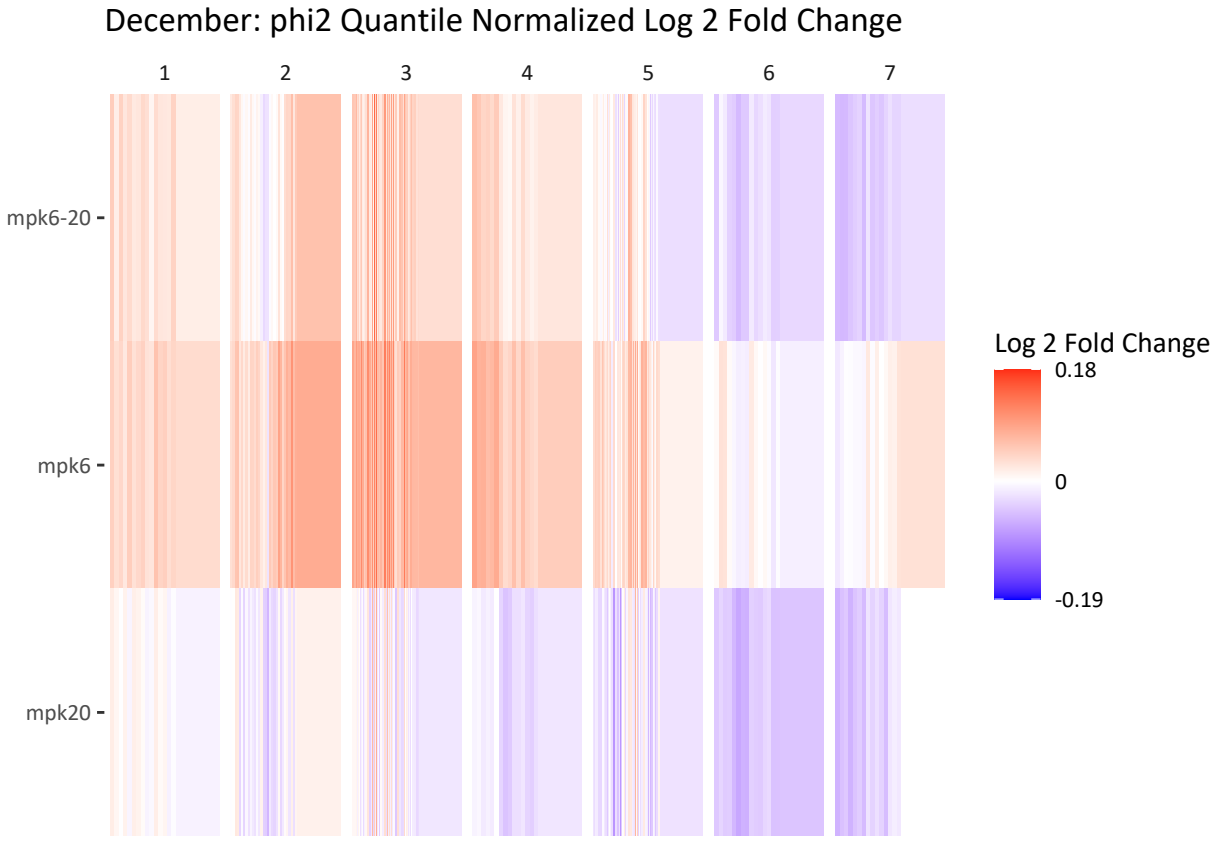

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-13
mpk1
mpk13
Log 2 Fold Change
0.18
0
-0.19

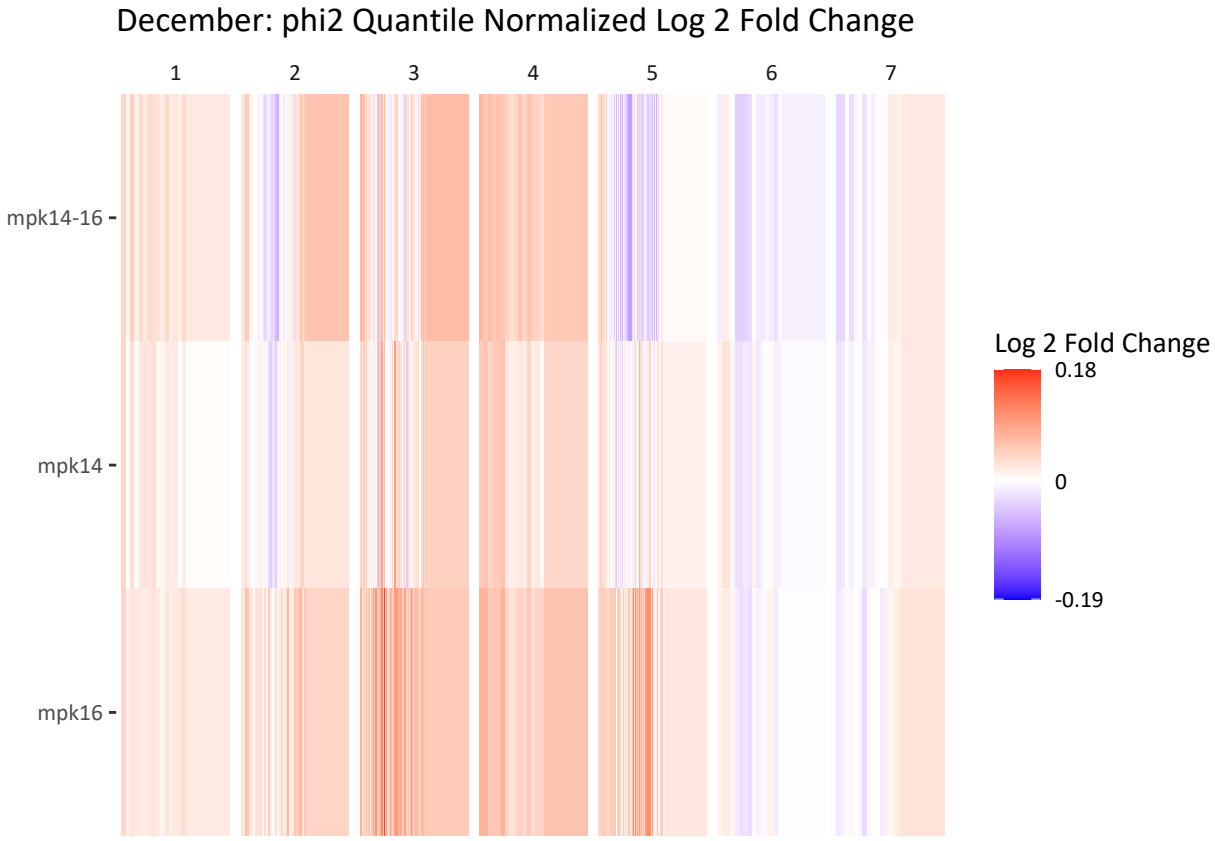

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk17-20
mpk17
mpk20
Log 2 Fold Change
0.18
0
-0.19

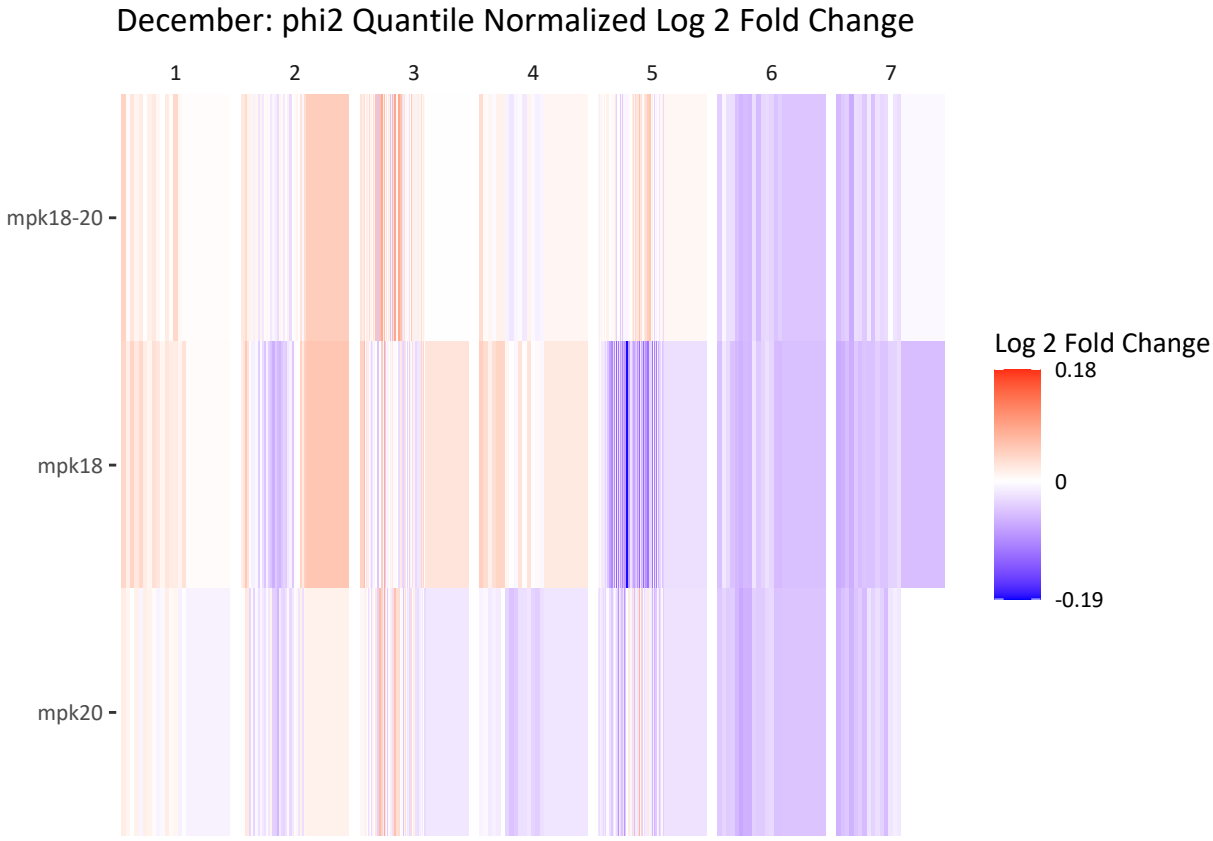

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk13-20
mpk13
mpk20
Log 2 Fold Change
0.18
0
-0.19

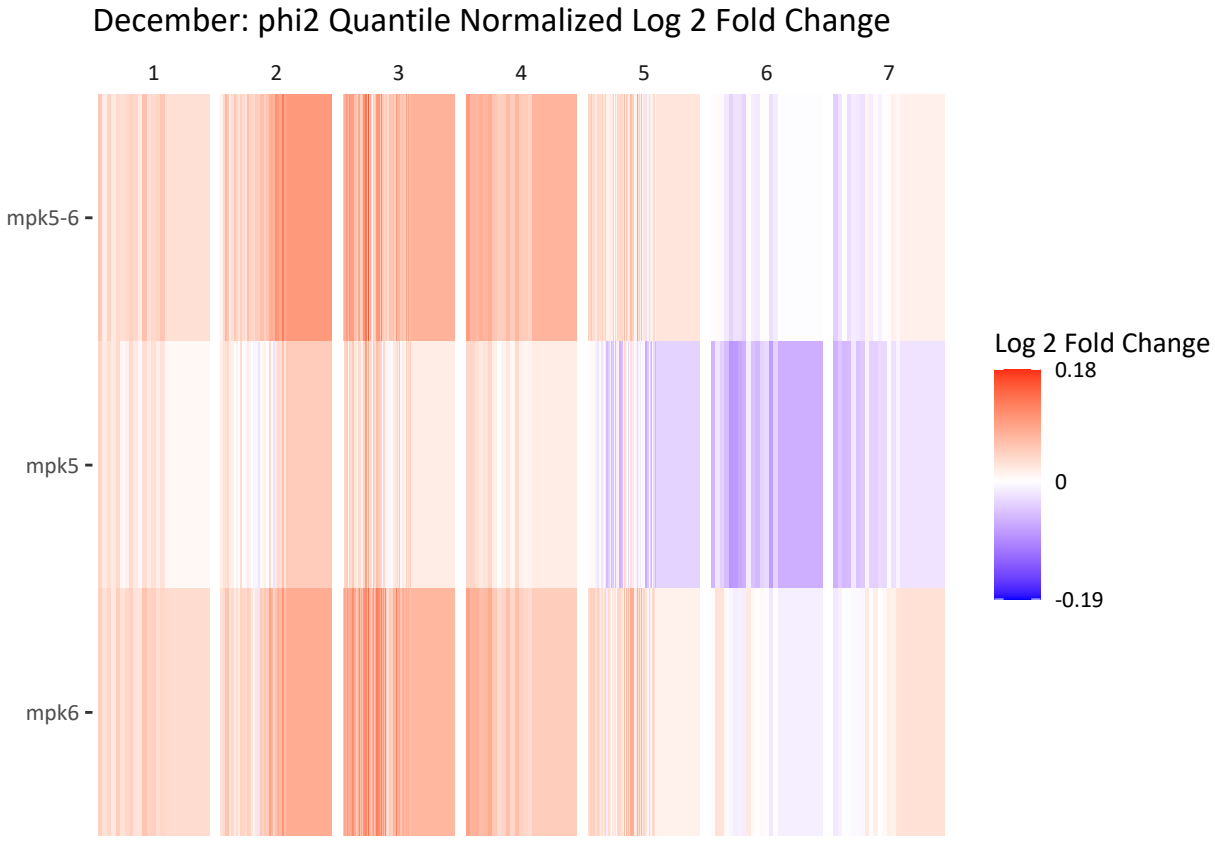

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-8
mpk6
mpk8
Log 2 Fold Change
0.18
0
-0.19

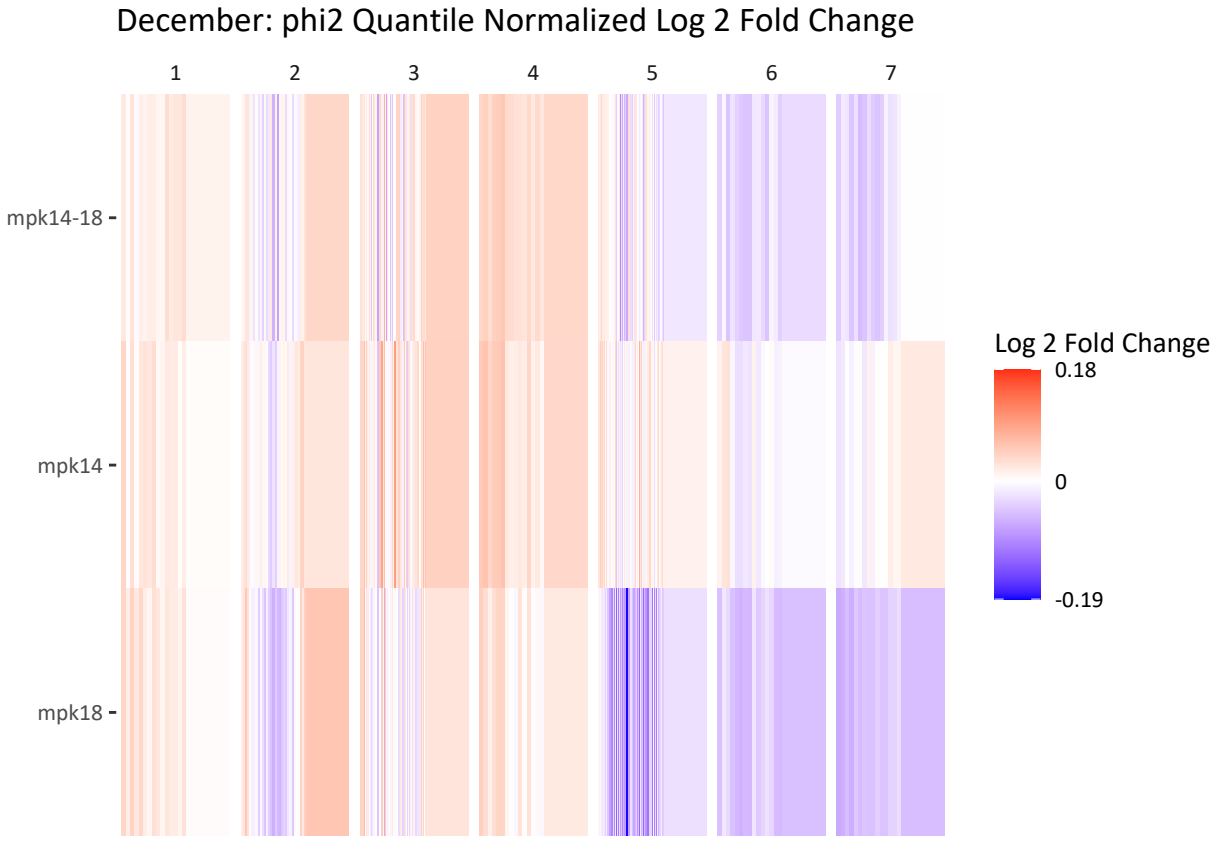

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-18
mpk9
mpk18
Log 2 Fold Change
0.18
0
-0.19

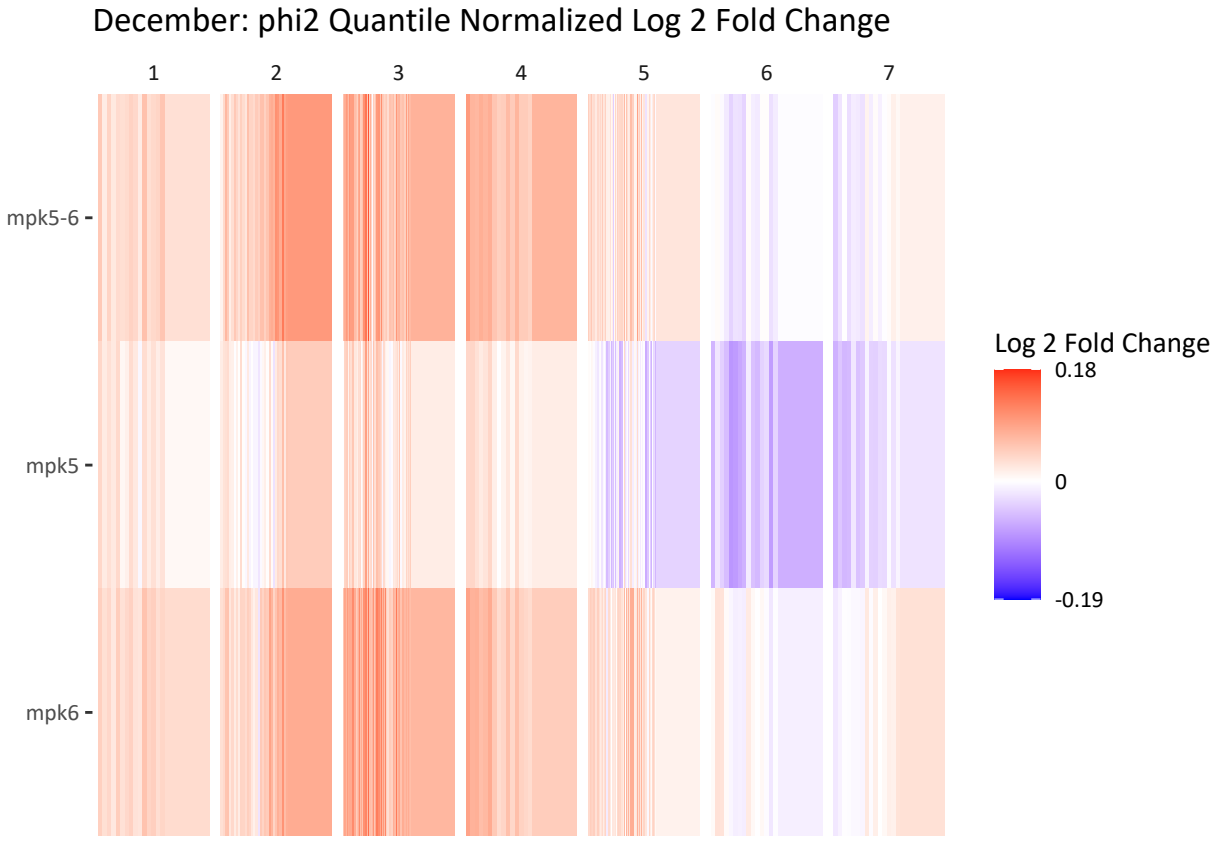

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk6-20
mpk6
mpk20
Log 2 Fold Change
0.18
0
-0.19

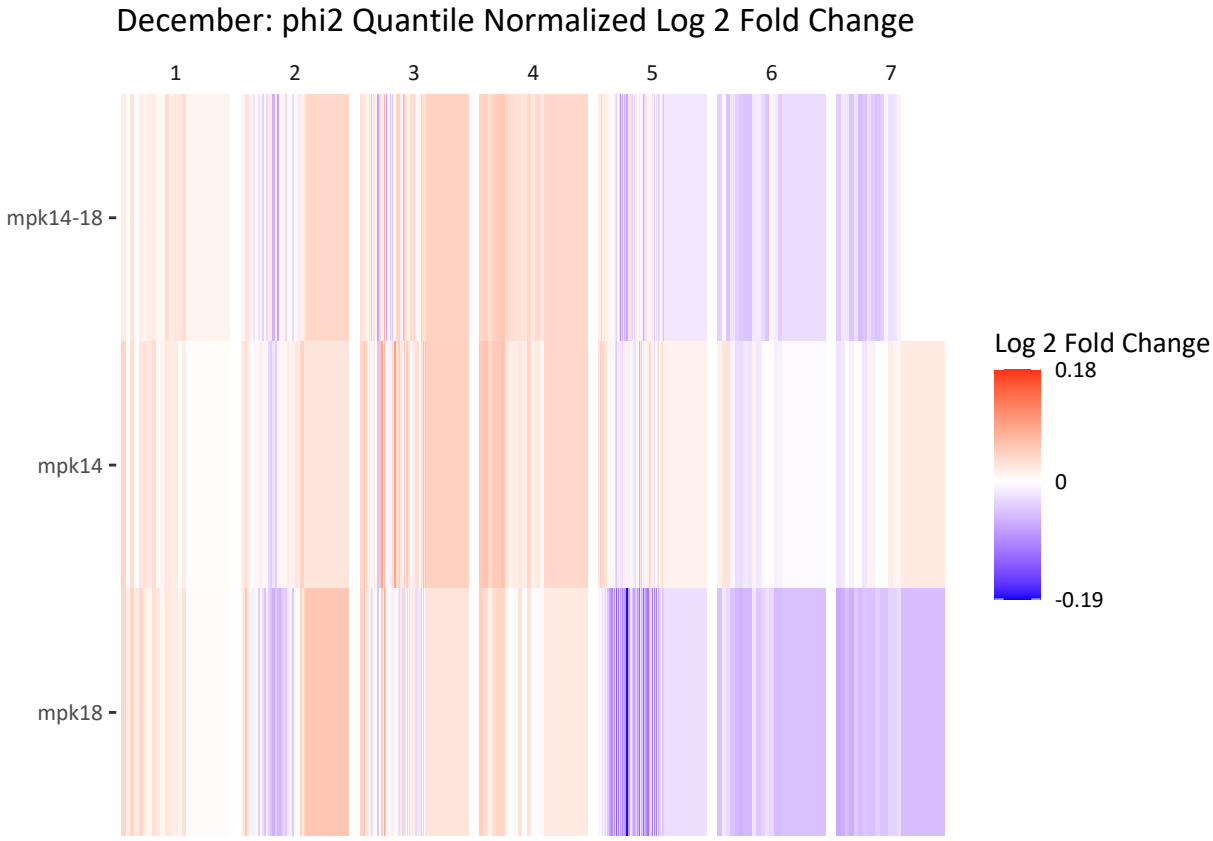

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-16
mpk14
mpk16
Log 2 Fold Change
0.18
0
-0.19

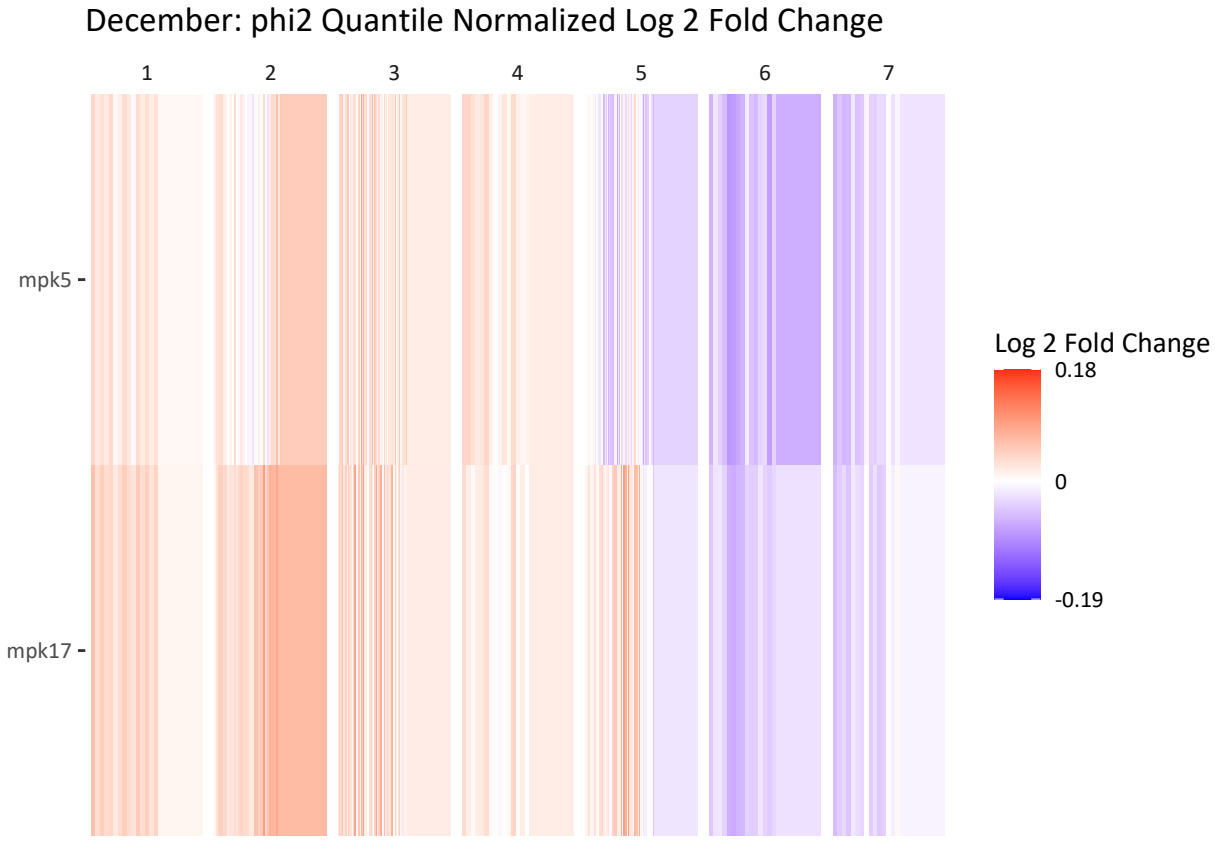

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk18-20
mpk18
mpk20
Log 2 Fold Change
0.18
0
-0.19

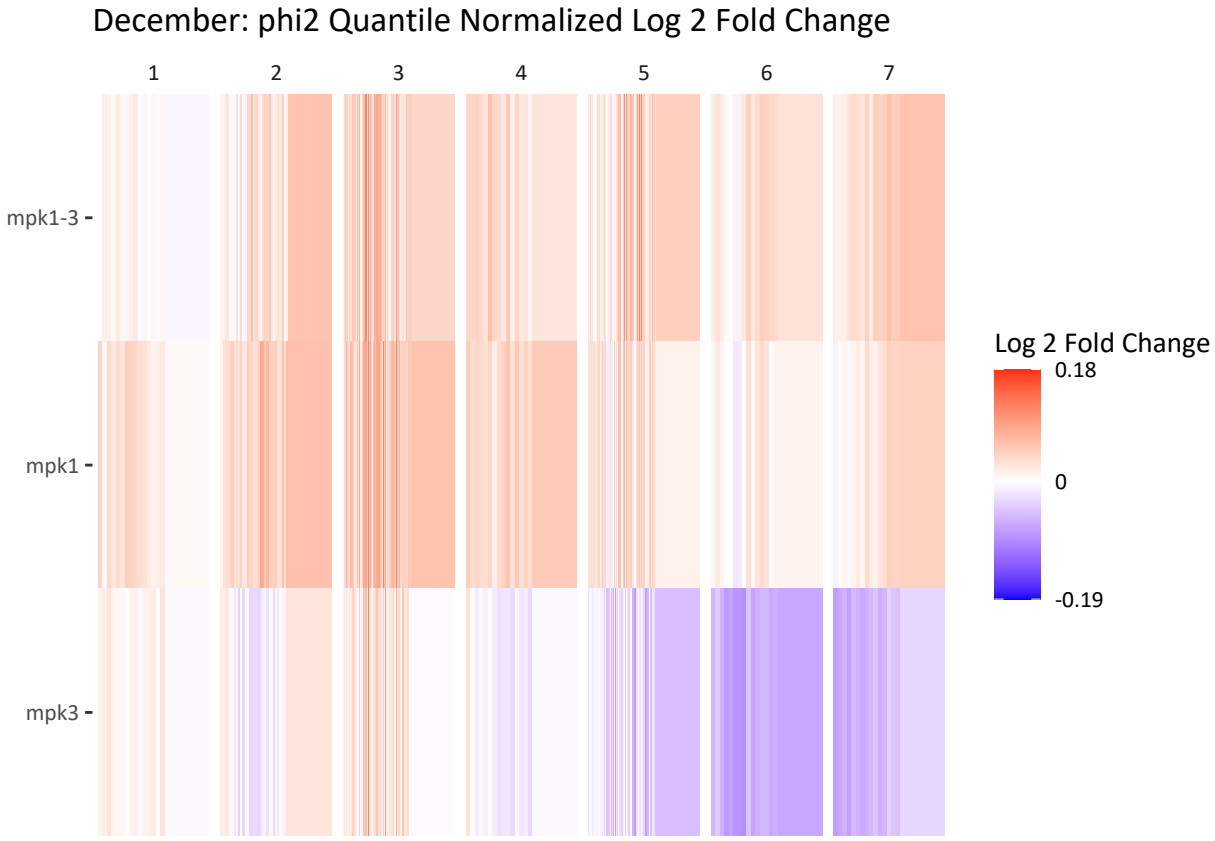

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.18
0
-0.19

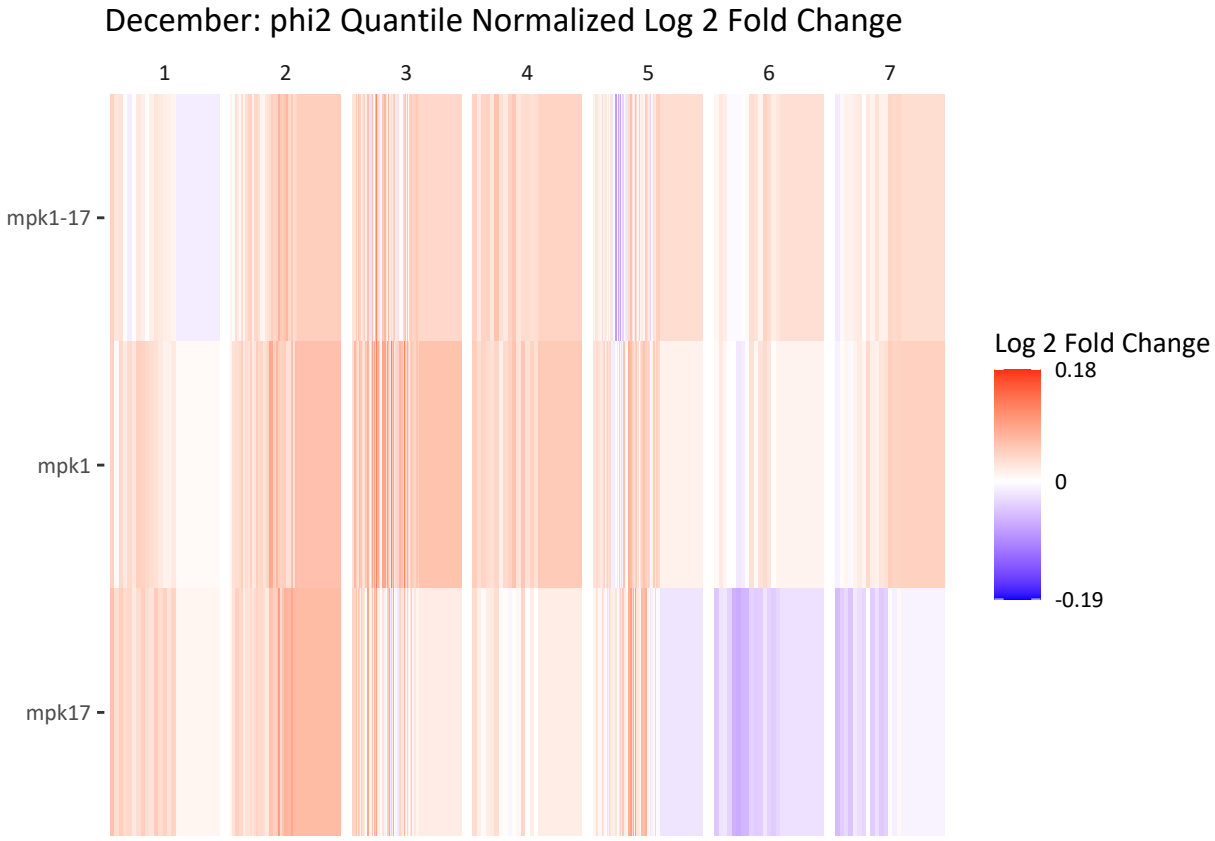

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.18
0
-0.19

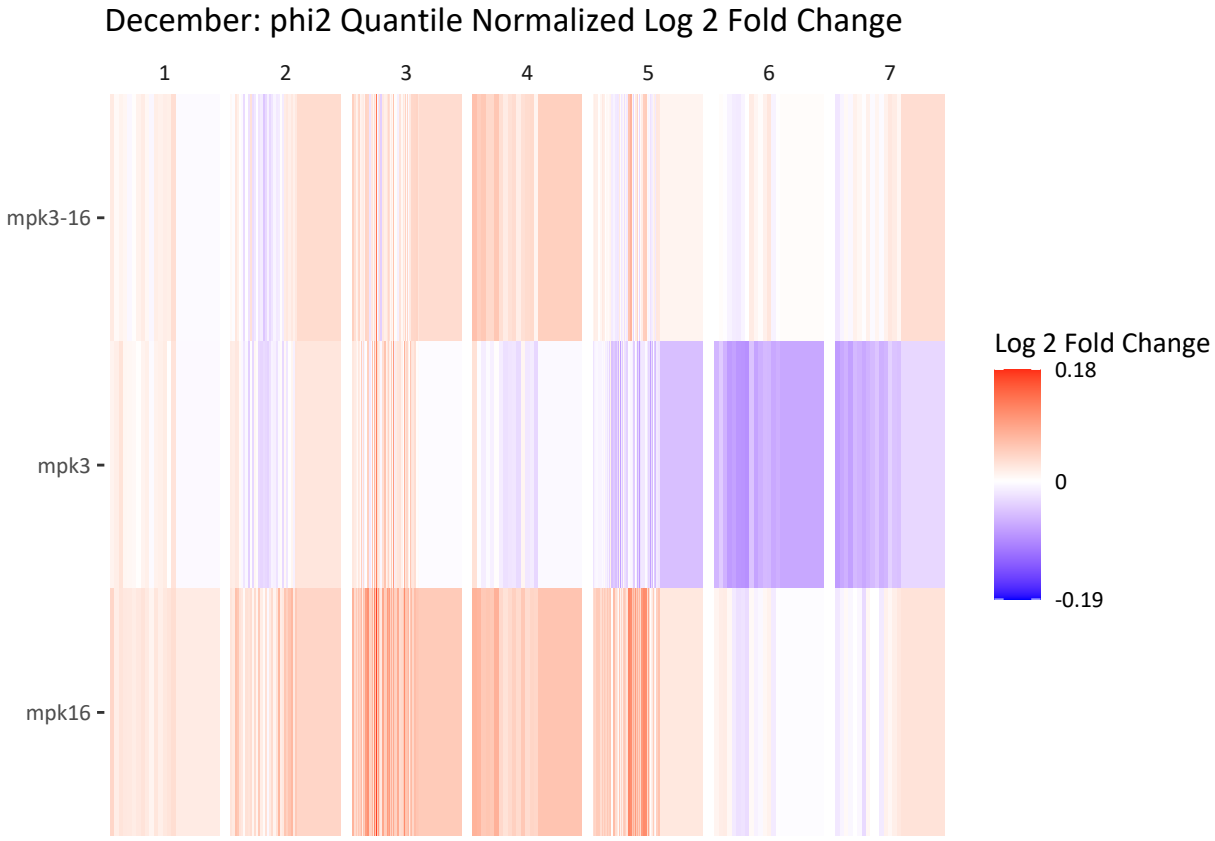

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5-6
mpk5
mpk6
Log 2 Fold Change
0.18
0
-0.19

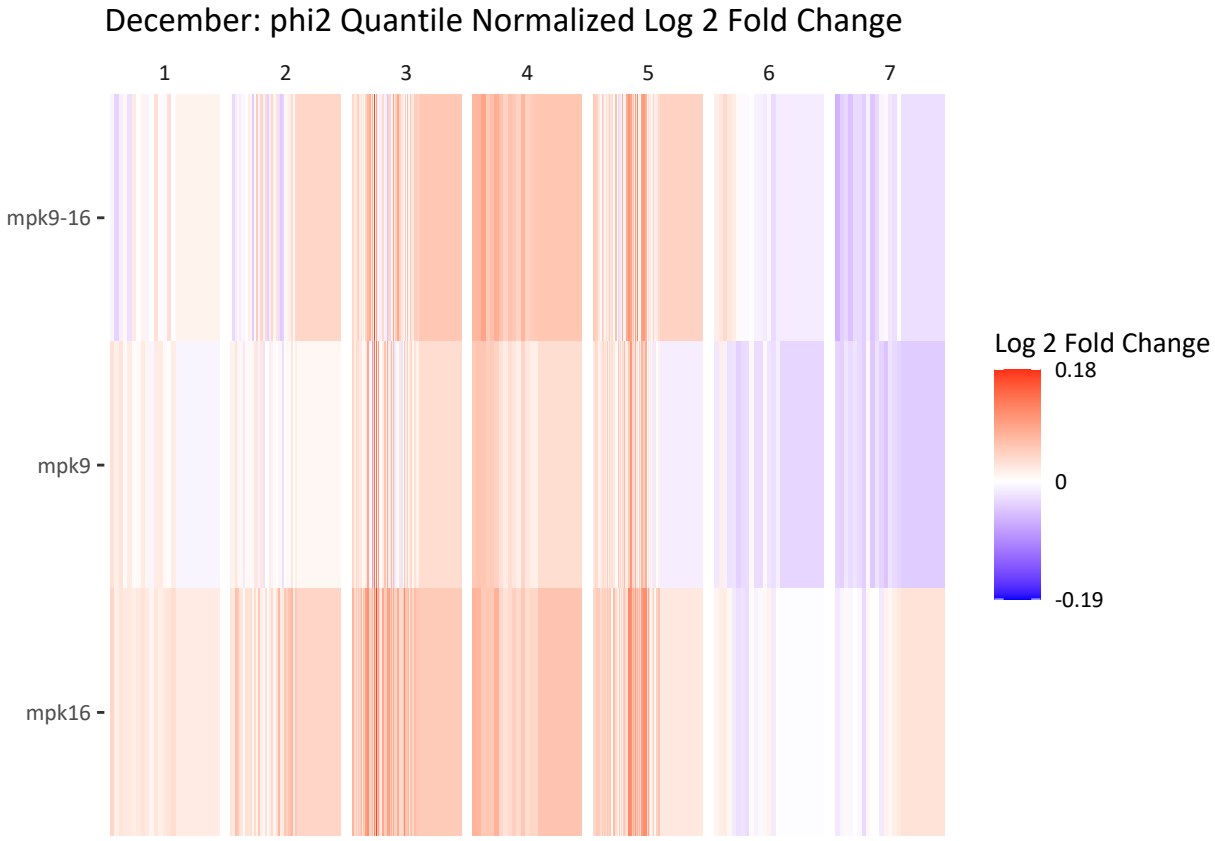

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk14-18
mpk14
mpk18
Log 2 Fold Change
0.18
0
-0.19

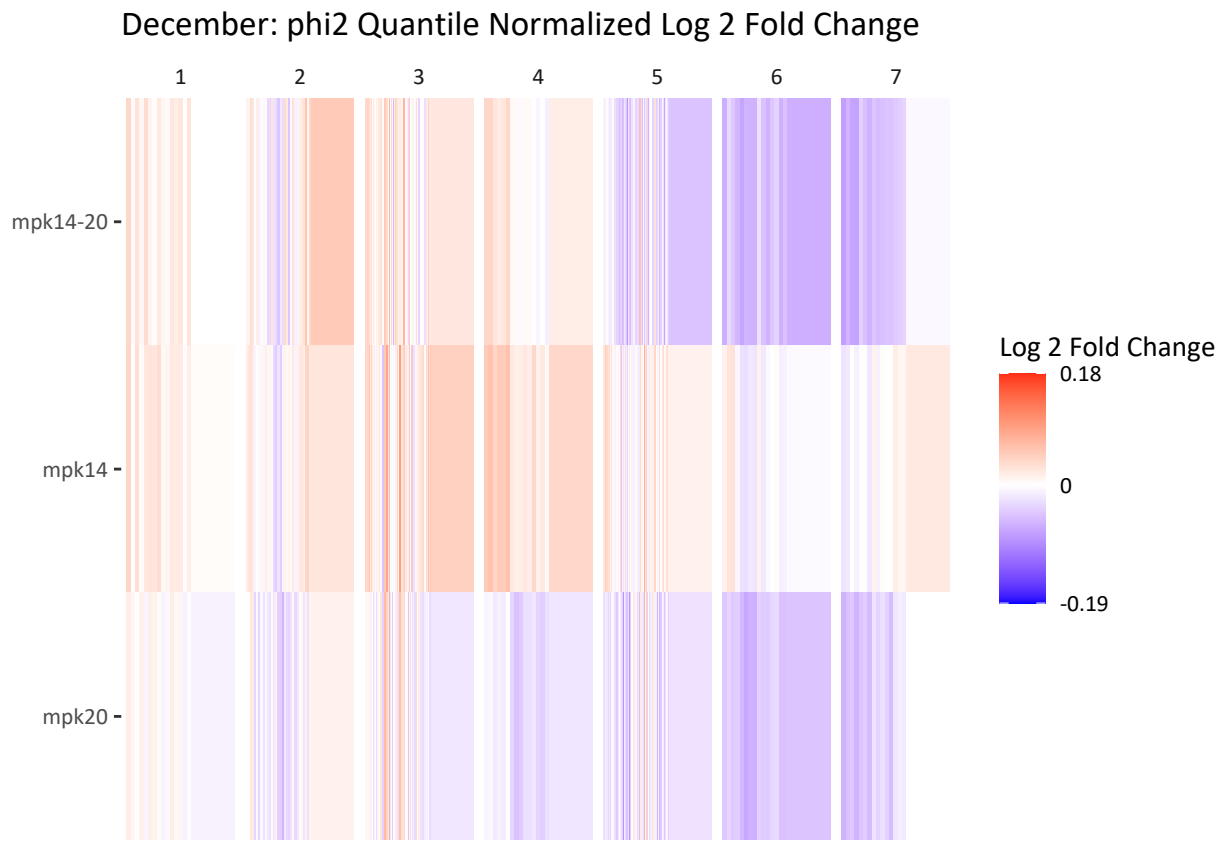

December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk5
mpk17
Log 2 Fold Change
0.18
0
-0.19


December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-3
mpk1
mpk3
Log 2 Fold Change
0.18
0
-0.19


December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk1-17
mpk1
mpk17
Log 2 Fold Change
0.18
0
-0.19


December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk3-16
mpk3
mpk16
Log 2 Fold Change
0.18
0
-0.19


December: phi2 Quantile Normalized Log 2 Fold Change
1
2
3
4
5
6
7
mpk9-16
mpk9
mpk16
Log 2 Fold Change
0.18
0
-0.19



Flucuating light days:

1. Look at the end of the day
2. Look at an "up" point
3. Look at a "down" point
4. Look at the start of the day

Epistasis on Day 3

All time points, all three additive, proportional, selection coefficients. (Three seperate plots, one for each month).