STAT850 HW4

Stewart Kerr

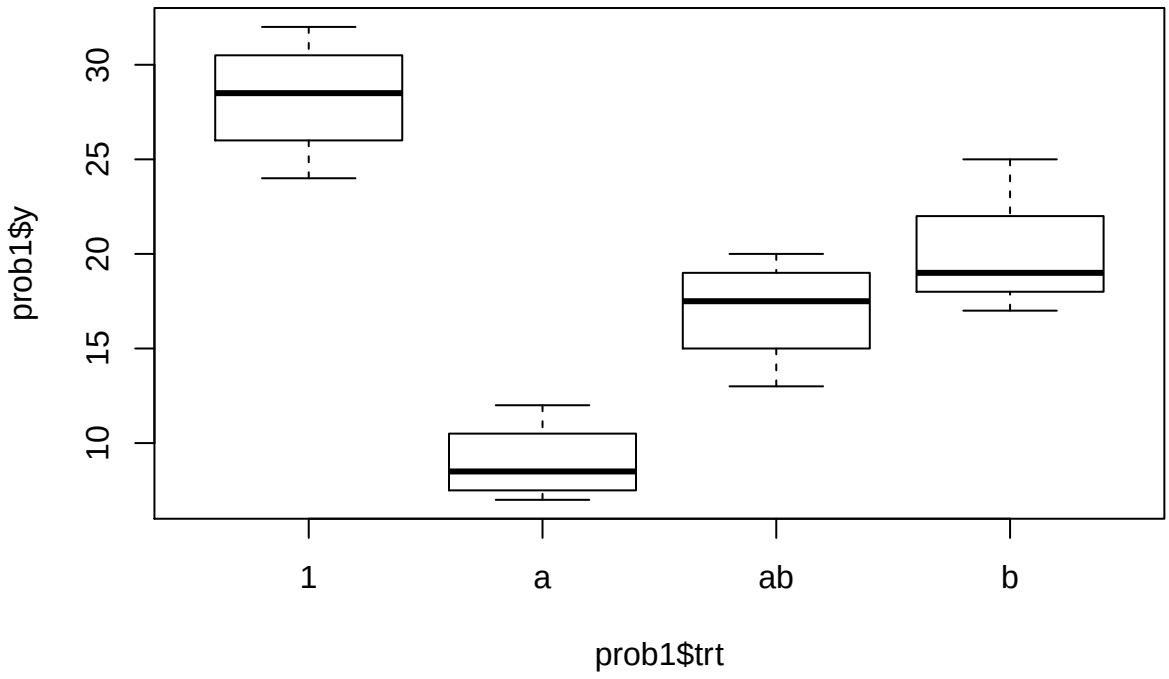
February 18, 2019

Problem 1

a.

```
#Problem 1a
#Load in data and contrasts
prob1 <- read.csv('hw04.csv')

#Plot the data
plot(prob1$y ~ prob1$trt)
```




```
#Build the contrasts for each test
c1 = c(-1/2,1/2,1/2,-1/2) #Main effect of A
c2 = c(-1/2,-1/2,1/2,1/2) #Main effect of B
c3 = c(1/2,-1/2,1/2,-1/2) #Interaction of AB
cmat = cbind(c1,c2,c3)

#Load the contrasts in
contrasts(prob1$trt) <- cmat
```

```
prob1_aov <- aov(y ~ trt, data = prob1)
summary.aov(prob1_aov, split = list(trt = list("Main effect of A" = 1, "Main effect of B" = 2, "Interac
```

```
##                                Df Sum Sq Mean Sq F value   Pr(>F)
## trt                             3  759.2   253.1  27.924 1.07e-05 ***
##   trt: Main effect of A         1  495.1   495.1  54.628 8.38e-06 ***
##   trt: Main effect of B         1    0.1     0.1   0.007 0.935185
##   trt: Interaction effect of AB 1  264.1   264.1  29.138 0.000161 ***
## Residuals                      12  108.8     9.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From the above aov output, we see that both the main effect of A and the interaction effect of AB are significant. However, the data does not indicate that the main effect of B is significant. Due to the interaction of AB having a significant effect on y, however, we will say that B *does* have an effect.

b.

```
#Problem 1b
prob1_fit = lm(y ~ a * b, data = prob1)
anova(prob1_fit)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## a          1 495.06  495.06 54.6276 8.375e-06 ***
## b          1   0.06    0.06  0.0069 0.9351846
## a:b        1 264.06  264.06 29.1379 0.0001606 ***
## Residuals 12 108.75    9.06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

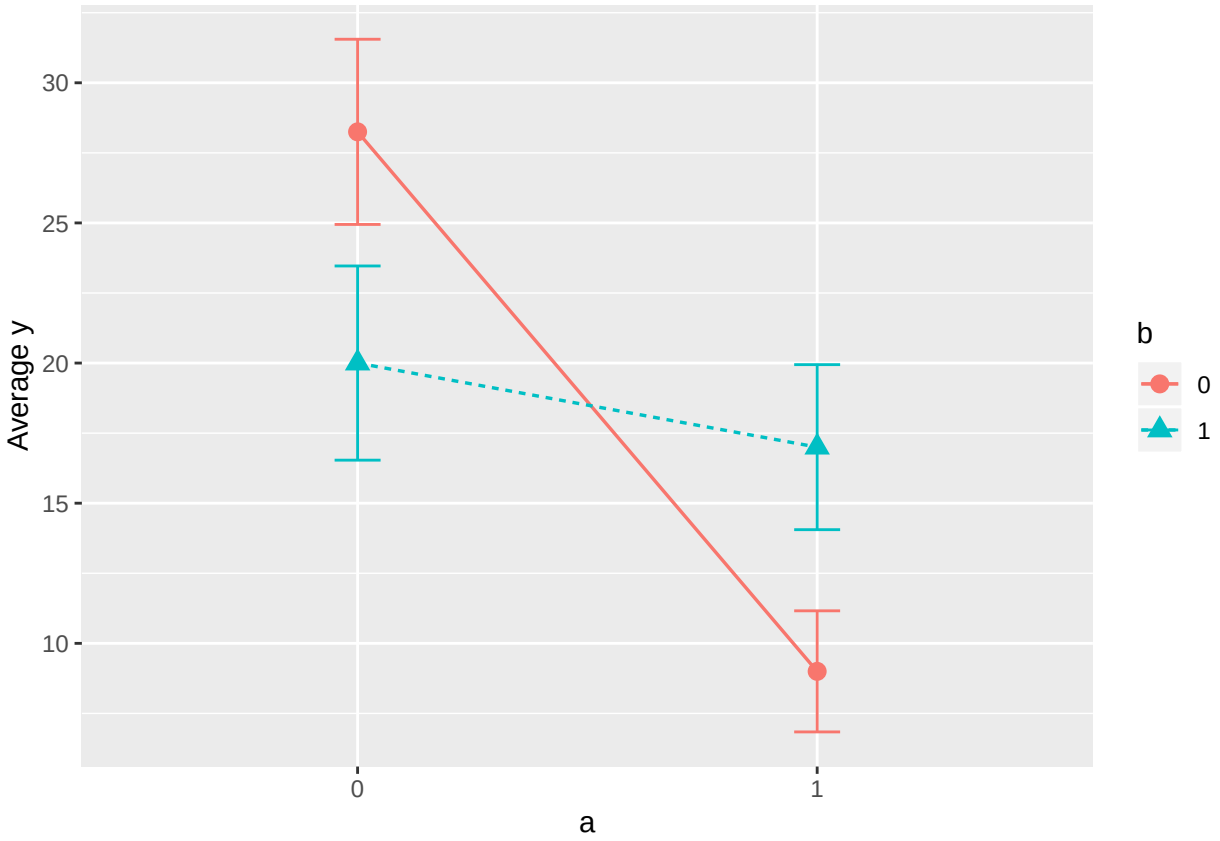
```
#Note that anova() gives type I SS but since our design is balanced this is equal to type III SS
```

The two-factor model for A,B, and interaction AB is given above. Note tat the F-values for the two-factor analysis are the same as the F values using contrasts in part a.

c.

```
#Problem 1c
#Calculate mean and se for each treatment
prob1_plot <- summarise(group_by(prob1,a,b), ybar = mean(y), sd = sd(y))
prob1_plot$a = factor(prob1_plot$a)
prob1_plot$b = factor(prob1_plot$b)

gp <- ggplot(prob1_plot, aes(x=a, y=ybar, colour=b, group=b))
gp + geom_line(aes(linetype=b), size=.6) +
     geom_point(aes(shape=b), size=3) +
     geom_errorbar(aes(ymax=ybar+sd, ymin=ybar-sd), width=.1) +
     ylab("Average y")
```



On my plot, I've included standard deviation bars. They represent the interval that is ±1 standard deviation from each treatment mean (which correspond to points on the line).

d.

```
#Problem 1d
#Perform regression and anova
prob1_reg <- lm(y ~ x1 + x2 + x3, data = prob1)
anova(prob1_reg)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## x1         1 495.06  495.06 54.6276 8.375e-06 ***
## x2         1   0.06    0.06  0.0069 0.9351846
## x3         1 264.06  264.06 29.1379 0.0001606 ***
## Residuals 12 108.75    9.06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the data is coded such that A = x1, B = x2, and AB = x3 (with no low A and low B being the intercept/residuals). The sum of squares error matches with the calculated SS in part a and b. Now I will change the fitting order.

```
#Problem 1d
prob1_reg <- lm(y ~ x3 + x2 + x1, data = prob1)
anova(prob1_reg)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value   Pr(>F)
## x3         1  13.02   13.02  1.4368   0.2538
## x2         1   5.04    5.04  0.5563   0.4701
## x1         1 741.12  741.12 81.7793 1.05e-06 ***
## Residuals 12 108.75    9.06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The sum of squares error changes when we change the fitting order. This occurs because the `anova()` function gives sequential SS, that is, it gives the sum of squares *given* the previous terms are in the model. In part a, we found *additional SS* which is the SS upon adding that particular term last (with all other terms in the model).

Problem 2

a. This is a CRD balanced two factor 4x3 experiment. The statistical model for this experiment is as follows:

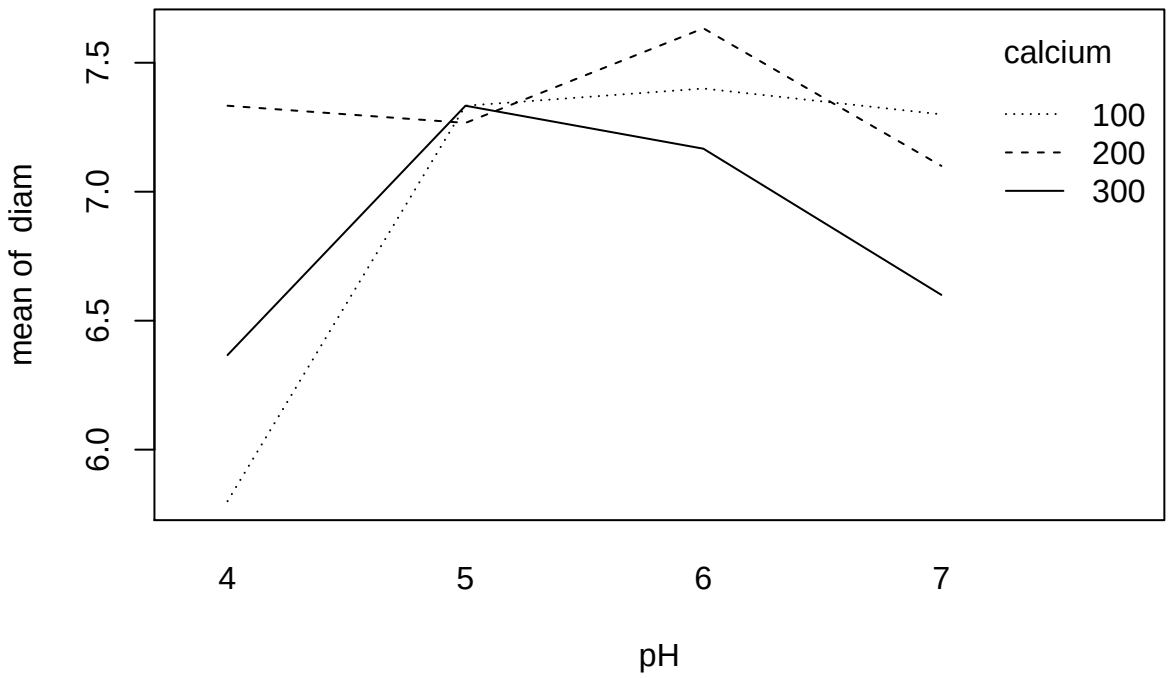
$$Y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}$$

In this model,
Y_{ijk} represents the response (increase in diameter) for a given experimental unit
μ represents the baseline term
α_i represents the different pH levels (i = 4,5,6, and 7)
β_j represents the different calcium levels (j = 100, 200, or 300)
$(\alpha\beta)_{ij}$ represents the interaction term for each of the factor-level combinations of pH and calcium
$\epsilon_{ijk} \sim$ i.i.d N(0,σ^2_ϵ) represents the within group variation/error

b.

```
#Problem 2b
#Load data
orange <- read.csv("~/2019spring/STAT850/hw4/orange.txt", sep="")
orange$calcium = factor(orange$calcium)
orange$pH = factor(orange$pH)

#Construct plots
with(orange, interaction.plot(pH, calcium, diam))
```



From the interaction diagram, I would conclude that there is likely an interaction between pH and calcium level on the tree diameter because the lines are not parallel. The interaction appears to be fairly strong. Both calcium level and pH also appear to have main effects. It's interesting to note that at low calcium, pH has a positive affect on diameter but at higher calcium pH's effect becomes negative.

c.

```
#Problem 2c
orange_aov <- aov(diam ~ pH * calcium, data = orange)
anova(orange_aov)
```

```
## Analysis of Variance Table
##
## Response: diam
##            Df Sum Sq Mean Sq F value    Pr(>F)
## pH          3 4.4608 1.48694 21.9385 4.635e-07 ***
## calcium     2 1.4672 0.73361 10.8238 0.0004462 ***
## pH:calcium  6 3.2550 0.54250  8.0041 8.186e-05 ***
## Residuals  24 1.6267 0.06778
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA table for this data is given above. From the table, we can see that each of the terms are significant - that is, both calcium and pH alone have an association with tree diameter and the interaction between calcium and pH also has an association with tree diameter. If the interaction term had not been significant, then I would have dropped it from the model and tested for significance of the individual factors pH and calcium, but since it is significant I will not do that.

d. To use Fisher's LSD, we first have to perform an F-test to determine that there is a difference in at least

one of the group means. This was demonstrated above. Now I will calculate the LSD that will be used to determine how diameter differs based on level of calcium and/or pH.

```
#Problem 2d
model.tables(orange_aov, type = "means", se = TRUE)
```

```
## Tables of means
## Grand mean
##
## 7.052778
##
##  pH
## pH
##     4     5     6     7
## 6.500 7.311 7.400 7.000
##
##  calcium
## calcium
##   100   200   300
## 6.958 7.333 6.867
##
##  pH:calcium
##    calcium
## pH  100   200   300
##   4 5.800 7.333 6.367
##   5 7.333 7.267 7.333
##   6 7.400 7.633 7.167
##   7 7.300 7.100 6.600
##
## Standard errors for differences of means
##             pH calcium pH:calcium
##         0.1227  0.1063     0.2126
## replic.      9      12          3
```

We can use the standard errors from `model.tables()` to calculate LSD for pH, levels of calcium, and combination means.

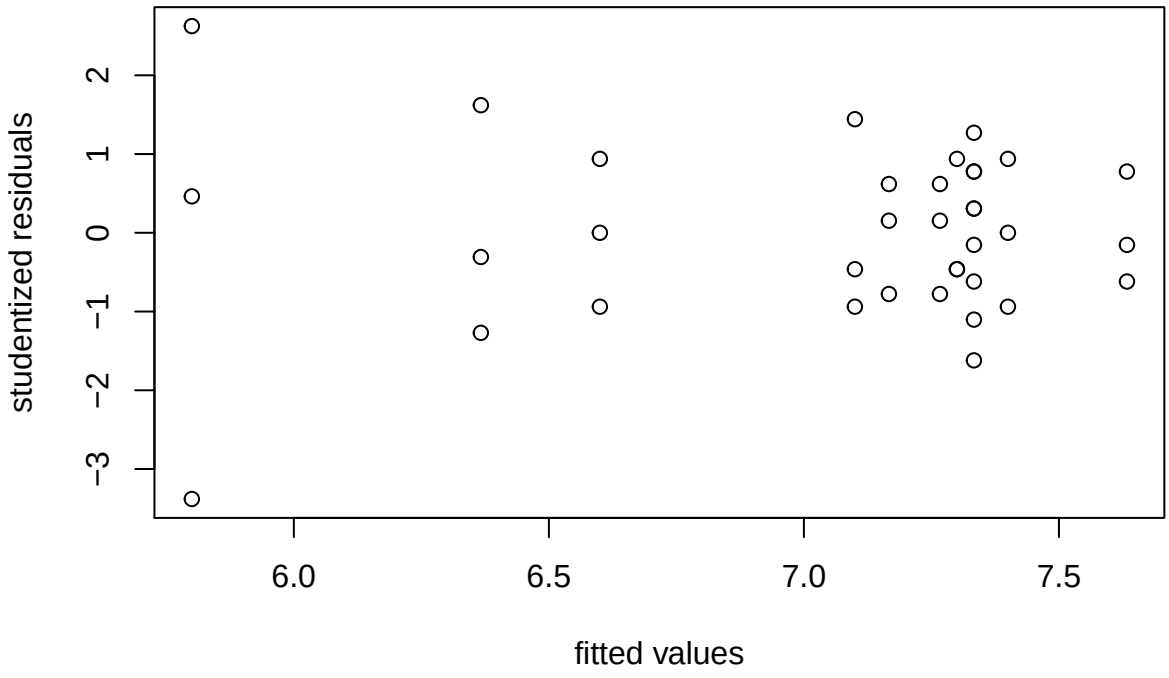
```
dfError = 24
LSD_pH = qt(0.975, df = dfError)*0.1227*sqrt(2/9)
LSD_Ca = qt(0.975, df = dfError)*0.1063*sqrt(2/12)
LSD_Comb = qt(0.975, df = dfError)*0.2126*sqrt(2/3)
kable(rbind(LSD_pH,LSD_Ca,LSD_Comb), col.names = "LSD" )
```

	LSD
LSD_pH	0.1193786
LSD_Ca	0.0895666
LSD_Comb	0.3582663

The table of LSD for each of the group means (pH, Ca, and combined) is given above. If we compare the LSD to our calculated means above, we see that only pH 5 and 6 do not differ significantly, all calcium levels differ significantly, and there are a number of significant differences in the combined group means.

e.

```
#Problem 2e
plot(rstudent(orange_aov) ~ fitted(orange_aov), xlab = "fitted values", ylab = "studentized residuals")
```



We have a potential outlier with a studentized residual less than -3 at a fitted value of about 5.5. Other than that, we appear to have fairly even spread about 0 and there are no obvious patterns in the residuals. Other than the potential outlier, these residuals look good.

Problem 3

See attached sheet

Problem 4

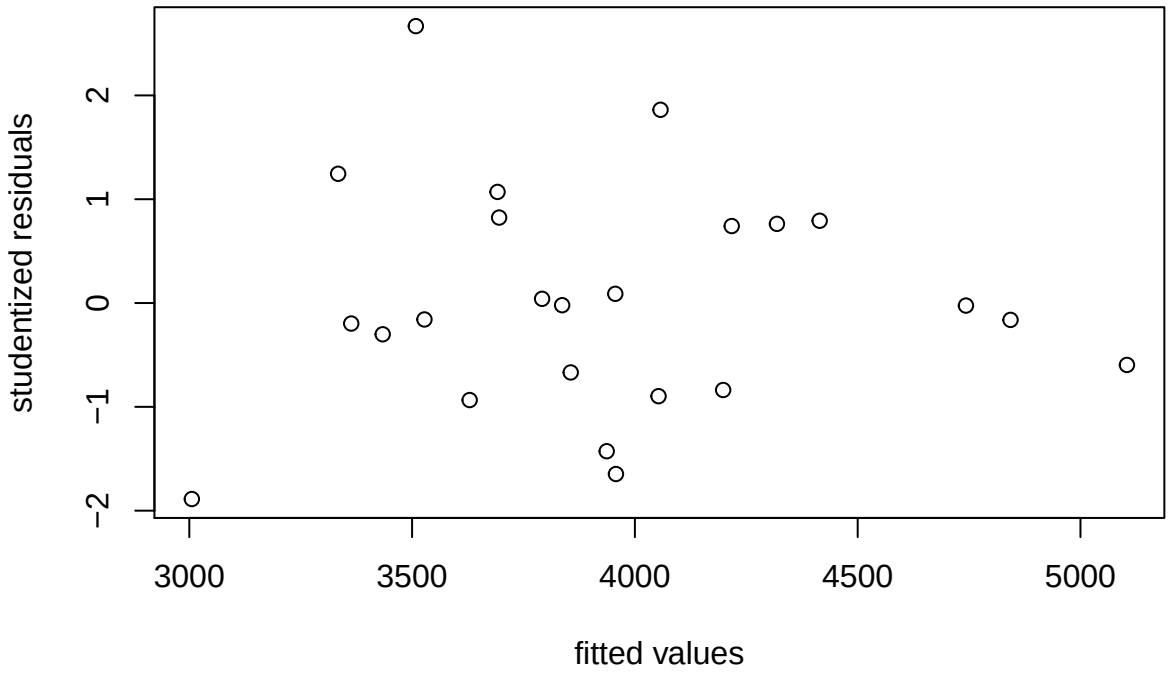
a.

```
#Problem 4a
#Load the data in and create sum variable
apple <- read.csv("~/2019spring/STAT850/hw4/apple.csv") %>%
  mutate(weight_tot = weight_b1+weight_b2)
apple$treatment = factor(apple$treatment)
apple$block = factor(apple$block)

#ANOVA LM with block
apple_lm <- lm(weight_tot ~ treatment + block, data = apple)
anova(apple_lm)
```

```
## Analysis of Variance Table
##
## Response: weight_tot
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## treatment  5 4334223  866845  5.4384 0.004738 **
## block      3 1463654  487885  3.0609 0.060541 .
## Residuals 15 2390887  159392
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Plot residuals
plot(rstudent(apple_lm) ~ fitted(apple_lm), xlab = "fitted values", ylab = "studentized residuals")
```

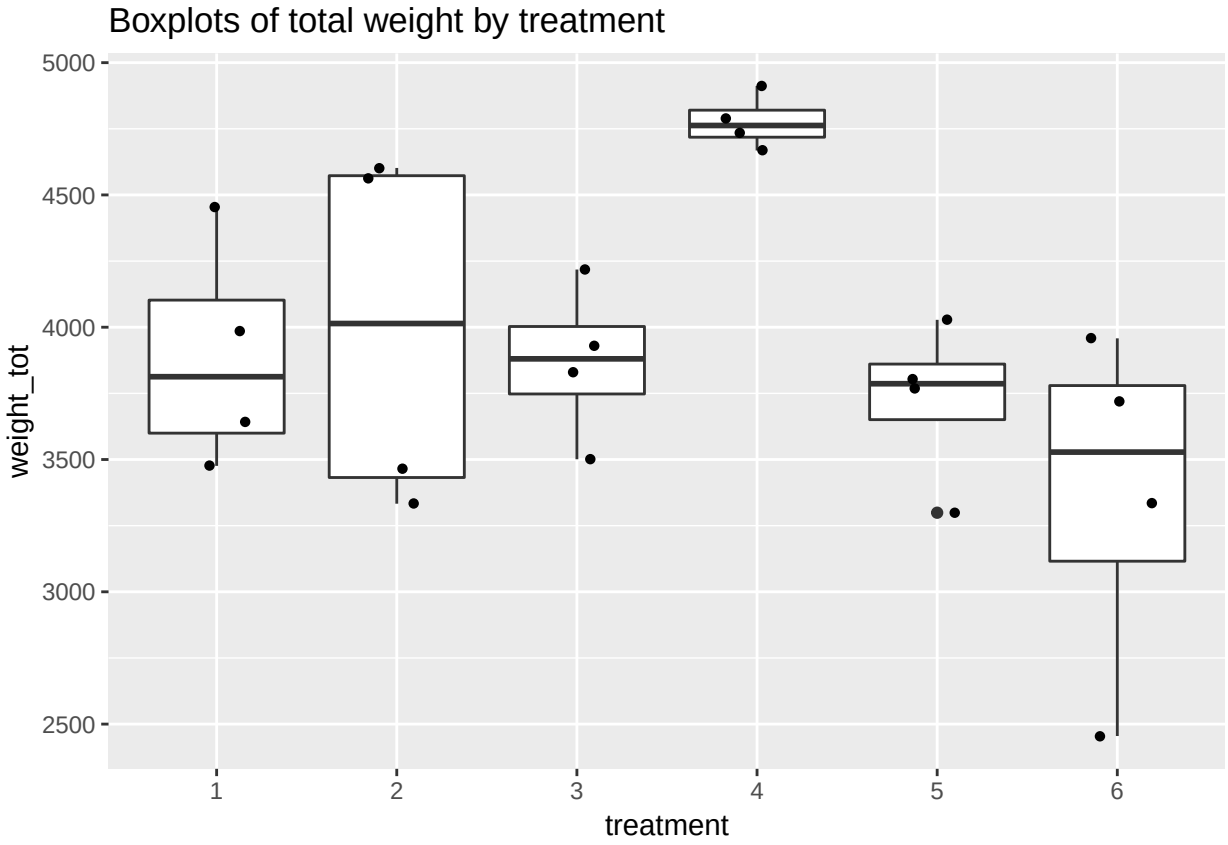


The null hypothesis is that each of the six different treatments will yield approximately the same total weight in apples. That is, $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5 = \mu_6$. From the ANOVA table, the F-statistic, which is MS_{Trt}/MS_{Error} is 5.4384 with 5 degrees of freedom in the numerator and 15 degrees of freedom for error. This corresponds to a p-value of 0.004. Thus, based on this data, there is strong evidence to reject the null hypothesis and suggest that at least one of the treatments has a different mean apple yield. Note that because we used a blocked design, our SSE was lower than if we had not accounted for blocks.

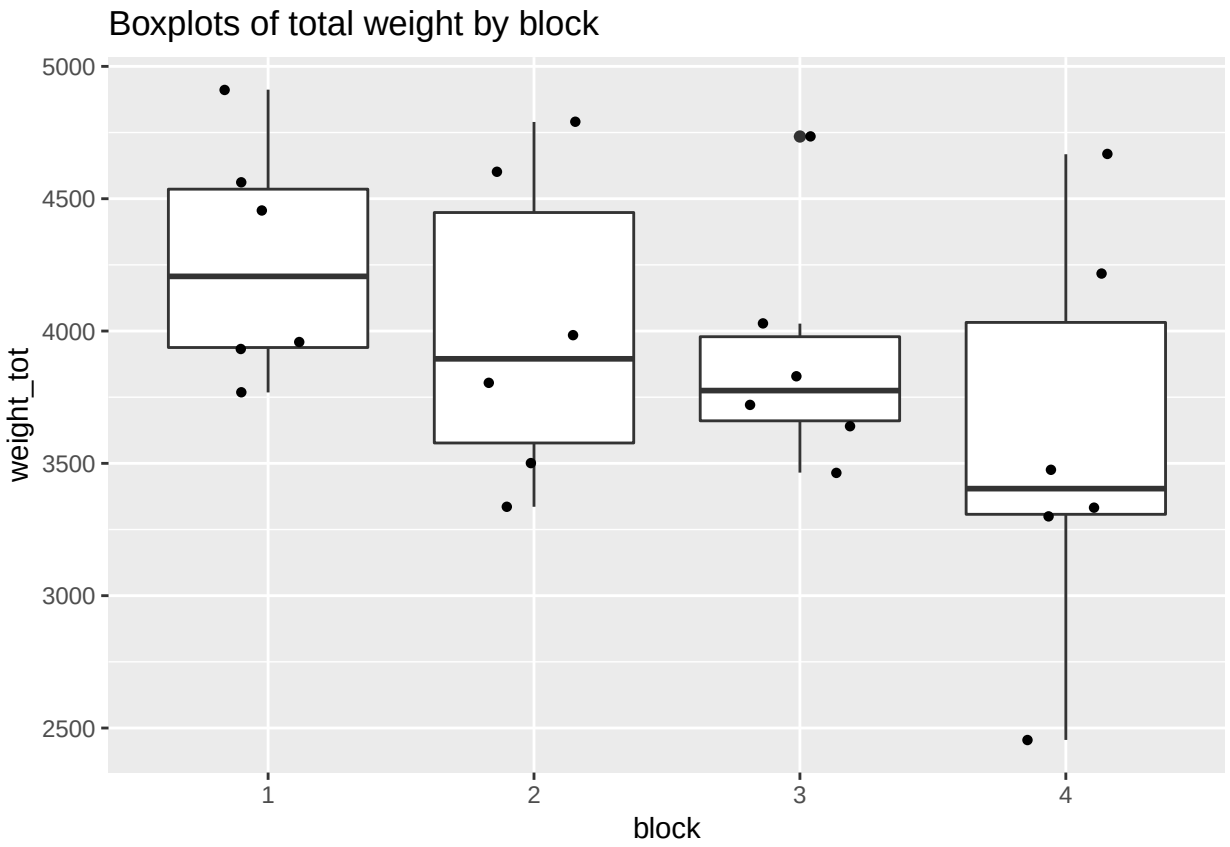
In order for our F-test to be valid, we must assume equal variance among each of the treatment groups. A plot of studentized residuals vs. fitted values is given above. There appears to be a difference in variance among the group at a fitted value of 4800 (treatment 4) and all of the other groups. This is potentially a problem and might necessitate further investigation. Other than that, the variance appears to be normally distributed around 0.

b.

```
#Problem 4b
#Plot the data
#Create a grouped boxplot
ggplot(apple, aes(x=treatment, y=weight_tot, group=treatment)) +
  geom_boxplot() +
  geom_jitter(shape=16, position=position_jitter(0.2)) +
  ggtitle('Boxplots of total weight by treatment')
```




```
ggplot(apple, aes(x=block, y=weight_tot, group=block)) +
  geom_boxplot() +
  geom_jitter(shape=16, position=position_jitter(0.2)) +
  ggtitle('Boxplots of total weight by block')
```



I would definitely recommend treatment 4 in order to maximize the apple yield. From the boxplots, it's clear that treatment 4 outperforms the other treatments. In fact, the worst total weight for treatment 4 for is about the same as the best total weight for the other treatments. Based on this data, I don't think that any of the other treatments can compare to treatment 4 in terms of maximizing yield.

It's also worth noting that block 4 appears to have performed much worse than the other blocks. While blocks 1-3 are on elevated ground, block 4 trees were planted in a low region. Therefore, I would suggest that future trees be planted in elevated regions if possible.

c. At first glance, it appears that there is a strong, positive interaction between planting flowers and mowing. This is because each of treatments 1-3 have about the same average branch weight (indicating little to no main effects of mowing and flowers by themselves) yet treatment 4 has a much larger average branch weight. In order to confirm this hypothesis, I will calculate Fisher's LSD and see if treatment 4 differs significantly from treatments 1,2, and 3.

```
#Problem 4c
apple_aov = aov(weight_tot ~ treatment + block, data = apple)
model.tables(apple_aov, type = "means", se = TRUE)
```

```
## Tables of means
## Grand mean
##
## 3936.417
##
##  treatment
## treatment
##    1    2    3    4    5    6
## 3889 3991 3870 4776 3725 3367
```

```
##
##  block
## block
##    1    2    3    4
## 4264 4003 3903 3575
##
## Standard errors for differences of means
##         treatment block
##             282.3 230.5
## replic.         4     6
```

```
LSD = qt(0.975, df = 18)*327.2*sqrt(2/4)
paste("The LSD for the apple data is: ", round(LSD,1))
```

```
## [1] "The LSD for the apple data is:  486.1"
```

Based on this calculated LSD, it appears that treatment 4 *does* differ significantly from the treatments 1,2, and 3 - indicating that there is a strong interaction between planting flowers and mowing.

d. We are assuming that the weight of a branch is normally distributed with mean μ and variance σ^2_ϵ. From STAT609, we know that the sum of two normal random variables is also normally distributed. Thus, it's also appropriate to use the sum of the weights of two branches as our dependent variable.

Code

```
## ----include = FALSE----------------------------------------------------------
knitr::opts_chunk$set(echo = TRUE)
library(dplyr)
library(knitr)
library(car)
library(ggplot2)

set.seed(1104)             # make random results reproducible

this_file <- "kerr_stat850_hw04.Rmd"  # used to automatically generate code appendix

## ------------------------------------------------------------------------------
#Problem 1a
#Load in data and contrasts
prob1 <- read.csv('hw04.csv')

#Plot the data
plot(prob1$y ~ prob1$trt)

#Build the contrasts for each test
c1 = c(-1/2,1/2,1/2,-1/2) #Main effect of A
c2 = c(-1/2,-1/2,1/2,1/2) #Main effect of B
c3 = c(1/2,-1/2,1/2,-1/2) #Interaction of AB
cmat = cbind(c1,c2,c3)

#Load the contrasts in
contrasts(prob1$trt) <- cmat

prob1_aov <- aov(y ~ trt, data = prob1)
summary.aov(prob1_aov, split = list(trt = list("Main effect of A" = 1, "Main effect of B" = 2, "Interac
```

```r
## ------------------------------------------------------------------------
#Problem 1b
prob1_fit = lm(y ~ a * b, data = prob1)
anova(prob1_fit)
#Note that anova() gives type I SS but since our design is balanced this is equal to type III SS

## ------------------------------------------------------------------------
#Problem 1c
#Calculate mean and se for each treatment
prob1_plot <- summarise(group_by(prob1,a,b), ybar = mean(y), sd = sd(y))
prob1_plot$a = factor(prob1_plot$a)
prob1_plot$b = factor(prob1_plot$b)

gp <- ggplot(prob1_plot, aes(x=a, y=ybar, colour=b, group=b))
gp + geom_line(aes(linetype=b), size=.6) +
     geom_point(aes(shape=b), size=3) +
     geom_errorbar(aes(ymax=ybar+sd, ymin=ybar-sd), width=.1) +
     ylab("Average y")

## ------------------------------------------------------------------------
#Problem 1d
#Perform regression and anova
prob1_reg <- lm(y ~ x1 + x2 + x3, data = prob1)
anova(prob1_reg)

## ------------------------------------------------------------------------
#Problem 1d
prob1_reg <- lm(y ~ x3 + x2 + x1, data = prob1)
anova(prob1_reg)

## ------------------------------------------------------------------------
#Problem 2b
#Load data
orange <- read.csv("~/2019spring/STAT850/hw4/orange.txt", sep="")
orange$calcium = factor(orange$calcium)
orange$pH = factor(orange$pH)

#Construct plots
with(orange, interaction.plot(pH, calcium, diam))

## ------------------------------------------------------------------------
#Problem 2c
orange_aov <- aov(diam ~ pH * calcium, data = orange)
anova(orange_aov)

## ------------------------------------------------------------------------
#Problem 2d
model.tables(orange_aov, type = "means", se = TRUE)

## ------------------------------------------------------------------------
dfError = 24
LSD_pH = qt(0.975, df = dfError)*0.1227*sqrt(2/9)
```

```r
LSD_Ca = qt(0.975, df = dfError)*0.1063*sqrt(2/12)
LSD_Comb = qt(0.975, df = dfError)*0.2126*sqrt(2/3)
kable(rbind(LSD_pH,LSD_Ca,LSD_Comb), col.names = "LSD" )

## ------------------------------------------------------------------------
#Problem 2e
plot(rstudent(orange_aov) ~ fitted(orange_aov), xlab = "fitted values", ylab = "studentized residuals")

## ------------------------------------------------------------------------
#Problem 4a
#Load the data in and create sum variable
apple <- read.csv("~/2019spring/STAT850/hw4/apple.csv") %>%
  mutate(weight_tot = weight_b1+weight_b2)
apple$treatment = factor(apple$treatment)
apple$block = factor(apple$block)

#ANOVA LM with block
apple_lm <- lm(weight_tot ~ treatment + block, data = apple)
anova(apple_lm)

#Plot residuals
plot(rstudent(apple_lm) ~ fitted(apple_lm), xlab = "fitted values", ylab = "studentized residuals")

## ------------------------------------------------------------------------
#Problem 4b
#Plot the data
#Create a grouped boxplot
ggplot(apple, aes(x=treatment, y=weight_tot, group=treatment)) +
  geom_boxplot() +
  geom_jitter(shape=16, position=position_jitter(0.2)) +
  ggtitle('Boxplots of total weight by treatment')
ggplot(apple, aes(x=block, y=weight_tot, group=block)) +
  geom_boxplot() +
  geom_jitter(shape=16, position=position_jitter(0.2)) +
  ggtitle('Boxplots of total weight by block')

## ------------------------------------------------------------------------
#Problem 4c
apple_aov = aov(weight_tot ~ treatment + block, data = apple)
model.tables(apple_aov, type = "means", se = TRUE)

LSD = qt(0.975, df = 18)*327.2*sqrt(2/4)
paste("The LSD for the apple data is: ", round(LSD,1))

## ----code = readLines(purl(this_file, documentation = 1)), echo = T, eval = F----
## # this R markdown chunk generates a code appendix
```