Titer Analysis

Charles Costanzo

June 2023

```r
setwd("/Users/charlescostanzo/College/Su 2023/Chopra/African Green Monkeys Plague/")

# load in packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----------------------- tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(readxl)
library(scales)
```

```
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
library(rstatix)
```

```
##
## Attaching package: 'rstatix'
##
## The following object is masked from 'package:stats':
##
##     filter
```

```r
library(reshape)
```

```
## 
## Attaching package: 'reshape'
## 
## The following object is masked from 'package:lubridate':
## 
##     stamp
## 
## The following object is masked from 'package:dplyr':
## 
##     rename
## 
## The following objects are masked from 'package:tidyr':
## 
##     expand, smiths
```

```r
library(dplyr)
library(ggpubr)
library(plyr)
```

```
## ------------------------------------------------------------------------------
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## ------------------------------------------------------------------------------
## 
## Attaching package: 'plyr'
## 
## The following object is masked from 'package:ggpubr':
## 
##     mutate
## 
## The following objects are masked from 'package:reshape':
## 
##     rename, round_any
## 
## The following objects are masked from 'package:rstatix':
## 
##     desc, mutate
## 
## The following objects are masked from 'package:dplyr':
## 
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
## 
## The following object is masked from 'package:purrr':
## 
##     compact
```

```r
library(datarium)
library(RColorBrewer)
library(afex)
```

```
## Loading required package: lme4
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following object is masked from 'package:reshape':
##
##      expand
##
## The following objects are masked from 'package:tidyr':
##
##      expand, pack, unpack
##
## ************
## Welcome to afex. For support visit: http://afex.singmann.science/
## - Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
## - Methods for calculating p-values with mixed(): 'S', 'KR', 'LRT', and 'PB'
## - 'afex_aov' and 'mixed' objects can be passed to emmeans() for follow-up tests
## - Get and set global package options with: afex_options()
## - Set sum-to-zero contrasts globally: set_sum_contrasts()
## - For example analyses see: browseVignettes("afex")
## ************
##
## Attaching package: 'afex'
##
## The following object is masked from 'package:lme4':
##
##      lmer
```

```r
library(performance)
library(see)
```

```r
# load in data
data <- read_excel("Data/primate.xlsx")

# create a new variable "id_treatment_sex" that combines "id",
# "treatment", and "sex" variables separated by a space
data2 <- data %>%
  mutate(id_treatment_sex = unite(data = data, col = "id_treatment_sex", c(id, treatment, sex), sep = "

# "make" time a factor variable with three levels
data2$time <- factor(data2$time, levels = c("Pre","Prime","Boost"))

# make "id_treatment_sex" a factor variable
data2$id_treatment_sex <- factor(data2$id_treatment_sex,
                                 levels = c("9501 PBS F1 Male",
                                            "9286 PBS F1 Male",
                                            "9035 F1 Male",
                                            "9050 F1 Male",
                                            "9330 F1 Male",
                                            "0303 PBS F1 Female",
                                            "0116 PBS F1 Female",
                                            "9022 F1 Female",
                                            "9051 F1 Female",
```

```r
                                           "9107 F1 Female",
                                           "9501 PBS LcrV Male",
                                           "9286 PBS LcrV Male",
                                           "9035 LcrV Male",
                                           "9050 LcrV Male",
                                           "9330 LcrV Male",
                                           "0303 PBS LcrV Female",
                                           "0116 PBS LcrV Female",
                                           "9022 LcrV Female",
                                           "9051 LcrV Female",
                                           "9107 LcrV Female"
                          ))

# filter out unvaccinated "control" monkeys (not sure if we can analyze those)
# convert columns sex, time, and treatment to factor
data2 <- data2 %>%
  filter(treatment == "F1" | treatment == "LcrV") %>%
  mutate(sex = as.factor(sex),
         time = as.factor(time),
         treatment = as.factor(treatment))

# create a new variable "log10_titer"
# that contains titer values on log base 10 scale
data2$log10_titer <- log(data2$titer, base = 10)
```

```
################################################################################
# Summary Statistics
################################################################################

summary <- data2 %>%
  group_by(time, sex) %>%
  get_summary_stats(titer, type = "mean_sd")
summary
```

```
## # A tibble: 6 x 6
##   sex    time  variable      n   mean      sd
##   <fct>  <fct> <fct>     <dbl>  <dbl>   <dbl>
## 1 Female Pre   titer         6    100       0
## 2 Male   Pre   titer         6    100       0
## 3 Female Prime titer         6  1833.   2313.
## 4 Male   Prime titer         6   6900  10363.
## 5 Female Boost titer         6   4600   4904.
## 6 Male   Boost titer         6 29467.  40282.
```

```
################################################################################
# Data Visualization
################################################################################

# generate a box-dotplot for titer by dose time
data2 %>%
  ggplot(aes(x = time, y = titer, fill = time)) +
  geom_boxplot() +
  scale_y_log10() +
  scale_fill_brewer(palette = "Paired",
                    name = "Dose") +
  theme_light() +
  ylab("Titer") +
  xlab("Dose") +
  ggtitle("Titer by Dose Time") +
  geom_dotplot(binaxis = "y",
               stackdir = "center",
               stackgroups = FALSE,
               dotsize = 0.5,
               position = position_dodge())
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```
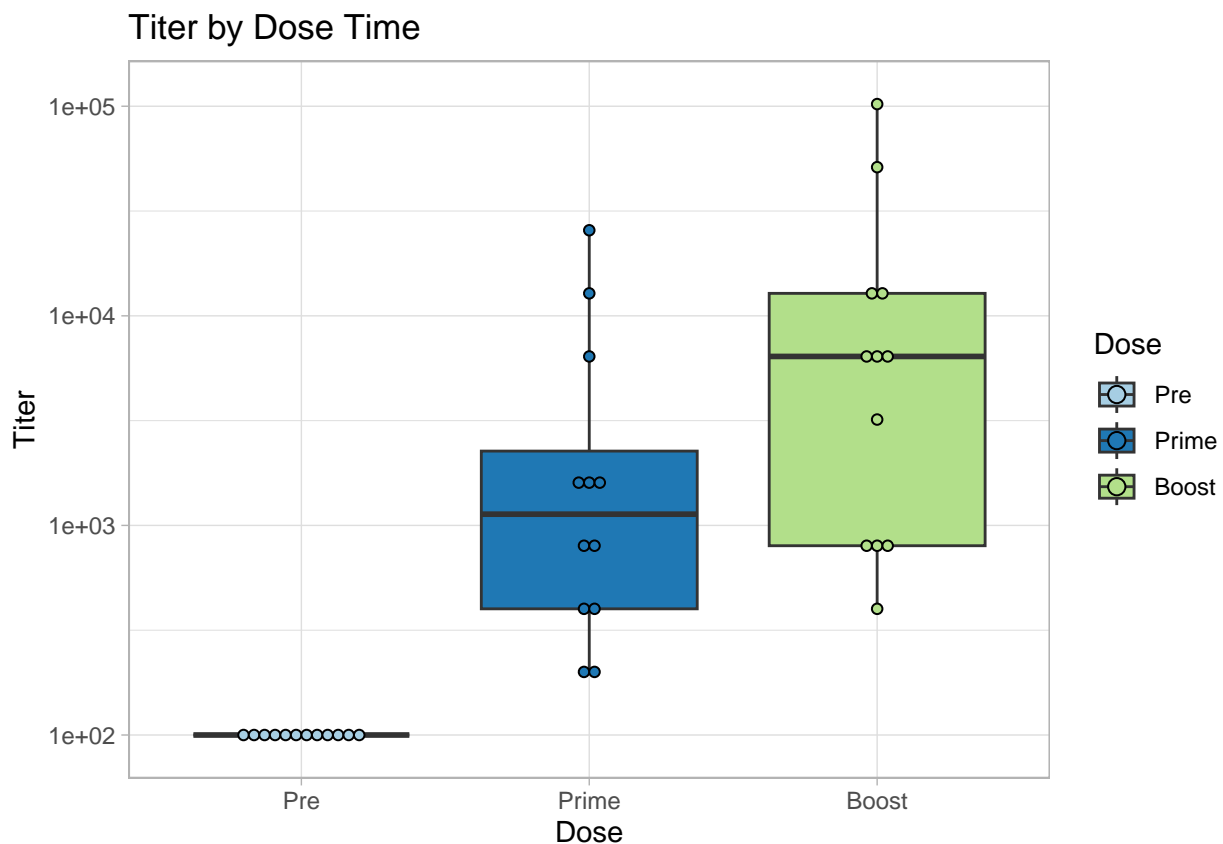
```
# check for outliers in titer column grouped by dose time
outlier_titer <- data2 %>%
  group_by(time) %>%
  identify_outliers(titer)
outlier_titer
```

```
## # A tibble: 4 x 10
##   time  name          id    treatment sex    titer id_treatment_sex log10_titer
##   <fct> <chr>         <chr> <fct>     <fct>  <dbl> <fct>                  <dbl>
## 1 Prime 9330 Vax Male~ 9330  LcrV      Male   25600 9330 LcrV Male          4.41
## 2 Prime 9050 Vax Male~ 9050  LcrV      Male   12800 9050 LcrV Male          4.11
## 3 Boost 9330 Vax Male~ 9330  LcrV      Male  102400 9330 LcrV Male          5.01
## 4 Boost 9050 Vax Male~ 9050  LcrV      Male   51200 9050 LcrV Male          4.71
## # i 2 more variables: is.outlier <lgl>, is.extreme <lgl>
```

```
# check for outliers in log10_titer column grouped by dose time
outlier_log10_titer <- data2 %>%
  filter(treatment == "F1") %>%
  group_by(time) %>%
  identify_outliers(log10_titer)
outlier_log10_titer
```

```
##  [1] time             name             id               treatment
##  [5] sex              titer            id_treatment_sex log10_titer
##  [9] is.outlier       is.extreme
## <0 rows> (or 0-length row.names)
```

```r
################################################################################
# Check Assumptions
################################################################################


#############
# Normality #
#############

# check for normality in F1 treatment group, excluding "Pre" dose rows
# (using Shapiro-Wilks test)
normality_f1 <- data2 %>%
  group_by(time) %>%
  filter(treatment == "F1") %>%
  filter(time != "Pre") %>%
  shapiro_test(log10_titer)
normality_f1
```

```
## # A tibble: 2 x 4
##   time  variable    statistic     p
##   <fct> <chr>           <dbl> <dbl>
## 1 Prime log10_titer     0.853 0.167
## 2 Boost log10_titer     0.866 0.212
```

```r
# check for normality in LcrV treatment group, excluding "Pre" dose rows
normality_lcrv <- data2 %>%
  group_by(time) %>%
  filter(treatment == "LcrV") %>%
  filter(time != "Pre") %>%
  shapiro_test(log10_titer)
normality_lcrv
```

```
## # A tibble: 2 x 4
##   time  variable    statistic     p
##   <fct> <chr>           <dbl> <dbl>
## 1 Prime log10_titer     0.832 0.111
## 2 Boost log10_titer     0.863 0.201
```

```r
# do we have to check for normality with both treatment (antigen) groups together?
normality_both <- data2 %>%
  group_by(time) %>%
  filter(time != "Pre") %>%
  shapiro_test(log10_titer)
normality_both
```

```
## # A tibble: 2 x 4
##   time  variable    statistic     p
##   <fct> <chr>           <dbl> <dbl>
## 1 Prime log10_titer     0.923 0.313
## 2 Boost log10_titer     0.938 0.475
```

```
s1 <- aov_ez("id", "log10_titer", data = filter(data2, treatment == "F1"),
             between = "sex",
             within = "time")
```
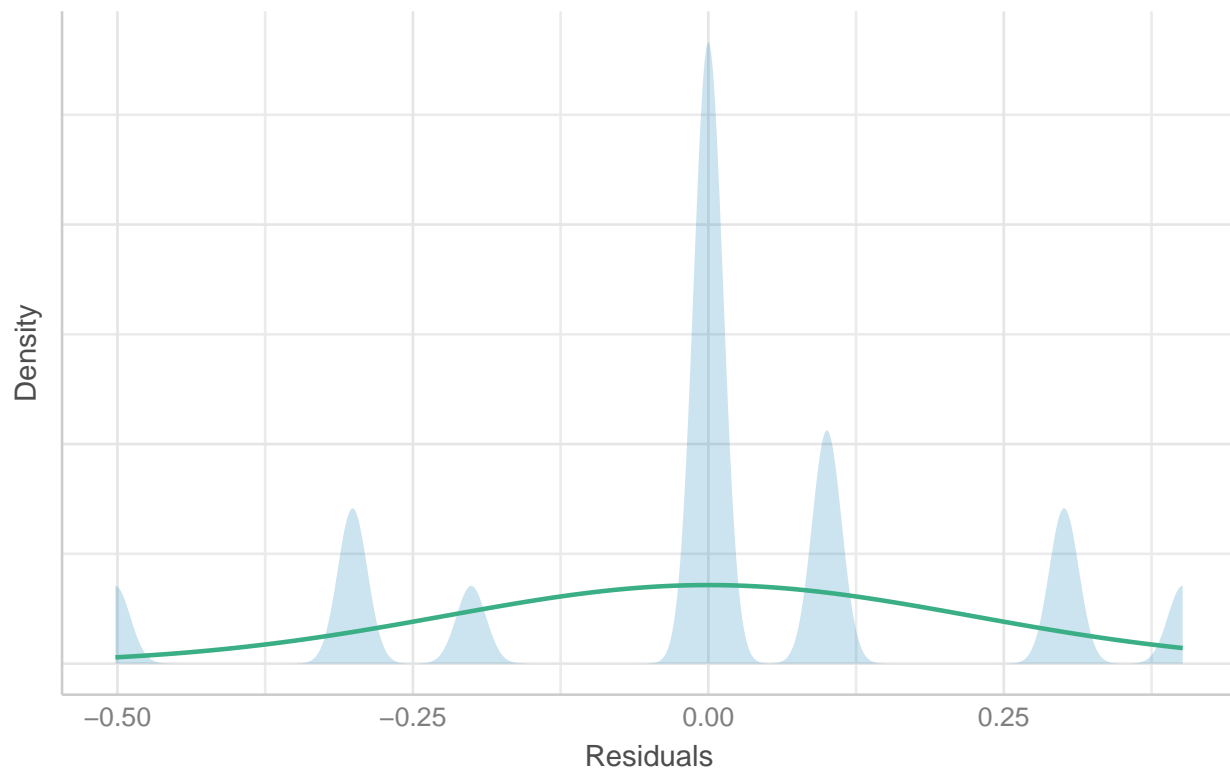
```
## Contrasts set to contr.sum for the following variables: sex
```

```
## Warning: More than one observation per design cell, aggregating data using `fun_aggregate = mean`.
## To turn off this warning, pass `fun_aggregate = mean` explicitly.
is_norm <- check_normality(s1)
```

```
plot(is_norm)
```
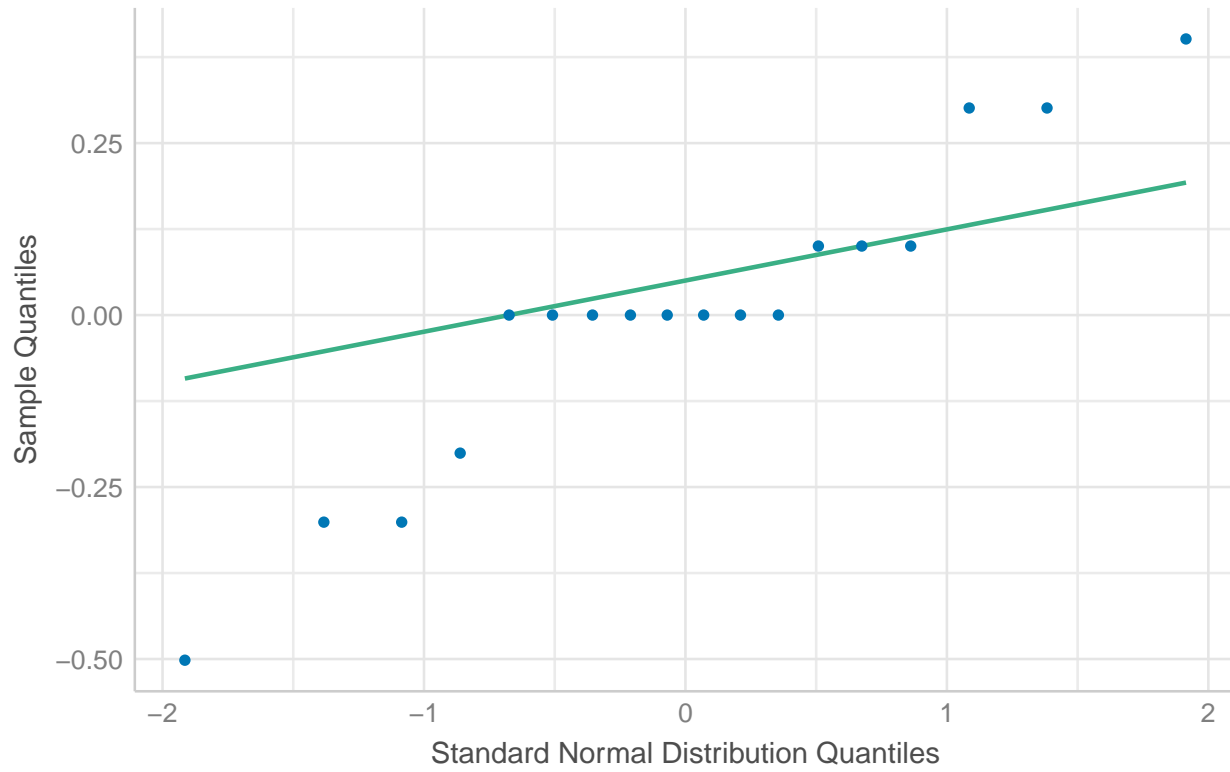
Normality of Residuals
Distribution should be close to the normal curve


```
plot(is_norm, type = "qq")
```

```
## For confidence bands, please install 'qqplotr'.
```

Normality of Residuals
Dots should fall along the line

```r
##############
# Sphericity #
##############
res_f1 <- anova_test(data = filter(data2, treatment == "F1"),
                     dv = log10_titer,
                     wid = id,
                     between = sex,
                     within = time)
res_f1
```

```
## ANOVA Table (type II tests)
##
## $ANOVA
##     Effect DFn DFd      F        p p<.05   ges
## 1      sex   1   4  2.571 0.184000       0.176
## 2     time   2   8 26.000 0.000316     * 0.812
## 3 sex:time   2   8  2.571 0.137000       0.300
##
## $`Mauchly's Test for Sphericity`
##     Effect     W    p p<.05
## 1     time 0.735 0.63
## 2 sex:time 0.735 0.63
##
## $`Sphericity Corrections`
##     Effect  GGe     DF[GG] p[GG] p[GG]<.05  HFe     DF[HF]     p[HF] p[HF]<.05
## 1     time 0.79 1.58, 6.32 0.001         * 1.22 2.44, 9.76 0.000316         *
## 2 sex:time 0.79 1.58, 6.32 0.155           1.22 2.44, 9.76 0.137000
```

```r
# sphericity is met for F1

res_lcrv <- anova_test(data = filter(data2, treatment == "LcrV"),
                       dv = log10_titer,
                       wid = id,
                       between = sex,
                       within = time)
res_lcrv
```

```
## ANOVA Table (type II tests)
##
## $ANOVA
##     Effect DFn DFd       F        p p<.05   ges
## 1      sex   1   4   3.000 1.58e-01       0.324
## 2     time   2   8 120.667 1.06e-06     * 0.916
## 3 sex:time   2   8   2.889 1.14e-01       0.206
##
## $`Mauchly's Test for Sphericity`
##     Effect     W     p p<.05
## 1     time 0.062 0.015     *
## 2 sex:time 0.062 0.015     *
##
## $`Sphericity Corrections`
##     Effect  GGe     DF[GG]    p[GG] p[GG]<.05  HFe     DF[HF]    p[HF]
## 1     time 0.516 1.03, 4.13 0.000323         * 0.532 1.06, 4.26 0.000265
```

```
## 2 sex:time 0.516 1.03, 4.13 0.163000          0.532 1.06, 4.26 0.161000
##   p[HF]<.05
## 1        *
## 2
```

```r
# sphericity is met for LcrV
```

```r
###############
# Equal Variance #
##############

library(afex)
library(performance) # for assumption checks

o1 <- aov_ez("id", "titer", filter(data2, treatment == "F1"),
            between = "sex",
            within = "time")
```

```
## Contrasts set to contr.sum for the following variables: sex
```

```r
check_homogeneity(o1)
```

```
## OK: There is not clear evidence for different variances across groups (Levene's Test, p = 0.175).
```

```
################################################################################
# ANOVA Testing - F1
################################################################################
#
#
# Two-Way
#
#
################################################################################
# Test effect of sex and time on log10_titer
two.way_f1 <- data2 %>%
  filter(treatment == "F1") %>%
  anova_test(dv = log10_titer,
             wid = id,
             between = sex,
             within = time,
             detailed = TRUE)
two.way_f1
```

```
## ANOVA Table (type II tests)
##
## $ANOVA
##         Effect DFn DFd     SSn   SSd        F        p p<.05   ges
## 1 (Intercept)   1   4 118.760 0.282 1684.979 2.10e-06     * 0.993
## 2         sex   1   4   0.181 0.282    2.571 1.84e-01       0.176
## 3        time   2   8   3.665 0.564   26.000 3.16e-04     * 0.812
## 4    sex:time   2   8   0.362 0.564    2.571 1.37e-01       0.300
##
## $`Mauchly's Test for Sphericity`
##     Effect     W    p p<.05
## 1     time 0.735 0.63
## 2 sex:time 0.735 0.63
##
## $`Sphericity Corrections`
##     Effect  GGe     DF[GG] p[GG] p[GG]<.05  HFe     DF[HF]    p[HF] p[HF]<.05
## 1     time 0.79 1.58, 6.32 0.001         * 1.22 2.44, 9.76 0.000316         *
## 2 sex:time 0.79 1.58, 6.32 0.155           1.22 2.44, 9.76 0.137000
```

```
# Pairwise comparison between each time point combination within each sex
pwc_two.way_f1 <- data2 %>%
  group_by(sex) %>%
  filter(treatment == "F1") %>%
  pairwise_t_test(
    log10_titer ~ time,
    paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc_two.way_f1
```

```
## # A tibble: 6 x 11
##   sex   .y.   group1 group2    n1    n2 statistic    df     p p.adj p.adj.signif
## * <fct> <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <dbl> <chr>
## 1 Fema~ log1~ Pre    Prime      3     3     -3.46     2 0.074 0.223 ns
```

```
## 2 Fema~ log1~ Pre    Boost    3    3    -8       2 0.015 0.046 *
## 3 Fema~ log1~ Prime  Boost    3    3    -2       2 0.184 0.552 ns
## 4 Male  log1~ Pre    Prime    3    3    -3.46    2 0.074 0.223 ns
## 5 Male  log1~ Pre    Boost    3    3    -5.29    2 0.034 0.102 ns
## 6 Male  log1~ Prime  Boost    3    3    -2.22    2 0.157 0.471 ns
```

```
################################################################################
#
#
# One-Way
#
#
################################################################################

# Test effect of time on log10_titer
one.way_f1 <- data2 %>%
  filter(treatment == "F1") %>%
  anova_test(dv = log10_titer,
             wid = id,
             within = time,
             detailed = TRUE)
one.way_f1
```

```
## ANOVA Table (type III tests)
##
## $ANOVA
##        Effect DFn DFd     SSn   SSd        F        p p<.05   ges
## 1 (Intercept)   1   5 118.760 0.463 1282.049 3.20e-07     * 0.988
## 2        time   2  10   3.665 0.926   19.783 3.34e-04     * 0.725
##
## $`Mauchly's Test for Sphericity`
##   Effect     W     p p<.05
## 1   time 0.578 0.335
##
## $`Sphericity Corrections`
##   Effect   GGe    DF[GG] p[GG] p[GG]<.05   HFe    DF[HF]    p[HF] p[HF]<.05
## 1   time 0.703 1.41, 7.03 0.002         * 0.896 1.79, 8.96 0.000617         *
```

```
# Pairwise Comparison Between Time Points
pwc_one.way_f1 <- data2 %>%
  filter(treatment == "F1") %>%
  pairwise_t_test(
    log10_titer ~ time,
    paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc_one.way_f1
```

```
## # A tibble: 3 x 10
##   .y.         group1 group2    n1    n2 statistic    df     p p.adj p.adj.signif
## * <chr>       <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <dbl> <chr>
## 1 log10_titer Pre    Prime      6     6     -5.48     5 0.003 0.008 **
## 2 log10_titer Pre    Boost      6     6     -5.97     5 0.002 0.006 **
## 3 log10_titer Prime  Boost      6     6     -2.33     5 0.067 0.201 ns
```

```
################################################################################
# ANOVA Testing - LcrV
################################################################################
#
#
# Two-Way
#
#
################################################################################
# Test effect of sex and time on log10_titer
two.way_lcrv <- data2 %>%
  filter(treatment == "LcrV") %>%
  anova_test(dv = log10_titer,
             wid = id,
             between = sex,
             within = time,
             detailed = TRUE)
two.way_lcrv
```

```
## ANOVA Table (type II tests)
##
## $ANOVA
##        Effect DFn DFd     SSn   SSd       F        p p<.05   ges
## 1 (Intercept)   1   4 196.551 0.967 813.366 9.00e-06     * 0.992
## 2         sex   1   4   0.725 0.967   3.000 1.58e-01       0.324
## 3        time   2   8  16.402 0.544 120.667 1.06e-06     * 0.916
## 4    sex:time   2   8   0.393 0.544   2.889 1.14e-01       0.206
##
## $`Mauchly's Test for Sphericity`
##     Effect     W     p p<.05
## 1     time 0.062 0.015     *
## 2 sex:time 0.062 0.015     *
##
## $`Sphericity Corrections`
##     Effect   GGe     DF[GG]    p[GG] p[GG]<.05   HFe     DF[HF]    p[HF]
## 1     time 0.516 1.03, 4.13 0.000323         * 0.532 1.06, 4.26 0.000265
## 2 sex:time 0.516 1.03, 4.13 0.163000           0.532 1.06, 4.26 0.161000
##   p[HF]<.05
## 1         *
## 2
```

```
# Pairwise comparison between each time point combination within each sex
pwc_two.way_lcrv <- data2 %>%
  group_by(sex) %>%
  filter(treatment == "LcrV") %>%
  pairwise_t_test(
    log10_titer ~ time,
    paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc_two.way_lcrv
```

```
## # A tibble: 6 x 11
```

```
##    sex    .y.    group1 group2    n1    n2 statistic   df       p p.adj p.adj.signif
## * <fct> <chr>  <chr>  <chr>  <int> <int>     <dbl> <dbl>   <dbl> <dbl> <chr>
## 1 Fema~ log1~ Pre    Prime      3     3      -7        2 0.02  0.059 ns
## 2 Fema~ log1~ Pre    Boost      3     3     -19.0      2 0.003 0.008 **
## 3 Fema~ log1~ Prime  Boost      3     3      -5.00     2 0.038 0.113 ns
## 4 Male  log1~ Pre    Prime      3     3      -5.27     2 0.034 0.103 ns
## 5 Male  log1~ Pre    Boost      3     3      -9.83     2 0.01  0.031 *
## 6 Male  log1~ Prime  Boost      3     3      -7.00     2 0.02  0.059 ns
```

```
################################################################################
#
#
# One-Way
#
#
################################################################################

# Test effect of time on log10_titer
one.way_lcrv <- data2 %>%
  filter(treatment == "LcrV") %>%
  anova_test(dv = log10_titer,
             wid = id,
             within = time,
             detailed = TRUE)
one.way_lcrv
```

```
## ANOVA Table (type III tests)
##
## $ANOVA
##        Effect DFn DFd     SSn   SSd       F        p p<.05   ges
## 1 (Intercept)   1   5 196.551 1.692 580.976 2.29e-06     * 0.987
## 2        time   2  10  16.402 0.936  87.581 4.59e-07     * 0.862
##
## $`Mauchly's Test for Sphericity`
##   Effect     W     p p<.05
## 1   time 0.337 0.114
##
## $`Sphericity Corrections`
##   Effect   GGe    DF[GG]  p[GG] p[GG]<.05   HFe    DF[HF]   p[HF] p[HF]<.05
## 1   time 0.601 1.2, 6.01 6.55e-05         * 0.687 1.37, 6.87 2.24e-05         *
```

```
# Pairwise Comparison Between Time Points
pwc_one.way_lcrv <- data2 %>%
  filter(treatment == "LcrV") %>%
  pairwise_t_test(
    log10_titer ~ time,
    paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc_one.way_lcrv
```

```
## # A tibble: 3 x 10
##   .y.       group1 group2    n1    n2 statistic    df       p p.adj p.adj.signif
## * <chr>     <chr>  <chr>  <int> <int>     <dbl> <dbl>   <dbl> <dbl> <chr>
## 1 log10_ti~ Pre    Prime      6     6     -7.65     5 6.07e-4  2e-3 **
## 2 log10_ti~ Pre    Boost      6     6    -11.2      5 9.99e-5  3e-4 ***
## 3 log10_ti~ Prime  Boost      6     6     -7.75     5 5.73e-4  2e-3 **
```

```
# Should we use Kruskal-Wallis/Dunn's Test since the "Pre" group is not normal?

library(permuco)
```

```
mod <- aovperm(log10_titer ~ time*sex + Error(id/(time)),
                    data = filter(data2, treatment == "F1"), method = "Rd_kheradPajouh_renaud")
mod
```

```
##
## Resampling test using Rd_kheradPajouh_renaud to handle nuisance variables and 5000 permutations.
##           SSn dfn    SSd dfd    MSEn    MSEd       F parametric P(>F)
## sex      0.1812   1 0.2819   4 0.1812 0.07048  2.571           0.184074
## time     3.6650   2 0.5639   8 1.8325 0.07048 26.000           0.000316
## time:sex 0.3625   2 0.5639   8 0.1812 0.07048  2.571           0.137278
##         resampled P(>F)
## sex             0.1794
## time            0.0004
## time:sex        0.1374
```

```
mod2 <- aovperm(log10_titer ~ time + sex + Error(id/(time)),
                data = filter(data2, treatment == "F1"), method = "Rd_kheradPajouh_renaud")
mod2
```

```
##
## Resampling test using Rd_kheradPajouh_renaud to handle nuisance variables and 5000 permutations.
##        SSn dfn    SSd dfd    MSEn    MSEd       F parametric P(>F)
## sex  0.1812   1 0.2819   4 0.1812 0.07048  2.571           0.1840740
## time 3.6650   2 0.9263  10 1.8325 0.09263 19.783           0.0003343
##      resampled P(>F)
## sex          0.1756
## time         0.0004
```

```
mod3 <- aovperm(log10_titer ~ time + Error(id/(time)),
                data = filter(data2, treatment == "F1"), method = "Rd_kheradPajouh_renaud")
mod3
```

```
##
## Resampling test using Rd_kheradPajouh_renaud to handle nuisance variables and 5000 permutations.
##        SSn dfn    SSd dfd  MSEn    MSEd      F parametric P(>F) resampled P(>F)
## time 3.665   2 0.9263  10 1.833 0.09263 19.78           0.0003343          0.0012
```

```
# Friedman
res.fried <- data2 %>%
  filter(treatment == "F1") %>% friedman_test(log10_titer ~ time |id)
res.fried
```

```
## # A tibble: 1 x 6
##   .y.              n statistic    df       p method
## * <chr>        <int>     <dbl> <dbl>   <dbl> <chr>
## 1 log10_titer      6      11.6     2 0.00308 Friedman test
```

```
# Effect size - overall difference among group
data2 %>%
  filter(treatment == "F1") %>%
  friedman_effsize(log10_titer ~ time |id)
```

19

```
## # A tibble: 1 x 5
##   .y.           n effsize method    magnitude
## * <chr>      <int>   <dbl> <chr>        <ord>
## 1 log10_titer    6   0.964 Kendall W large
```

```r
# pairwise comparisons
pwc <- data2 %>%
  filter(treatment == "F1") %>%
# filter(time == "Pre" | time == "Prime") %>%
  wilcox_test(log10_titer ~ time,
              paired = TRUE,
              p.adjust.method = "bonferroni")
pwc
```

```
## # A tibble: 3 x 9
##   .y.          group1 group2   n1    n2 statistic     p p.adj p.adj.signif
## * <chr>        <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <chr>
## 1 log10_titer Pre    Prime      6     6         0 0.034 0.104 ns
## 2 log10_titer Pre    Boost      6     6         0 0.034 0.102 ns
## 3 log10_titer Prime  Boost      6     6         0 0.054 0.164 ns
```

```r
# connect with mean, sd (numeric table)
# give p-values and adjusted p-values

# for LcrV
res.fried2 <- data2 %>%
  filter(treatment == "LcrV") %>% friedman_test(log10_titer ~ time |id)
res.fried2
```

```
## # A tibble: 1 x 6
##   .y.           n statistic    df       p method
## * <chr>      <int>     <dbl> <dbl>   <dbl> <chr>
## 1 log10_titer    6        12     2 0.00248 Friedman test
```

```r
# Effect size
data2 %>%
  filter(treatment == "LcrV") %>%
  friedman_effsize(log10_titer ~ time |id)
```

```
## # A tibble: 1 x 5
##   .y.           n effsize method    magnitude
## * <chr>      <int>   <dbl> <chr>        <ord>
## 1 log10_titer    6       1 Kendall W large
```

```r
# pairwise comparisons
pwc2 <- data2 %>%
  filter(treatment == "LcrV") %>%
  wilcox_test(log10_titer ~ time,
              paired = TRUE,
              p.adjust.method = "bonferroni")
pwc2
```

```
## # A tibble: 3 x 9
##   .y.        group1 group2    n1    n2 statistic     p p.adj p.adj.signif
## * <chr>      <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <chr>
## 1 log10_titer Pre   Prime      6     6         0 0.034 0.102 ns
## 2 log10_titer Pre   Boost      6     6         0 0.035 0.105 ns
## 3 log10_titer Prime Boost      6     6         0 0.034 0.102 ns
```