COVID MICE

Jackson Maines

2024-07-31

Packages Used for Project

Here are the packages used in our project.

```
library(tidyverse)
library(Hmisc)
library(ez)
library(lme4)
library(lmerTest)
library(performance)
library(ggrepel)
```

Tidyverse has a variety of packages included inside of it. We will mostly be using dplyr and ggplot2.

Hmisc has a variety of packages included inside of it. We will use this for ease of use in coding.

ez is for our repeated measure anovas.

lme4 and lmerTest are for our mixed models.

performance is also to test our mixed models.

ggrepel is for volcano plots to move point labels away from eachother

Data Cleaning

Some initial data cleaning. The data wasn't formatted nice. Probably a better way to do all of this but idc.

```
  #Calls Data Set
Mice_Weight <- read.csv("Endsley Data 1.1.csv")

  #Removes first two columns
Mice_Weight <- Mice_Weight[,-(1:2)]

  #Replaces column names with first row
names(Mice_Weight) <- Mice_Weight %>%
  slice(1) %>%
  unlist()
Mice_Weight <- Mice_Weight %>%
  slice(-1)

  #Removes extra rows at bottom
Mice_Weight <- Mice_Weight[-(26:36),]

  #Rename Animal # to something less annoying
Mice_Weight <- Mice_Weight %>%
  rename(Animal_Num =`Animal #`)

  #Split Sex and Genotype into 2 columns
Mice_Weight <- Mice_Weight %>%
  separate(col=`Sex/Genotype`, into=c("Sex", "GenoType"), sep="/")

  #Factor Character columns
Mice_Weight <- Mice_Weight %>%
  mutate(across(c(1, 2:4), as.factor))

  #Change character columns to numeric
Mice_Weight <- Mice_Weight %>%
  mutate(across(5:12, as.numeric))

  #Pivot Longer, puts data into long format
    #Fix Date format
Mice_Weight_Long <- Mice_Weight %>%
  pivot_longer(!c("Animal_Num", "Sex", "GenoType", "Infection"),
               names_to = "Dates",
               values_to = "Weight") %>%
  mutate(Dates = mdy(Dates))
```

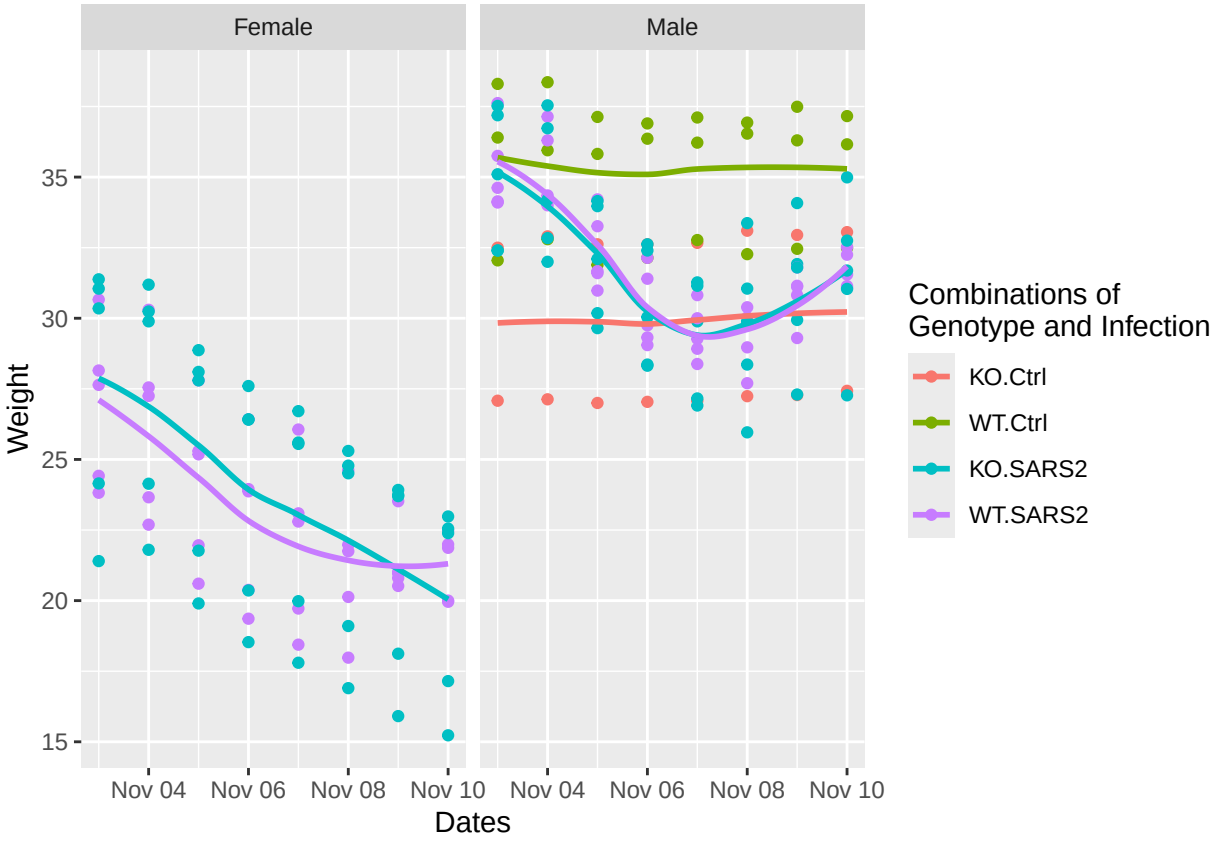
First Plots

These are some of our first exploratory plots of the data.

Sex Split and Combinations of Genotype and Infection

```
#This Plots shows a female and male split. Each colour represents all the
#   combination of genotype and infection a mice could have.
#   The lines represent average weights of each combination.
ggplot(data = Mice_Weight_Long, aes(x = Dates,
                                    y = Weight,
                                    color = interaction(GenoType, Infection))) +
  geom_point() +
  geom_smooth(mapping = aes(group = interaction(GenoType, Infection)),
              se = FALSE) +
  facet_wrap( ~ Sex) +
  labs(color = "Combinations of
Genotype and Infection")
```

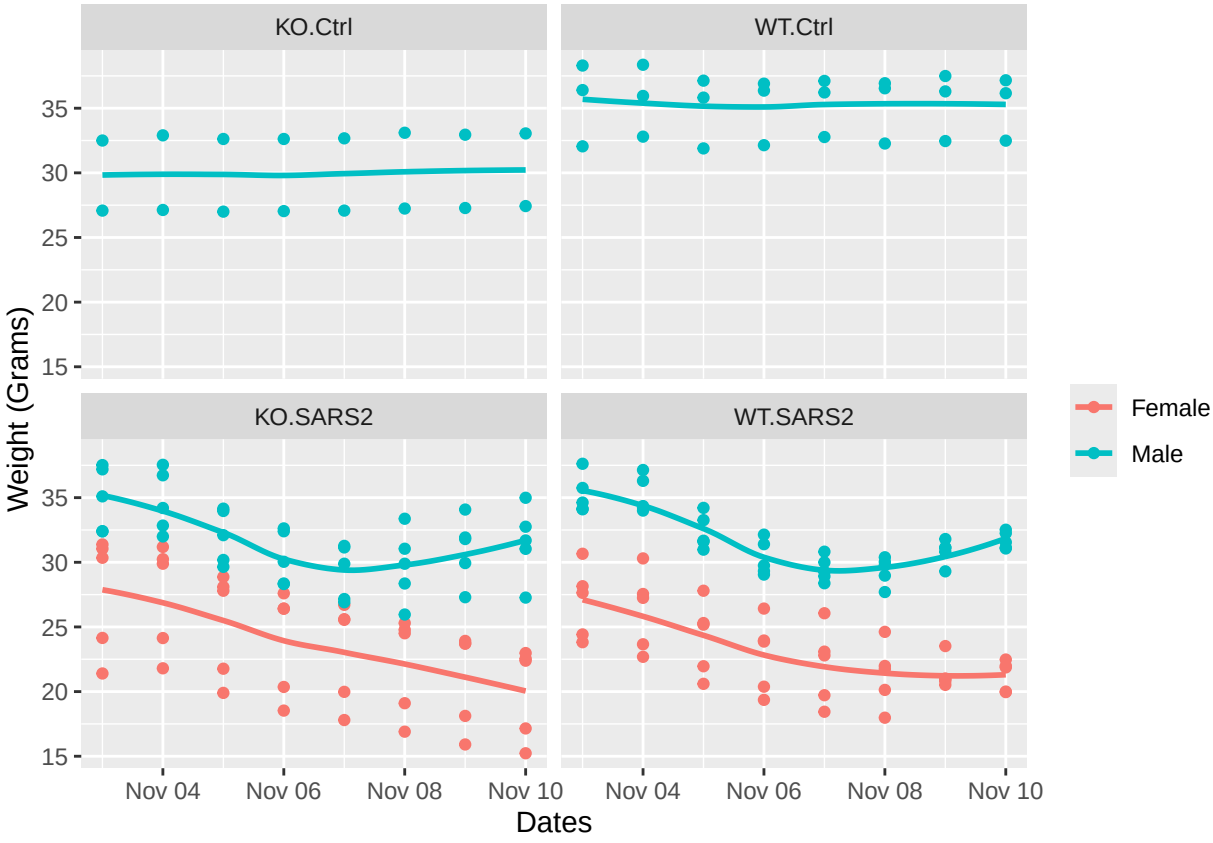
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
ggplot(data = Mice_Weight_Long, aes(x = Dates,
                                    y = Weight,
                                    color = Sex)) +
  geom_point() +
  geom_smooth(mapping = aes(group = Sex),
              se = FALSE) +
```

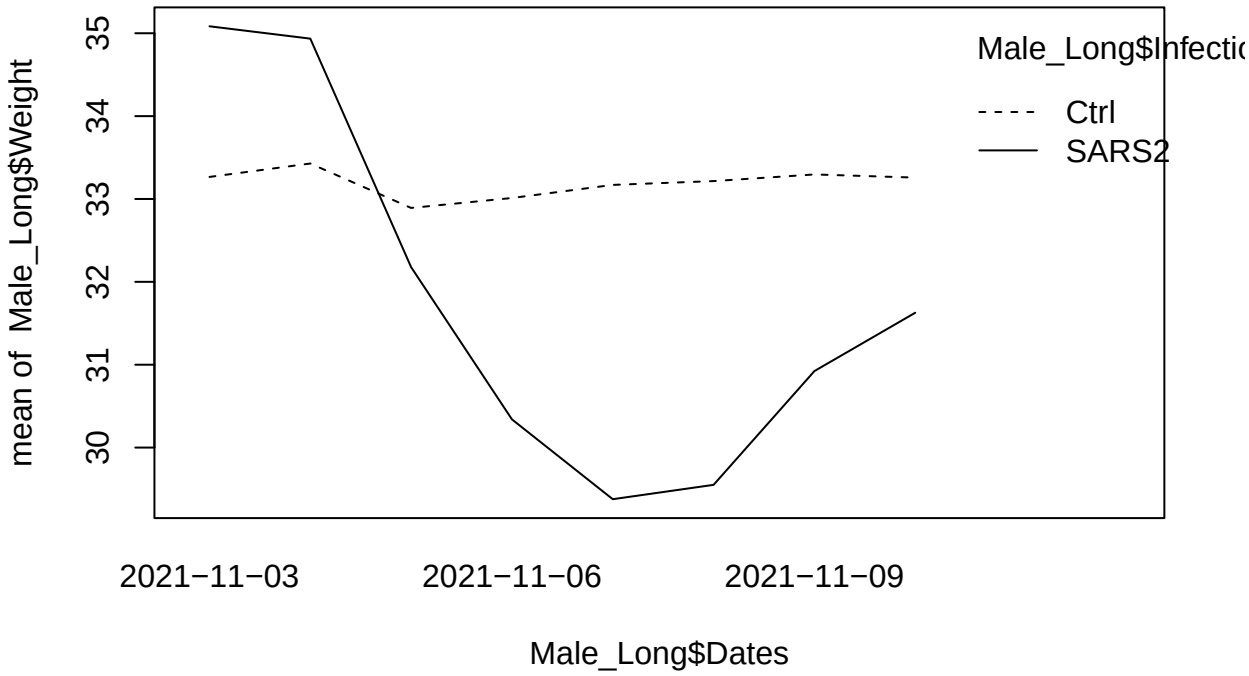
```
  facet_wrap( ~ interaction(GenoType, Infection)) +
  labs(y = "Weight (Grams)",
       color = "")
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Interactions Plots

```
  #This filter our long format data to just male mice.
Male_Long <- Mice_Weight_Long %>%
  filter(Sex == "Male")

#Interaction Plots
#   This shows average weights on the y and dates on the x.
#   With splits in either infection or genotype.
interaction.plot(Male_Long$Dates,
                 Male_Long$Infection,
                 Male_Long$Weight)
```

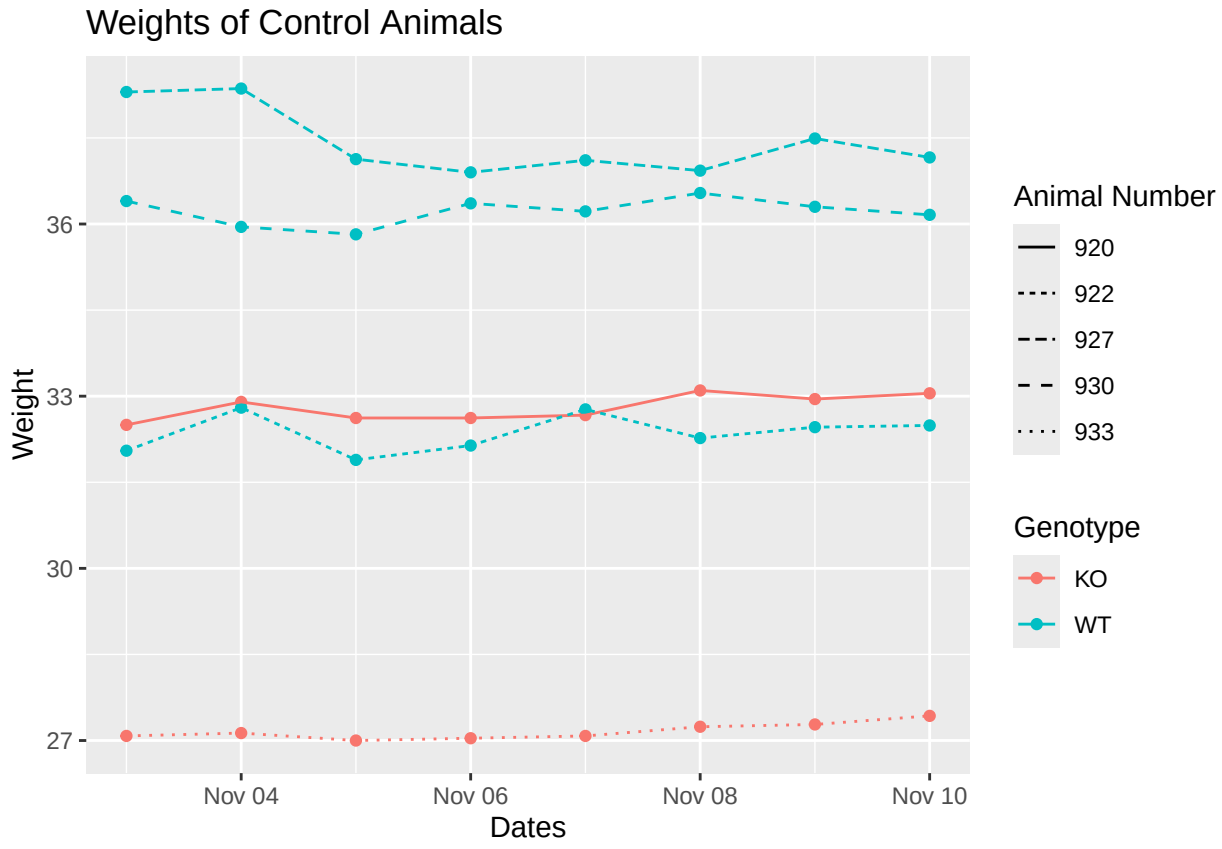



```
interaction.plot(Male_Long$Dates,
                 Male_Long$GenoType,
                 Male_Long$Weight)
```


mean of Male_Long$Weight
30
31
32
33
34
35
2021−11−03
2021−11−05
2021−11−07
2021−11−09
Male_Long$Dates
Male_Long$Gen
WT
KO

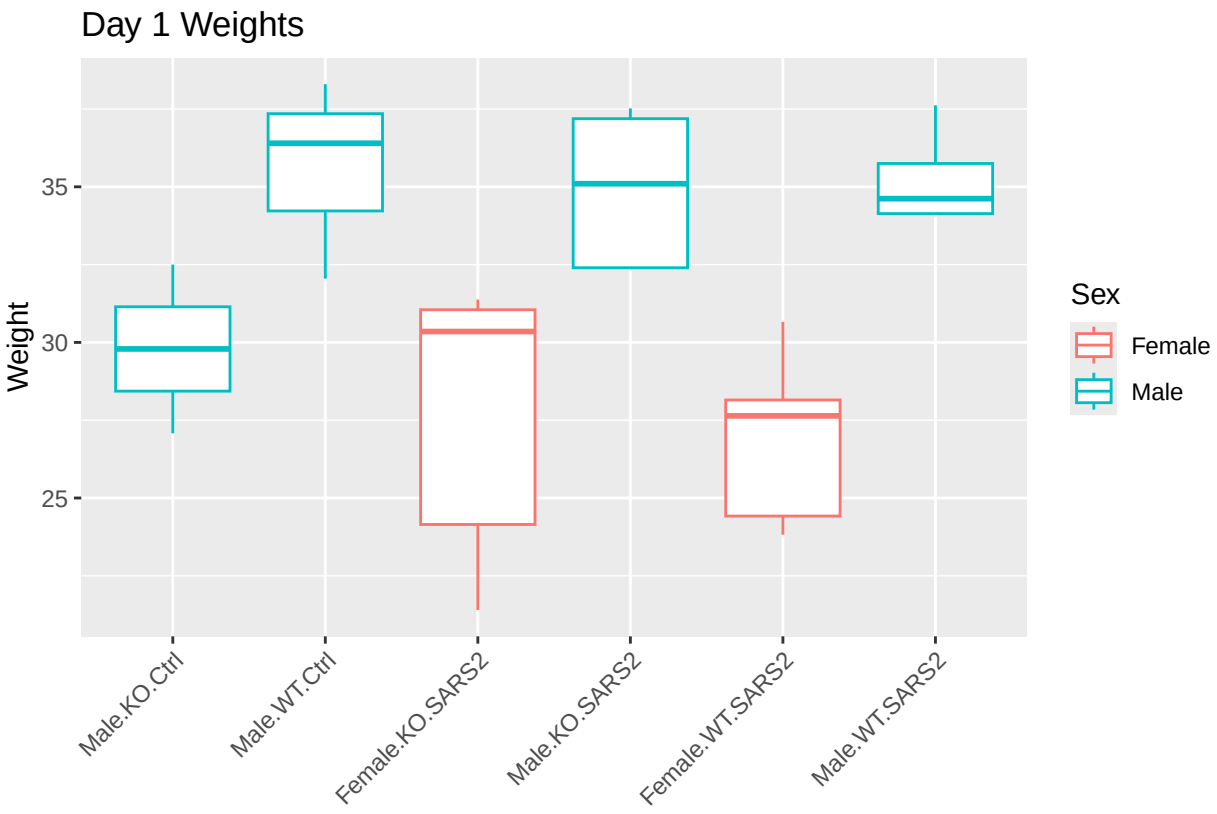
Weights of Male Controls

```
  #This filter our long format data to male control mice.
Male_Long_Ctrl <- Mice_Weight_Long %>%
  filter(Sex == "Male") %>%
  filter(Infection == "Ctrl")

#This plot shows just male control. With the colour based on genotype and
#   the linetype for each individual subject.
ggplot(data = Male_Long_Ctrl, aes(x = Dates,
                                  y = Weight,
                                  color = GenoType,
                                  linetype = Animal_Num)) +
  geom_point() +
  geom_line() +
  labs(color = "Genotype",
       linetype = "Animal Number",
       title = "Weights of Control Animals")
```



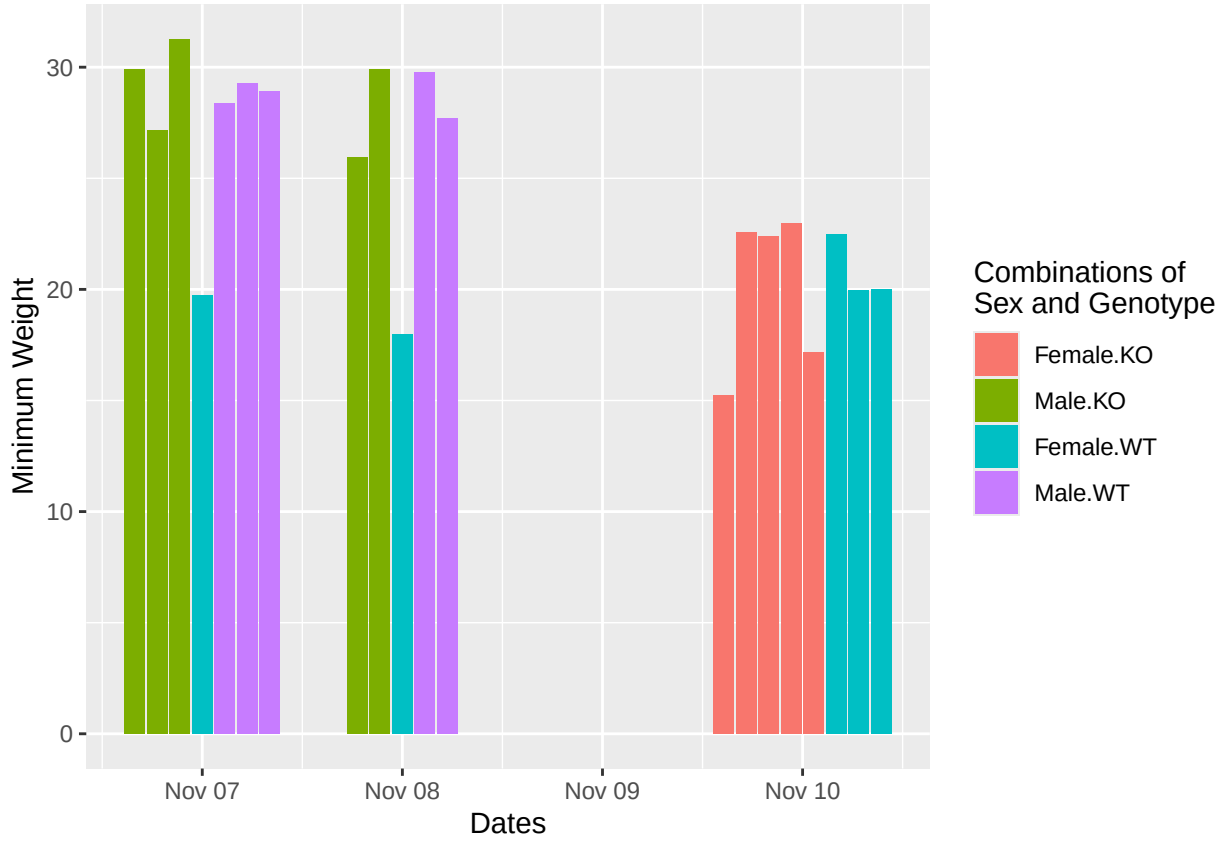
Boxplots of Day 1 Weight

```
Mice_Day_1 <- Mice_Weight_Long %>%
  filter(Dates == "2021-11-03")

ggplot(data = Mice_Day_1, aes(x = interaction(Sex, GenoType, Infection),
                              y = Weight,
                              color = Sex)) +
  geom_boxplot() +
  labs(x = " ",
       y = "Weight",
       title = "Day 1 Weights") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```



Minimum Weight of Infected Mice with Date

```
#This groups each animal num and finds where they have the lowest weight.
# Then we remove all the days except the for the day where they are the lightest.
# We also remove the mice that werent infected.

Mice_Min <- Mice_Weight_Long %>%
  group_by(Animal_Num) %>%
  filter(Weight == min(Weight)) %>%
  filter(Infection != "Ctrl")

#This plot shows what days infected mice bottom out in weight and their weight.
# The colour is the combinations of Sex and Genotype.

ggplot(data = Mice_Min, aes(x = Dates,
                            y = Weight,
                            fill = interaction(Sex, GenoType))) +
  geom_bar(stat = "identity",
           position = position_dodge2(preserve = "single")) +
  labs(x = "Dates",
       y  = "Minimum Weight",
       fill = "Combinations of
Sex and Genotype")
```



2-Way ANOVA of Weights on Day 1

```r
# Mice_Day_1 filters to just day 1 of the Mice_Weight

# 2-Way ANOVA, Weight on GenoType and Infection
two_way_aov <- aov(Weight ~ GenoType + Infection, data = Mice_Day_1)

summary(two_way_aov)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## GenoType     1    7.3   7.348   0.309  0.584
## Infection    1   15.5  15.536   0.653  0.428
## Residuals   22  523.6  23.800
```

2-Way ANOVA of Weights on Day 7

```
#Filter to last day of study
Mice_Day_7 <- Mice_Weight_Long %>%
  filter(Dates == "2021-11-10")

# 2-Way ANOVA, Weight on GenoType and Infection
two_way_aov <- aov(Weight ~ GenoType + Infection, data = Mice_Day_7)

summary(two_way_aov)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## GenoType     1   24.2   24.15   0.718 0.4059
## Infection    1  192.7  192.66   5.727 0.0257 *
## Residuals   22  740.1   33.64
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Repeated Measure ANOVA

We did Repeated Measure ANOVA for both females and males. Since females didnt have any controls we only need to do a two-way repeated measures ANOVA. Because we had only male controls we needed to do a three-way repeated measures ANOVA.

2-Way Repeated Measures ANOVA

```
#2-Way Repeated Measures ANOVA
    #This filter our long format data to just female mice.
Female_Long <- Mice_Weight_Long %>%
  filter(Sex == "Female")

ezANOVA(data = Female_Long,
        dv = Weight,
        wid = Animal_Num,
        within = Dates,
        between = GenoType,
        detailed = TRUE)
```

```
## Warning: You have removed one or more Ss from the analysis. Refactoring
## "Animal_Num" for ANOVA.
```

```
## Warning: Converting "Dates" to factor for ANOVA.
```

```
## $ANOVA
##           Effect DFn DFd          SSn       SSd            F            p p<.05
## 1    (Intercept)   1   8 44364.432080 680.96371 521.19584175 1.436797e-08     *
## 2       GenoType   1   8     6.395805 680.96371   0.07513828 7.909422e-01
## 3          Dates   7  56   446.736120  50.29228  71.06236990 1.561816e-25     *
## 4 GenoType:Dates   7  56    12.860595  50.29228   2.04573644 6.506663e-02
##           ges
## 1 0.983784348
## 2 0.008670493
## 3 0.379235236
## 4 0.017283037
##
## $`Mauchly's Test for Sphericity`
##           Effect            W           p p<.05
## 3          Dates 8.578261e-10 1.437487e-11     *
## 4 GenoType:Dates 8.578261e-10 1.437487e-11     *
##
## $`Sphericity Corrections`
##           Effect       GGe        p[GG] p[GG]<.05       HFe        p[HF]
## 3          Dates 0.1616483 1.047419e-05         * 0.1702157 6.499806e-06
## 4 GenoType:Dates 0.1616483 1.872512e-01           0.1702157 1.856474e-01
##   p[HF]<.05
## 3         *
## 4
```

This 2-way repeated measure anova shows that only dates is significant. This makes since logcially since the mices weights change day to day.

3-Way Repeated Measures ANOVA

```
#3-Way Repeated Measures ANOVA
    #This filter our long format data to just male mice.
Male_Long <- Mice_Weight_Long %>%
  filter(Sex == "Male")

ezANOVA(data = Male_Long,
        dv = Weight,
        wid = Animal_Num,
        within = Dates,
        between = c(GenoType, Infection),
        detailed = TRUE)
```

```
## Warning: You have removed one or more Ss from the analysis. Refactoring
## "Animal_Num" for ANOVA.
```

```
## Warning: Converting "Dates" to factor for ANOVA.
```

```
## Warning: Data is unbalanced (unequal N per group). Make sure you specified a
## well-considered value for the type argument to ezANOVA().
```

```
## $ANOVA
##                     Effect DFn DFd          SSn      SSd            F
## 1              (Intercept)   1  11 1.246669e+05 439.7986 3118.1008017
## 2                 GenoType   1  11 9.833481e+01 439.7986    2.4594963
## 3                Infection   1  11 4.184872e+01 439.7986    1.0466972
## 5                    Dates   7  77 2.378645e+02  36.1607   72.3578271
## 4       GenoType:Infection   1  11 1.770484e+02 439.7986    4.4282379
## 6           GenoType:Dates   7  77 1.781352e+00  36.1607    0.5418829
## 7          Infection:Dates   7  77 1.102655e+02  36.1607   33.5424957
## 8 GenoType:Infection:Dates   7  77 3.931701e-01  36.1607    0.1196014
##              p p<.05          ges
## 1 7.496974e-15     * 0.9961966736
## 2 1.451145e-01       0.1712272796
## 3 3.282424e-01       0.0808189967
## 5 3.005414e-31     * 0.3332258387
## 4 5.915429e-02       0.2711276252
## 6 8.001135e-01       0.0037287004
## 7 6.413156e-21     * 0.1880942015
## 8 9.967889e-01       0.0008253765
##
## $`Mauchly's Test for Sphericity`
##                     Effect            W            p p<.05
## 5                    Dates 0.0002265625 2.152578e-05     *
## 6           GenoType:Dates 0.0002265625 2.152578e-05     *
## 7          Infection:Dates 0.0002265625 2.152578e-05     *
## 8 GenoType:Infection:Dates 0.0002265625 2.152578e-05     *
##
## $`Sphericity Corrections`
##                     Effect       GGe        p[GG] p[GG]<.05       HFe
## 5                    Dates 0.2857111 2.113511e-10         * 0.3492012
## 6           GenoType:Dates 0.2857111 5.892164e-01           0.3492012
## 7          Infection:Dates 0.2857111 2.083826e-07         * 0.3492012
## 8 GenoType:Infection:Dates 0.2857111 8.878449e-01           0.3492012
##          p[HF] p[HF]<.05
```

```
## 5 2.865625e-12         *
## 6 6.225926e-01
## 7 1.269971e-08         *
## 8 9.208081e-01
```

This 3-way repeated measure anova shows that dates and the interaction between dates and infection are significant. For the same reason as before dates makes logical since. The interaction between dates and infection is understandable as mice with SARS2 are going to loose more weight.

Mixed Models

These are some of the mixed models we created for the project.

More to come...

```
#Mixed Models

  #Test Models
test_model <- lmer(Weight ~ Dates + (1|Animal_Num), data = Mice_Weight_Long)
test_model_1 <- lmer(Weight ~ Sex + GenoType + Infection + (1 | Dates) + (1 | Animal_Num),
                     data = Mice_Weight_Long)
test_model_1.1 <- lmer(Weight ~ Sex + GenoType + Infection + (1 | Animal_Num),
                       data = Mice_Weight_Long)
test_model_2 <- lmer(Weight ~ Sex + GenoType + Infection + (1 | Dates),
                     data = Mice_Weight_Long)
test_model_3 <- lmer(Weight ~ 1 +Sex + GenoType + Infection + Dates + (1 + Dates | Animal_Num),
                     REML = T,
                     data = Mice_Weight_Long)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
anova(test_model_1, test_model_3, test = "Chisq", refit = T)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: Mice_Weight_Long
## Models:
## test_model_1: Weight ~ Sex + GenoType + Infection + (1 | Dates) + (1 | Animal_Num)
## test_model_3: Weight ~ 1 + Sex + GenoType + Infection + Dates + (1 + Dates | Animal_Num)
##              npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## test_model_1    7 817.54 840.63 -401.77   803.54
## test_model_3    9 849.06 878.74 -415.53   831.06     0  2          1
```

```
summary(test_model_3)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Weight ~ 1 + Sex + GenoType + Infection + Dates + (1 + Dates |
##     Animal_Num)
##    Data: Mice_Weight_Long
##
## REML criterion at convergence: 828.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.31639 -0.73889 -0.07401  0.81595  2.22433
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.  Corr
##  Animal_Num (Intercept) 2.487e+00 1.5771675
##             Dates       1.544e-08 0.0001243 0.31
##  Residual               2.489e+00 1.5777414
## Number of obs: 200, groups:  Animal_Num, 25
##
## Fixed effects:
```

```
##                  Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)     1.242e+04  9.221e+02  1.153e+01  13.471 2.09e-08 ***
## SexMale         8.200e+00  1.456e+00  1.369e+01   5.631 6.74e-05 ***
## GenoTypeWT      8.594e-01  1.308e+00  1.369e+01   0.657    0.522
## InfectionSARS2 -1.351e+00  1.789e+00  1.369e+01  -0.755    0.463
## Dates          -6.546e-01  4.869e-02  1.153e+01 -13.444 2.14e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) SexMal GnTyWT ISARS2
## SexMale     -0.002
## GenoTypeWT  -0.001  0.000
## InfctnSARS2 -0.002  0.407  0.073
## Dates       -1.000  0.000  0.000  0.000
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
AIC(logLik(test_model_1))
```

```
## [1] 809.3267
```

```
BIC(logLik(test_model_1))
```

```
## [1] 832.4149
```

```
compare_performance(test_model,
                    test_model_1,
                    test_model_2,
                    test_model_3)
```

```
## # Comparison of Model Performance Indices
##
## Name         |           Model |  AIC (weights) | AICc (weights) |  BIC (weights) | R2 (cond.) | R2
## ------------------------------------------------------------------------------------------------------
## test_model   | lmerModLmerTest |  871.9 (<.001) |  872.1 (<.001) |  885.1 (<.001) |      0.917 |
## test_model_1 | lmerModLmerTest |  817.8 (>.999) |  818.4 (>.999) |  840.9 (>.999) |      0.941 |
## test_model_2 | lmerModLmerTest | 1030.2 (<.001) | 1030.6 (<.001) | 1050.0 (<.001) |      0.709 |
## test_model_3 | lmerModLmerTest |  851.1 (<.001) |  852.1 (<.001) |  880.8 (<.001) |      0.926 |
```

We decide that either model_test_1 or test_model_3 are the best. I believe model_1 has random intercept for both date and animal number. While model_3 has dates as a correlated random slope and animal number as a random intercept.

Dilution Data Cleaning

This is the second page of the spreadsheet.

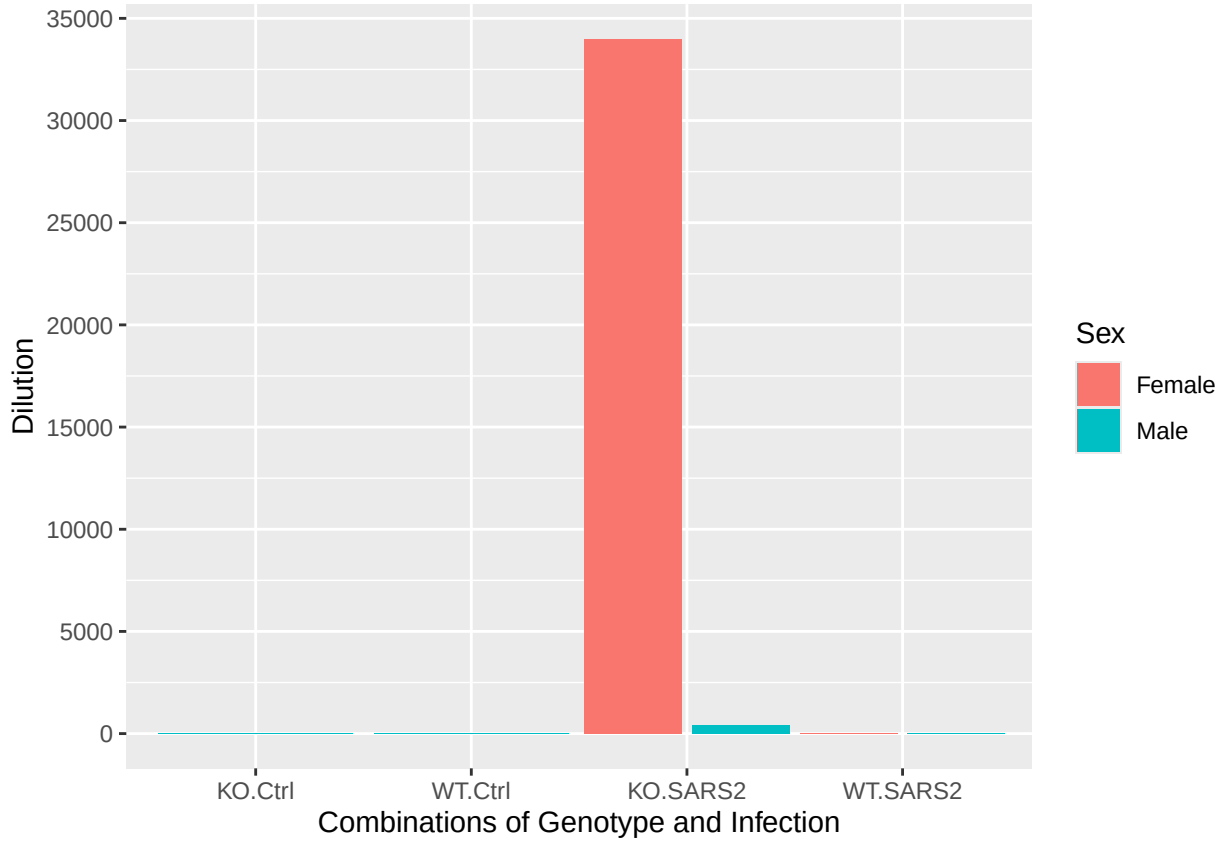
The data gets cleaned then our corrected for dilution row gets moved to the data set we have been work with.

```
#This is the second data, Dilution info
Mice_Dilution <- read.csv("Endsley Data 1.2.csv")

#Removes First Row
Mice_Dilution <- Mice_Dilution[-1,]

#Shift Dilution columns up one
shift <- function(x, n){
  c(x[-(seq(n))], rep(NA, n))
}
Mice_Dilution$X.6 <- shift(Mice_Dilution$X.6, 1)

#Avg Double Dillution and replace
Mice_Dilution$X.6[13] <- mean(c(Mice_Dilution$X.6[13], Mice_Dilution$X.6[14]))

#Move one value up that was missed placed
Mice_Dilution$X.6[21] <- Mice_Dilution$X.6[22]

#Remove extra rows
Mice_Dilution <- Mice_Dilution %>%
  filter(!(X.2 == ""))

#Combine Dilution columns with OG Data Set
Mice_Weight$Dilution <- Mice_Dilution$X.6

#Change NA's to 0's
Mice_Weight <- Mice_Weight %>%
  mutate(Dilution = replace_na(Dilution, 0))
```

Plots with New Dilution Data

Interaction of Dilution with Genotype, Infection,

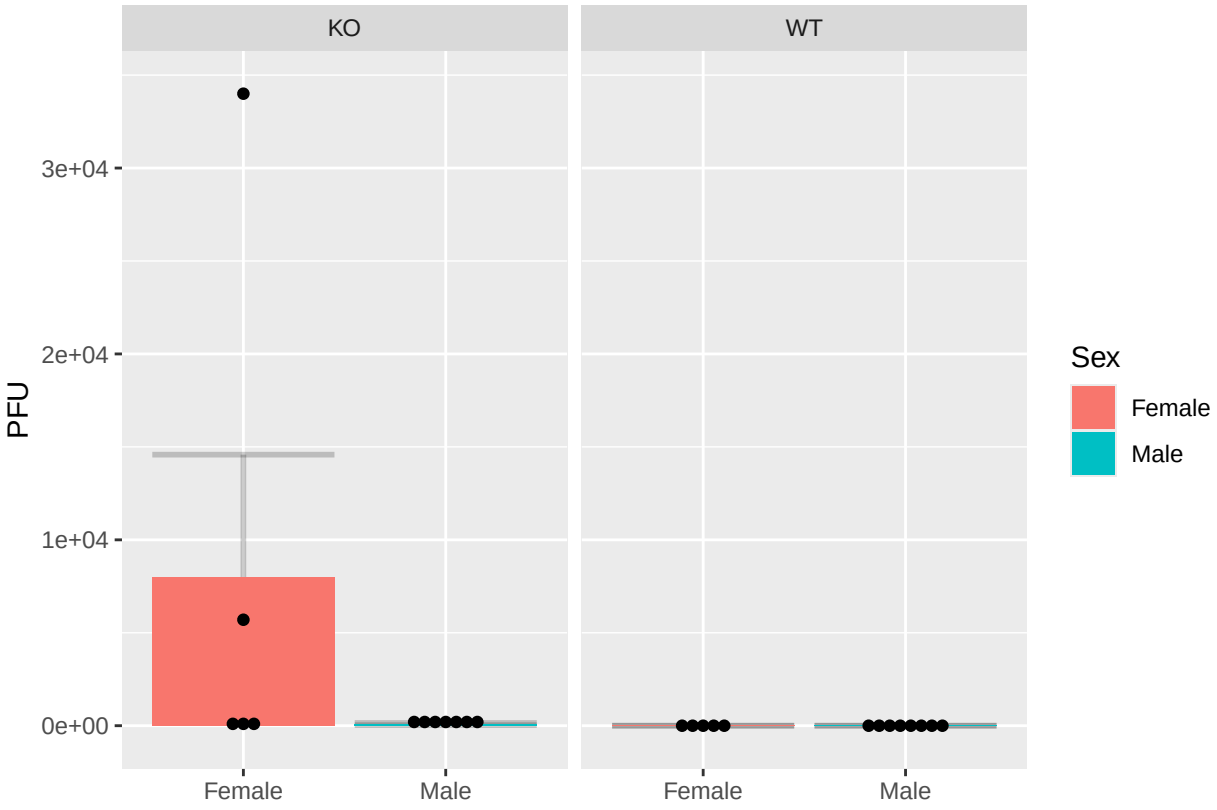
```
#Plot data with interaction with new Dilution
  #This plot shows the max dilution for each sex for each combination of
  #genotype and infection.
ggplot(data = Mice_Weight, aes(x = interaction(GenoType, Infection),
                               y = Dilution,
                               fill = Sex)) +
  geom_bar(stat = "identity",
           position = position_dodge(width = 1)) +
  scale_y_continuous(n.breaks = 10) +
  labs(x = "Combinations of Genotype and Infection")
```



Barplot of PFU with Error Bars and Points

```
#This makes a new column of average dilution based on group
Mice_Weight <- Mice_Weight %>%
  group_by(interaction(GenoType, Sex)) %>%
  mutate(Avg_Dil = mean(Dilution))


# This plots a average barplot, with every invidual subject plotted
# with an error bar.
ggplot(data = Mice_Weight, aes(x = Sex,
                               y = Dilution)) +
  facet_grid(~ GenoType) +
  stat_summary(linewidth = 1,
               geom = "errorbar",
               colour = "BLACK",
               alpha = .2) +
  geom_bar(mapping = aes(y = Avg_Dil,
                         fill = Sex),
           stat = "identity",
           position = position_dodge(width = 1)) +
  geom_dotplot(binaxis='y',
               dotsize = .5,
               stackgroups = TRUE,
               binpositions = "bygroup",
               stackdir = "center") +
  scale_y_continuous(n.breaks = 10) +
  labs(x = "",
       y = "PFU") +
  scale_y_continuous(labels = scales::scientific)
```

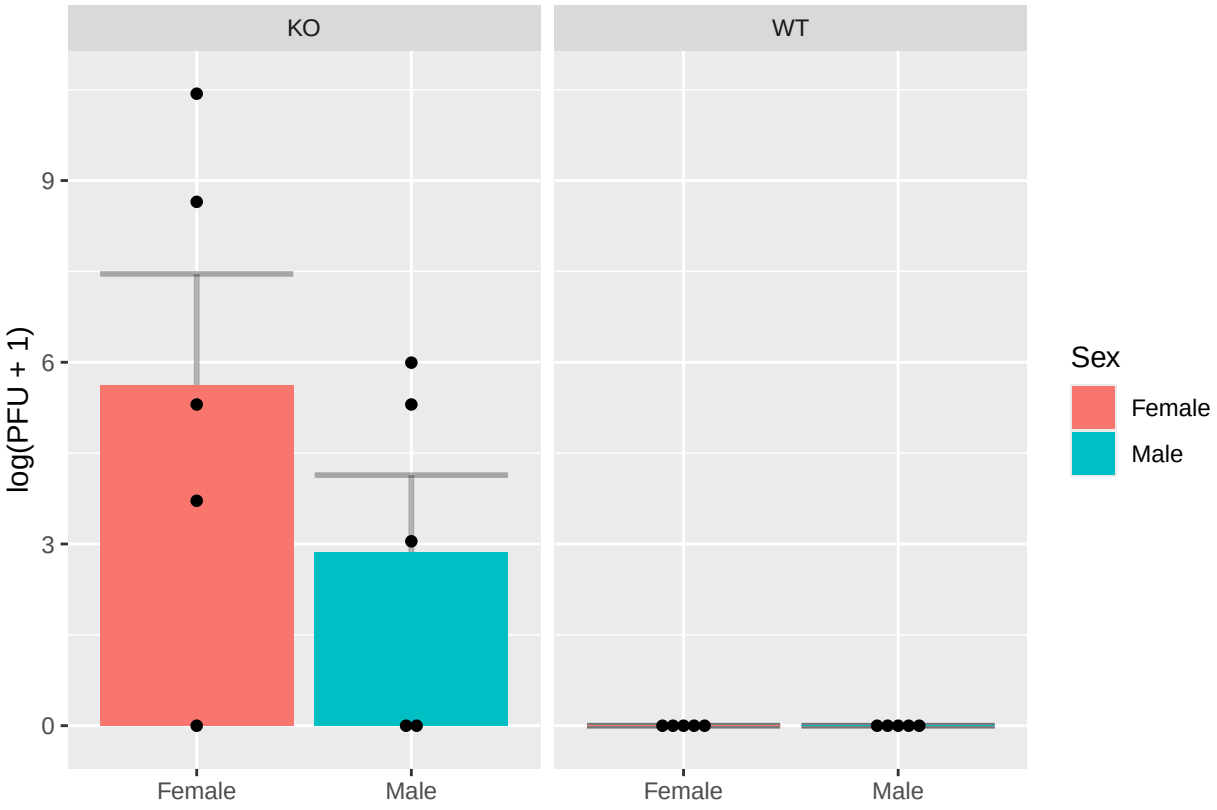
```
## `geom_dotplot()` called with `stackgroups = TRUE` and `method =
## "dotdensity"`.", i = "Do you want `binpositions = "all"` instead?
## Scale for y is already present.
## Adding another scale for y, which will replace the existing scale.
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```


KO
WT
PFU
3e+04
2e+04
1e+04
0e+00
Female
Male
Female
Male
Sex
Female
Male

Barplot of log(PFU + 1) with Error Bars and Points

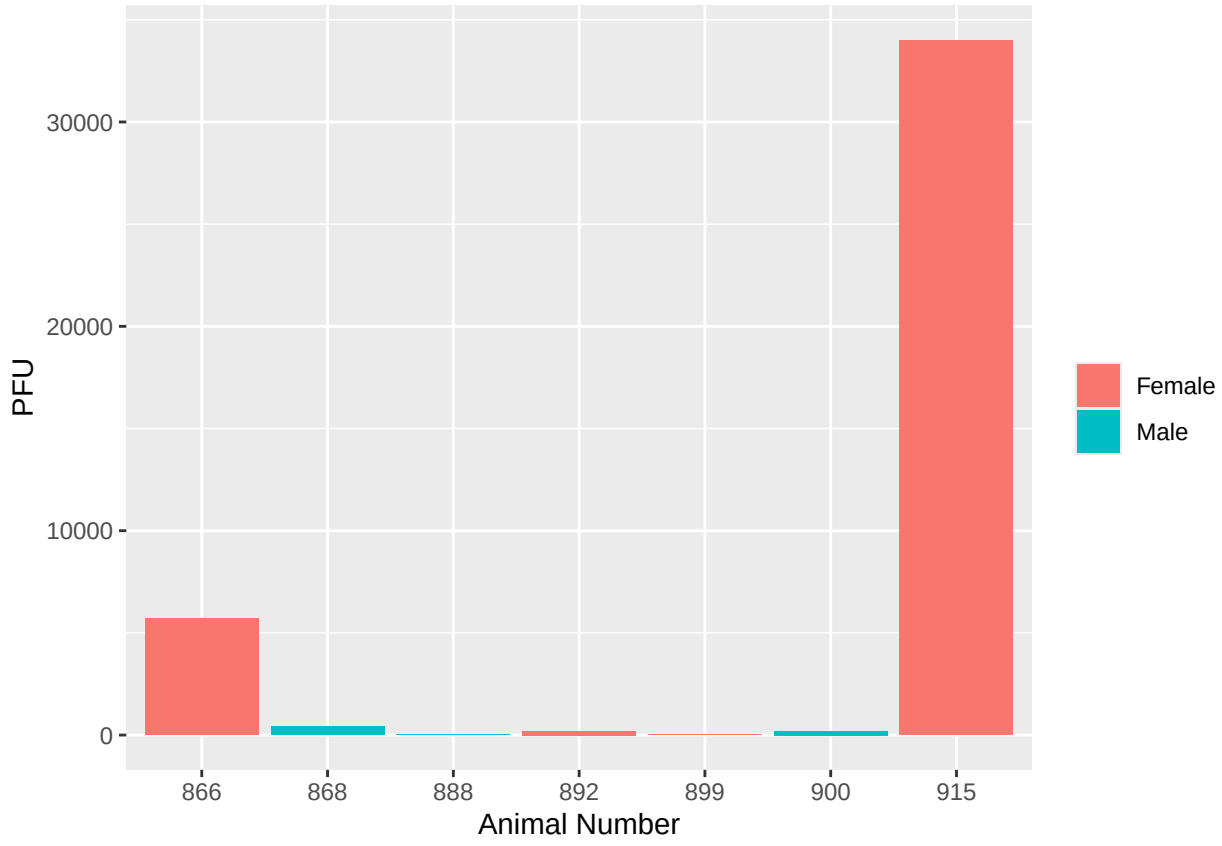
```
#Redid graph from above to fix skewedness.
# with log( + 1)
Mice_WDIL <- Mice_Weight %>%
  filter(Infection == "SARS2") %>%
  group_by(interaction(GenoType, Sex)) %>%
  mutate(Avg_Dil = mean(log(Dilution + 1)))




ggplot(data = Mice_WDIL, aes(x = Sex,
                             y = log(Dilution + 1))) +
  facet_grid(~ GenoType) +
   stat_summary(linewidth = 1,
                geom = "errorbar",
                colour = "BLACK",
                alpha = .3) +
  geom_bar(mapping = aes(y = Avg_Dil,
                         fill = Sex),
           stat = "identity",
           position = position_dodge(width = 1)) +
  geom_dotplot(binaxis='y',
                dotsize = .5,
                stackgroups = TRUE,
                binpositions = "bygroup",
                stackdir = "center") +
  labs(x = "",
       y = "log(PFU + 1)")
```

```
## `geom_dotplot()` called with `stackgroups = TRUE` and `method =
## "dotdensity"`.", i = "Do you want `binpositions = "all"` instead?
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```


KO
WT
log(PFU + 1)
9
6
3
0
Female
Male
Female
Male
Sex
Female
Male

Corrected Dilution of Mice

```
#This filters the dilutions not equal to zero.
Mice_Dilution <- Mice_Weight %>%
  filter(!(Dilution == 0))

#This plot shows the corrected dilution of mice with dilution.
ggplot(data = Mice_Dilution, aes(x = Animal_Num,
                                 y = Dilution,
                                 fill = Sex)) +
  geom_bar(stat = "identity",
           position = position_dodge(width = 1)) +
  labs(x = "Animal Number",
       y = "PFU",
       fill = "")
```



Wilcox Rank Sum Test

```
#Wilcox Rank Sum Test
Mice_KO <- Mice_Weight %>%
  filter(GenoType == "KO") %>%
  filter(Infection == "SARS2")
Mice_KFD <- Mice_KO %>%
  filter(Sex == "Female")
Mice_KMD <- Mice_KO %>%
  filter(Sex == "Male")

wilcox.test(Mice_KFD$Dilution, Mice_KMD$Dilution, paired = FALSE, alternative = "greater")
```

```
## Warning in wilcox.test.default(Mice_KFD$Dilution, Mice_KMD$Dilution, paired =
## FALSE, : cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Mice_KFD$Dilution and Mice_KMD$Dilution
## W = 17.5, p-value = 0.1699
## alternative hypothesis: true location shift is greater than 0
```

```
Mice_WT <- Mice_Weight %>%
  filter(GenoType == "WT") %>%
  filter(Infection == "SARS2")

wilcox.test(Mice_KO$Dilution, Mice_WT$Dilution, paired = FALSE, alternative = "greater")
```

```
## Warning in wilcox.test.default(Mice_KO$Dilution, Mice_WT$Dilution, paired =
## FALSE, : cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Mice_KO$Dilution and Mice_WT$Dilution
## W = 85, p-value = 0.0011
## alternative hypothesis: true location shift is greater than 0
```

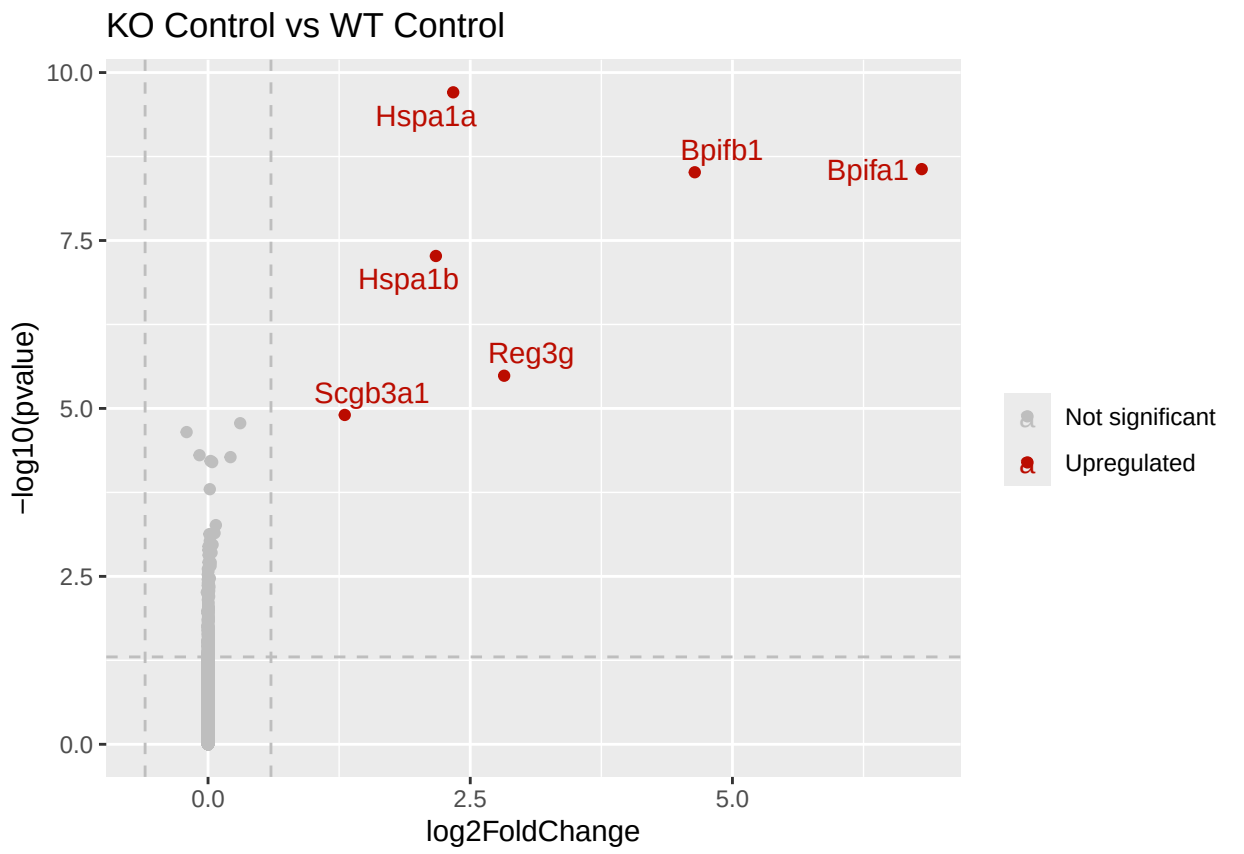
The wilcox test was against male and female knock out mice that were infected with SARS2. We found that there wasn't a significant difference in the dilutions between the males and females.

RNASeq Data

KO Control vs WT Control Volcano Plot

```
#Control KO vs WT

Geno_Cntrl <- read.csv("KO_Ctrl.vs.WT_Ctrl.csv")

#Filters to only protein coding genes
# removes all NA pvals and NA adjusted pvals
Geno_Cntrl <- Geno_Cntrl %>%
  filter(biotype == "protein_coding") %>%
  filter(!(is.na(pvalue))) %>%
  filter(!(is.na(padj)))

#Makes new col that says if gene is up or down regulated
Geno_Cntrl$diffexpressed <- "NO"
Geno_Cntrl$diffexpressed[Geno_Cntrl$log2FoldChange > 0.6 & Geno_Cntrl$pvalue < 0.05] <- "UP"
Geno_Cntrl$diffexpressed[Geno_Cntrl$log2FoldChange < -0.6 & Geno_Cntrl$pvalue < 0.05] <- "DOWN"

#Makes col of top 6 Genes to name on volcano plot
Geno_Cntrl$delabel <- ifelse(Geno_Cntrl$name %in% head(Geno_Cntrl[order(Geno_Cntrl$padj), "name"], 6), (

#Volcano Plot
ggplot(data = Geno_Cntrl, aes(x = log2FoldChange,
                              y = -log10(pvalue),
                              col = diffexpressed,
                              label = delabel)) +
  geom_point() +
  geom_vline(xintercept = c(-0.6, 0.6), col = "gray", linetype = 'dashed') +
  geom_hline(yintercept = -log10(0.05), col = "gray", linetype = 'dashed') +
  geom_text_repel(max.overlaps = Inf) +
  scale_color_manual(values = c("grey", "#bb0c00", "#00AFBB"),
                     labels = c("Not significant", "Upregulated", "Downregulated")) +
  labs(col = "",
       title = "KO Control vs WT Control")
```

```
## Warning: Removed 15565 rows containing missing values or values outside the scale range
## (`geom_text_repel()`).
```

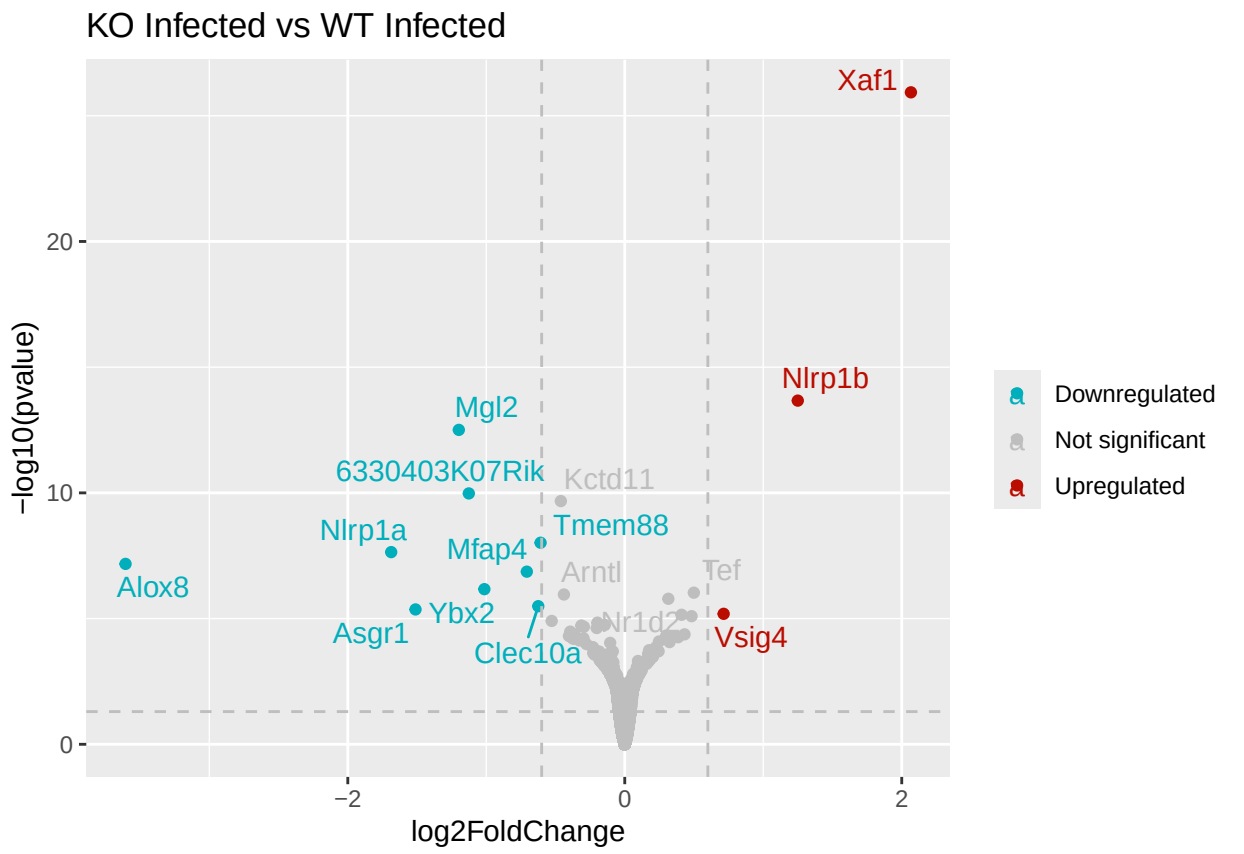



```
#dput(Geno_Cntrl$name)
```

KO Infected vs WT Infected Volcano Plot

```
#WT VS KO Infected
Geno_Infc <- read.csv("KO_SARS2.vs.WT_SARS2.csv")

#Filters to only protein coding genes
# removes all NA pvals and NA adjusted pvals
Geno_Infc <- Geno_Infc %>%
  filter(biotype == "protein_coding") %>%
  filter(!(is.na(pvalue))) %>%
  filter(!(is.na(padj)))


#Makes new col that says if gene is up or down regulated
Geno_Infc$diffexpressed <- "NO"
Geno_Infc$diffexpressed[Geno_Infc$log2FoldChange > 0.6 & Geno_Infc$pvalue < 0.05] <- "UP"
Geno_Infc$diffexpressed[Geno_Infc$log2FoldChange < -0.6 & Geno_Infc$pvalue < 0.05] <- "DOWN"

#Makes col of top 16 Genes to name on volcano plot
Geno_Infc$delabel <- ifelse(Geno_Infc$name %in% head(Geno_Infc[order(Geno_Infc$padj), "name"], 16), Gen


#Volcano Plots
ggplot(data = Geno_Infc, aes(x = log2FoldChange,
                             y = -log10(pvalue),
                             col = diffexpressed,
                             label = delabel)) +
  geom_point() +
  geom_vline(xintercept = c(-0.6, 0.6), col = "gray", linetype = 'dashed') +
  geom_hline(yintercept = -log10(0.05), col = "gray", linetype = 'dashed') +
  geom_text_repel(max.overlaps = Inf) +
  scale_color_manual(values = c("#00AFBB", "grey", "#bb0c00"),
                     labels = c("Downregulated", "Not significant", "Upregulated")) +
  labs(col = "",
       title = "KO Infected vs WT Infected")
```

```
## Warning: Removed 16160 rows containing missing values or values outside the scale range
## (`geom_text_repel()`).
```



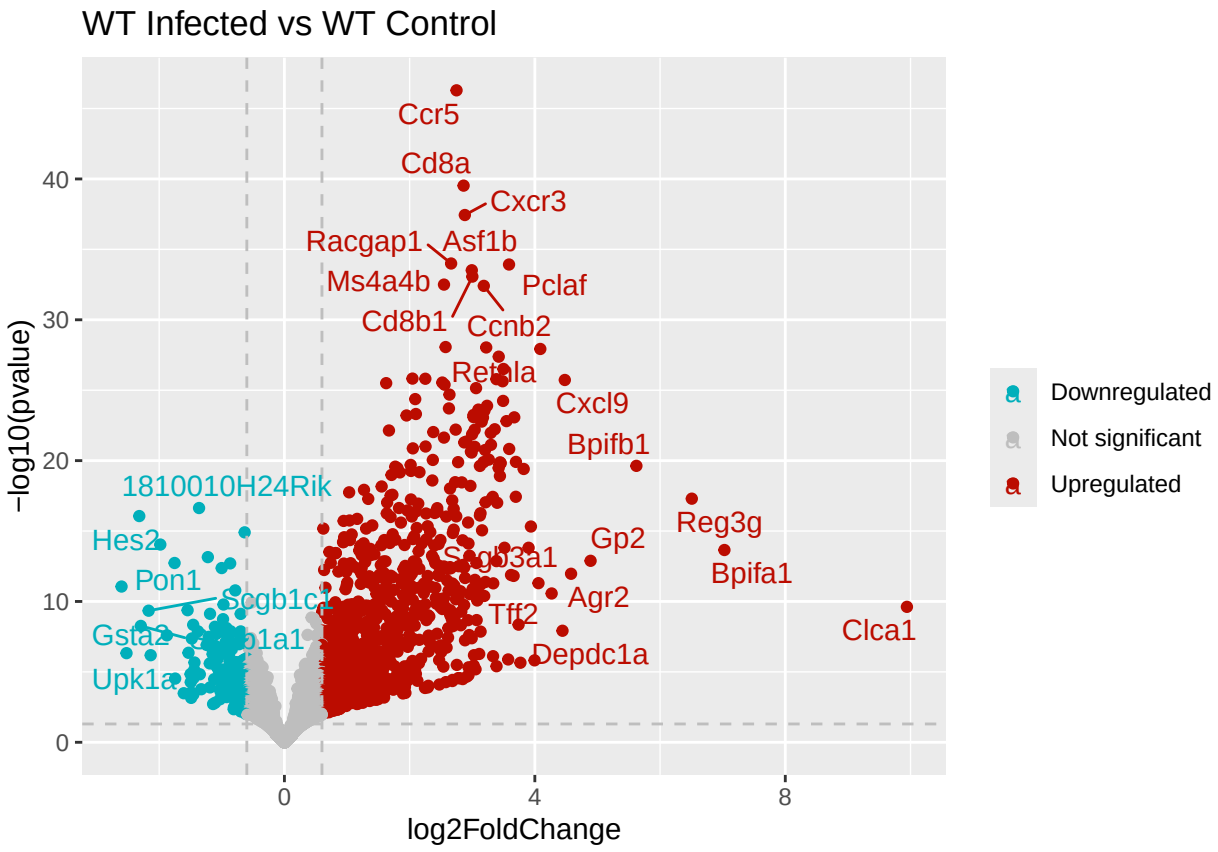

```
dput((Geno_Infc %>%
  filter(diffexpressed != "NO"))$name)
```

```
## c("Xaf1", "Nlrp1b", "Vsig4", "Tmem88", "Clec10a", "Mfap4", "Ybx2",
## "6330403K07Rik", "Mgl2", "Asgr1", "Nlrp1a", "Alox8")
```

WT Infected vs WT Control Volcano Plot

```
Geno_Mix <- read.csv("WT_SARS2.vs.WT_Ctrl.csv")

#Filters to only protein coding genes
# removes all NA pvals and NA adjusted pvals
Geno_Mix <- Geno_Mix %>%
  filter(biotype == "protein_coding") %>%
  filter(!(is.na(pvalue))) %>%
  filter(!(is.na(padj)))

#Makes new col that says if gene is up or down regulated
Geno_Mix$diffexpressed <- "NO"
Geno_Mix$diffexpressed[Geno_Mix$log2FoldChange > 0.6 & Geno_Mix$pvalue < 0.05] <- "UP"
Geno_Mix$diffexpressed[Geno_Mix$log2FoldChange < -0.6 & Geno_Mix$pvalue < 0.05] <- "DOWN"

#Makes col of top upregulated/downregulated Genes to name on volcano plot
Geno_Mix <- Geno_Mix %>%
  mutate(diffexpressed2 = case_when(
    log2FoldChange > 2 & -log10(pvalue) > 30 ~ "Upregulated",
    log2FoldChange > 4 ~ "Upregulated",
    (log2FoldChange < -2 | -log10(pvalue) > 15) & log2FoldChange < -.6 ~ "Downregulated",
    TRUE ~ "Not of Interest"
  )) %>%
  mutate(labeled_genes = case_when(
    diffexpressed2 == "Upregulated" ~ name,
    diffexpressed2 == "Downregulated" ~ name,
    TRUE ~ NA
  ))


#Volcano Plot
ggplot(data = Geno_Mix, aes(x = log2FoldChange,
                            y = -log10(pvalue),
                            col = diffexpressed,
                            label = labeled_genes)) +
  geom_point() +
  geom_vline(xintercept = c(-0.6, 0.6), col = "gray", linetype = 'dashed') +
  geom_hline(yintercept = -log10(0.05), col = "gray", linetype = 'dashed') +
  geom_text_repel(max.overlaps = Inf) +
  scale_color_manual(values = c("#00AFBB", "grey", "#bb0c00"),
                     labels = c("Downregulated", "Not significant", "Upregulated")) +
  labs(col = "",
       title = "WT Infected vs WT Control")
```

```
## Warning: Removed 15064 rows containing missing values or values outside the scale range
## (`geom_text_repel()`).
```


WT Infected vs WT Control
Ccr5
Cd8a
Cxcr3
Racgap1
Asf1b
Ms4a4b
Pclaf
Cd8b1
Ccnb2
Retnla
Cxcl9
Bpifb1
1810010H24Rik
Reg3g
Hes2
Gp2
Scgb3a1
Bpifa1
Pon1
Agr2
Scgb1c1
Tff2
Clca1
Gsta2
Depdc1a
Upk1a
40
30
20
10
0
0
4
8
−log10(pvalue)
log2FoldChange
Downregulated
Not significant
Upregulated

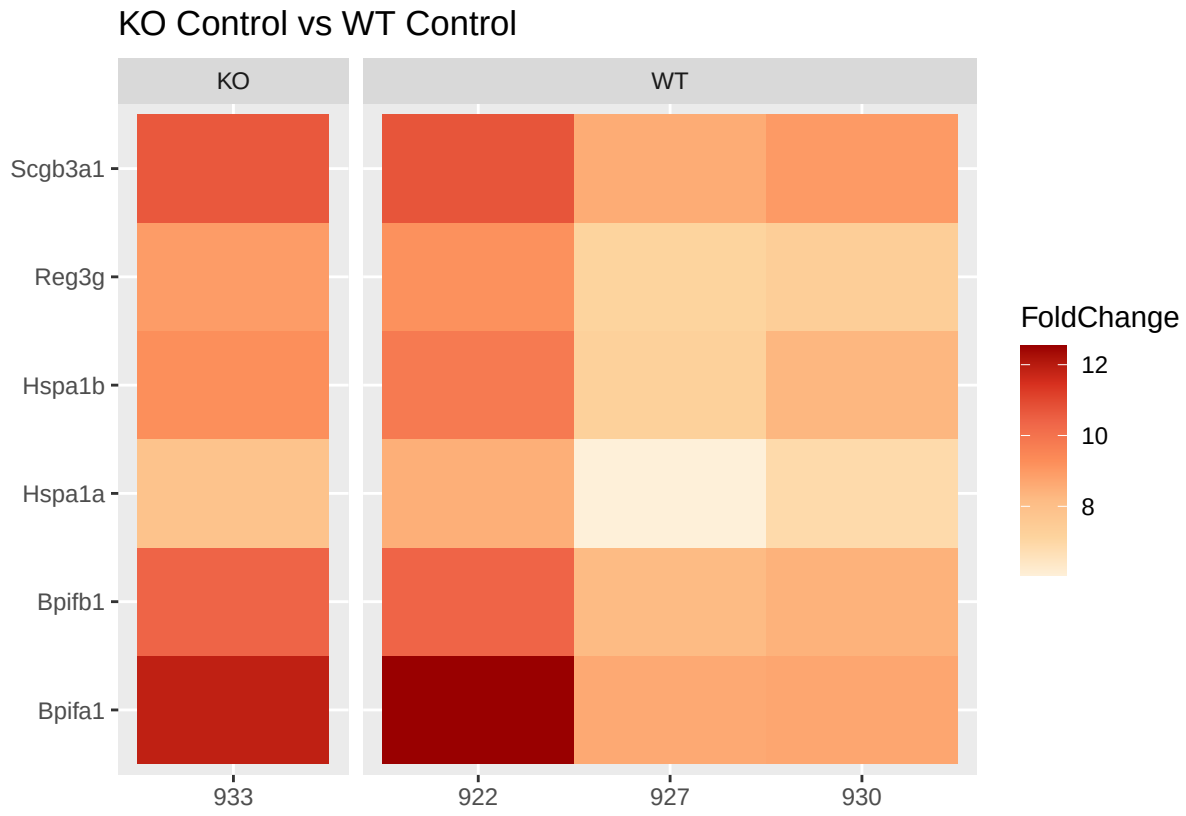
RNASeq Data Cleaning

```
RNASeq <- read.csv("Log2.normalized.read.counts.csv")


#Filters Data to only protein coding genes
# Pivot Longer
RNASeq <- RNASeq %>%
  filter(biotype == "protein_coding") %>%
  pivot_longer(!c(ENSEMBL_id, biotype, name, gene_id),
               names_to = "Animal_Num",
               values_to = "FoldChange")

# Remove M before all animal numbers
RNASeq$Animal_Num <- parse_number(RNASeq$Animal_Num)

# Makes sure that my Animal Num in OG data is a numeric value
Mice_Weight$Animal_Num <- as.numeric(levels(Mice_Weight$Animal_Num)[as.numeric(Mice_Weight$Animal_Num)]

# Join Mice Weight and RNASeq by the animal numbers.
# Has all col from both data sets
RNASeq <- inner_join(RNASeq, Mice_Weight, by = "Animal_Num")

RNASeq$Animal_Num <- as.factor(RNASeq$Animal_Num)
```

Heatmap of KO Control vs WT Control

```
#Filter RNASeq data to Ctrl
# Then find which genes are up and down regulated in the
# KO Control vs WT Control Volcano Plot
# pipe to ggplot and plot
# Facet grid to genotype and remove the empty columns




RNASeq %>%
  filter(name %in% dput((Geno_Cntrl %>%
                           filter(diffexpressed != "NO"))$name)) %>%
  filter(Infection == "Ctrl") %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~ GenoType, scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1) +
  labs(title = "KO Control vs WT Control",
       x= "",
       y = "",
       fill = "FoldChange")
```

```
## c("Hspa1a", "Bpifa1", "Bpifb1", "Hspa1b", "Reg3g", "Scgb3a1")
```

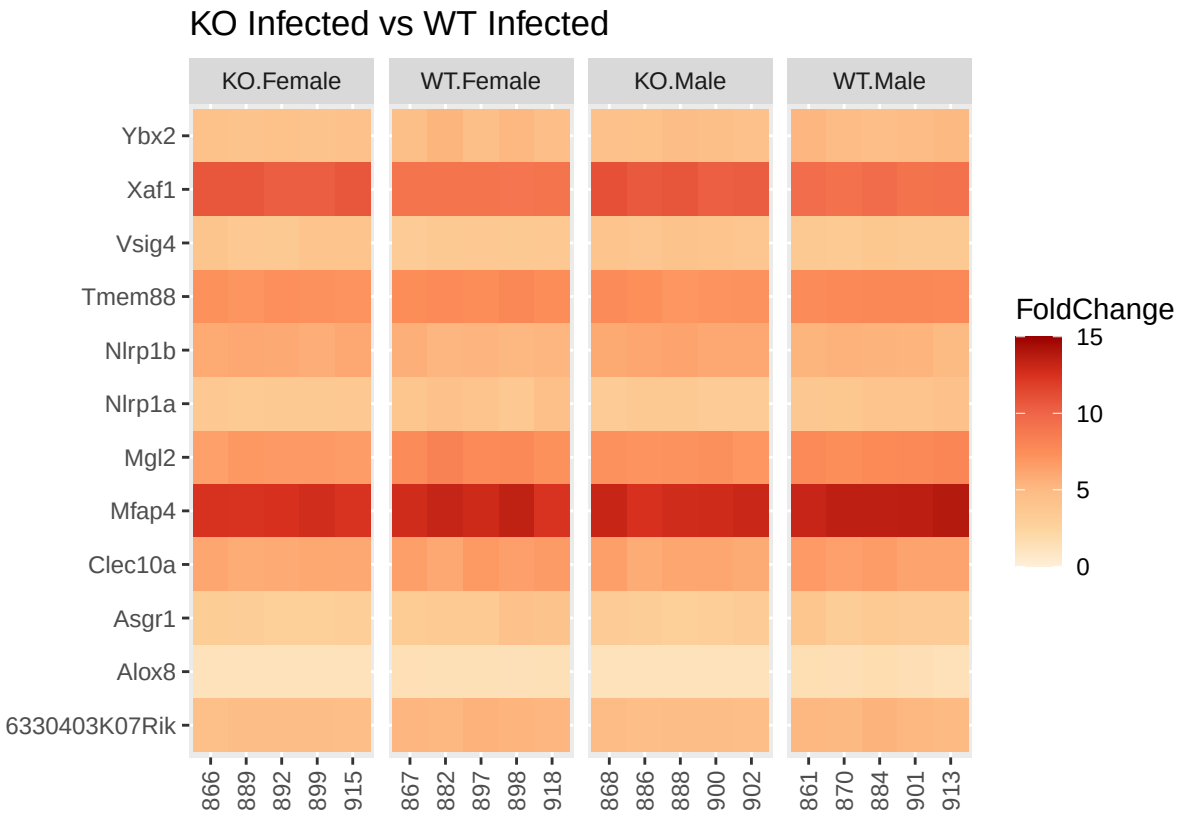



```
# NOTE: dput function outputs a written concatenate
#         so you dont have to write it yourself.
# NOTE: If your Fold Change crosses zero you can use
#         palette = "RdBu",
#                       limits = c(-1,1)*max(abs(RNASeq$FoldChange))
#         in the scale_fill_distiller
```

Heatmap of KO Infected vs WT Infected

```
#Filter RNASeq data to SARS2
# Then find which genes are up and down regulated in the
# KO Infected vs WT Infected Volcano Plot
# pipe to ggplot and plot
# Facet grid to genotype and remove the empty columns

RNASeq %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Infc %>%
                           filter(diffexpressed != "NO"))$name)) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(GenoType, Sex), scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1,
                       limits = c(0, 15)) +
  labs(title = "KO Infected vs WT Infected",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## c("Xaf1", "Nlrp1b", "Vsig4", "Tmem88", "Clec10a", "Mfap4", "Ybx2",
## "6330403K07Rik", "Mgl2", "Asgr1", "Nlrp1a", "Alox8")
```

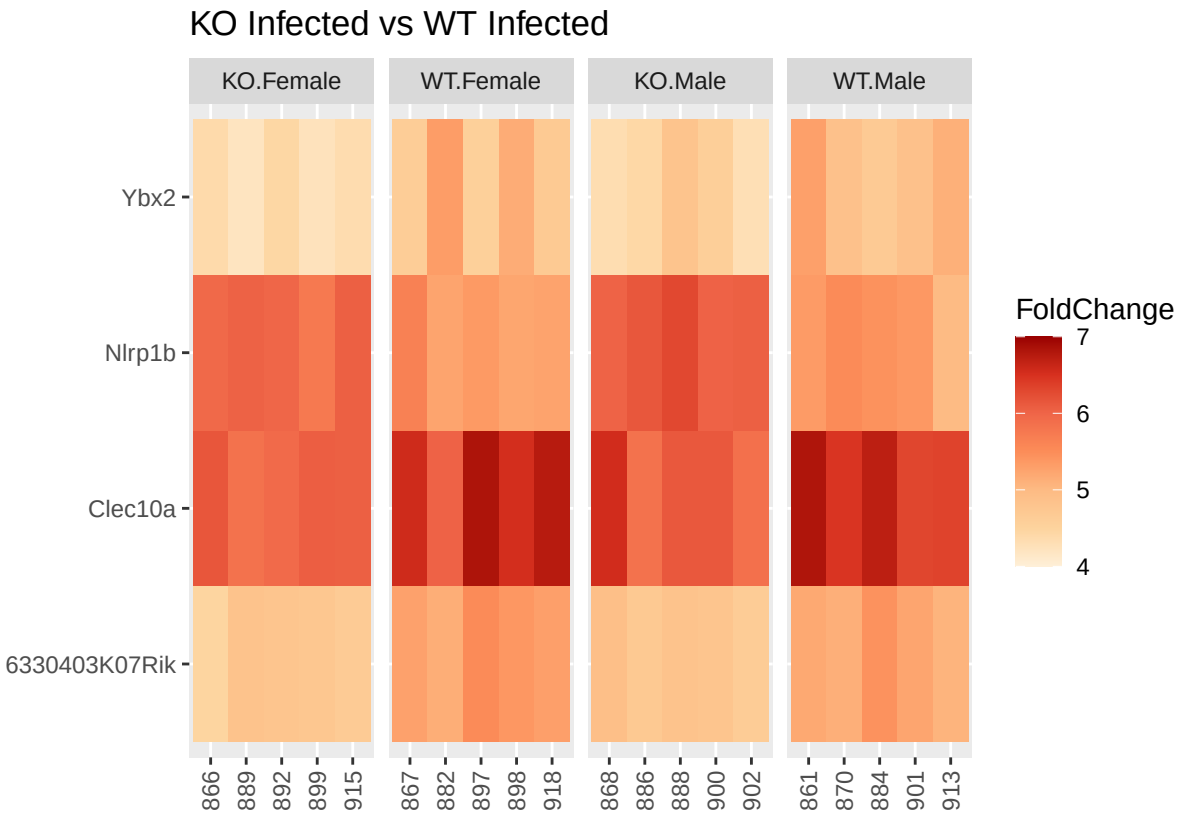



```
RNASeq$Dil_Label <- ifelse(RNASeq$Dilution > 0, "Y", "N")
```

Reduced Heatmap of KO Infected vs WT Infected

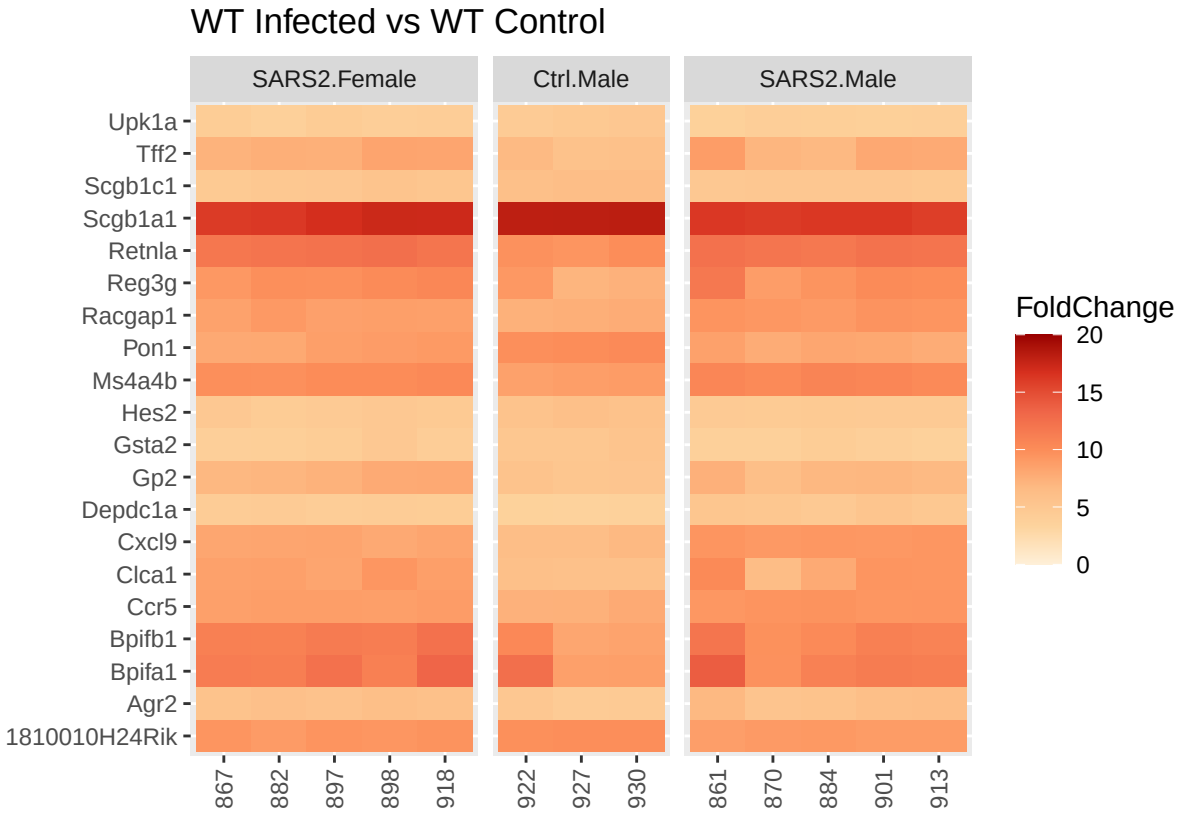
```
#Filter RNASeq data to SARS2
# Then find which genes are up and down regulated in the
# KO Infected vs WT Control Volcano Plot
# Take the baseMean < 100 and baseMean > 20 to plot
# pipe to ggplot and plot
# Facet grid to genotype and remove the empty columns

RNASeq %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Infc %>%
                           filter(diffexpressed != "NO") %>%
                           filter(baseMean < 100,
                                  baseMean  > 20))$name)) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(GenoType, Sex), scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1,
                       limits = c(4, 7)) +
  labs(title = "KO Infected vs WT Infected",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## c("Nlrp1b", "Clec10a", "Ybx2", "6330403K07Rik")
```


KO Infected vs WT Infected
KO.Female
WT.Female
KO.Male
WT.Male
Ybx2
Nlrp1b
Clec10a
6330403K07Rik
866
889
892
899
915
867
882
897
898
918
868
886
888
900
902
861
870
884
901
913
FoldChange
7
6
5
4

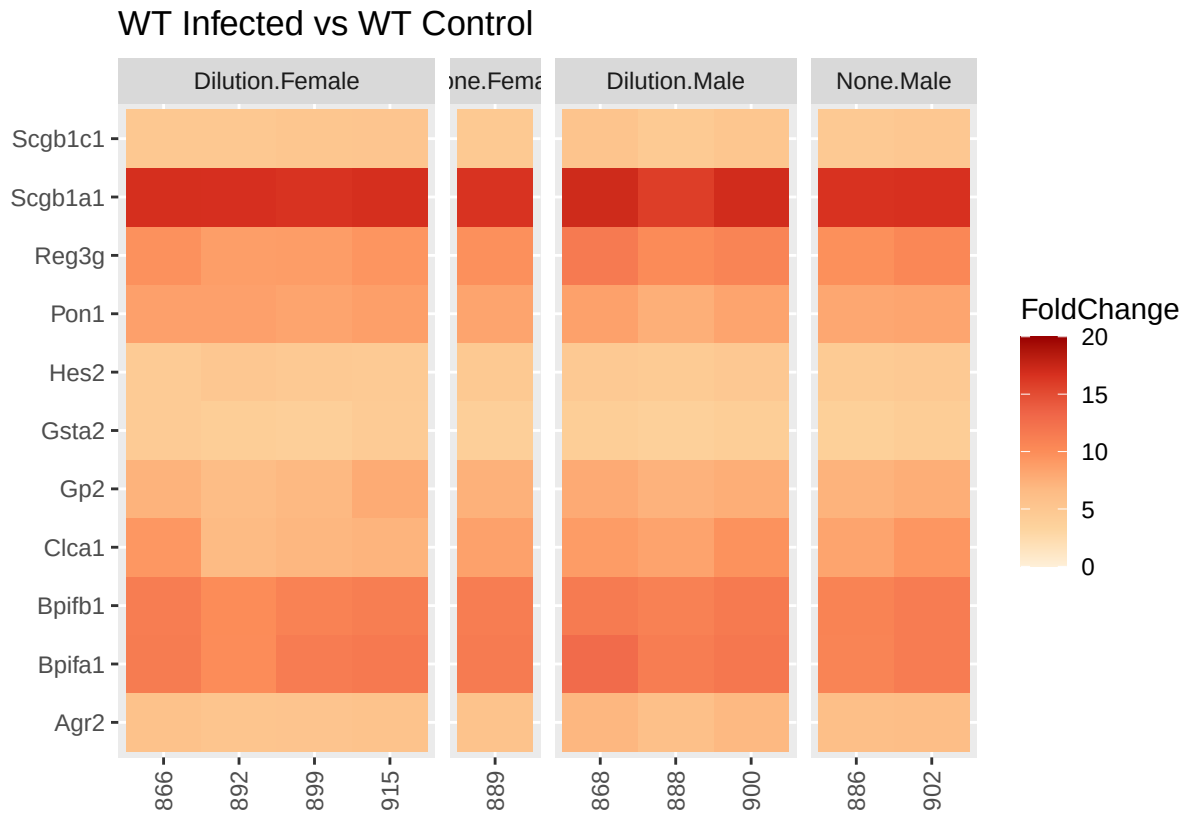
Heatmap of WT Infected vs WT Control

```
#Filter RNASeq data to SARS2
# Then find which genes are up and down regulated in the
# WT Infected vs WT Control Volcano Plot
# Take the first 10 up regulated and bottom 10 down regulated, try using
# pipe to ggplot and plot
# Facet grid to genotype and remove the empty columns

RNASeq %>%
  filter(GenoType == "WT") %>%
  filter(name %in% dput((Geno_Mix %>%
  filter(!is.na(labeled_genes)) %>%
    arrange(log2FoldChange))$labeled_genes[c(1:10, 18:27)])) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(Infection, Sex), scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1,
                       limits = c(0, 20)) +
  labs(title = "WT Infected vs WT Control",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```


WT Infected vs WT Control
SARS2.Female
Ctrl.Male
SARS2.Male
Upk1a
Tff2
Scgb1c1
Scgb1a1
Retnla
Reg3g
Racgap1
Pon1
Ms4a4b
Hes2
Gsta2
Gp2
Depdc1a
Cxcl9
Clca1
Ccr5
Bpifb1
Bpifa1
Agr2
1810010H24Rik
867
882
897
898
918
922
927
930
861
870
884
901
913
FoldChange
20
15
10
5
0

```r
# Dilution label to see which Subjects have Dilution
RNASeq$Dil_Label <- ifelse(RNASeq$Dilution > 0, "Dilution", "None")

#Filter RNASeq data to SARS2
# Then find which genes are up and down regulated in the
# WT Infected vs WT Control Volcano Plot
# Take the first 5 up regulated and bottom 5 down regulated, try using
# pipe to ggplot and plot
# Facet grid to genotype and remove the empty columns

RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Mix %>%
  filter(!is.na(labeled_genes)) %>%
    arrange(log2FoldChange))$labeled_genes[c(1:5, 22:27)])) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(Dil_Label, Sex), scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1,
                       limits = c(0, 20)) +
  labs(title = "WT Infected vs WT Control",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

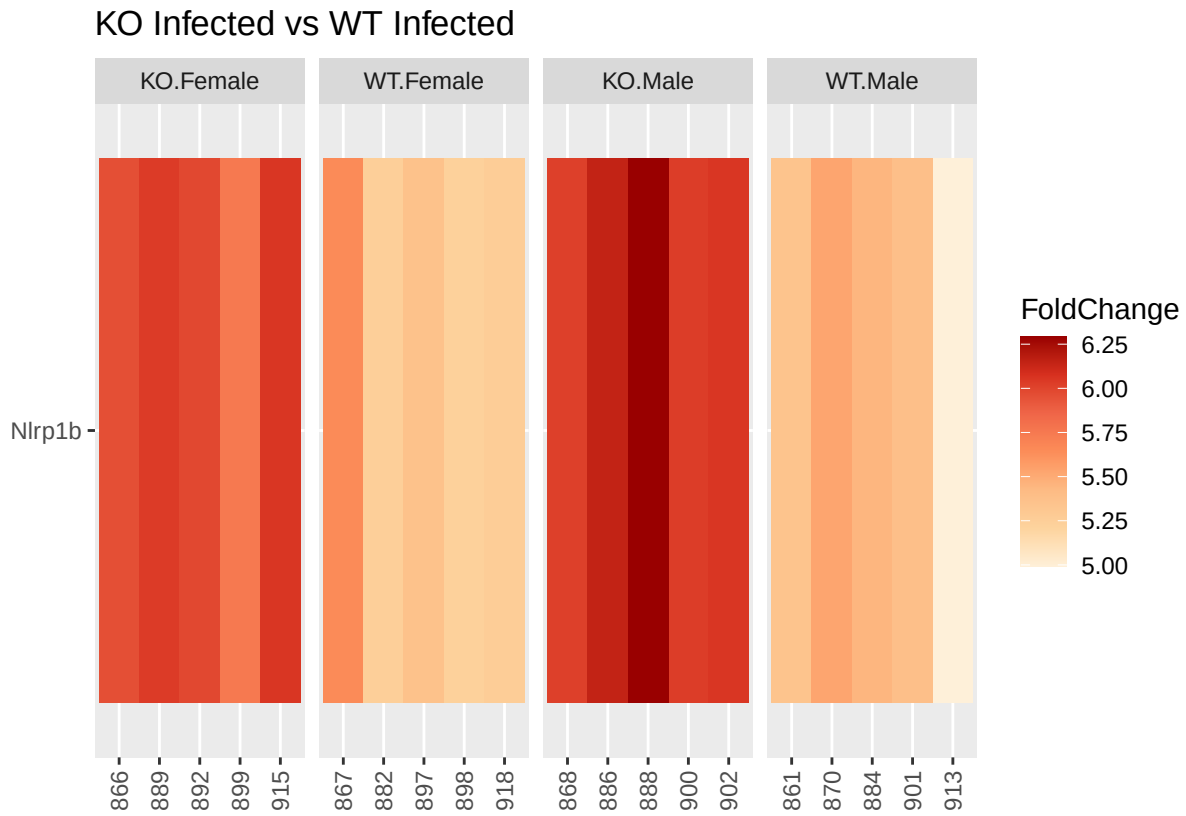
```
## c("Pon1", "Gsta2", "Hes2", "Scgb1a1", "Scgb1c1", "Agr2", "Gp2",
## "Bpifb1", "Reg3g", "Bpifa1", "Clca1")
```


WT Infected vs WT Control
Dilution.Female
Dilution.Male
None.Male
Scgb1c1
Scgb1a1
Reg3g
Pon1
Hes2
Gsta2
Gp2
Clca1
Bpifb1
Bpifa1
Agr2
866
892
899
915
889
868
888
900
886
902
FoldChange
20
15
10
5
0

Heatmap of Nlrp1b Infected vs WT Infected

```
#RNASeq filtered to just SARS2 and Nlrp1b
# piped into ggplot, tile plot
# color change by fold change and labeled by animal num
# facet grid based on genotype and sex
# labels rotated for readability

RNASeq %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Infc %>%
                           filter(name == c("Nlrp1b")))$name)) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(GenoType, Sex),
             scales = "free",
             space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1) +
  labs(title = "KO Infected vs WT Infected",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

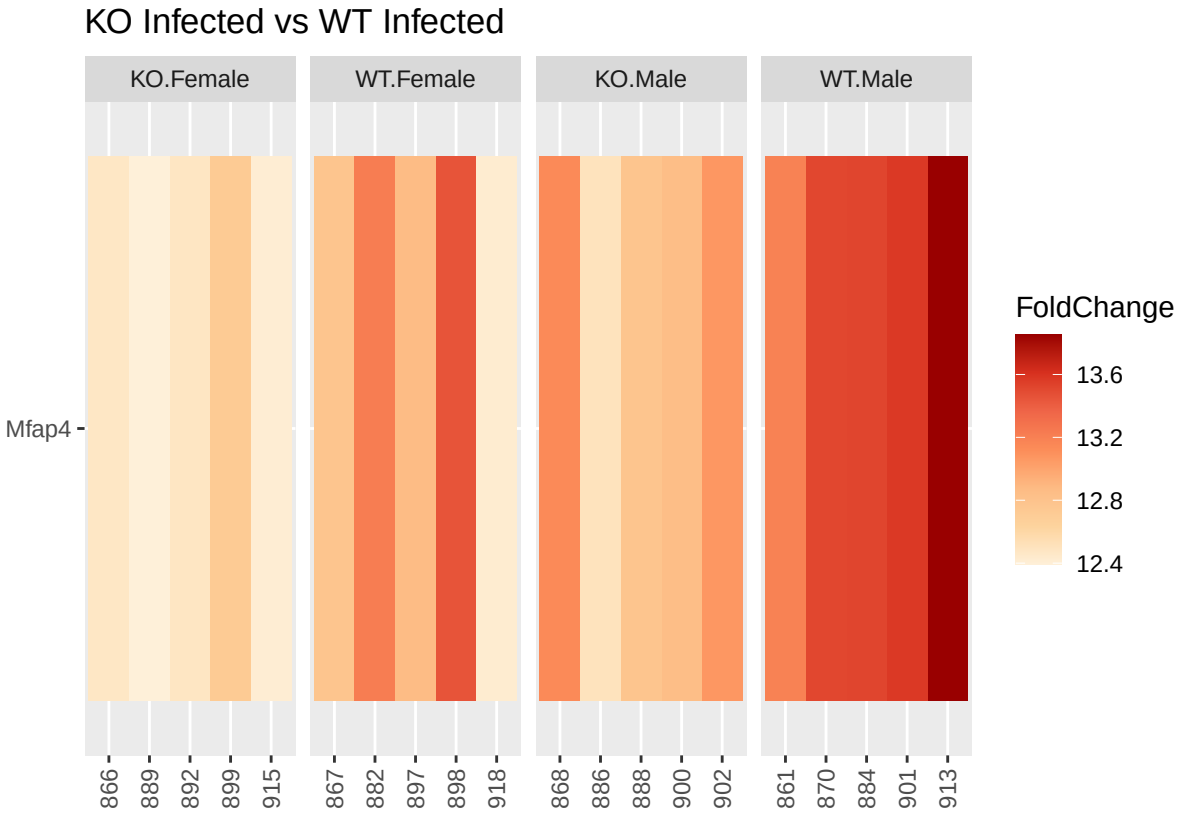
```
## "Nlrp1b"
```



Heatmap of Mfap4 KO Infected vs WT Infected

```
#RNASeq filtered to just SARS2 and Mfap4
# piped into ggplot, tile plot
# color change by fold change and labeled by animal num
# facet grid based on genotype and sex
# labels rotated for readability
RNASeq %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Infc %>%
                           filter(name == c("Mfap4")))$name)) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(GenoType, Sex), scales = "free", space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1) +
  labs(title = "KO Infected vs WT Infected",
       x= "",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

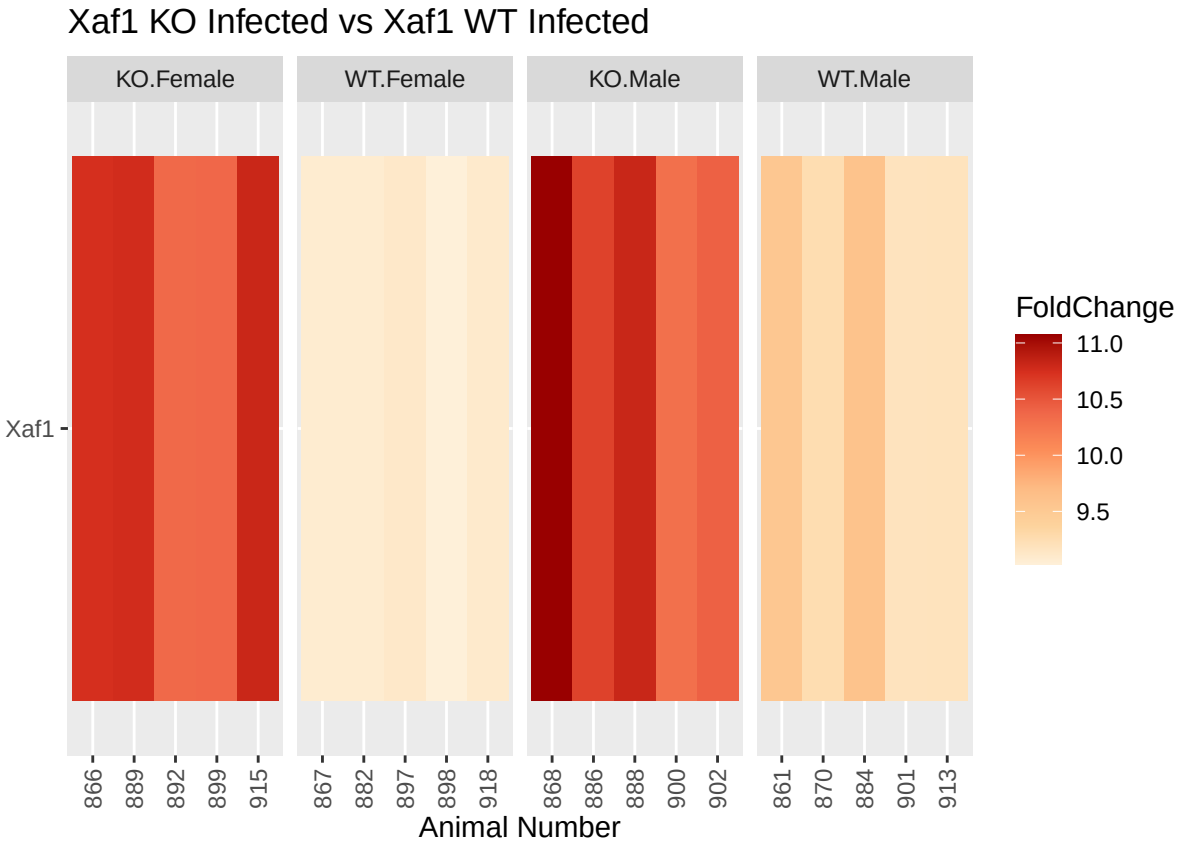
```
## "Mfap4"
```


KO Infected vs WT Infected
KO.Female
WT.Female
KO.Male
WT.Male
Mfap4
866
889
892
899
915
867
882
897
898
918
868
886
888
900
902
861
870
884
901
913
FoldChange
13.6
13.2
12.8
12.4

Heatmap of Xaf1 Infected vs WT Infected

```
#RNASeq filtered to just SARS2 and Xaf1
# piped into ggplot, tile plot
# color change by fold change and labeled by animal num
# facet grid based on genotype and sex
# labels rotated for readability

RNASeq %>%
  filter(Infection == "SARS2") %>%
  filter(name %in% dput((Geno_Infc %>%
                           filter(name == c("Xaf1")))$name)) %>%
  ggplot(aes(x = Animal_Num,
             y = name,
             fill = FoldChange)) +
    geom_tile() +
  facet_grid(~interaction(GenoType, Sex),
             scales = "free",
             space = "free") +
  scale_fill_distiller(palette = "OrRd",
                       direction = 1) +
  labs(title = "Xaf1 KO Infected vs Xaf1 WT Infected",
       x= "Animal Number",
       y = "",
       fill = "FoldChange") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## "Xaf1"
```


Xaf1 KO Infected vs Xaf1 WT Infected
KO.Female
WT.Female
KO.Male
WT.Male
Xaf1
866
889
892
899
915
867
882
897
898
918
868
886
888
900
902
861
870
884
901
913
Animal Number
FoldChange
11.0
10.5
10.0
9.5

Dilution Correlation Test

NOTE: One issues on the following correlation tests is that our data is small. So it will be difficult to find any associations.

Correlation Of KO Dilution Female vs Xaf1 Fold Change

```
#Filters RNASeq data to KO Females with Xaf1 gene
Dil_XAF1 <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(Sex == "Female") %>%
  filter(name == "Xaf1")

#Runs Correlation test on Xaf1 fold change and KO Females Dilutions
cor.test(Dil_XAF1$FoldChange, Dil_XAF1$Dilution)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Dil_XAF1$FoldChange and Dil_XAF1$Dilution
## t = 1.183, df = 3, p-value = 0.322
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.6334876  0.9657242
## sample estimates:
##       cor
## 0.5640027
```

H0: r = 0

H1: r != 0

The p-val is equal to .322 with a significant level of .05. We fail to reject the null, thus we dont have significant evidence to suggest that there is association between KO dilution female and Xaf1 fold change.

Correlation Of KO Dilution Male vs Xaf1 Fold Change

```
#Filters RNASeq Data to KO Males with Xaf1 Gene
Dil_XAF1 <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(Sex == "Male") %>%
  filter(name == "Xaf1")

#Runs Correlation test on Xaf1 fold change and KO Males Dilutions
cor.test(Dil_XAF1$FoldChange, Dil_XAF1$Dilution)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Dil_XAF1$FoldChange and Dil_XAF1$Dilution
## t = 1.3148, df = 4, p-value = 0.2589
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4731991  0.9412601
## sample estimates:
##       cor
## 0.5493161
```

H0: r = 0

H1: r != 0

The p-val is equal to .2589 with a significant level of .05. We fail to reject the null, thus we dont have significant evidence to suggest that there is association between KO dilution male and Xaf1 fold change.

Correlation Of KO Dilution vs Xaf1 Fold Change

```
#Filters KO Mice with Xaf1 genes
Dil_XAF1 <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(name == "Xaf1")

#Runs Correlation test on Xaf1 fold change and KO Mice Dilutions
cor.test(Dil_XAF1$FoldChange, Dil_XAF1$Dilution)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Dil_XAF1$FoldChange and Dil_XAF1$Dilution
## t = 0.9306, df = 9, p-value = 0.3763
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3692219  0.7609138
## sample estimates:
##       cor
## 0.2962739
```

H0: r = 0

H1: r != 0

The p-val is equal to .2763 with a significant level of .05. We fail to reject the null, thus we dont have significant evidence to suggest that there is association between KO dilution and Xaf1 fold change.

T-Test for Xaf1 Fold Change

T-Test for Xaf1, KO vs WT

```
# Filter RNASeq to both WT and Xaf1
WT_XAF1 <- RNASeq %>%
  filter(GenoType == "WT") %>%
  filter(name == "Xaf1")

# Filter RNASeq to both KO and Xaf1
KO_XAF1 <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(name == "Xaf1")

# T test with a greater than alternative
t.test(KO_XAF1$FoldChange, WT_XAF1$FoldChange, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  KO_XAF1$FoldChange and WT_XAF1$FoldChange
## t = 12.525, df = 21.007, p-value = 1.64e-11
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  1.270228      Inf
## sample estimates:
## mean of x mean of y
## 10.579827  9.107291
```

H0: Mean of KO Xaf1 fold change is less or equal than WT Xaf1 fold change

H1: Mean of KO Xaf1 fold change is greater than WT Xaf1 fold change

The p-val is 1.64e-11 with significant level of .05. We reject the null, meaning we have sufficient evidence to suggest that the mean of KO Xaf1 fold change is greater than WT Xaf1 fold change

T-Test for Xaf1 KO, Females vs Males

```
# Filter RNASeq to KO Male and Xaf1
KO_XAF1_M <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(Sex == "Male") %>%
  filter(name == "Xaf1")

# Filter RNASeq to KO Female and Xaf1
KO_XAF1_F <- RNASeq %>%
  filter(GenoType == "KO") %>%
  filter(Sex == "Female") %>%
  filter(name == "Xaf1")

# T test with a greater than alternative
t.test(KO_XAF1_F$FoldChange, KO_XAF1_M$FoldChange, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  KO_XAF1_F$FoldChange and KO_XAF1_M$FoldChange
## t = 0.3213, df = 8.4475, p-value = 0.3779
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -0.2710588        Inf
## sample estimates:
## mean of x mean of y
##  10.61097  10.55388
```

H0: Mean of KO Females Xaf1 fold change is less than or equal to mean of KO Male Xaf1 fold change.

H1: Mean of KO Females Xaf1 fold change is greater than the mean of KO Male Xaf1 fold change.

The p-val is .3779 with a significant level of .05. We fail to reject the null, meaning do not have sufficient evidence to suggest that the mean of KO Females Xaf1 fold change is greater than the mean of KO Male Xaf1 fold change.

T-Test for Xaf1 WT, Females vs Males

```
# Filter RNASeq to KO Male and Xaf1
WT_XAF1_M <- RNASeq %>%
  filter(GenoType == "WT") %>%
  filter(Sex == "Male") %>%
  filter(name == "Xaf1")

# Filter RNASeq to KO Female and Xaf1
WT_XAF1_F <- RNASeq %>%
  filter(GenoType == "WT") %>%
  filter(Sex == "Female") %>%
  filter(name == "Xaf1")

# T test with a greater than alternative
t.test(WT_XAF1_F$FoldChange, WT_XAF1_M$FoldChange, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  WT_XAF1_F$FoldChange and WT_XAF1_M$FoldChange
## t = -0.3133, df = 7.2027, p-value = 0.6185
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -0.2859798        Inf
## sample estimates:
## mean of x mean of y
##  9.082228  9.122956
```

H0: Mean of WT Females Xaf1 fold change is less than or equal to mean of WT Male Xaf1 fold change.

H1: Mean of WT Females Xaf1 fold change is greater than the mean of WT Male Xaf1 fold change.

The p-val is .6185 with a significant level of .05. We fail to reject the null, meaning do not have sufficient evidence to suggest that the mean of WT Females Xaf1 fold change is greater than the mean of WT Male Xaf1 fold change.

2-Way ANOVA of Xaf1 Fold Change

```
# Filter RNASeq to just Xaf1
XAF1 <- RNASeq %>%
  filter(name == "Xaf1")

# 2-Way Anova, testing FoldChange on GenoType and Sex
two_way_aov <- aov(FoldChange ~ GenoType + Sex, data = XAF1)

summary(two_way_aov)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## GenoType     1 12.920  12.920 150.869 4.73e-11 ***
## Sex          1  0.000   0.000   0.002    0.966
## Residuals   21  1.798   0.086
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Because GenoType p-vals is 4.73e-11 we can conclude that a change in genotype will impact significantly the mean of Xaf1 fold change.

.

Hi Jackson cool code you got here. Would be a shame if someone were to delete it all. Good thing I'd never do that to you. :)

I like your plots they look cool. I'm jealous because I want to do a heat map but our stuff is insanely simple. I wish I knew what you were doing to my code cause I'm not actually doing anything. Also I don't know how to use the % stuff so I will not be using it at all. I appreciate the effort though. Okay you can have your stuff back :)

Hello again Jackson. It's your favorite code writer here to say I'm better at coding than you are. You have to agree. I agree.

I know I said you're a hater but I just don't know to scroll down so it's okay you're not a hater. You're just smarter than me and I'm JEALOUS.

I hope you weren't doing anything too important because now I'm taking up your time on purpose. I'm the best coder alive by the way. Make sure you tell people that when we leave the program. Hi Jackson.

Okay hello again Jackson. How are you? I hope you're doing good it would kinda suck if you weren't. Just wanted to fix your code because it's horrendous. No need to worry though, I got it covered. Also you're pretty cool. Also you're secretly a British man. That's all for now. Bye. :)

Hello again Jackson. Hope you didn't miss me too much. Your code sure is coding. I wish my project was cooler like yours is. I'm just doing very basic introductory statistics things and you're doing cool stuff. I'm still a better coder than you are though. You're an effective coder but I'm a cool coder. Don't be too jealous though. I'll teach you how to code if you'd like. Okay bye. :)

Thanks for helping me with my code I really appreciate it. I don't know what I'd do if you didn't help me out. Probably go ask Dr. Villasante but that's not the point.