2015S

Fountain*, Crain

2015F

2015F1

[2017S1]
Find the best model for predicting Y based on X1 and X2. Y is the amount of profit that a company makes in a month. X1 is the number of months that the company has been in business. X2 is the amount spent on advertising.
Consider as predictors all possible linear and quadratic terms ($X1$, $X1^2$, $X2$, $X2^2$, and $X1X2$). Consider possible transformations of Y. Include all appropriate diagnostics. When you have found your "best" model, predict a new Y when X1 = 20 and X2 = $1,500, giving a 95% prediction interval. The data set, shown below, appears in "Profits.xlsx".

2015F2

A replicated fractional factorial design is used to investigate the effect of five factors on the free height of leaf springs used in an automotive application. The factors are (A) furnace temperature, (B) heating time, (C) transfer time, (D) hold down time, and (E) quench oil temperature. There are 3 observations at each setting.
Write out the alias structure for this design. What is the resolution of this design? Analyze the data. What factors influence the mean free height? The data set appears in the file "Springs.xlsx".

2016S

Fountain, Tableman*

2016S1

2017F1
Find the best model for predicting Y (weight) based on X1 (age), X2 (height), and X3 (indicator for male). Consider as predictors all possible linear and quadratic terms. Consider possible transformations of Y. Include all appropriate diagnostics. When you have found your "best" model, predict a new Y when X1 = 26, X2 = 70, and X3 = 1, giving a 95% prediction interval. The data set, shown below, appears in "RegressionSpr16.xlsx".

2016S2

2017F2
A process engineer is testing the yield of a product manufactured on three specific machines. Each machine can be operated at fixed high and low power settings, although the actual settings differ from one machine to the next. Furthermore, a machine has three stations on which the product is formed, and these are the same for each machine. An experiment is conducted in which each machine is tested at both power settings, and three observations on yield are taken from each station. The runs are made in random order. Analyze this experiment. The data set, shown below, appears in "DesignSpr16.xlsx".

```
DesignSpr16 <- readxl::read_excel("qe_lab/DesignSpr16.xlsx")
```

```
## New names:
## * `` -> ...1
## * `` -> ...2
## * `` -> ...4
## * `` -> ...5
## * `` -> ...6
## * ... and 4 more problems
```

```
library(tidyverse)
```

```
## -- Attaching packages -----------------------------------------------------------------------------------
## v ggplot2 3.2.0     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## -- Conflicts --------------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

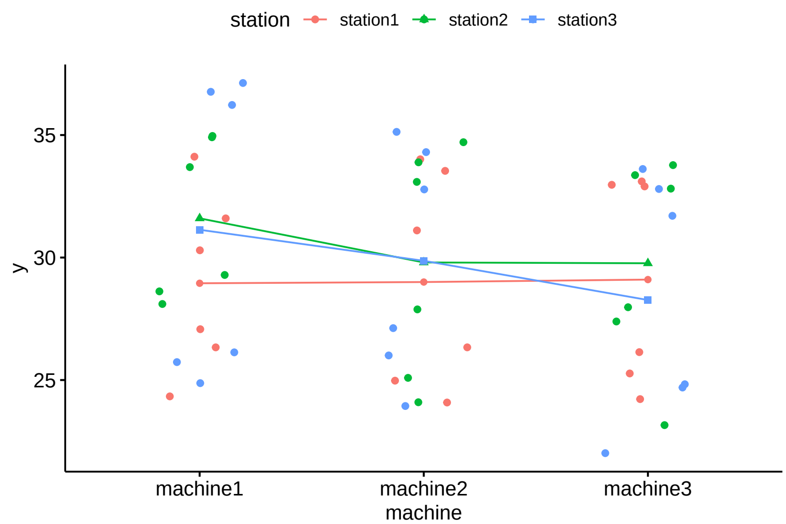
```
table_2016s2 <- gather(DesignSpr16[c(2:4,6:8),c(2:4,6:8,10:12)])
names(table_2016s2) <- c("machine","y")
table_2016s2 <- table_2016s2[c("y","machine")]
table_2016s2$machine <- as.factor(c(rep("machine1",18),rep("machine2",18),rep("machine3",18)) )
table_2016s2$station <- as.factor(rep(c(rep("station1",6),rep("station2",6),rep("station3",6)),3) )
table_2016s2$power <- as.factor(rep(c(rep("power1",3),rep("power2",3)),9) )
str(table_2016s2)
```

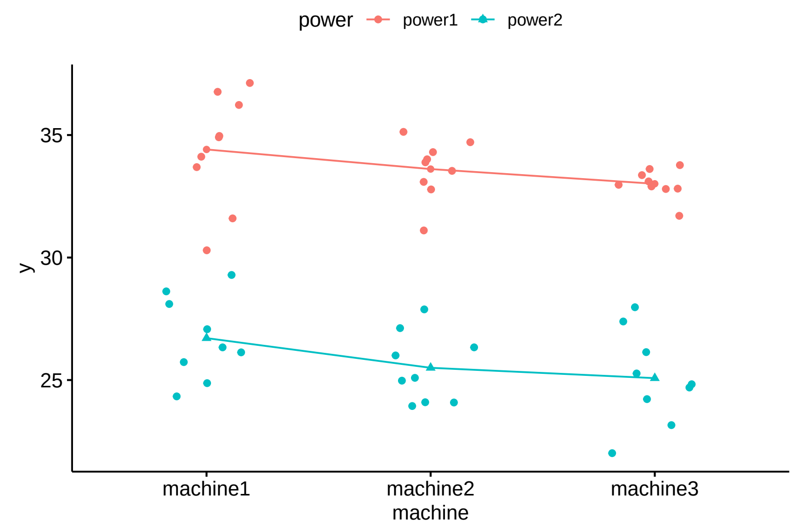
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    54 obs. of  4 variables:
##  $ y      : num  34.1 30.3 31.6 24.3 26.3 27.1 33.7 34.9 35 28.1 ...
##  $ machine: Factor w/ 3 levels "machine1","machine2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ station: Factor w/ 3 levels "station1","station2",..: 1 1 1 1 1 1 2 2 2 2 ...
##  $ power  : Factor w/ 2 levels "power1","power2": 1 1 1 2 2 2 1 1 1 2 ...
```

```
library(ggpubr)
```

```
## Loading required package: magrittr
##
## Attaching package: 'magrittr'
## The following object is masked from 'package:purrr':
##
##     set_names
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
ggline(table_2016s2,"machine","y",add = c("mean","jitter"),color = "station",shape = "station")
ggline(table_2016s2,"machine","y",add = c("mean","jitter"),color = "power",shape = "power")
```






```
model_2016s2 <- lm(y~machine*station*power, table_2016s2)
summary(model_2016s2)
```

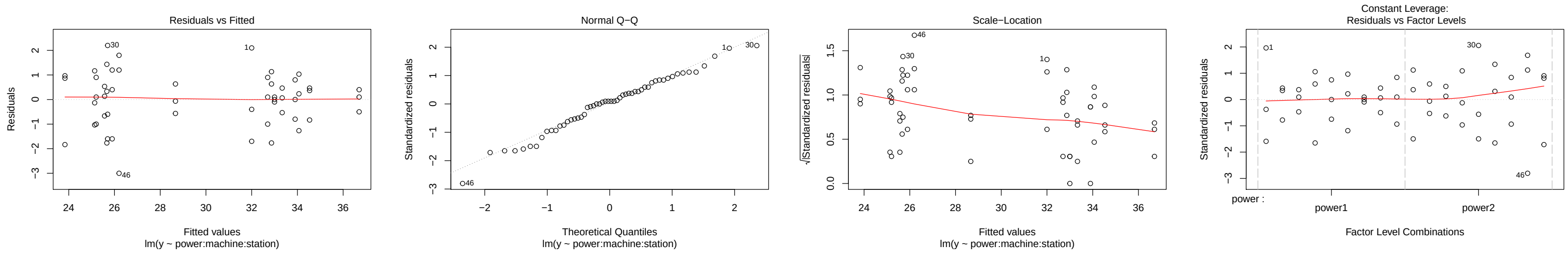
```
##
## Call:
## lm(formula = y ~ machine * station * power, data = table_2016s2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0000 -0.6500  0.1000  0.7583  2.2000
##
## Coefficients:
##                                        Estimate Std. Error t value
## (Intercept)                             32.0000     0.7562  42.316
## machinemachine2                          0.8667     1.0694   0.810
## machinemachine3                          1.0000     1.0694   0.935
## stationstation2                          2.5333     1.0694   2.369
## stationstation3                          4.7000     1.0694   4.395
## powerpower2                             -6.1000     1.0694  -5.704
## machinemachine2:stationstation2         -1.5000     1.5124  -0.992
## machinemachine3:stationstation2         -2.2000     1.5124  -1.455
## machinemachine2:stationstation3         -3.5000     1.5124  -2.314
## machinemachine3:stationstation3         -5.0000     1.5124  -3.306
## machinemachine2:powerpower2             -1.6333     1.5124  -1.080
## machinemachine3:powerpower2             -1.7000     1.5124  -1.124
## stationstation2:powerpower2              0.2333     1.5124   0.154
## stationstation3:powerpower2             -5.0333     1.5124  -3.328
## machinemachine2:stationstation2:powerpower2  -0.7000     2.1389  -0.327
## machinemachine3:stationstation2:powerpower2   0.4333     2.1389   0.203
## machinemachine2:stationstation3:powerpower2   4.3667     2.1389   2.042
## machinemachine3:stationstation3:powerpower2   3.9667     2.1389   1.855
##                                        Pr(>|t|)
## (Intercept)                             < 2e-16 ***
## machinemachine2                         0.42304
## machinemachine3                         0.35598
## stationstation2                         0.02333 *
## stationstation3                        9.38e-05 ***
## powerpower2                            1.73e-06 ***
## machinemachine2:stationstation2         0.32792
## machinemachine3:stationstation2         0.15444
## machinemachine2:stationstation3         0.02648 *
## machinemachine3:stationstation3         0.00215 **
## machinemachine2:powerpower2             0.28735
## machinemachine3:powerpower2             0.26844
## stationstation2:powerpower2             0.87825
## stationstation3:powerpower2             0.00203 **
## machinemachine2:stationstation2:powerpower2  0.74536
## machinemachine3:stationstation2:powerpower2  0.84059
## machinemachine2:stationstation3:powerpower2  0.04858 *
## machinemachine3:stationstation3:powerpower2  0.07187 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.31 on 36 degrees of freedom
## Multiple R-squared:  0.9377, Adjusted R-squared:  0.9083
## F-statistic:  31.9 on 17 and 36 DF,  p-value: < 2.2e-16
```

```r
anova(model_2016s2)
## Analysis of Variance Table
##
## Response: y
##                        Df Sum Sq Mean Sq  F value    Pr(>F)
## machine                 2  21.44   10.72   6.2475  0.004687 **
## station                 2  16.98    8.49   4.9489  0.012623 *
## power                   1 845.70  845.70 492.9587 < 2.2e-16 ***
## machine:station         4  16.60    4.15   2.4195  0.066255 .
## machine:power           2   0.38    0.19   0.1115  0.894793
## station:power           2  16.30    8.15   4.7514  0.014749 *
## machine:station:power   4  12.91    3.23   1.8806  0.135072
## Residuals              36  61.76    1.72
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
model_2016s2_1 <- lm(y~power:machine:station, table_2016s2)
summary(model_2016s2_1)
##
## Call:
## lm(formula = y ~ power:machine:station, data = table_2016s2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0000 -0.6500  0.1000  0.7583  2.2000
##
## Coefficients: (1 not defined because of singularities)
##                                          Estimate Std. Error t value
## (Intercept)                               23.8333     0.7562  31.517
## powerpower1:machinemachine1:stationstation1   8.1667     1.0694   7.636
## powerpower2:machinemachine1:stationstation1   2.0667     1.0694   1.932
## powerpower1:machinemachine2:stationstation1   9.0333     1.0694   8.447
## powerpower2:machinemachine2:stationstation1   1.3000     1.0694   1.216
## powerpower1:machinemachine3:stationstation1   9.1667     1.0694   8.571
## powerpower2:machinemachine3:stationstation1   1.3667     1.0694   1.278
## powerpower1:machinemachine1:stationstation2  10.7000     1.0694  10.005
## powerpower2:machinemachine1:stationstation2   4.8333     1.0694   4.519
## powerpower1:machinemachine2:stationstation2  10.0667     1.0694   9.413
## powerpower2:machinemachine2:stationstation2   1.8667     1.0694   1.745
## powerpower1:machinemachine3:stationstation2   9.5000     1.0694   8.883
## powerpower2:machinemachine3:stationstation2   2.3667     1.0694   2.213
## powerpower1:machinemachine1:stationstation3  12.8667     1.0694  12.031
## powerpower2:machinemachine1:stationstation3   1.7333     1.0694   1.621
## powerpower1:machinemachine2:stationstation3  10.2333     1.0694   9.569
## powerpower2:machinemachine2:stationstation3   1.8333     1.0694   1.714
## powerpower1:machinemachine3:stationstation3   8.8667     1.0694   8.291
## powerpower2:machinemachine3:stationstation3       NA         NA      NA
##                                          Pr(>|t|)
## (Intercept)                               < 2e-16 ***
## powerpower1:machinemachine1:stationstation1 4.89e-09 ***
## powerpower2:machinemachine1:stationstation1   0.0612 .
## powerpower1:machinemachine2:stationstation1 4.60e-10 ***
## powerpower2:machinemachine2:stationstation1   0.2321
## powerpower1:machinemachine3:stationstation1 3.22e-10 ***
## powerpower2:machinemachine3:stationstation1   0.2095
## powerpower1:machinemachine1:stationstation2 6.13e-12 ***
## powerpower2:machinemachine1:stationstation2 6.46e-05 ***
## powerpower1:machinemachine2:stationstation2 3.05e-11 ***
## powerpower2:machinemachine2:stationstation2   0.0894 .
## powerpower1:machinemachine3:stationstation2 1.33e-10 ***
## powerpower2:machinemachine3:stationstation2   0.0333 *
## powerpower1:machinemachine1:stationstation3 3.57e-14 ***
## powerpower2:machinemachine1:stationstation3   0.1138
## powerpower1:machinemachine2:stationstation3 1.99e-11 ***
## powerpower2:machinemachine2:stationstation3   0.0951 .
## powerpower1:machinemachine3:stationstation3 7.21e-10 ***
## powerpower2:machinemachine3:stationstation3       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.31 on 36 degrees of freedom
## Multiple R-squared:  0.9377, Adjusted R-squared:  0.9083
## F-statistic:  31.9 on 17 and 36 DF,  p-value: < 2.2e-16
```

```r
anova(model_2016s2_1)
## Analysis of Variance Table
##
## Response: y
##                       Df Sum Sq Mean Sq F value    Pr(>F)
## power:machine:station 17 930.31  54.724  31.899 < 2.2e-16 ***
## Residuals             36  61.76   1.716
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
plot(model_2016s2_1)
```




```r
model_2016s2_2 <- lm(y~machine+power:machine+power:station:machine, table_2016s2)
summary(model_2016s2_2)
```

```
##
## Call:
## lm(formula = y ~ machine + power:machine + power:station:machine,
##     data = table_2016s2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0000 -0.6500  0.1000  0.7583  2.2000
##
## Coefficients:
##                                         Estimate Std. Error t value
## (Intercept)                              32.0000     0.7562  42.316
## machinemachine2                           0.8667     1.0694   0.810
## machinemachine3                           1.0000     1.0694   0.935
## machinemachine1:powerpower2              -6.1000     1.0694  -5.704
## machinemachine2:powerpower2              -7.7333     1.0694  -7.231
## machinemachine3:powerpower2              -7.8000     1.0694  -7.294
## machinemachine1:powerpower1:stationstation2   2.5333     1.0694   2.369
## machinemachine2:powerpower1:stationstation2   1.0333     1.0694   0.966
## machinemachine3:powerpower1:stationstation2   0.3333     1.0694   0.312
## machinemachine1:powerpower2:stationstation2   2.7667     1.0694   2.587
## machinemachine2:powerpower2:stationstation2   0.5667     1.0694   0.530
## machinemachine3:powerpower2:stationstation2   1.0000     1.0694   0.935
## machinemachine1:powerpower1:stationstation3   4.7000     1.0694   4.395
## machinemachine2:powerpower1:stationstation3   1.2000     1.0694   1.122
## machinemachine3:powerpower1:stationstation3  -0.3000     1.0694  -0.281
## machinemachine1:powerpower2:stationstation3  -0.3333     1.0694  -0.312
## machinemachine2:powerpower2:stationstation3   0.5333     1.0694   0.499
## machinemachine3:powerpower2:stationstation3  -1.3667     1.0694  -1.278
##                                         Pr(>|t|)
## (Intercept)                              < 2e-16 ***
## machinemachine2                           0.4230
## machinemachine3                           0.3560
## machinemachine1:powerpower2              1.73e-06 ***
## machinemachine2:powerpower2              1.64e-08 ***
## machinemachine3:powerpower2              1.36e-08 ***
## machinemachine1:powerpower1:stationstation2   0.0233 *
## machinemachine2:powerpower1:stationstation2   0.3404
## machinemachine3:powerpower1:stationstation2   0.7571
## machinemachine1:powerpower2:stationstation2   0.0139 *
## machinemachine2:powerpower2:stationstation2   0.5995
## machinemachine3:powerpower2:stationstation2   0.3560
## machinemachine1:powerpower1:stationstation3 9.38e-05 ***
## machinemachine2:powerpower1:stationstation3   0.2693
## machinemachine3:powerpower1:stationstation3   0.7807
## machinemachine1:powerpower2:stationstation3   0.7571
## machinemachine2:powerpower2:stationstation3   0.6210
## machinemachine3:powerpower2:stationstation3   0.2095
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.31 on 36 degrees of freedom
## Multiple R-squared:  0.9377, Adjusted R-squared:  0.9083
## F-statistic:  31.9 on 17 and 36 DF,  p-value: < 2.2e-16
```

```r
anova(model_2016s2_2)
```

```
## Analysis of Variance Table
##
## Response: y
##                       Df  Sum Sq Mean Sq  F value    Pr(>F)
## machine                2   21.44  10.718   6.2475  0.004687 **
## machine:power          3  846.08 282.027 164.3939 < 2.2e-16 ***
## machine:power:station 12   62.79   5.233   3.0501  0.004742 **
## Residuals             36   61.76   1.716
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2016F

Jong Sung Kim*, Brad Crain

2016F1

A national insurance organization wanted to study the consumption pattern of cigarettes in all 50 states and the District of Columbia. Data were collected for 1960, 1970, and 1980, but we will focus here on 1970. Using data from 1970, the organization wanted to construct a regression equation that relates statewide cigarette consumption (on a per capita basis) to various socioeconomic and demographic variables, and to determine whether these variables were useful in predicting the consumption of cigarettes. The variables chosen for study are given below. Age, x1: Median age of a person living in the state

Education, x2: Percentage of people over 25 years of age in a state that had completed high school

Income, x3: Per capita personal income for a state (in dollars)

Perblack, x4: Percentage of blacks living in a state

Perfem, x5: Percentage of females living in a state

Price, x6: Average price of a pack of cigarettes in a state (in cents)

Scig, y: Number of packs of cigarettes sold in a state on a per capita basis.

The data on these variables are stored in 8 columns in the same order as listed above; a two-letter alphabetic code is given first, however.

The data are saved as "cigcons.xlsx"

Perform a complete regression analysis on these data; including checking of model assumptions and attempting appropriate remedies, if needed. The main objective of the analysis is to find the smallest number of variables that describes the state sale of cigarettes meaningfully and adequately. You might want to consider among others partial regression plot, interaction terms, outliers and influential cases analysis, Box-Cox transformation, and explanation of your final model.

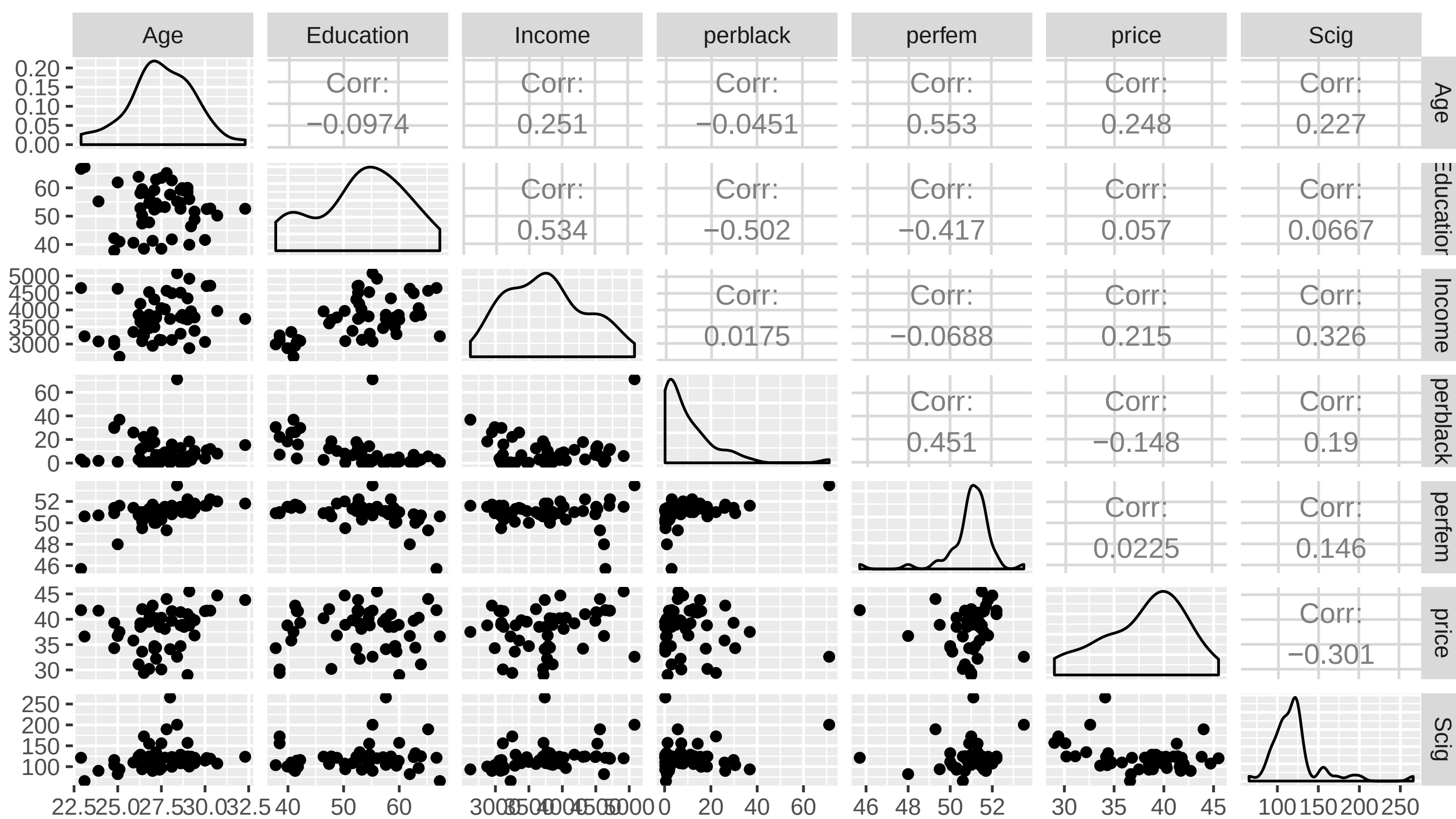
```
table_2016f1 <- readxl::read_xlsx("qe_lab/cigcons.xlsx")
table_2016f1$State <- as.factor(table_2016f1$State)
str(table_2016f1)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    51 obs. of  8 variables:
##  $ State    : Factor w/ 51 levels "AK","AL","AR",..: 2 1 4 3 5 6 7 9 8 10 ...
##  $ Age      : num  27 22.9 26.3 29.1 28.1 26.2 29.1 26.8 28.4 32.3 ...
##  $ Education: num  41.3 66.7 58.1 39.9 62.6 63.9 56 54.6 55.2 52.6 ...
##  $ Income   : num  2948 4644 3655 2878 4493 ...
##  $ perblack : num  26.2 3 3 18.3 7 3 6 14.3 71.1 15.3 ...
##  $ perfem   : num  51.7 45.7 50.8 51.5 50.8 50.7 51.5 51.3 53.5 51.8 ...
##  $ price    : num  42.7 41.8 38.5 38.8 39.7 31.1 45.5 41.3 32.6 43.8 ...
##  $ Scig     : num  89.8 121.3 115.2 100.3 123 ...
```

```
library(GGally)
```

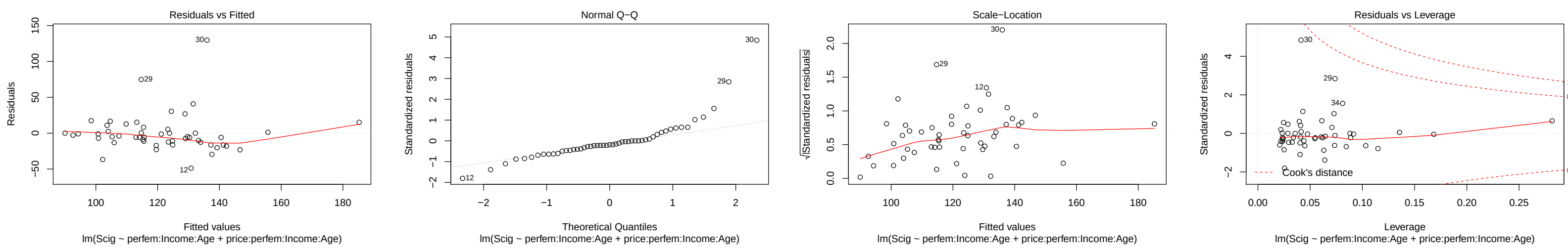
```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
##
## Attaching package: 'GGally'
## The following object is masked from 'package:dplyr':
##
##     nasa
```

```
ggpairs(table_2016f1[,-1])
```

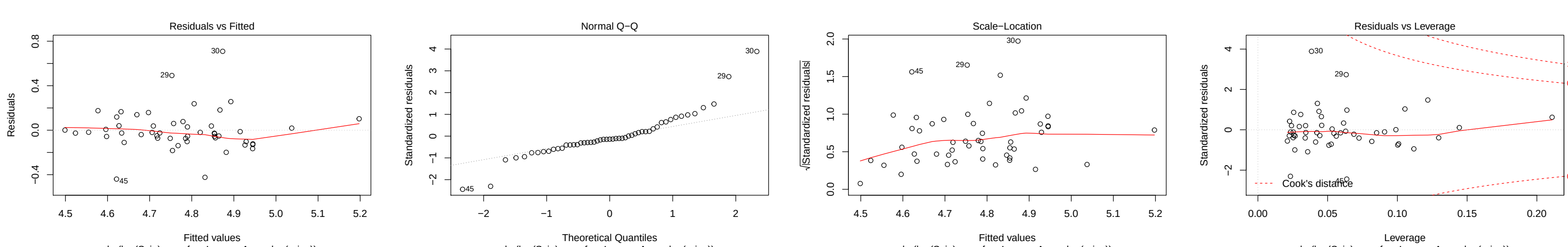


```r
model_2016f1 <- lm(Scig~price*perfem*perblack*Income*Education*Age, table_2016f1)
ols_step_both_aic(model_2016f1)
ols_step_both_p(model_2016f1)
```

```
##       perfem:Income:Age perfem:Income:Age:price
##                5.380033                5.380033
##
## Call:
## lm(formula = Scig ~ perfem:Income:Age + price:perfem:Income:Age,
##     data = table_2016f1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -48.743 -12.457  -4.995   3.835 129.705
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              4.067e+01  2.127e+01   1.912  0.06187 .
## perfem:Income:Age        3.847e-05  8.784e-06   4.379 6.43e-05 ***
## perfem:Income:Age:price -6.053e-07  1.770e-07  -3.420  0.00129 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.36 on 48 degrees of freedom
## Multiple R-squared:  0.3013, Adjusted R-squared:  0.2721
## F-statistic: 10.35 on 2 and 48 DF,  p-value: 0.0001835
## Analysis of Variance Table
##
## Response: Scig
##                         Df Sum Sq Mean Sq F value   Pr(>F)
## perfem:Income:Age        1   6735  6734.6  8.9961 0.004279 **
## perfem:Income:Age:price  1   8758  8757.6 11.6985 0.001286 **
## Residuals               48  35933   748.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```




```
##         log(price) perfem:Income:Age
##           1.066268          1.066268
##
## Call:
## lm(formula = log(Scig) ~ perfem:Income:Age + log(price), data = table_2016f1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43922 -0.07364 -0.02540  0.05006  0.70893
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.407e+00  8.503e-01   8.711 1.89e-11 ***
## log(price)        -8.993e-01  2.405e-01  -3.739 0.000493 ***
## perfem:Income:Age  1.197e-07  2.657e-08   4.506 4.24e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1859 on 48 degrees of freedom
## Multiple R-squared:  0.3651, Adjusted R-squared:  0.3386
## F-statistic:  13.8 on 2 and 48 DF,  p-value: 1.843e-05
## Analysis of Variance Table
##
## Response: log(Scig)
##                   Df  Sum Sq Mean Sq F value    Pr(>F)
## log(price)         1 0.25199 0.25199  7.2931  0.009534 **
## perfem:Income:Age  1 0.70156 0.70156 20.3043 4.236e-05 ***
## Residuals         48 1.65850 0.03455
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



2016F2

An experiment is conducted to compare the water quality of three creeks in an area. Five water samples are selected from each creek. Each sample is divided into two parts, and the dissolved oxygen content is measured for each part. (Higher dissolved oxygen contents indicate higher water quality.) The results are given as follows:

Creek/Water Sample	1	2	3	4	5
1	5.2, 5.4	5.6, 5.7	5.4, 5.4	5.6, 5.5	5.8, 5.5
2	5.1, 5.3	5.1, 5.0	5.3, 5.2	5.0, 5.0	4.9, 5.1
3	5.9, 5.8	5.8, 5.8	5.7, 5.8	5.8, 5.9	5.9, 5.9

a. Write down an appropriate model with assumptions (including normality).

Two-stage nested design

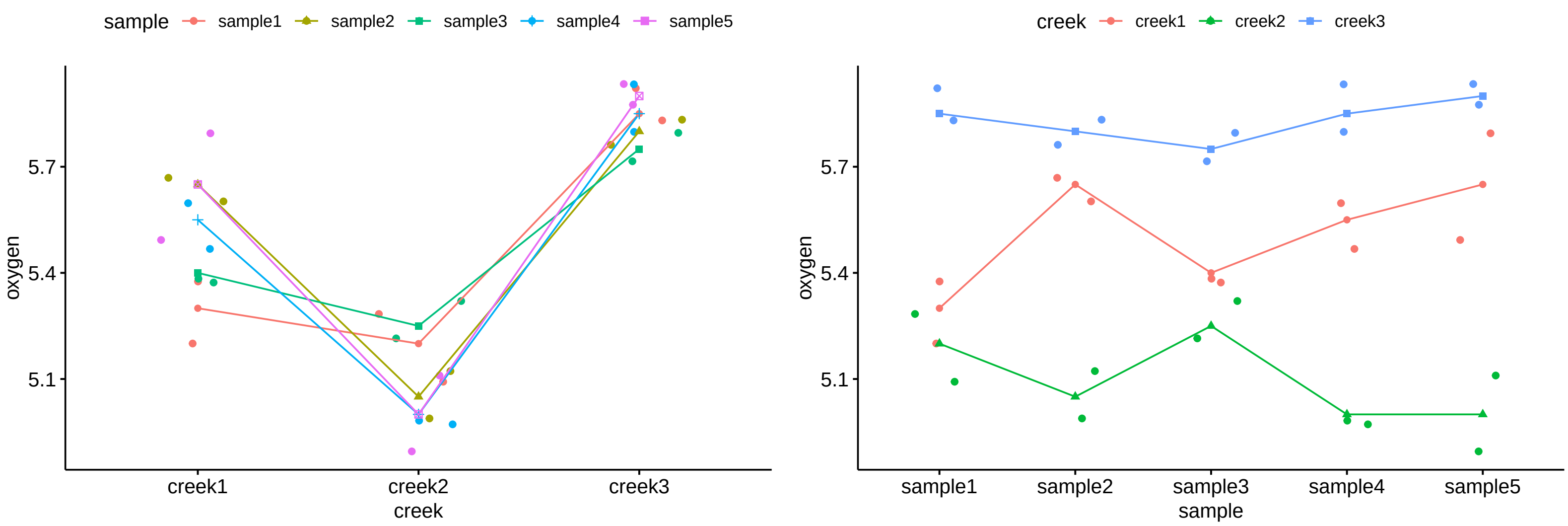
$y = \mu + \tau_i + \beta_{j(i)} + \varepsilon_{k(ij)}, i = 1,2,3; j = 1,2,3,4,5; k = 1,2$

b. Find the ANOVA table for the data.

c. Perform the F-test comparing the creeks using a .05 level.

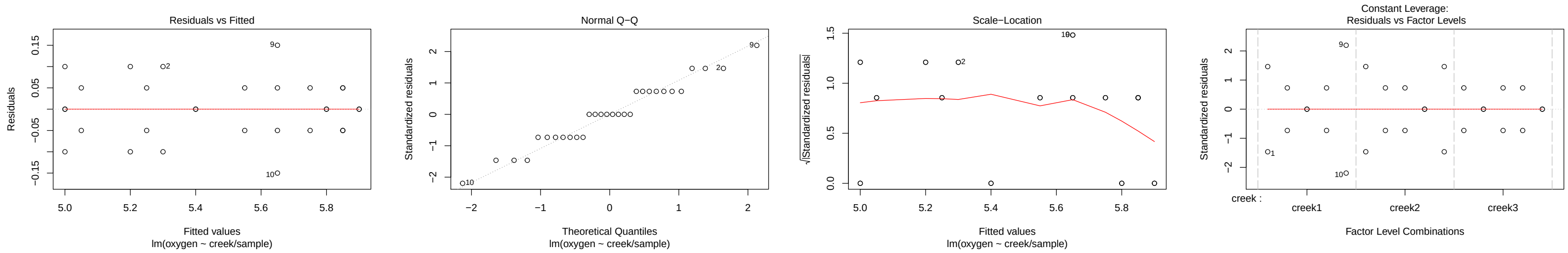
d. Perform a Tukey multiple comparison on the creeks using a .05 level.

```
## 'data.frame':    30 obs. of  4 variables:
##  $ creek : Factor w/ 3 levels "creek1","creek2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ oxygen: num  5.2 5.4 5.6 5.7 5.4 5.4 5.6 5.5 5.8 5.5 ...
##  $ sample: Factor w/ 5 levels "sample1","sample2",..: 1 1 2 2 3 3 4 4 5 5 ...
##  $ rep   : Factor w/ 2 levels "rep1","rep2": 1 2 1 2 1 2 1 2 1 2 ...
```

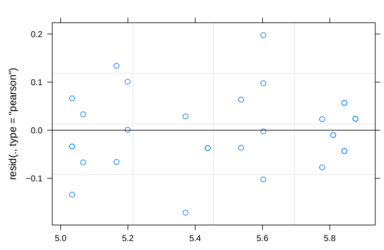



```
##
## Call:
## lm(formula = oxygen ~ creek/sample, data = table_2016f2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -0.15  -0.05   0.00   0.05   0.15
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                5.300e+00  6.831e-02  77.584  < 2e-16 ***
## creekcreek2               -1.000e-01  9.661e-02  -1.035  0.31702
## creekcreek3                5.500e-01  9.661e-02   5.693 4.26e-05 ***
## creekcreek1:samplesample2  3.500e-01  9.661e-02   3.623  0.00251 **
## creekcreek2:samplesample2 -1.500e-01  9.661e-02  -1.553  0.14135
## creekcreek3:samplesample2 -5.000e-02  9.661e-02  -0.518  0.61232
## creekcreek1:samplesample3  1.000e-01  9.661e-02   1.035  0.31702
## creekcreek2:samplesample3  5.000e-02  9.661e-02   0.518  0.61232
## creekcreek3:samplesample3 -1.000e-01  9.661e-02  -1.035  0.31702
## creekcreek1:samplesample4  2.500e-01  9.661e-02   2.588  0.02060 *
## creekcreek2:samplesample4 -2.000e-01  9.661e-02  -2.070  0.05611 .
## creekcreek3:samplesample4 -2.764e-17  9.661e-02   0.000  1.00000
## creekcreek1:samplesample5  3.500e-01  9.661e-02   3.623  0.00251 **
## creekcreek2:samplesample5 -2.000e-01  9.661e-02  -2.070  0.05611 .
## creekcreek3:samplesample5  5.000e-02  9.661e-02   0.518  0.61232
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09661 on 15 degrees of freedom
## Multiple R-squared:  0.9555, Adjusted R-squared:  0.914
## F-statistic: 23.02 on 14 and 15 DF,  p-value: 1.305e-07
## Analysis of Variance Table
##
## Response: oxygen
##              Df Sum Sq Mean Sq  F value    Pr(>F)
## creek         2  2.678 1.33900 143.4643 1.665e-10 ***
## creek:sample 12  0.330 0.02750   2.9464   0.02559 *
## Residuals    15  0.140 0.00933
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##              GVIF Df GVIF^(1/(2*Df))
## creek          25  2        2.236068
## creek:sample   25 12        1.143530
```

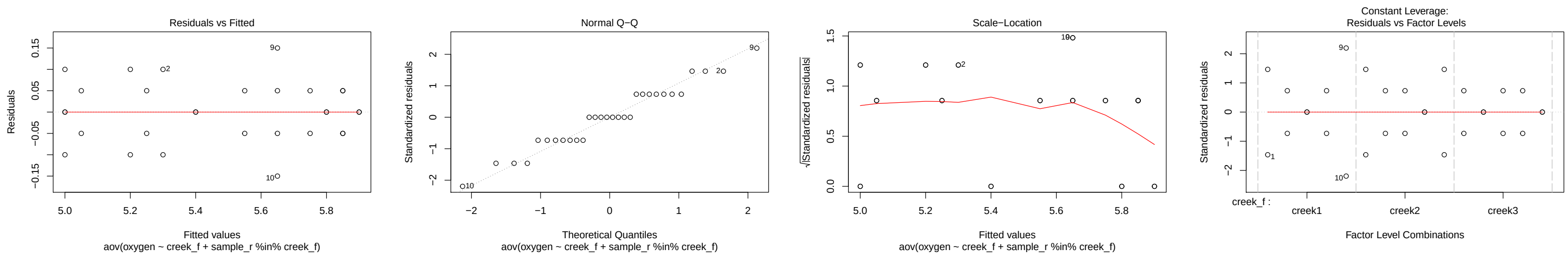



```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
## Registered S3 methods overwritten by 'lme4':
##   method                          from
##   cooks.distance.influence.merMod car
##   influence.merMod                car
##   dfbeta.influence.merMod         car
##   dfbetas.influence.merMod        car
## Linear mixed model fit by REML ['lmerMod']
## Formula: oxygen ~ creek + (1 | creek:sample)
##    Data: table_2016f2
##
## REML criterion at convergence: -29.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.77284 -0.43208 -0.06556  0.52633  2.04448
##
## Random effects:
##  Groups       Name        Variance Std.Dev.
##  creek:sample (Intercept) 0.009083 0.09531
##  Residual                 0.009333 0.09661
## Number of obs: 30, groups:  creek:sample, 15
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  5.51000    0.05244 105.071
## creekcreek2 -0.41000    0.07416  -5.528
## creekcreek3  0.32000    0.07416   4.315
##
## Correlation of Fixed Effects:
##             (Intr) crkcr2
## creekcreek2 -0.707
## creekcreek3 -0.707  0.500
## Analysis of Variance Table
##       Df  Sum Sq Mean Sq F value
## creek  2 0.90889 0.45444  48.691
## [1] 1.743538e-06
## Computing profile confidence intervals ...
##                   2.5 %     97.5 %
## .sig01       0.00000000  0.1425879
## .sigma       0.07016639  0.1450729
## (Intercept)  5.41185731  5.6081427
## creekcreek2 -0.54879472 -0.2712053
## creekcreek3  0.18120528  0.4587947
```




```
## Loading required package: matrixStats
##
## Attaching package: 'matrixStats'
## The following object is masked from 'package:dplyr':
##
##     count
## Loading required package: R.methodsS3
## R.methodsS3 v1.7.1 (2016-02-15) successfully loaded. See ?R.methodsS3 for help.
## Analysis of Variance Table
##
## Response: oxygen
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## creek_f            2  2.678 1.33900 48.6909 1.743e-06 ***
## creek_f:sample_r  12  0.330 0.02750  2.9464   0.02559 *
```

```
## Residual          15  0.140 0.00933
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```




```
library(emmeans)
cre_sam <- pairs(lsmeans(model_2016f2_1,~creek|sample))
sam_cre <- pairs(lsmeans(model_2016f2_1,~sample|creek))
library(kableExtra)
kable(test(rbind(cre_sam,sam_cre),adjust="tukey"),format="latex")%>%kable_styling("condensed",full_width=F,font_size
```

creek	contrast	estimate	SE	df	t.ratio	p.value
.	sample1,creek1 - sample2,creek1	-0.35	0.0966092	15	-3.6228442	0.1101055
.	**sample1,creek1 - sample3,creek1**	**-0.10**	**0.0966092**	**15**	**-1.0350983**	**0.9991798**
.	sample1,creek1 - sample4,creek1	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample1,creek1 - sample5,creek1	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample1,creek1 - sample1,creek2	0.10	0.0966092	15	1.0350983	0.9991798
.	sample1,creek1 - sample2,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample1,creek1 - sample3,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample1,creek1 - sample4,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample1,creek1 - sample5,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample1,creek1 - sample1,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek1 - sample2,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample1,creek1 - sample3,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample1,creek1 - sample4,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek1 - sample5,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample2,creek1 - sample3,creek1	0.25	0.0966092	15	2.5877458	0.4869333
.	sample2,creek1 - sample4,creek1	0.10	0.0966092	15	1.0350983	0.9991798
.	sample2,creek1 - sample5,creek1	0.00	0.0966092	15	0.0000000	1.0000000
.	sample2,creek1 - sample1,creek2	0.45	0.0966092	15	4.6579425	0.0183213
.	sample2,creek1 - sample2,creek2	0.60	0.0966092	15	6.2105900	0.0012178
.	sample2,creek1 - sample3,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample2,creek1 - sample4,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample2,creek1 - sample5,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample2,creek1 - sample1,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek1 - sample2,creek3	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample2,creek1 - sample3,creek3	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample2,creek1 - sample4,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek1 - sample5,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample3,creek1 - sample4,creek1	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample3,creek1 - sample5,creek1	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample3,creek1 - sample1,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample3,creek1 - sample2,creek2	0.35	0.0966092	15	3.6228442	0.1101055
.	sample3,creek1 - sample3,creek2	0.15	0.0966092	15	1.5526475	0.9626296
.	sample3,creek1 - sample4,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample3,creek1 - sample5,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample3,creek1 - sample1,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample3,creek1 - sample2,creek3	-0.40	0.0966092	15	-4.1403934	0.0456168
.	sample3,creek1 - sample3,creek3	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample3,creek1 - sample4,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample3,creek1 - sample5,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample4,creek1 - sample5,creek1	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample4,creek1 - sample1,creek2	0.35	0.0966092	15	3.6228442	0.1101055
.	sample4,creek1 - sample2,creek2	0.50	0.0966092	15	5.1754917	0.0073200
.	sample4,creek1 - sample3,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample4,creek1 - sample4,creek2	0.55	0.0966092	15	5.6930409	0.0029559
.	sample4,creek1 - sample5,creek2	0.55	0.0966092	15	5.6930409	0.0029559
.	sample4,creek1 - sample1,creek3	-0.30	0.0966092	15	-3.1052950	0.2473710
.	sample4,creek1 - sample2,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample4,creek1 - sample3,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample4,creek1 - sample4,creek3	-0.30	0.0966092	15	-3.1052950	0.2473710
.	sample4,creek1 - sample5,creek3	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample5,creek1 - sample1,creek2	0.45	0.0966092	15	4.6579425	0.0183213
.	sample5,creek1 - sample2,creek2	0.60	0.0966092	15	6.2105900	0.0012178
.	sample5,creek1 - sample3,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample5,creek1 - sample4,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample5,creek1 - sample5,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample5,creek1 - sample1,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample5,creek1 - sample2,creek3	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample5,creek1 - sample3,creek3	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample5,creek1 - sample4,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample5,creek1 - sample5,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample1,creek2 - sample2,creek2	0.15	0.0966092	15	1.5526475	0.9626296
.	sample1,creek2 - sample3,creek2	-0.05	0.0966092	15	-0.5175492	0.9999999
.	sample1,creek2 - sample4,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample1,creek2 - sample5,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample1,creek2 - sample1,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample1,creek2 - sample2,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample1,creek2 - sample3,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek2 - sample4,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample1,creek2 - sample5,creek3	-0.70	0.0966092	15	-7.2456884	0.0002237
.	sample2,creek2 - sample3,creek2	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek2 - sample4,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample2,creek2 - sample5,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample2,creek2 - sample1,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample2,creek2 - sample2,creek3	-0.75	0.0966092	15	-7.7632375	0.0001001
.	sample2,creek2 - sample3,creek3	-0.70	0.0966092	15	-7.2456884	0.0002237
.	sample2,creek2 - sample4,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample2,creek2 - sample5,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample3,creek2 - sample4,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample3,creek2 - sample5,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample3,creek2 - sample1,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample3,creek2 - sample2,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample3,creek2 - sample3,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample3,creek2 - sample4,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample3,creek2 - sample5,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample4,creek2 - sample5,creek2	0.00	0.0966092	15	0.0000000	1.0000000
.	sample4,creek2 - sample1,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample4,creek2 - sample2,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample4,creek2 - sample3,creek3	-0.75	0.0966092	15	-7.7632375	0.0001001
.	sample4,creek2 - sample4,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample4,creek2 - sample5,creek3	-0.90	0.0966092	15	-9.3158851	0.0000107
.	sample5,creek2 - sample1,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample5,creek2 - sample2,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462

```r
TukeyHSD(model_2016f2_3,conf.level=0.95)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = oxygen ~ creek_f + sample_r %in% creek_f, data = table_2016f2)
##
## $creek_f
##                diff        lwr        upr   p adj
## creek2-creek1 -0.41 -0.5222235 -0.2977765 3.0e-07
## creek3-creek1  0.32  0.2077765  0.4322235 6.2e-06
## creek3-creek2  0.73  0.6177765  0.8422235 0.0e+00
##
## $`creek_f:sample_r`
##                                diff        lwr        upr     p adj
## creek2:sample1-creek1:sample1 -0.10 -0.4859204  0.2859204 0.9982090
## creek3:sample1-creek1:sample1  0.55  0.1640796  0.9359204 0.0024451
## creek1:sample2-creek1:sample1  0.35 -0.0359204  0.7359204 0.0948215
## creek2:sample2-creek1:sample1 -0.25 -0.6359204  0.1359204 0.4419456
## creek3:sample2-creek1:sample1  0.50  0.1140796  0.8859204 0.0060915
## creek1:sample3-creek1:sample1  0.10 -0.2859204  0.4859204 0.9982090
## creek2:sample3-creek1:sample1 -0.05 -0.4359204  0.3359204 0.9999994
## creek3:sample3-creek1:sample1  0.45  0.0640796  0.8359204 0.0153692
## creek1:sample4-creek1:sample1  0.25 -0.1359204  0.6359204 0.4419456
## creek2:sample4-creek1:sample1 -0.30 -0.6859204  0.0859204 0.2177180
## creek3:sample4-creek1:sample1  0.55  0.1640796  0.9359204 0.0024451
## creek1:sample5-creek1:sample1  0.35 -0.0359204  0.7359204 0.0948215
## creek2:sample5-creek1:sample1 -0.30 -0.6859204  0.0859204 0.2177180
## creek3:sample5-creek1:sample1  0.60  0.2140796  0.9859204 0.0010028
## creek3:sample1-creek2:sample1  0.65  0.2640796  1.0359204 0.0004223
## creek1:sample2-creek2:sample1  0.45  0.0640796  0.8359204 0.0153692
## creek2:sample2-creek2:sample1 -0.15 -0.5359204  0.2359204 0.9460679
## creek3:sample2-creek2:sample1  0.60  0.2140796  0.9859204 0.0010028
## creek1:sample3-creek2:sample1  0.20 -0.1859204  0.5859204 0.7351714
## creek2:sample3-creek2:sample1  0.05 -0.3359204  0.4359204 0.9999994
## creek3:sample3-creek2:sample1  0.55  0.1640796  0.9359204 0.0024451
## creek1:sample4-creek2:sample1  0.35 -0.0359204  0.7359204 0.0948215
## creek2:sample4-creek2:sample1 -0.20 -0.5859204  0.1859204 0.7351714
## creek3:sample4-creek2:sample1  0.65  0.2640796  1.0359204 0.0004223
## creek1:sample5-creek2:sample1  0.45  0.0640796  0.8359204 0.0153692
## creek2:sample5-creek2:sample1 -0.20 -0.5859204  0.1859204 0.7351714
## creek3:sample5-creek2:sample1  0.70  0.3140796  1.0859204 0.0001830
## creek1:sample2-creek3:sample1 -0.20 -0.5859204  0.1859204 0.7351714
## creek2:sample2-creek3:sample1 -0.80 -1.1859204 -0.4140796 0.0000376
## creek3:sample2-creek3:sample1 -0.05 -0.4359204  0.3359204 0.9999994
## creek1:sample3-creek3:sample1 -0.45 -0.8359204 -0.0640796 0.0153692
## creek2:sample3-creek3:sample1 -0.60 -0.9859204 -0.2140796 0.0010028
## creek3:sample3-creek3:sample1 -0.10 -0.4859204  0.2859204 0.9982090
## creek1:sample4-creek3:sample1 -0.30 -0.6859204  0.0859204 0.2177180
## creek2:sample4-creek3:sample1 -0.85 -1.2359204 -0.4640796 0.0000178
## creek3:sample4-creek3:sample1  0.00 -0.3859204  0.3859204 1.0000000
## creek1:sample5-creek3:sample1 -0.20 -0.5859204  0.1859204 0.7351714
## creek2:sample5-creek3:sample1 -0.85 -1.2359204 -0.4640796 0.0000178
## creek3:sample5-creek3:sample1  0.05 -0.3359204  0.4359204 0.9999994
## creek2:sample2-creek1:sample2 -0.60 -0.9859204 -0.2140796 0.0010028
## creek3:sample2-creek1:sample2  0.15 -0.2359204  0.5359204 0.9460679
## creek1:sample3-creek1:sample2 -0.25 -0.6359204  0.1359204 0.4419456
## creek2:sample3-creek1:sample2 -0.40 -0.7859204 -0.0140796 0.0386879
## creek3:sample3-creek1:sample2  0.10 -0.2859204  0.4859204 0.9982090
## creek1:sample4-creek1:sample2 -0.10 -0.4859204  0.2859204 0.9982090
## creek2:sample4-creek1:sample2 -0.65 -1.0359204 -0.2640796 0.0004223
## creek3:sample4-creek1:sample2  0.20 -0.1859204  0.5859204 0.7351714
## creek1:sample5-creek1:sample2  0.00 -0.3859204  0.3859204 1.0000000
## creek2:sample5-creek1:sample2 -0.65 -1.0359204 -0.2640796 0.0004223
## creek3:sample5-creek1:sample2  0.25 -0.1359204  0.6359204 0.4419456
## creek3:sample2-creek2:sample2  0.75  0.3640796  1.1359204 0.0000817
## creek1:sample3-creek2:sample2  0.35 -0.0359204  0.7359204 0.0948215
## creek2:sample3-creek2:sample2  0.20 -0.1859204  0.5859204 0.7351714
## creek3:sample3-creek2:sample2  0.70  0.3140796  1.0859204 0.0001830
## creek1:sample4-creek2:sample2  0.50  0.1140796  0.8859204 0.0060915
## creek2:sample4-creek2:sample2 -0.05 -0.4359204  0.3359204 0.9999994
## creek3:sample4-creek2:sample2  0.80  0.4140796  1.1859204 0.0000376
## creek1:sample5-creek2:sample2  0.60  0.2140796  0.9859204 0.0010028
## creek2:sample5-creek2:sample2 -0.05 -0.4359204  0.3359204 0.9999994
## creek3:sample5-creek2:sample2  0.85  0.4640796  1.2359204 0.0000178
## creek1:sample3-creek3:sample2 -0.40 -0.7859204 -0.0140796 0.0386879
## creek2:sample3-creek3:sample2 -0.55 -0.9359204 -0.1640796 0.0024451
## creek3:sample3-creek3:sample2 -0.05 -0.4359204  0.3359204 0.9999994
## creek1:sample4-creek3:sample2 -0.25 -0.6359204  0.1359204 0.4419456
## creek2:sample4-creek3:sample2 -0.80 -1.1859204 -0.4140796 0.0000376
## creek3:sample4-creek3:sample2  0.05 -0.3359204  0.4359204 0.9999994
## creek1:sample5-creek3:sample2 -0.15 -0.5359204  0.2359204 0.9460679
## creek2:sample5-creek3:sample2 -0.80 -1.1859204 -0.4140796 0.0000376
## creek3:sample5-creek3:sample2  0.10 -0.2859204  0.4859204 0.9982090
```

```
## creek2:sample3-creek1:sample3 -0.15 -0.5359204  0.2359204 0.9460679
## creek3:sample3-creek1:sample3  0.35 -0.0359204  0.7359204 0.0948215
## creek1:sample4-creek1:sample3  0.15 -0.2359204  0.5359204 0.9460679
## creek2:sample4-creek1:sample3 -0.40 -0.7859204 -0.0140796 0.0386879
## creek3:sample4-creek1:sample3  0.45  0.0640796  0.8359204 0.0153692
## creek1:sample5-creek1:sample3  0.25 -0.1359204  0.6359204 0.4419456
## creek2:sample5-creek1:sample3 -0.40 -0.7859204 -0.0140796 0.0386879
## creek3:sample5-creek1:sample3  0.50  0.1140796  0.8859204 0.0060915
## creek3:sample3-creek2:sample3  0.50  0.1140796  0.8859204 0.0060915
## creek1:sample4-creek2:sample3  0.30 -0.0859204  0.6859204 0.2177180
## creek2:sample4-creek2:sample3 -0.25 -0.6359204  0.1359204 0.4419456
## creek3:sample4-creek2:sample3  0.60  0.2140796  0.9859204 0.0010028
## creek1:sample5-creek2:sample3  0.40  0.0140796  0.7859204 0.0386879
## creek2:sample5-creek2:sample3 -0.25 -0.6359204  0.1359204 0.4419456
## creek3:sample5-creek2:sample3  0.65  0.2640796  1.0359204 0.0004223
## creek1:sample4-creek3:sample3 -0.20 -0.5859204  0.1859204 0.7351714
## creek2:sample4-creek3:sample3 -0.75 -1.1359204 -0.3640796 0.0000817
## creek3:sample4-creek3:sample3  0.10 -0.2859204  0.4859204 0.9982090
## creek1:sample5-creek3:sample3 -0.10 -0.4859204  0.2859204 0.9982090
## creek2:sample5-creek3:sample3 -0.75 -1.1359204 -0.3640796 0.0000817
## creek3:sample5-creek3:sample3  0.15 -0.2359204  0.5359204 0.9460679
## creek2:sample4-creek1:sample4 -0.55 -0.9359204 -0.1640796 0.0024451
## creek3:sample4-creek1:sample4  0.30 -0.0859204  0.6859204 0.2177180
## creek1:sample5-creek1:sample4  0.10 -0.2859204  0.4859204 0.9982090
## creek2:sample5-creek1:sample4 -0.55 -0.9359204 -0.1640796 0.0024451
## creek3:sample5-creek1:sample4  0.35 -0.0359204  0.7359204 0.0948215
## creek3:sample4-creek2:sample4  0.85  0.4640796  1.2359204 0.0000178
## creek1:sample5-creek2:sample4  0.65  0.2640796  1.0359204 0.0004223
## creek2:sample5-creek2:sample4  0.00 -0.3859204  0.3859204 1.0000000
## creek3:sample5-creek2:sample4  0.90  0.5140796  1.2859204 0.0000087
## creek1:sample5-creek3:sample4 -0.20 -0.5859204  0.1859204 0.7351714
## creek2:sample5-creek3:sample4 -0.85 -1.2359204 -0.4640796 0.0000178
## creek3:sample5-creek3:sample4  0.05 -0.3359204  0.4359204 0.9999994
## creek2:sample5-creek1:sample5 -0.65 -1.0359204 -0.2640796 0.0004223
## creek3:sample5-creek1:sample5  0.25 -0.1359204  0.6359204 0.4419456
## creek3:sample5-creek2:sample5  0.90  0.5140796  1.2859204 0.0000087
```

```
cre_sam <- pairs(lsmeans(model_2016f2_3,~creek_f|sample_r))
sam_cre <- pairs(lsmeans(model_2016f2_3,~sample_r|creek_f))
kable(test(rbind(cre_sam,sam_cre),adjust="tukey"),format="latex")%>%kable_styling("condensed",full_width=F,font_size
```

creek_f	contrast	estimate	SE	df	t.ratio	p.value
.	sample1,creek1 - sample2,creek1	-0.35	0.0966092	15	-3.6228442	0.1101055
.	**sample1,creek1 - sample3,creek1**	**-0.10**	**0.0966092**	**15**	**-1.0350983**	**0.9991798**
.	sample1,creek1 - sample4,creek1	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample1,creek1 - sample5,creek1	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample1,creek1 - sample1,creek2	0.10	0.0966092	15	1.0350983	0.9991798
.	sample1,creek1 - sample2,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample1,creek1 - sample3,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample1,creek1 - sample4,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample1,creek1 - sample5,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample1,creek1 - sample1,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek1 - sample2,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample1,creek1 - sample3,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample1,creek1 - sample4,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek1 - sample5,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample2,creek1 - sample3,creek1	0.25	0.0966092	15	2.5877458	0.4869333
.	sample2,creek1 - sample4,creek1	0.10	0.0966092	15	1.0350983	0.9991798
.	sample2,creek1 - sample5,creek1	0.00	0.0966092	15	0.0000000	1.0000000
.	sample2,creek1 - sample1,creek2	0.45	0.0966092	15	4.6579425	0.0183213
.	sample2,creek1 - sample2,creek2	0.60	0.0966092	15	6.2105900	0.0012178
.	sample2,creek1 - sample3,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample2,creek1 - sample4,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample2,creek1 - sample5,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample2,creek1 - sample1,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek1 - sample2,creek3	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample2,creek1 - sample3,creek3	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample2,creek1 - sample4,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek1 - sample5,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample3,creek1 - sample4,creek1	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample3,creek1 - sample5,creek1	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample3,creek1 - sample1,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample3,creek1 - sample2,creek2	0.35	0.0966092	15	3.6228442	0.1101055
.	sample3,creek1 - sample3,creek2	0.15	0.0966092	15	1.5526475	0.9626296
.	sample3,creek1 - sample4,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample3,creek1 - sample5,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample3,creek1 - sample1,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample3,creek1 - sample2,creek3	-0.40	0.0966092	15	-4.1403934	0.0456168
.	sample3,creek1 - sample3,creek3	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample3,creek1 - sample4,creek3	-0.45	0.0966092	15	-4.6579425	0.0183213
.	sample3,creek1 - sample5,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample4,creek1 - sample5,creek1	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample4,creek1 - sample1,creek2	0.35	0.0966092	15	3.6228442	0.1101055
.	sample4,creek1 - sample2,creek2	0.50	0.0966092	15	5.1754917	0.0073200
.	sample4,creek1 - sample3,creek2	0.30	0.0966092	15	3.1052950	0.2473710
.	sample4,creek1 - sample4,creek2	0.55	0.0966092	15	5.6930409	0.0029559
.	sample4,creek1 - sample5,creek2	0.55	0.0966092	15	5.6930409	0.0029559
.	sample4,creek1 - sample1,creek3	-0.30	0.0966092	15	-3.1052950	0.2473710
.	sample4,creek1 - sample2,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample4,creek1 - sample3,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample4,creek1 - sample4,creek3	-0.30	0.0966092	15	-3.1052950	0.2473710
.	sample4,creek1 - sample5,creek3	-0.35	0.0966092	15	-3.6228442	0.1101055
.	sample5,creek1 - sample1,creek2	0.45	0.0966092	15	4.6579425	0.0183213
.	sample5,creek1 - sample2,creek2	0.60	0.0966092	15	6.2105900	0.0012178
.	sample5,creek1 - sample3,creek2	0.40	0.0966092	15	4.1403934	0.0456168
.	sample5,creek1 - sample4,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample5,creek1 - sample5,creek2	0.65	0.0966092	15	6.7281392	0.0005147
.	sample5,creek1 - sample1,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample5,creek1 - sample2,creek3	-0.15	0.0966092	15	-1.5526475	0.9626296
.	sample5,creek1 - sample3,creek3	-0.10	0.0966092	15	-1.0350983	0.9991798
.	sample5,creek1 - sample4,creek3	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample5,creek1 - sample5,creek3	-0.25	0.0966092	15	-2.5877458	0.4869333
.	sample1,creek2 - sample2,creek2	0.15	0.0966092	15	1.5526475	0.9626296
.	sample1,creek2 - sample3,creek2	-0.05	0.0966092	15	-0.5175492	0.9999999
.	sample1,creek2 - sample4,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample1,creek2 - sample5,creek2	0.20	0.0966092	15	2.0701967	0.7780478
.	sample1,creek2 - sample1,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample1,creek2 - sample2,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample1,creek2 - sample3,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample1,creek2 - sample4,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample1,creek2 - sample5,creek3	-0.70	0.0966092	15	-7.2456884	0.0002237
.	sample2,creek2 - sample3,creek2	-0.20	0.0966092	15	-2.0701967	0.7780478
.	sample2,creek2 - sample4,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample2,creek2 - sample5,creek2	0.05	0.0966092	15	0.5175492	0.9999999
.	sample2,creek2 - sample1,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample2,creek2 - sample2,creek3	-0.75	0.0966092	15	-7.7632375	0.0001001
.	sample2,creek2 - sample3,creek3	-0.70	0.0966092	15	-7.2456884	0.0002237
.	sample2,creek2 - sample4,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample2,creek2 - sample5,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample3,creek2 - sample4,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample3,creek2 - sample5,creek2	0.25	0.0966092	15	2.5877458	0.4869333
.	sample3,creek2 - sample1,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample3,creek2 - sample2,creek3	-0.55	0.0966092	15	-5.6930409	0.0029559
.	sample3,creek2 - sample3,creek3	-0.50	0.0966092	15	-5.1754917	0.0073200
.	sample3,creek2 - sample4,creek3	-0.60	0.0966092	15	-6.2105900	0.0012178
.	sample3,creek2 - sample5,creek3	-0.65	0.0966092	15	-6.7281392	0.0005147
.	sample4,creek2 - sample5,creek2	0.00	0.0966092	15	0.0000000	1.0000000
.	sample4,creek2 - sample1,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample4,creek2 - sample2,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462
.	sample4,creek2 - sample3,creek3	-0.75	0.0966092	15	-7.7632375	0.0001001
.	sample4,creek2 - sample4,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample4,creek2 - sample5,creek3	-0.90	0.0966092	15	-9.3158851	0.0000107
.	sample5,creek2 - sample1,creek3	-0.85	0.0966092	15	-8.7983359	0.0000219
.	sample5,creek2 - sample2,creek3	-0.80	0.0966092	15	-8.2807867	0.0000462

2017S

Brad Crain, Jong Sung Kim*

2017SR1

2015F1
Find the best model for predicting Y based on X1 and X2. Y is the amount of profit that a company makes in a month. X1 is the number of months that the company has been in business. X2 is the amount spent on advertising. Consider as predictors all possible linear and quadratic terms ($X1, X1^2, X2, X2^2$, and $X1X2$). Consider possible transformations of Y. Include all appropriate diagnostics. When you have found your "best" model, predict a new Y when X1 = 20 and X2 = /$1,500, giving a 95% prediction interval. The data set, shown below, appears in "Profits.xlsx".

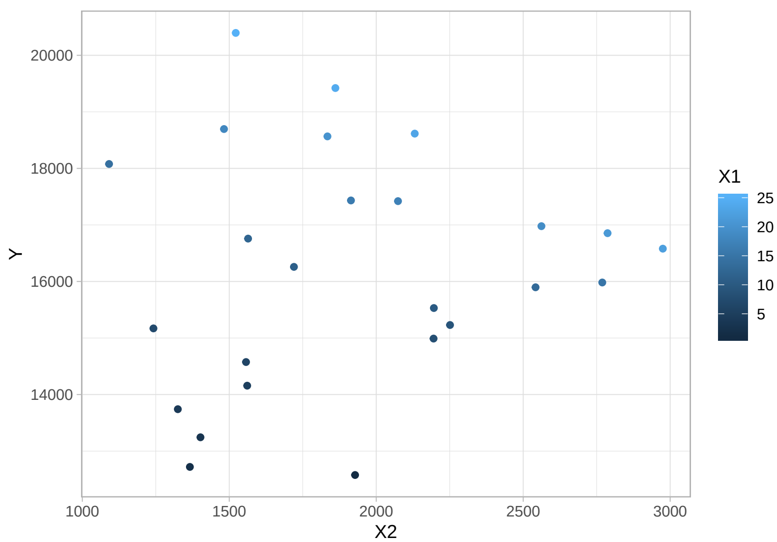
```r
table_2017sr1 <- readxl::read_xlsx("qe_lab/Profits_2017s.xlsx")
# table_2017sr1$X1 <- as.factor(table_2017sr1$X1)
str(table_2017sr1)
```

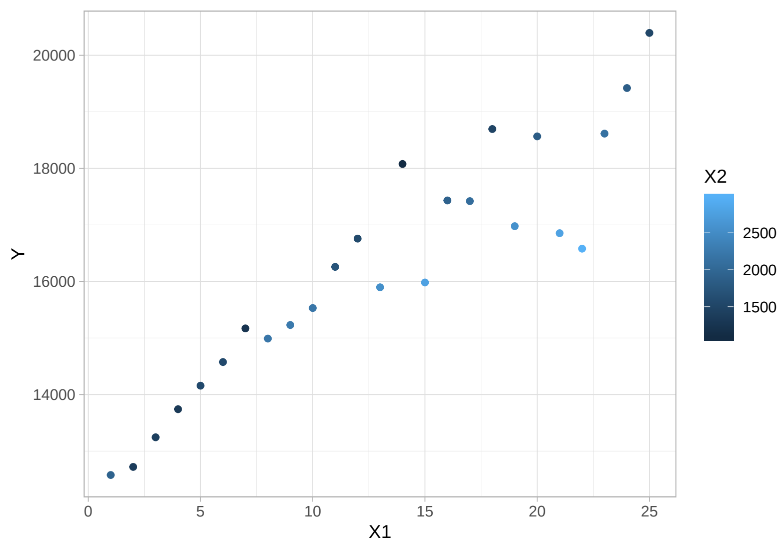
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    25 obs. of  3 variables:
##  $ X1: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ X2: num  1928 1366 1402 1325 1561 ...
##  $ Y : num  12577 12720 13244 13741 14157 ...
```

```r
summary(table_2017sr1)
```

```
##        X1           X2             Y        
##  Min.   : 1   Min.   :1091   Min.   :12577  
##  1st Qu.: 7   1st Qu.:1522   1st Qu.:14990  
##  Median :13   Median :1861   Median :16258  
##  Mean   :13   Mean   :1914   Mean   :16235  
##  3rd Qu.:19   3rd Qu.:2196   3rd Qu.:17433  
##  Max.   :25   Max.   :2975   Max.   :20396
```

```r
library(ggplot2)
ggplot(table_2017sr1, aes(X2,Y,color=X1))+geom_point()+theme_light()
ggplot(table_2017sr1, aes(X1,Y,color=X2))+geom_point()+theme_light()
```






```r
model_2017sr1 <- lm(Y^2~X1+X2, table_2017sr1)
# car::vif(model_2017sr1)
summary(model_2017sr1)
```

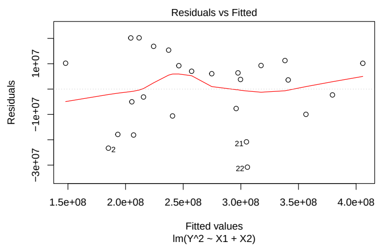
```
## 
## Call:
## lm(formula = Y^2 ~ X1 + X2, data = table_2017sr1)
## 
## Residuals:
##       Min        1Q    Median        3Q       Max 
## -30805386  -9969025   3791394  10176772  20218197 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 231392970   11535085  20.060 1.25e-15 ***
## X1            9924495     455383  21.794  < 2e-16 ***
## X2             -48414       6402  -7.563 1.48e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 14900000 on 22 degrees of freedom
## Multiple R-squared:  0.956,  Adjusted R-squared:  0.952 
## F-statistic:   239 on 2 and 22 DF,  p-value: 1.194e-15
```

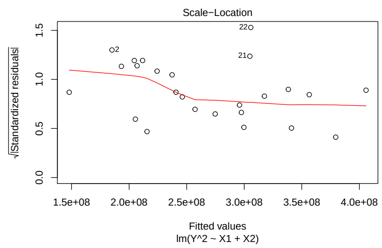
```r
anova(model_2017sr1)
```

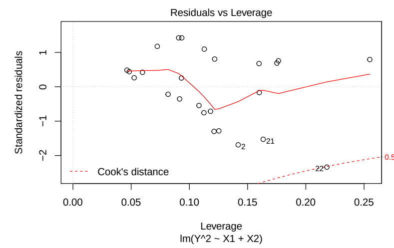
```
## Analysis of Variance Table
## 
## Response: Y^2
##           Df     Sum Sq    Mean Sq F value    Pr(>F)    
## X1         1 9.3403e+16 9.3403e+16 420.892 7.822e-16 ***
## X2         1 1.2692e+16 1.2692e+16  57.192 1.482e-07 ***
## Residuals 22 4.8822e+15 2.2192e+14
```

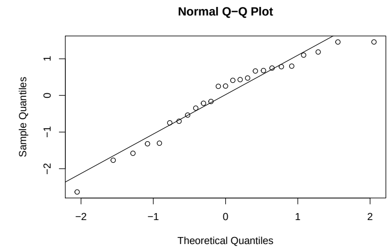
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

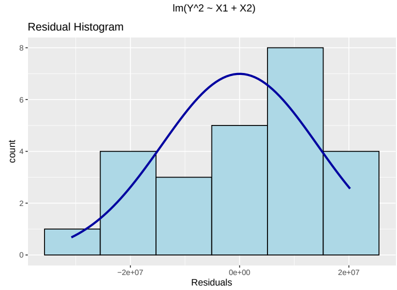
```
plot(model_2017sr1,c(1,3,5))
residual_2017sr1 <- rstudent(model_2017sr1)
qqnorm(residual_2017sr1)
qqline(residual_2017sr1)
olsrr::ols_plot_resid_hist(model_2017sr1)
hist(residual_2017sr1)
```

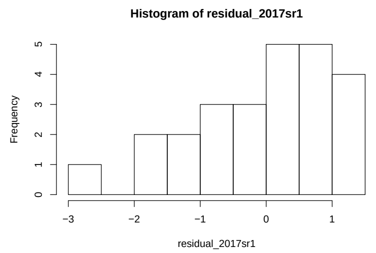













```
sqrt(predict(model_2017sr1,newdata = data.frame(X1=20,X2=1500),interval = "prediction", level = 0.95))
```

```
##        fit      lwr      upr
## 1 18901.39 18003.89 19758.17
```

2017SD1

Review the data provided in 'NBalance.xlsx'. Note, there were nine distinct treatments [Feed Rations] and three distinct animals. An experimental design was used to examine the means differences in the Nitrogen balance in ruminants. Provide the following in your answer

1. Which design was used, include the required parameters of the experimental design $[t; b; k; r; \lambda]$

$y = t + b + k + r + \lambda + \varepsilon$ Latin Square? BIBD?

2. An appropriate ANOVA
3. A TukeyHSD analysis of the proper means differences
4. Conclusions on the impact of Feed Rations on Nitrogen Balance in Ruminants

Source: J.L. Gill (1978), Design and analysis of experiments in the animal and medical sciences, Vol2. Ames, Iowa: Iowa State University Press

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    27 obs. of  4 variables:
##  $ Block   : Factor w/ 9 levels "Blk1","Blk2",..: 1 1 1 2 2 2 3 3 3 4 ...
##  $ Animal  : Factor w/ 3 levels "Animal1","Animal2",..: 1 2 3 1 2 3 1 2 3 1 ...
##  $ Ration  : Factor w/ 9 levels "Ration1","Ration2",..: 1 2 3 1 4 6 1 5 7 2 ...
##  $ Nitrogen: num  33.7 37.8 42.2 38.6 45.4 ...
```

```
library(ggpubr)
ggline(table_2017sd1, "Animal","Nitrogen", add = c("mean","jitter"),color = "Ration",shape = "Ration")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```
ggline(table_2017sd1, "Block","Nitrogen", add = c("mean","jitter"),color = "Ration",shape = "Ration")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```r
ggline(table_2017sd1, "Animal","Nitrogen", add = c("mean","jitter"),color = "Block",shape = "Block")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```r
ggline(table_2017sd1, "Ration","Nitrogen", add = c("mean","jitter"),color = "Block",shape = "Block")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

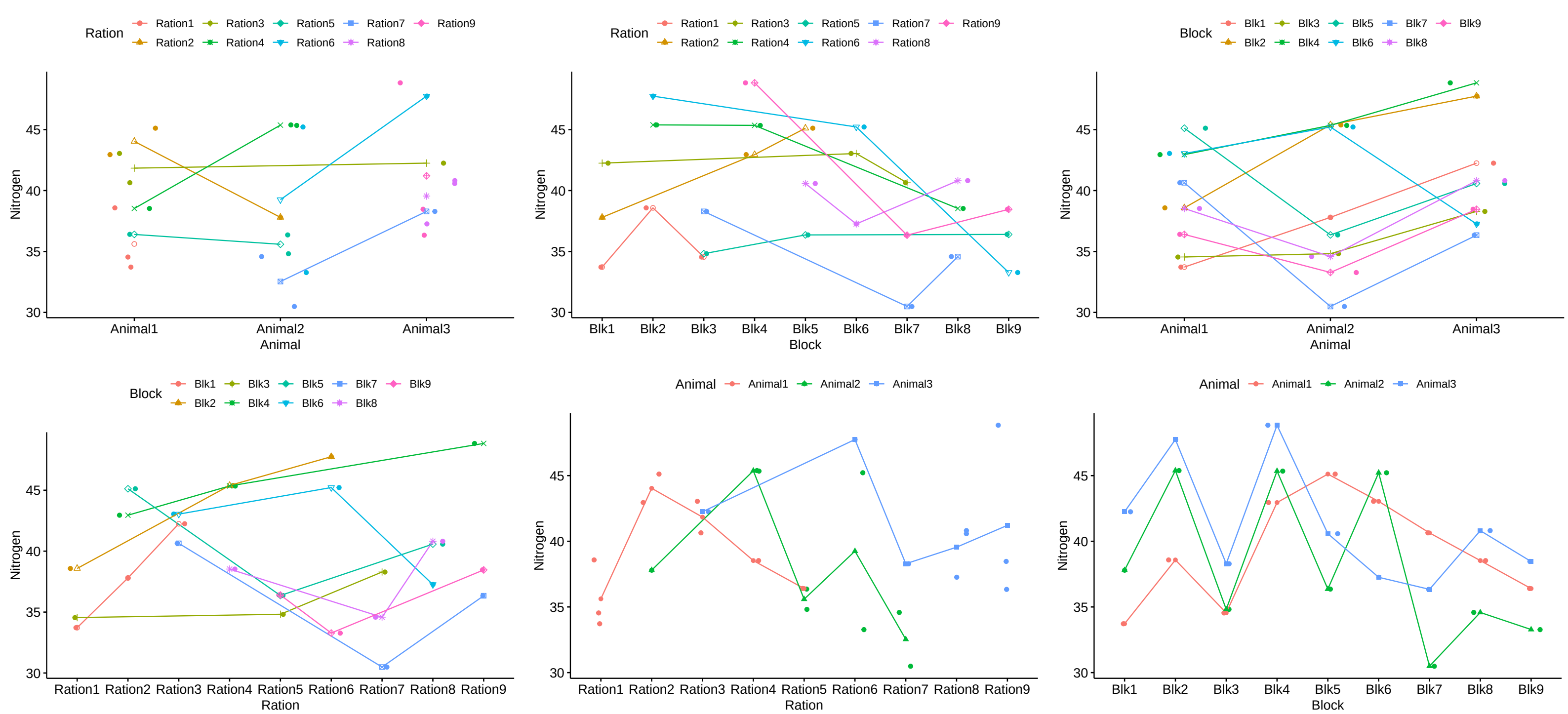
```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```r
ggline(table_2017sd1, "Ration","Nitrogen", add = c("mean","jitter"),color = "Animal",shape = "Animal")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```r
ggline(table_2017sd1, "Block","Nitrogen", add = c("mean","jitter"),color = "Animal",shape = "Animal")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```




```r
model_2017sd1 <- aov(Nitrogen~Animal+Ration, table_2017sd1)
summary(model_2017sd1)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Animal       2  42.23   21.12   1.237  0.317
## Ration       8 274.92   34.36   2.013  0.111
## Residuals   16 273.10   17.07
```

```r
anova(model_2017sd1)
```

```
## Analysis of Variance Table
```

```
##
## Response: Nitrogen
##           Df  Sum Sq Mean Sq F value Pr(>F)
## Animal     2  42.233  21.117  1.2372 0.3165
## Ration     8 274.919  34.365  2.0133 0.1112
## Residuals 16 273.096  17.069
```

```
TukeyHSD(model_2017sd1,conf.level = 0.95)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Nitrogen ~ Animal + Ration, data = table_2017sd1)
##
## $Animal
##                       diff       lwr      upr     p adj
## Animal2-Animal1 -1.137778 -6.163134 3.887579 0.8304041
## Animal3-Animal1  1.894444 -3.130912 6.919801 0.6039032
## Animal3-Animal2  3.032222 -1.993134 8.057579 0.2921780
##
## $Ration
##                       diff        lwr       upr     p adj
## Ration2-Ration1  6.7192593  -5.281040 18.719558 0.5684881
## Ration3-Ration1  5.7285185  -6.271780 17.728817 0.7397514
## Ration4-Ration1  8.2318519  -3.768447 20.232151 0.3269033
## Ration5-Ration1  1.0018519 -10.998447 13.002151 0.9999968
## Ration6-Ration1  6.5937037  -5.406595 18.594003 0.5905654
## Ration7-Ration1 -1.0362963 -13.036595 10.964003 0.9999958
## Ration8-Ration1  2.0388889  -9.961410 14.039188 0.9993103
## Ration9-Ration1  3.7022222  -8.298077 15.702521 0.9663240
## Ration3-Ration2 -0.9907407 -12.991040 11.009558 0.9999970
## Ration4-Ration2  1.5125926 -10.487706 13.512892 0.9999235
## Ration5-Ration2 -5.7174074 -17.717706  6.282892 0.7415690
## Ration6-Ration2 -0.1255556 -12.125854 11.874743 1.0000000
## Ration7-Ration2 -7.7555556 -19.755854  4.244743 0.3959375
## Ration8-Ration2 -4.6803704 -16.680669  7.319929 0.8870802
## Ration9-Ration2 -3.0170370 -15.017336  8.983262 0.9901325
## Ration4-Ration3  2.5033333  -9.496966 14.503632 0.9971040
## Ration5-Ration3 -4.7266667 -16.726966  7.273632 0.8818327
## Ration6-Ration3  0.8651852 -11.135114 12.865484 0.9999990
## Ration7-Ration3 -6.7648148 -18.765114  5.235484 0.5605018
## Ration8-Ration3 -3.6896296 -15.689929  8.310669 0.9669660
## Ration9-Ration3 -2.0262963 -14.026595  9.974003 0.9993402
## Ration5-Ration4 -7.2300000 -19.230299  4.770299 0.4805005
## Ration6-Ration4 -1.6381481 -13.638447 10.362151 0.9998610
## Ration7-Ration4 -9.2681481 -21.268447  2.732151 0.2059952
## Ration8-Ration4 -6.1929630 -18.193262  5.807336 0.6609465
## Ration9-Ration4 -4.5296296 -16.529929  7.470669 0.9032344
## Ration6-Ration5  5.5918519  -6.408447 17.592151 0.7618186
## Ration7-Ration5 -2.0381481 -14.038447  9.962151 0.9993120
## Ration8-Ration5  1.0370370 -10.963262 13.037336 0.9999958
## Ration9-Ration5  2.7003704  -9.299929 14.700669 0.9951866
## Ration7-Ration6 -7.6300000 -19.630299  4.370299 0.4154158
## Ration8-Ration6 -4.5548148 -16.555114  7.445484 0.9006356
## Ration9-Ration6 -2.8914815 -14.891780  9.108817 0.9924771
## Ration8-Ration7  3.0751852  -8.925114 15.075484 0.9888726
## Ration9-Ration7  4.7385185  -7.261780 16.738817 0.8804680
## Ration9-Ration8  1.6633333 -10.336966 13.663632 0.9998443
```

2017F

Robert Fountain*, Daniel Taylor-Rodriguez

2017F1

2016S1
Find the best model for predicting Y (weight) based on X1 (age), X2 (height), and X3 (indicator for male). Consider as predictors all possible linear and quadratic terms. Consider possible transformations of Y. Include all appropriate diagnostics. When you have found your "best" model, predict a new Y when X1 = 26, X2 = 70, and X3 = 1, giving a 95% prediction interval. The data set, shown below, appears in "RegressionFall17.xlsx".

```
table_2017f1 <- readxl::read_xlsx("qe_lab/RegressionFall17.xlsx")[-1,]
table_2017f1$weight <- round(as.numeric(table_2017f1$weight),2)
table_2017f1$age <- as.numeric(table_2017f1$age)
table_2017f1$height <- round(as.numeric(table_2017f1$height),2)
table_2017f1$male <- factor(table_2017f1$male, labels = c("female", "male"))
str(table_2017f1)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    30 obs. of  4 variables:
##  $ weight: num  240 100 233 108 239 ...
##  $ age   : num  20 20 20 20 20 21 21 21 21 21 ...
##  $ height: num  71 67.2 68.1 67.7 68.6 65.2 67.6 67.4 67.5 69.4 ...
##  $ male  : Factor w/ 2 levels "female","male": 2 1 2 1 2 1 1 1 1 2 ...
```

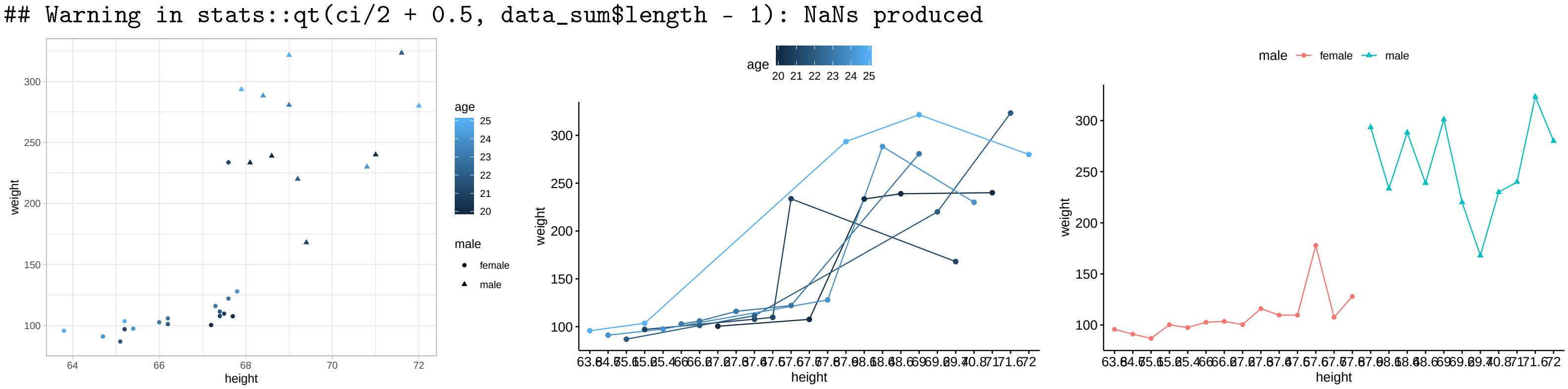
```
library(ggplot2)
ggplot(table_2017f1, aes(height,weight,color=age,shape=male))+geom_point()+theme_light()
library(ggpubr)
ggline(table_2017f1,"height","weight",add=c("mean","jetter"),color="age")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```

```
ggline(table_2017f1,"height","weight",add=c("mean","jetter"),color="male",shape = "male")
```

```
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
```




```
model_2017f1 <- lm(weight~height*age*male,table_2017f1)
library(olsrr)
ols_step_both_aic(model_2017f1)
```

```
## Stepwise Selection Method
## -------------------------
##
## Candidate Terms:
##
## 1 . height
## 2 . age
## 3 . male
## 4 . height:age
## 5 . height:male
```

```
## 6 . age:male
## 7 . height:age:male
##
##
## Variables Entered/Removed:
##
## - height:age:male added
## - age:male added
## - height:age added
##
## No more variables to be added or removed.
##
##
##                                     Stepwise Summary
## -------------------------------------------------------------------------------------------
## Variable            Method       AIC          RSS          Sum Sq        R-Sq      Adj. R-Sq
## -------------------------------------------------------------------------------------------
## height:age:male    addition    304.169    34051.024    163429.310    0.82757      0.81480
## age:male           addition    303.786    29423.052    168057.281    0.85101      0.82717
## height:age         addition    303.786    29423.052    168057.281    0.85101      0.82717
## -------------------------------------------------------------------------------------------
```

```
ols_step_both_p(model_2017f1)
```

```
## Stepwise Selection Method
## ---------------------------
##
## Candidate Terms:
##
## 1. height
## 2. age
## 3. male
## 4. height:age
## 5. height:male
## 6. age:male
## 7. height:age:male
##
## We are selecting variables based on p value...
##
## Variables Entered/Removed:
##
## - male added
## - age:male added
## - age added
## - height added
##
## No more variables to be added/removed.
##
##
## Final Model Output
## ------------------
##
##                           Model Summary
## ----------------------------------------------------------------
## R                        0.921       RMSE                 34.629
## R-Squared                0.848       Coef. Var            20.228
## Adj. R-Squared           0.824       MSE                1199.151
## Pred R-Squared           0.788       MAE                  20.901
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                 ANOVA
## ----------------------------------------------------------------------
##                  Sum of
##                 Squares        DF    Mean Square      F          Sig.
## ----------------------------------------------------------------------
## Regression    167501.551        4      41875.388    34.921    0.0000
## Residual       29978.783       25       1199.151
## Total         197480.333       29
## ----------------------------------------------------------------------
##
##                                      Parameter Estimates
## ---------------------------------------------------------------------------------------------
##        model         Beta    Std. Error    Std. Beta        t         Sig         lower      upper
## ---------------------------------------------------------------------------------------------
##  (Intercept)     -321.525       435.768                  -0.738      0.467    -1219.006    575.956
##     malemale     -191.733       172.899       -1.158     -1.109      0.278     -547.826    164.360
##          age       -1.245         5.847       -0.026     -0.213      0.833      -13.287     10.797
##       height        6.951         5.487        0.172      1.267      0.217       -4.349     18.252
## malemale:age       14.058         7.892        1.929      1.781      0.087       -2.195     30.311
## ---------------------------------------------------------------------------------------------
##
##                               Stepwise Selection Summary
## ---------------------------------------------------------------------------------------
##                      Added/                   Adj.
## Step    Variable    Removed     R-Square    R-Square     C(p)        AIC         RMSE
## ---------------------------------------------------------------------------------------
##    1      male      addition       0.795       0.788    6.9980    307.3529    38.0198
##    2    age:male    addition       0.838       0.820    2.0100    304.2144    35.0293
##    3      age       addition       0.838       0.820    4.0100    304.2144    35.0293
##    4     height     addition       0.848       0.824    4.4410    304.3477    34.6288
## ---------------------------------------------------------------------------------------
```

```r
model_2017f1_1 <- lm(weight~height+age:male,table_2017f1)
model_2017f1_2 <- lm(log(weight)~height+age:male,table_2017f1)
car::vif(model_2017f1_2)
```

```
##            GVIF Df GVIF^(1/(2*Df))
## height   2.8472  1        1.687365
## age:male 2.8472  2        1.298986
```

```r
summary(model_2017f1_2)
```

```
##
## Call:
## lm(formula = log(weight) ~ height + age:male, data = table_2017f1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35019 -0.06823 -0.03331  0.08138  0.70476
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.592786   2.103225   0.282   0.7803
## height         0.058601   0.028409   2.063   0.0493 *
## age:malefemale 0.009281   0.021315   0.435   0.6668
## age:malemale   0.038784   0.020107   1.929   0.0647 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1852 on 26 degrees of freedom
## Multiple R-squared:  0.8599, Adjusted R-squared:  0.8438
## F-statistic:  53.2 on 3 and 26 DF,  p-value: 3.121e-11
```

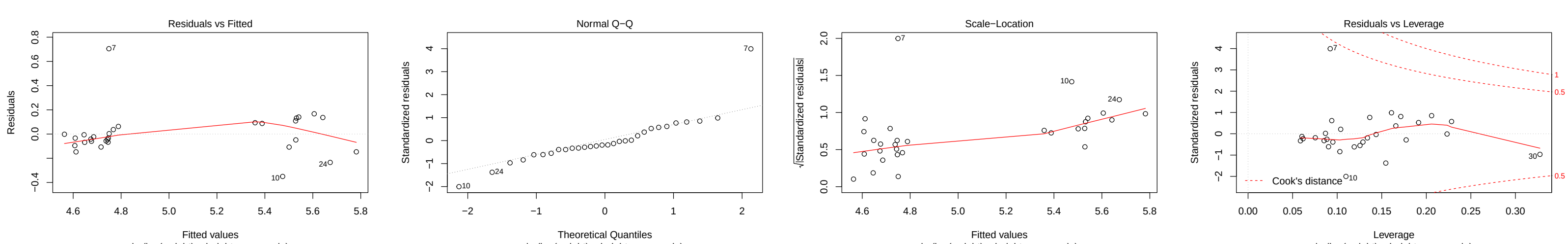
```r
anova(model_2017f1_2)
```

```
## Analysis of Variance Table
##
## Response: log(weight)
##           Df Sum Sq Mean Sq F value    Pr(>F)
## height     1 4.1007  4.1007 119.540 3.203e-11 ***
## age:male   2 1.3744  0.6872  20.033 5.432e-06 ***
## Residuals 26 0.8919  0.0343
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
ols_regress(model_2017f1_2)
```

```
##                         Model Summary
## -------------------------------------------------------------
## R                       0.927       RMSE                0.185
## R-Squared               0.860       Coef. Var           3.679
## Adj. R-Squared          0.844       MSE                 0.034
## Pred R-Squared          0.819       MAE                 0.112
## -------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                ANOVA
## -------------------------------------------------------------------
##                Sum of
##               Squares        DF    Mean Square      F         Sig.
## -------------------------------------------------------------------
## Regression      5.475         3          1.825    53.202     0.0000
## Residual        0.892        26          0.034
## Total           6.367        29
## -------------------------------------------------------------------
##
##                                   Parameter Estimates
## ---------------------------------------------------------------------------------------
##          model      Beta    Std. Error    Std. Beta       t        Sig      lower    upper
## ---------------------------------------------------------------------------------------
##    (Intercept)     0.593         2.103                  0.282    0.780    -3.730    4.916
##         height     0.059         0.028        0.255     2.063    0.049     0.000    0.117
## age:malefemale     0.009         0.021        0.223     0.435    0.667    -0.035    0.053
##   age:malemale     0.039         0.020        0.937     1.929    0.065    -0.003    0.080
## ---------------------------------------------------------------------------------------
```

```r
plot(model_2017f1_2)
```




```r
predict(model_2017f1_2, newdata=data.frame(age= 26, height= 70, male= "male"),interval = "prediction",level = 0.95)
```

```
##        fit      lwr      upr
## 1 5.703233 5.278615 6.127851
```

2017F2

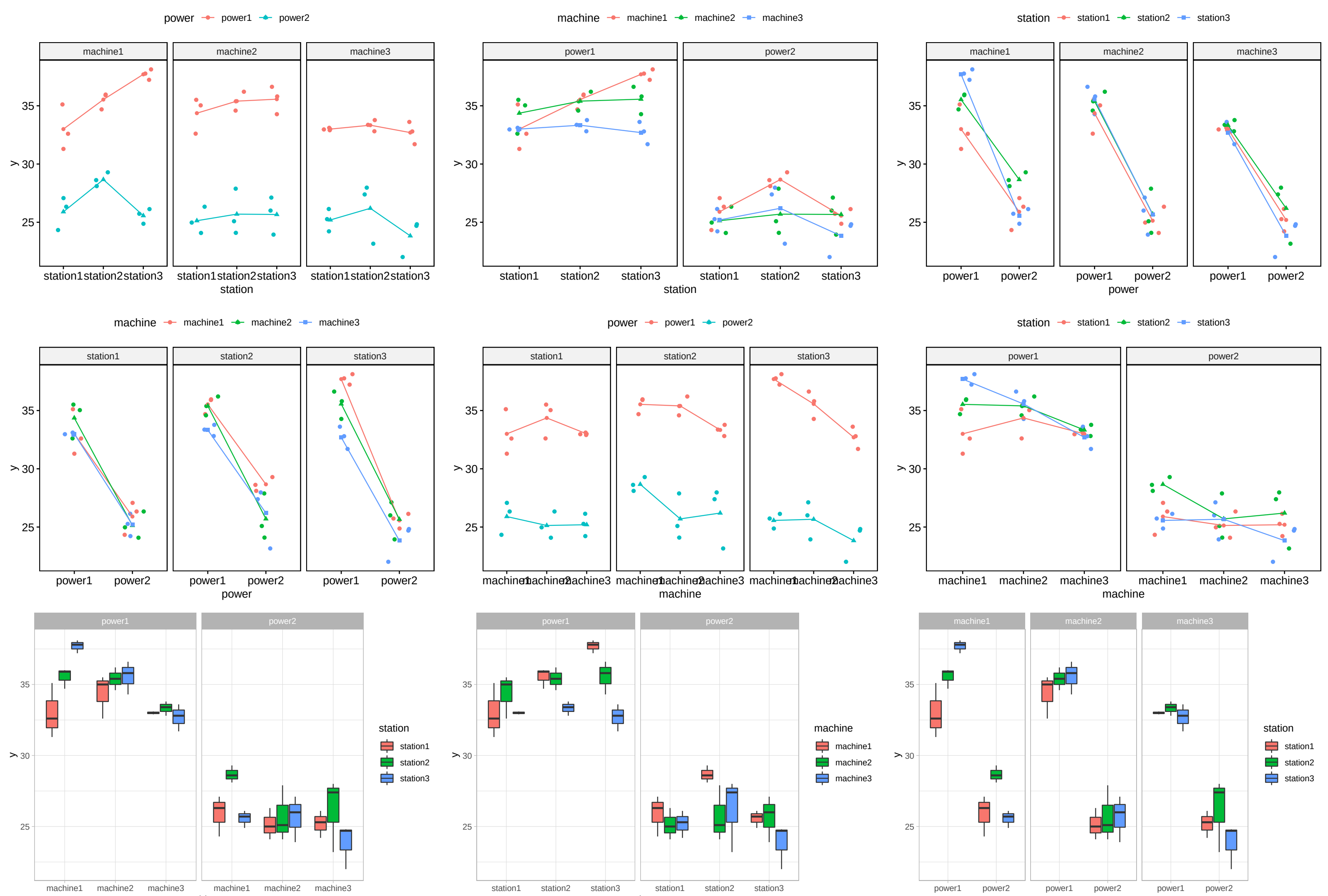
A process engineer is testing the yield of a product manufactured on three specific machines. Each machine can be operated at fixed high and low power settings, although the actual settings differ from one machine to the next. Furthermore, a machine has three stations on which the product is formed, and these are the same for each machine. An experiment is conducted in which each machine is tested at both power settings, and three observations on yield are taken from each station. The runs are made in random order. Analyze this experiment. The data set, shown below, appears in “DesignFall17.xlsx”.

```
DesignFall17 <- readxl::read_excel("qe_lab/DesignFall17.xlsx")
```

```
## New names:
## * `` -> ...1
## * `` -> ...2
## * `` -> ...4
## * `` -> ...5
## * `` -> ...6
## * ... and 4 more problems
```

```
library(tidyverse)
table_2017f2 <- gather(DesignFall17[c(2:4,6:8),c(2:4,6:8,10:12)])
names(table_2017f2)<- c("machine","y")
table_2017f2<- table_2017f2[c("y","machine")]
table_2017f2$machine <- as.factor(c(rep("machine1",18),rep("machine2",18),rep("machine3",18)))
table_2017f2$station <- as.factor(rep(c(rep("station1",6),rep("station2",6),rep("station3",6)),3))
table_2017f2$power <- as.factor(rep(c(rep("power1",3),rep("power2",3)),9))
str(table_2017f2)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    54 obs. of  4 variables:
##  $ y      : num  35.1 31.3 32.6 24.3 26.3 27.1 34.7 35.9 36 28.1 ...
##  $ machine: Factor w/ 3 levels "machine1","machine2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ station: Factor w/ 3 levels "station1","station2",..: 1 1 1 1 1 1 2 2 2 2 ...
##  $ power  : Factor w/ 2 levels "power1","power2": 1 1 1 2 2 2 1 1 1 2 ...
```



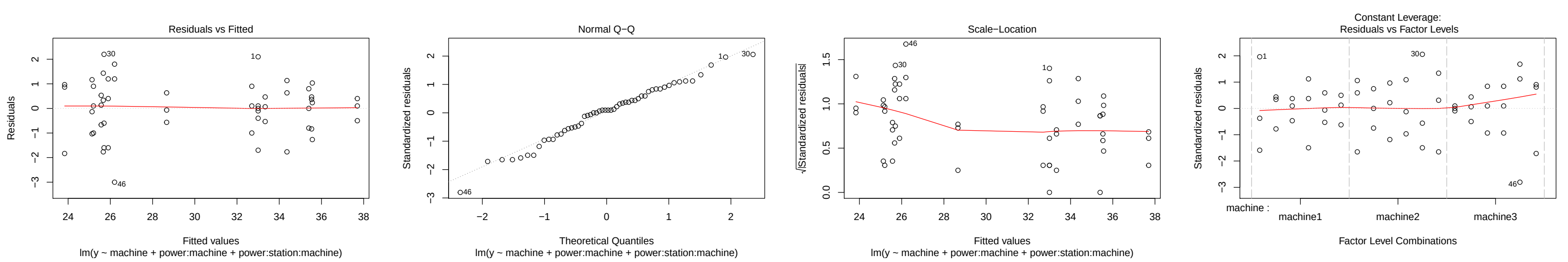

```
model_2017f2 <- lm(y~power*station*machine, table_2017f2)
library(olsrr)
# ols_step_both_aic(model_2017f2)
# ols_step_both_p(model_2017f2)
model_2017f2_2 <- lm(y~machine+power:machine+power:station:machine, table_2017f2)
summary(model_2017f2_2)
```

```
##
## Call:
```

```
## lm(formula = y ~ machine + power:machine + power:station:machine,
##     data = table_2017f2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0000 -0.6500  0.1000  0.7583  2.2000
##
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                             3.300e+01  7.562e-01  43.639
## machinemachine2                         1.367e+00  1.069e+00   1.278
## machinemachine3                        -2.546e-14  1.069e+00   0.000
## machinemachine1:powerpower2            -7.100e+00  1.069e+00  -6.639
## machinemachine2:powerpower2            -9.233e+00  1.069e+00  -8.634
## machinemachine3:powerpower2            -7.800e+00  1.069e+00  -7.294
## machinemachine1:powerpower1:stationstation2  2.533e+00  1.069e+00   2.369
## machinemachine2:powerpower1:stationstation2  1.033e+00  1.069e+00   0.966
## machinemachine3:powerpower1:stationstation2  3.333e-01  1.069e+00   0.312
## machinemachine1:powerpower2:stationstation2  2.767e+00  1.069e+00   2.587
## machinemachine2:powerpower2:stationstation2  5.667e-01  1.069e+00   0.530
## machinemachine3:powerpower2:stationstation2  1.000e+00  1.069e+00   0.935
## machinemachine1:powerpower1:stationstation3  4.700e+00  1.069e+00   4.395
## machinemachine2:powerpower1:stationstation3  1.200e+00  1.069e+00   1.122
## machinemachine3:powerpower1:stationstation3 -3.000e-01  1.069e+00  -0.281
## machinemachine1:powerpower2:stationstation3 -3.333e-01  1.069e+00  -0.312
## machinemachine2:powerpower2:stationstation3  5.333e-01  1.069e+00   0.499
## machinemachine3:powerpower2:stationstation3 -1.367e+00  1.069e+00  -1.278
##                                          Pr(>|t|)
## (Intercept)                               < 2e-16 ***
## machinemachine2                            0.2095
## machinemachine3                            1.0000
## machinemachine1:powerpower2              9.82e-08 ***
## machinemachine2:powerpower2              2.70e-10 ***
## machinemachine3:powerpower2              1.36e-08 ***
## machinemachine1:powerpower1:stationstation2   0.0233 *
## machinemachine2:powerpower1:stationstation2   0.3404
## machinemachine3:powerpower1:stationstation2   0.7571
## machinemachine1:powerpower2:stationstation2   0.0139 *
## machinemachine2:powerpower2:stationstation2   0.5995
## machinemachine3:powerpower2:stationstation2   0.3560
## machinemachine1:powerpower1:stationstation3 9.38e-05 ***
## machinemachine2:powerpower1:stationstation3   0.2693
## machinemachine3:powerpower1:stationstation3   0.7807
## machinemachine1:powerpower2:stationstation3   0.7571
## machinemachine2:powerpower2:stationstation3   0.6210
## machinemachine3:powerpower2:stationstation3   0.2095
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.31 on 36 degrees of freedom
## Multiple R-squared:  0.9486, Adjusted R-squared:  0.9243
## F-statistic: 39.08 on 17 and 36 DF,  p-value: < 2.2e-16
```

```r
anova(model_2017f2_2)
```

```
## Analysis of Variance Table
##
## Response: y
##                       Df  Sum Sq Mean Sq  F value    Pr(>F)
## machine                2   37.37   18.68  10.8913  0.000200 ***
## machine:power          3 1039.51  346.50 201.9765 < 2.2e-16 ***
## machine:power:station 12   62.79    5.23   3.0501  0.004742 **
## Residuals             36   61.76    1.72
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```




```r
# When some factors are random
library(GAD)
table_2019s2$Run_r <- as.random(table_2019s2$Run)
table_2019s2$Trt_f <- as.fixed(table_2019s2$Trt)
table_2019s2$Rev_f <- as.fixed(table_2019s2$Rev)
model_2019s2_1 <- aov(formula = Shrink ~ Run_r+Trt_f + Trt_f%in%Run_r+ Rev_f%in%Run_r + Rev_f + Trt_f:Rev_f, data=ta
pander::pander(gad(model_2019s2_1))
```

```r
# When some factors are random
library("lme4")
model_2017f2_3 <- lmer(formula = y ~ (1|machine) + station + power+ (1|machine:station) + (1|machine:station:power)
```

```r
summary(model_2017f2_3)$varcor
pander::pander(confint(model_2019s2_2)[1:4,1:2])
```

2018S

Robert Fountain*, Daniel Taylor-Rodriguez

2018S1

The data for this problem was obtained from research relating children smoking to pulmonary function. Today it is well established that smoking cigarettes is a very unhealthy habit, especially for children; however, this was not well-known in the past. This data corresponds to one of the first studies of the effects of smoking on pulmonary (i.e., lung) function, an observational study of 654 youths aged 3 to 19. The variables in the study are displayed in Table 1 below. The outcome variable is volume, which measures the liters of air exhaled by the child in the first second of a forced breath. Some evidence in the literature suggests that children under age 6 may not understand the instructions of the breath exhalation test, so that the quality of volume measurements for those children is suspect. We are interested in the relationship between smoking, gender and the volume of air exhaled. Smoking is expected to impair pulmonary function (i.e., decrease volume).
Find the best model to predict volume considering as predictors all possible linear, quadratic and pairwise interaction terms. Additionally, consider possible transformations of the response (i.e., volume), and include all relevant diagnostic measures. Once you select the best model, write down and test the hypothesis to determine if the volume is influenced by the smoking status in terms of your best model's parameters. Using this same model, predict the volume for a 16-yearold male smoker who is 61 inches high, and provide a 95% prediction interval. A description of the variables is found in the table below, and the data is included in the file Problem1_ChildSmoking.xlsx.
Variable Name and Description
age: age of child in years
volume: volume of air in exhaled breath in liters
height: height of child in inches
male=1 if child is male, and =0 otherwise
smoker=1 if child reports that he or she smokes cigarettes regularly, and =0 otherwise

```r
table_2018s1 <- readxl::read_xlsx("qe_lab/Problem1_ChildSmoking.xlsx")
table_2018s1_above6 <- table_2018s1[which(table_2018s1$age>5),]
table_2018s1_above6$age <- factor(table_2018s1_above6$age)
table_2018s1_above6$male <- factor(table_2018s1_above6$male, labels = c("female","male"))
table_2018s1_above6$smoker <- factor(table_2018s1_above6$smoker, labels = c("not regu","regularly"))
str(table_2018s1)
```

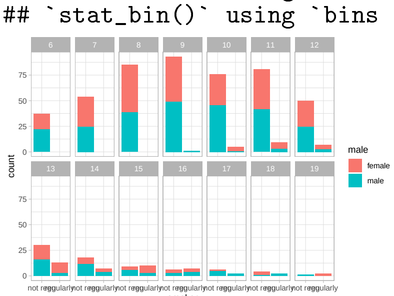
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    654 obs. of  5 variables:
##  $ age   : num  9 8 7 9 9 8 6 6 8 9 ...
##  $ volume: num  1.71 1.72 1.72 1.56 1.9 ...
##  $ height: num  57 67.5 54.5 53 57 61 58 56 58.5 60 ...
##  $ male  : num  0 0 0 1 1 0 0 0 0 0 ...
##  $ smoker: num  0 0 0 0 0 0 0 0 0 0 ...
```

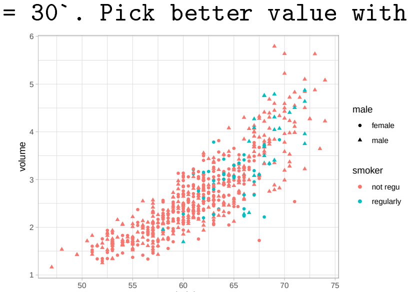
```r
str(table_2018s1_above6)
```

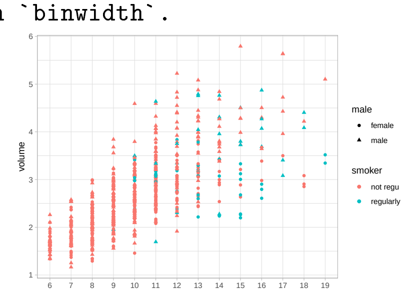
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    615 obs. of  5 variables:
##  $ age   : Factor w/ 14 levels "6","7","8","9",..: 4 3 2 4 4 3 1 1 3 4 ...
##  $ volume: num  1.71 1.72 1.72 1.56 1.9 ...
##  $ height: num  57 67.5 54.5 53 57 61 58 56 58.5 60 ...
##  $ male  : Factor w/ 2 levels "female","male": 1 1 1 2 2 1 1 1 1 1 ...
##  $ smoker: Factor w/ 2 levels "not regu","regularly": 1 1 1 1 1 1 1 1 1 1 ...
```

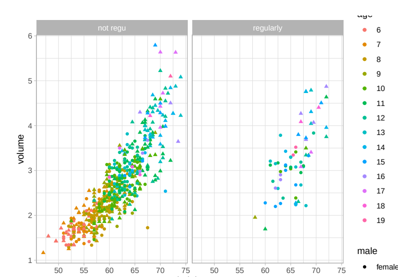
```r
summary(table_2018s1$height)
```

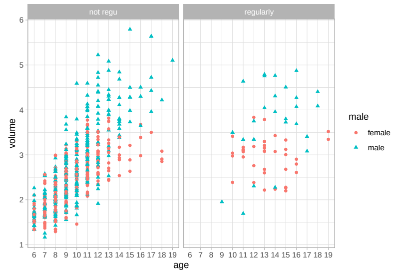
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   46.00   57.00   61.50   61.14   65.50   74.00
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## Warning in stats::qt(ci/2 + 0.5, data_sum$length - 1): NaNs produced
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

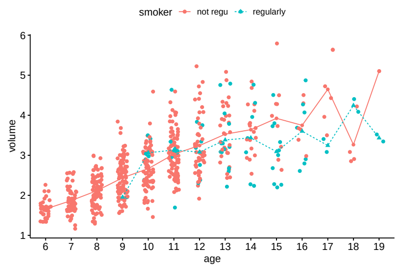


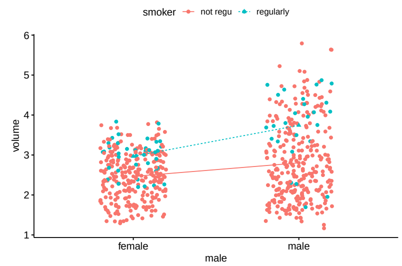


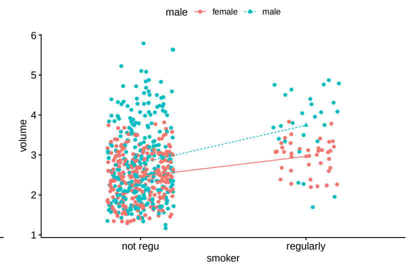


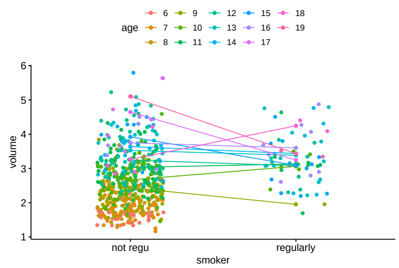


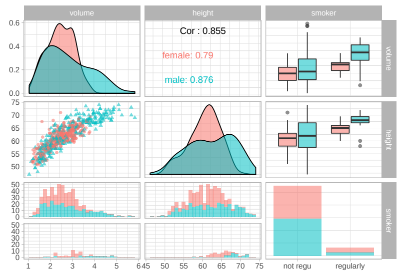













```
model_2018s1 <- lm(volume~height*age*male*smoker,table_2018s1_above6)
ols_step_both_aic(model_2018s1)
ols_step_both_p(model_2018s1)
```

```
model_2018s1_2 <- lm(log(volume)~log(height):age:male+smoker,table_2018s1_above6)
summary(model_2018s1_2)
```

```
##
## Call:
## lm(formula = log(volume) ~ log(height):age:male + smoker, data = table_2018s1_above6)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.53620 -0.08805  0.01031  0.08931  0.32758
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -9.55944    0.50972 -18.754   <2e-16 ***
## smokerregularly             -0.04161    0.02162  -1.925   0.0547 .
## log(height):age6:malefemale  2.52311    0.12828  19.669   <2e-16 ***
## log(height):age7:malefemale  2.53067    0.12722  19.892   <2e-16 ***
## log(height):age8:malefemale  2.52611    0.12503  20.205   <2e-16 ***
## log(height):age9:malefemale  2.53901    0.12444  20.403   <2e-16 ***
## log(height):age10:malefemale  2.55392    0.12371  20.644   <2e-16 ***
## log(height):age11:malefemale  2.55951    0.12321  20.773   <2e-16 ***
## log(height):age12:malefemale  2.56509    0.12310  20.838   <2e-16 ***
## log(height):age13:malefemale  2.56979    0.12291  20.907   <2e-16 ***
## log(height):age14:malefemale  2.54910    0.12263  20.788   <2e-16 ***
## log(height):age15:malefemale  2.54670    0.12327  20.659   <2e-16 ***
## log(height):age16:malefemale  2.56462    0.12322  20.813   <2e-16 ***
## log(height):age17:malefemale  2.61978    0.12811  20.450   <2e-16 ***
## log(height):age18:malefemale  2.56344    0.12435  20.615   <2e-16 ***
## log(height):age19:malefemale  2.58822    0.12449  20.791   <2e-16 ***
## log(height):age6:malemale    2.53430    0.12861  19.705   <2e-16 ***
## log(height):age7:malemale    2.54115    0.12729  19.964   <2e-16 ***
## log(height):age8:malemale    2.53915    0.12608  20.139   <2e-16 ***
## log(height):age9:malemale    2.54426    0.12419  20.487   <2e-16 ***
## log(height):age10:malemale   2.54385    0.12324  20.642   <2e-16 ***
## log(height):age11:malemale   2.55859    0.12183  21.002   <2e-16 ***
## log(height):age12:malemale   2.56512    0.12138  21.133   <2e-16 ***
## log(height):age13:malemale   2.58523    0.12074  21.412   <2e-16 ***
## log(height):age14:malemale   2.58262    0.12086  21.368   <2e-16 ***
## log(height):age15:malemale   2.60392    0.12106  21.509   <2e-16 ***
## log(height):age16:malemale   2.59467    0.12097  21.449   <2e-16 ***
## log(height):age17:malemale   2.59767    0.12076  21.511   <2e-16 ***
## log(height):age18:malemale   2.60975    0.12237  21.327   <2e-16 ***
## log(height):age19:malemale   2.61631    0.12363  21.163   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1404 on 585 degrees of freedom
## Multiple R-squared:  0.7988, Adjusted R-squared:  0.7888
## F-statistic: 80.08 on 29 and 585 DF,  p-value: < 2.2e-16
```

```
anova(model_2018s1_2)
```

```
## Analysis of Variance Table
##
## Response: log(volume)
##                       Df Sum Sq Mean Sq F value    Pr(>F)
## smoker                 1  3.197  3.1974 162.111 < 2.2e-16 ***
## log(height):age:male  28 42.605  1.5216  77.148 < 2.2e-16 ***
```

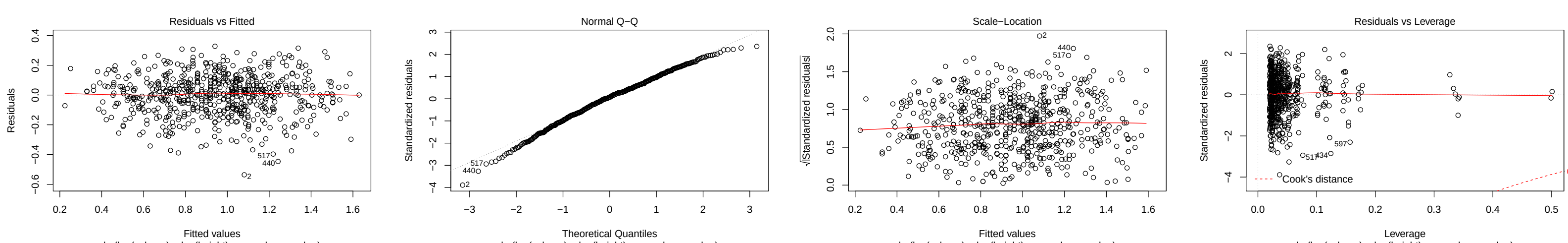
```
## Residuals           585 11.538  0.0197
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
library(olsrr)
ols_regress(model_2018s1_2)
```

```
##                          Model Summary
## ----------------------------------------------------------------
## R                       0.894       RMSE               0.140
## R-Squared               0.799       Coef. Var         14.772
## Adj. R-Squared          0.789       MSE                0.020
## Pred R-Squared           -Inf       MAE                0.108
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                 ANOVA
## ----------------------------------------------------------------------
##                Sum of
##               Squares         DF    Mean Square       F          Sig.
## ----------------------------------------------------------------------
## Regression     45.803         29          1.579    80.078      0.0000
## Residual       11.538        585          0.020
## Total          57.341        614
## ----------------------------------------------------------------------
##
##                                               Parameter Estimates
## ----------------------------------------------------------------------------------------------------------------
##                          model      Beta    Std. Error    Std. Beta         t        Sig       lower      upper
## ----------------------------------------------------------------------------------------------------------------
##                    (Intercept)    -9.559         0.510                  -18.754    0.000     -10.561     -8.558
##                smokerregularly    -0.042         0.022       -0.042      -1.925    0.055      -0.084      0.001
##  log(height):age6:malefemale       2.523         0.128        5.078      19.669    0.000       2.271      2.775
##  log(height):age7:malefemale       2.531         0.127        7.048      19.892    0.000       2.281      2.781
##  log(height):age8:malefemale       2.526         0.125        8.879      20.205    0.000       2.281      2.772
##  log(height):age9:malefemale       2.539         0.124        8.785      20.403    0.000       2.295      2.783
## log(height):age10:malefemale       2.554         0.124        7.886      20.644    0.000       2.311      2.797
## log(height):age11:malefemale       2.560         0.123        9.041      20.773    0.000       2.318      2.802
## log(height):age12:malefemale       2.565         0.123        7.386      20.838    0.000       2.323      2.807
## log(height):age13:malefemale       2.570         0.123        6.778      20.907    0.000       2.328      2.811
## log(height):age14:malefemale       2.549         0.123        4.189      20.788    0.000       2.308      2.790
## log(height):age15:malefemale       2.547         0.123        4.388      20.659    0.000       2.305      2.789
## log(height):age16:malefemale       2.565         0.123        3.442      20.813    0.000       2.323      2.807
## log(height):age17:malefemale       2.620         0.128        1.427      20.450    0.000       2.368      2.871
## log(height):age18:malefemale       2.563         0.124        2.428      20.615    0.000       2.319      2.808
## log(height):age19:malefemale       2.588         0.124        2.020      20.791    0.000       2.344      2.833
##    log(height):age6:malemale       2.534         0.129        6.119      19.705    0.000       2.282      2.787
##    log(height):age7:malemale       2.541         0.127        6.591      19.964    0.000       2.291      2.791
##    log(height):age8:malemale       2.539         0.126        8.200      20.139    0.000       2.292      2.787
##    log(height):age9:malemale       2.544         0.124        9.353      20.487    0.000       2.300      2.788
##   log(height):age10:malemale       2.544         0.123        9.162      20.642    0.000       2.302      2.786
##   log(height):age11:malemale       2.559         0.122        9.139      21.002    0.000       2.319      2.798
##   log(height):age12:malemale       2.565         0.121        7.366      21.133    0.000       2.327      2.804
##   log(height):age13:malemale       2.585         0.121        6.200      21.412    0.000       2.348      2.822
##   log(height):age14:malemale       2.583         0.121        5.695      21.368    0.000       2.345      2.820
##   log(height):age15:malemale       2.604         0.121        4.334      21.509    0.000       2.366      2.842
##   log(height):age16:malemale       2.595         0.121        3.825      21.449    0.000       2.357      2.832
##   log(height):age17:malemale       2.598         0.121        3.833      21.511    0.000       2.361      2.835
##   log(height):age18:malemale       2.610         0.122        2.517      21.327    0.000       2.369      2.850
##   log(height):age19:malemale       2.616         0.124        1.476      21.163    0.000       2.373      2.859
## ----------------------------------------------------------------------------------------------------------------
```

```r
plot(model_2018s1_2)
```

```
## Warning: not plotting observations with leverage one:
##   570, 591
## Warning: not plotting observations with leverage one:
##   570, 591
```



$y = \mu + \beta_1 \ln(H) * Age * Male + \beta_2 Smoker + \varepsilon$

$H_0 : \beta_2 = 0, H_1 : \beta_2 \neq 0$

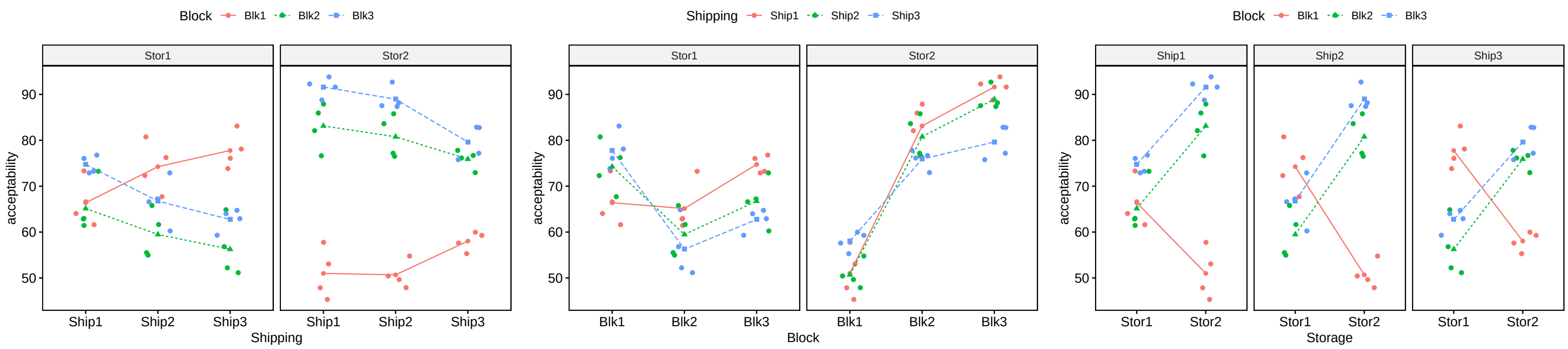
```r
predict(model_2018s1_2, newdata =data.frame(age="16",male="male",smoker="regularly",height=61), interval = "predict
```

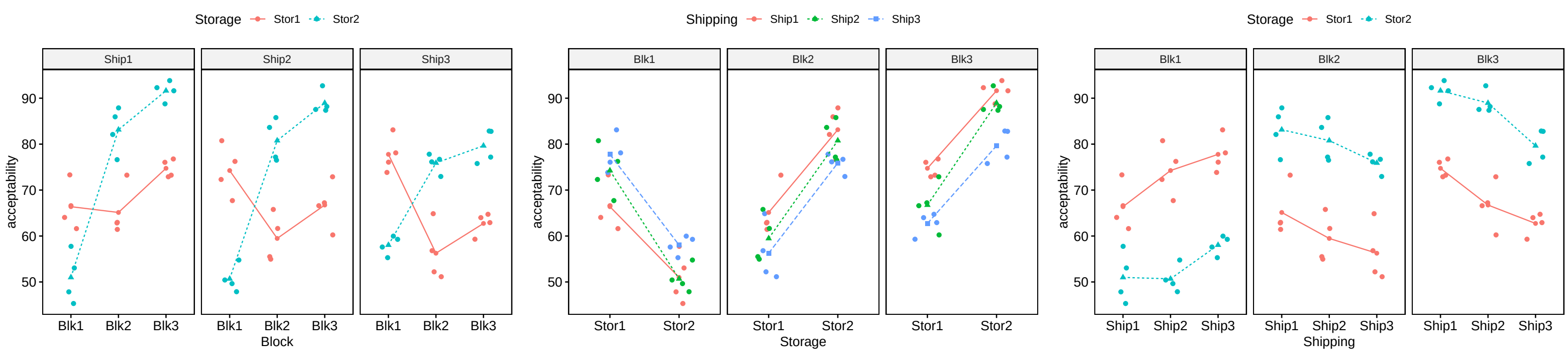
```
##        fit       lwr      upr
## 1 1.065319 0.7691739 1.361463
```

2018S2

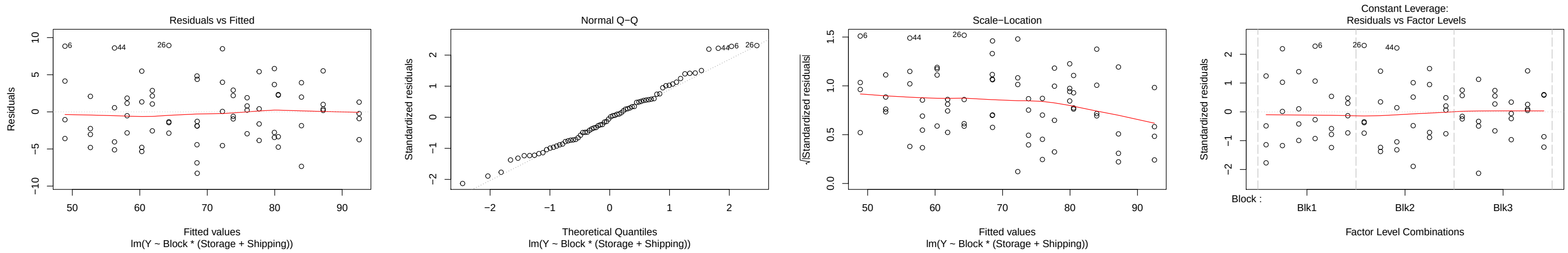
[RCBD]
An experiment is conducted to assess the effect of shipping and storage on the acceptability of avocados. Three shipping methods (labeled 1, 2 and 3) and two storage methods (labeled 1 and 2) were considered. Each combination of shipping x storage was applied to a group of four crates. Additionally, three different shipments were made. The experiment's configuration is shown below. Analyze this experiment. The data set can be found in the file Problem2_Avocado.xlsx.

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    72 obs. of  4 variables:
##  $ Block    : Factor w/ 3 levels "Blk1","Blk2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Shipping: Factor w/ 3 levels "Ship1","Ship2",..: 1 1 1 1 1 1 1 1 2 2 ...
##  $ Storage : Factor w/ 2 levels "Stor1","Stor2": 1 1 1 1 2 2 2 2 1 1 ...
##  $ Y       : num  73.3 66.6 61.6 64 53 ...
```






```
##
## Call:
## lm(formula = Y ~ Block * (Storage + Shipping), data = table_2018s2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2704 -2.8865 -0.0842  2.2082  8.9433
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)              68.476      1.735  39.478  < 2e-16 ***
## BlockBlk2                -4.170      2.453  -1.700  0.09436 .
## BlockBlk3                 5.370      2.453   2.189  0.03248 *
## StorageStor2            -19.573      1.735 -11.284  < 2e-16 ***
## ShippingShip2             3.776      2.124   1.778  0.08054 .
## ShippingShip3             9.217      2.124   4.339 5.58e-05 ***
## BlockBlk2:StorageStor2   39.227      2.453  15.991  < 2e-16 ***
## BlockBlk3:StorageStor2   38.242      2.453  15.589  < 2e-16 ***
## BlockBlk2:ShippingShip2  -7.786      3.004  -2.592  0.01198 *
## BlockBlk3:ShippingShip2  -9.112      3.004  -3.033  0.00357 **
## BlockBlk2:ShippingShip3 -17.272      3.004  -5.749 3.21e-07 ***
## BlockBlk3:ShippingShip3 -21.206      3.004  -7.059 1.99e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.249 on 60 degrees of freedom
## Multiple R-squared:  0.9054, Adjusted R-squared:  0.8881
## F-statistic: 52.23 on 11 and 60 DF,  p-value: < 2.2e-16
## Analysis of Variance Table
##
## Response: Y
##                Df Sum Sq Mean Sq  F value    Pr(>F)
## Block           2 2483.3 1241.65  68.7810 2.975e-16 ***
## Storage         1  703.2  703.19  38.9529 4.841e-08 ***
## Shipping        2  156.3   78.16   4.3297   0.01752 *
## Block:Storage   2 6004.3 3002.13 166.3021 < 2.2e-16 ***
## Block:Shipping  4 1024.0  256.00  14.1809 3.329e-08 ***
## Residuals      60 1083.1   18.05
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```




```
library(emmeans)
Blk_Stor <- pairs(lsmeans(model_2018s2,~Block|Storage))
Stor_Blk <- pairs(lsmeans(model_2018s2,~Storage|Block))
Blk_Ship <- pairs(lsmeans(model_2018s2,~Block|Shipping))
Ship_Blk <- pairs(lsmeans(model_2018s2,~Shipping|Block))
Stor_Ship <- pairs(lsmeans(model_2018s2,~Storage|Shipping))
Ship_Stor <- pairs(lsmeans(model_2018s2,~Shipping|Storage))
library(kableExtra)
kable(test(rbind(Blk_Stor,Stor_Blk),adjust="tukey"),format="latex")%>%kable_styling("condensed",full_width=F,font_s
```

Storage	Block	contrast	estimate	SE	df	t.ratio	p.value
Stor1	.	Blk1 - Blk2	12.522500	1.734563	60	7.219399	0.0000000
Stor1	**.**	**Blk1 - Blk3**	**4.735833**	**1.734563**	**60**	**2.730275**	**0.0561879**
Stor1	.	Blk2 - Blk3	-7.786667	1.734563	60	-4.489124	0.0002805
Stor2	.	Blk1 - Blk2	-26.704167	1.734563	60	-15.395331	0.0000000
Stor2	.	Blk1 - Blk3	-33.505833	1.734563	60	-19.316588	0.0000000
Stor2	.	Blk2 - Blk3	-6.801667	1.734563	60	-3.921257	0.0018712
.	Blk1	Stor1 - Stor2	19.572500	1.734563	60	11.283824	0.0000000
.	Blk2	Stor1 - Stor2	-19.654167	1.734563	60	-11.330906	0.0000000
.	Blk3	Stor1 - Stor2	-18.669167	1.734563	60	-10.763039	0.0000000

```
kable(test(rbind(Blk_Ship,Ship_Blk),adjust="tukey"),format="latex")%>%kable_styling("condensed",full_width=F,font_s
```

Shipping	Block	contrast	estimate	SE	df	t.ratio	p.value
Ship1	.	Blk1 - Blk2	-15.44375	2.124397	60	-7.2697107	0.0000000
Ship1	.	Blk1 - Blk3	-24.49125	2.124397	60	-11.5285668	0.0000000
Ship1	.	Blk2 - Blk3	-9.04750	2.124397	60	-4.2588560	0.0011835
Ship2	.	Blk1 - Blk2	-7.65750	2.124397	60	-3.6045526	0.0094034
Ship2	.	Blk1 - Blk3	-15.37875	2.124397	60	-7.2391138	0.0000000
Ship2	.	Blk2 - Blk3	-7.72125	2.124397	60	-3.6345612	0.0085960
Ship3	**.**	**Blk1 - Blk2**	**1.82875**	**2.124397**	**60**	**0.8608326**	**0.9679465**
Ship3	**.**	**Blk1 - Blk3**	**-3.28500**	**2.124397**	**60**	**-1.5463213**	**0.6800338**
Ship3	**.**	**Blk2 - Blk3**	**-5.11375**	**2.124397**	**60**	**-2.4071539**	**0.1928931**
.	Blk1	Ship1 - Ship2	-3.77625	2.124397	60	-1.7775634	0.5309345
.	**Blk1**	**Ship1 - Ship3**	**-9.21750**	**2.124397**	**60**	**-4.3388788**	**0.0009051**
.	Blk1	Ship2 - Ship3	-5.44125	2.124397	60	-2.5613153	0.1405152
.	Blk2	Ship1 - Ship2	4.01000	2.124397	60	1.8875947	0.4606427
.	**Blk2**	**Ship1 - Ship3**	**8.05500**	**2.124397**	**60**	**3.7916646**	**0.0053248**
.	Blk2	Ship2 - Ship3	4.04500	2.124397	60	1.9040699	0.4503529
.	Blk3	Ship1 - Ship2	5.33625	2.124397	60	2.5118895	0.1559587
.	**Blk3**	**Ship1 - Ship3**	**11.98875**	**2.124397**	**60**	**5.6433667**	**0.0000083**
.	**Blk3**	**Ship2 - Ship3**	**6.65250**	**2.124397**	**60**	**3.1314772**	**0.0357208**

```
kable(test(rbind(Stor_Ship,Ship_Stor),adjust="tukey"),format="latex")%>%kable_styling("condensed",full_width=F,font_
```

Shipping	Storage	contrast	estimate	SE	df	t.ratio	p.value
Ship1	**.**	**Stor1 - Stor2**	**-6.250278**	**1.001450**	**60**	**-6.241226**	**0.0000004**
Ship2	**.**	**Stor1 - Stor2**	**-6.250278**	**1.001450**	**60**	**-6.241226**	**0.0000004**
Ship3	**.**	**Stor1 - Stor2**	**-6.250278**	**1.001450**	**60**	**-6.241226**	**0.0000004**
.	Stor1	Ship1 - Ship2	1.856667	1.226521	60	1.513767	0.5320163
.	**Stor1**	**Ship1 - Ship3**	**3.608750**	**1.226521**	**60**	**2.942265**	**0.0329359**
.	Stor1	Ship2 - Ship3	1.752083	1.226521	60	1.428498	0.5860213
.	Stor2	Ship1 - Ship2	1.856667	1.226521	60	1.513767	0.5320163
.	**Stor2**	**Ship1 - Ship3**	**3.608750**	**1.226521**	**60**	**2.942265**	**0.0329359**
.	Stor2	Ship2 - Ship3	1.752083	1.226521	60	1.428498	0.5860213

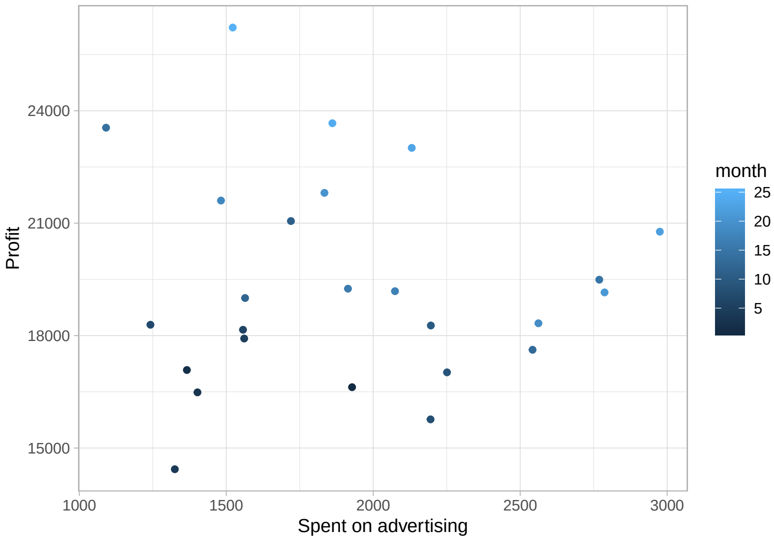
2018F

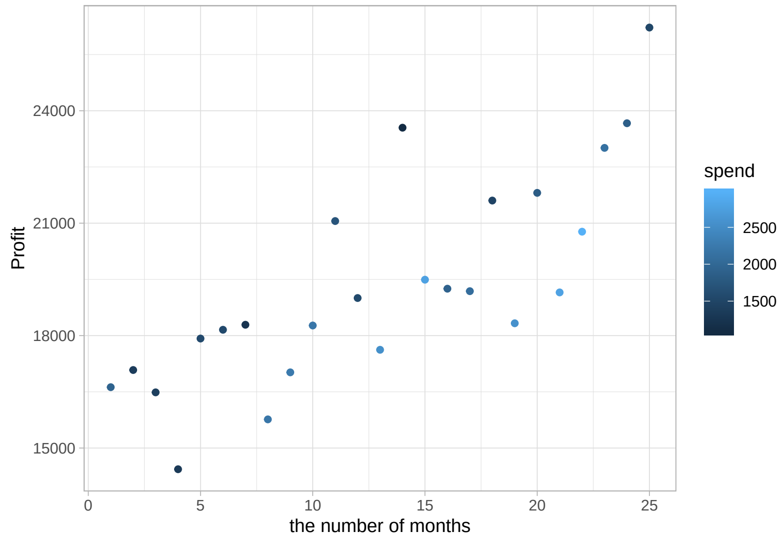
Robert Fountain*, Daniel Taylor-Rodriguez

2018F1

2015F1 [2017S1]
Find the best model for predicting Y based on X1 and X2. Y is the amount of profit that a company makes in a month. X1 is the number of months that the company has been in business. X2 is the amount spent on advertising. Consider as predictors all possible linear and quadratic terms ($X1, X1^2, X2, X2^2$, and $X1X2$). Consider possible transformations of Y. Include all appropriate diagnostics. When you have found your “best” model, predict a new Y when X1 = 20 and X2 = $1,900, giving a 95% prediction interval. The data set, shown below, appears in “Profits.xlsx”.

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    25 obs. of  3 variables:
##  $ X1: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ X2: num  1928 1366 1402 1325 1561 ...
##  $ Y : num  16624 17082 16486 14435 17922 ...
```






```r
model_2018f1 <- lm(Y~X1/X2, table_2018f1)
summary(model_2018f1)
```

```
##
## Call:
## lm(formula = Y ~ X1/X2, data = table_2018f1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2404.22  -904.25    51.59   918.49  1821.40
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 15080.0896   517.9820  29.113  < 2e-16 ***
## X1            647.7910    79.5268   8.146 4.37e-08 ***
## X1:X2          -0.1570     0.0322  -4.875 7.13e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1241 on 22 degrees of freedom
## Multiple R-squared:  0.818,  Adjusted R-squared:  0.8014
## F-statistic: 49.43 on 2 and 22 DF,  p-value: 7.266e-09
```

```r
library(olsrr)
ols_regress(model_2018f1)
```

```
##                          Model Summary
## ------------------------------------------------------------------
## R                       0.904       RMSE                 1240.942
## R-Squared               0.818       Coef. Var               6.413
## Adj. R-Squared          0.801       MSE               1539937.532
## Pred R-Squared          0.754       MAE                   990.759
## ------------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                 ANOVA
## -------------------------------------------------------------------------
##                    Sum of
##                   Squares        DF     Mean Square      F         Sig.
## -------------------------------------------------------------------------
## Regression    152247073.255       2     76123536.628    49.433    0.0000
## Residual       33878625.705      22      1539937.532
## Total         186125698.960      24
## -------------------------------------------------------------------------
##
##                                      Parameter Estimates
## ----------------------------------------------------------------------------------------------
##       model          Beta    Std. Error    Std. Beta       t        Sig        lower       upper
## ----------------------------------------------------------------------------------------------
## (Intercept)     15080.090       517.982                 29.113    0.000    14005.861   16154.318
##          X1       647.791        79.527        1.712     8.146    0.000      482.863     812.720
##       X1:X2        -0.157         0.032       -1.025    -4.875    0.000       -0.224      -0.090
## ----------------------------------------------------------------------------------------------
```

```r
car::Anova(model_2018f1)
```

```
## Anova Table (Type II tests)
##
## Response: Y
##              Sum Sq Df F value    Pr(>F)
## X1        115643157  1  75.096 1.527e-08 ***
## X1:X2      36603916  1  23.770 7.127e-05 ***
## Residuals  33878626 22
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

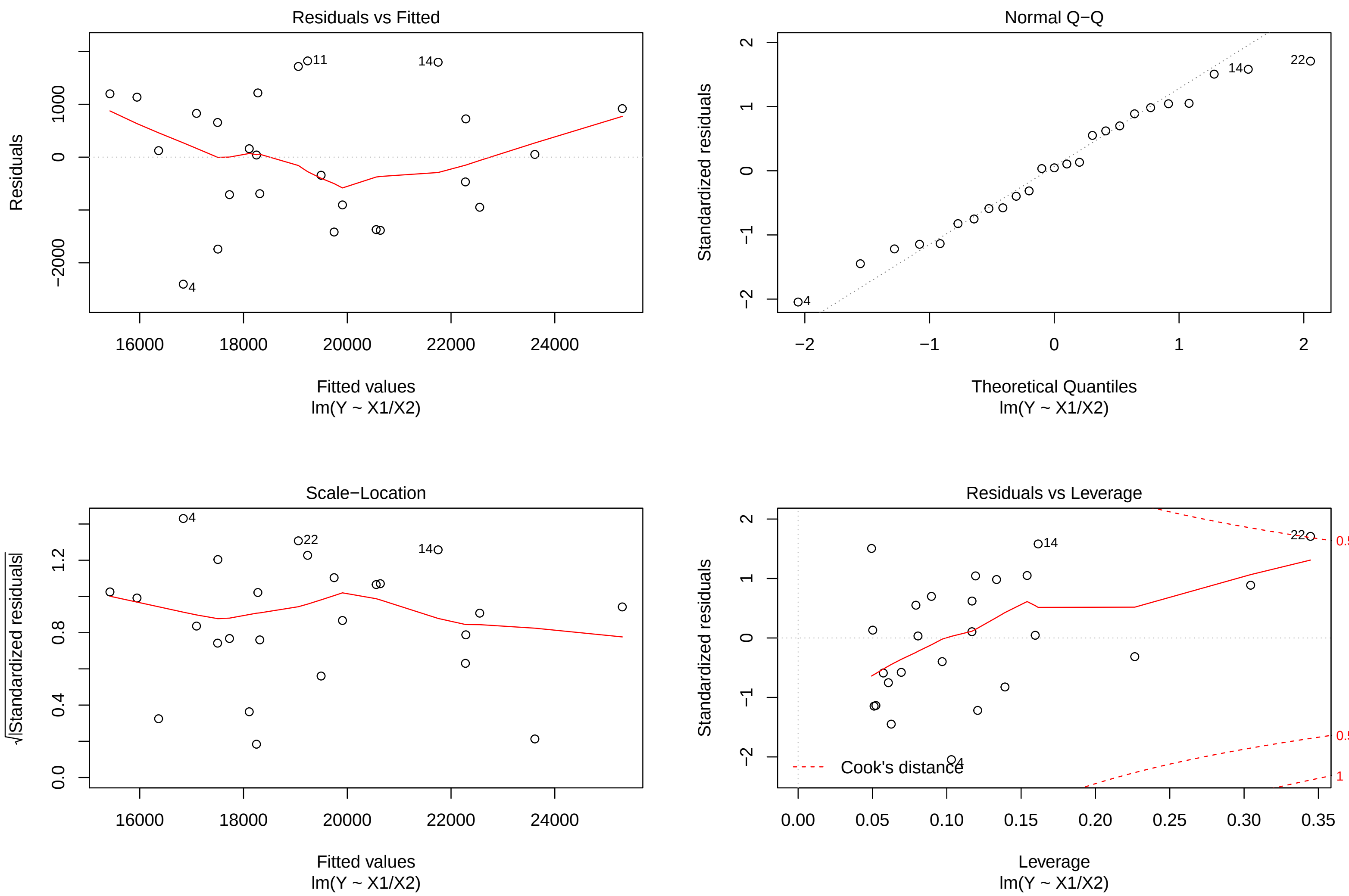
```r
car::vif(model_2018f1)
```

```
##       X1    X1:X2
## 5.339091 5.339091
```

```r
anova(model_2018f1)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df    Sum Sq   Mean Sq F value    Pr(>F)
## X1         1 115643157 115643157  75.096 1.527e-08 ***
## X1:X2      1  36603916  36603916  23.770 7.127e-05 ***
## Residuals 22  33878626   1539938
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(model_2018f1)
```




```
predict(model_2018f1, newdata = data.frame(X1=20,X2=1900), interval = "prediction", level=0.95)
```

```
##        fit      lwr      upr
## 1 22070.41 19385.27 24755.55
```

2018F2

2015F2 [7.4] [8.E.10]
A replicated fractional factorial design is used to investigate the effect of four factors on the free height of leaf springs used in an automotive application. The factors are (A) furnace temperature, (B) heating time, (C) transfer time, and (D) hold down time. There are 3 observations at each setting.
Write out the alias structure for this design. What is the resolution of this design?
I=ABCD, AB=CD, AC=BD, BC=AD; A=BCD, B=ACD, C=ABD, D=ABC; III
Analyze the data. What factors influence the mean free height? The data set appears in the file "Springs.xlsx".
A, B

```
table_2018f2 <- readxl::read_xlsx("qe_lab/Springs_2018f.xlsx")
table_2018f2 <- table_2018f2[order(table_2018f2$D ,table_2018f2$C ,table_2018f2$B,table_2018f2$A),]
str(table_2018f2)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    48 obs. of  5 variables:
##  $ A      : num  -1 -1 -1 -1 -1 -1 1 1 1 1 ...
##  $ B      : num  -1 -1 -1 -1 -1 -1 1 1 1 1 ...
##  $ C      : num  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ D      : num  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ Heights: num  8.56 8 8.56 7.5 8.62 7.24 8.18 8.26 8.12 8.5 ...
```

```
kableExtra::kable(table_2018f2)
```

A	B	C	D	Heights
-1	-1	-1	-1	8.56
-1	-1	-1	-1	8.00
-1	-1	-1	-1	8.56
-1	-1	-1	-1	7.50
-1	-1	-1	-1	8.62
-1	-1	-1	-1	7.24
1	1	-1	-1	8.18
1	1	-1	-1	8.26
1	1	-1	-1	8.12
1	1	-1	-1	8.50
1	1	-1	-1	8.50
1	1	-1	-1	8.12
1	-1	1	-1	8.38
1	-1	1	-1	8.12
1	-1	1	-1	9.18
1	-1	1	-1	8.38
1	-1	1	-1	9.12
1	-1	1	-1	8.24
-1	1	1	-1	8.12
-1	1	1	-1	7.36
-1	1	1	-1	8.04
-1	1	1	-1	7.36
-1	1	1	-1	7.88
-1	1	1	-1	7.50
1	-1	-1	1	9.30
1	-1	-1	1	8.76
1	-1	-1	1	9.36
1	-1	-1	1	8.76
1	-1	-1	1	8.76
1	-1	-1	1	7.88
-1	1	-1	1	8.00
-1	1	-1	1	8.00
-1	1	-1	1	8.12
-1	1	-1	1	8.12
-1	1	-1	1	8.00
-1	1	-1	1	8.00
-1	-1	1	1	8.08
-1	-1	1	1	7.64
-1	-1	1	1	9.00
-1	-1	1	1	7.88
-1	-1	1	1	8.76
-1	-1	1	1	7.88
1	1	1	1	8.12
1	1	1	1	8.62
1	1	1	1	8.62
1	1	1	1	8.00
1	1	1	1	8.38
1	1	1	1	8.18

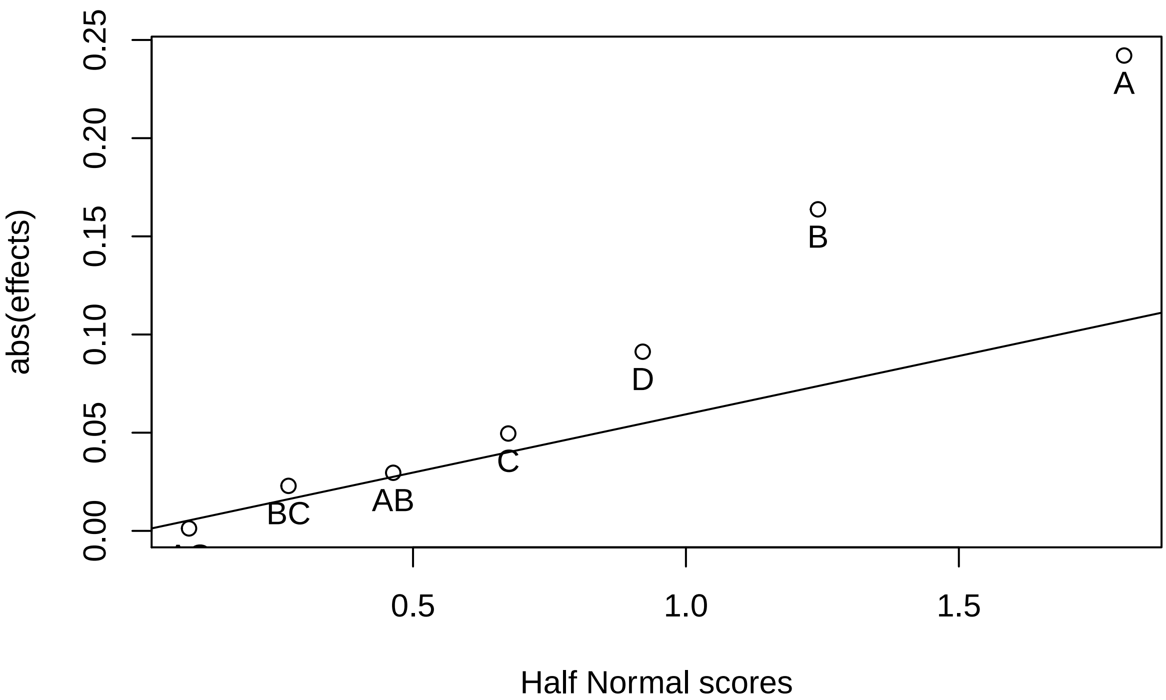
```r
model_2018f2 <- aov(Heights~A*B*C*D, table_2018f2)
model_2018f2$coefficients
```

```
## (Intercept)           A           B           C           D         A:B
##  8.25125000  0.24208333 -0.16375000 -0.04958333  0.09125000 -0.02958333
##         A:C         B:C         A:D         B:D         C:D       A:B:C
##  0.00125000 -0.02291667          NA          NA          NA          NA
##       A:B:D       A:C:D       B:C:D     A:B:C:D
##          NA          NA          NA          NA
```

```r
library(daewr)
```

```
## Registered S3 method overwritten by 'partitions':
##   method            from
##   print.equivalence lava
## Registered S3 method overwritten by 'DoE.base':
##   method           from
##   factorize.factor conf.design
##
## Attaching package: 'daewr'
## The following object is masked from 'package:olsrr':
##
##     cement
## The following object is masked from 'package:lme4':
##
##     cake
## The following object is masked from 'package:magrittr':
##
##     mod
```

```r
halfnorm(model_2018f2$coefficients[2:8],alpha=1)
```




```
## zscore= 0.08964235 0.27188 0.4637078 0.6744898 0.920823 1.241867 1.802743effp= 0.00125 0.02291667 0.02958333 0.04
```

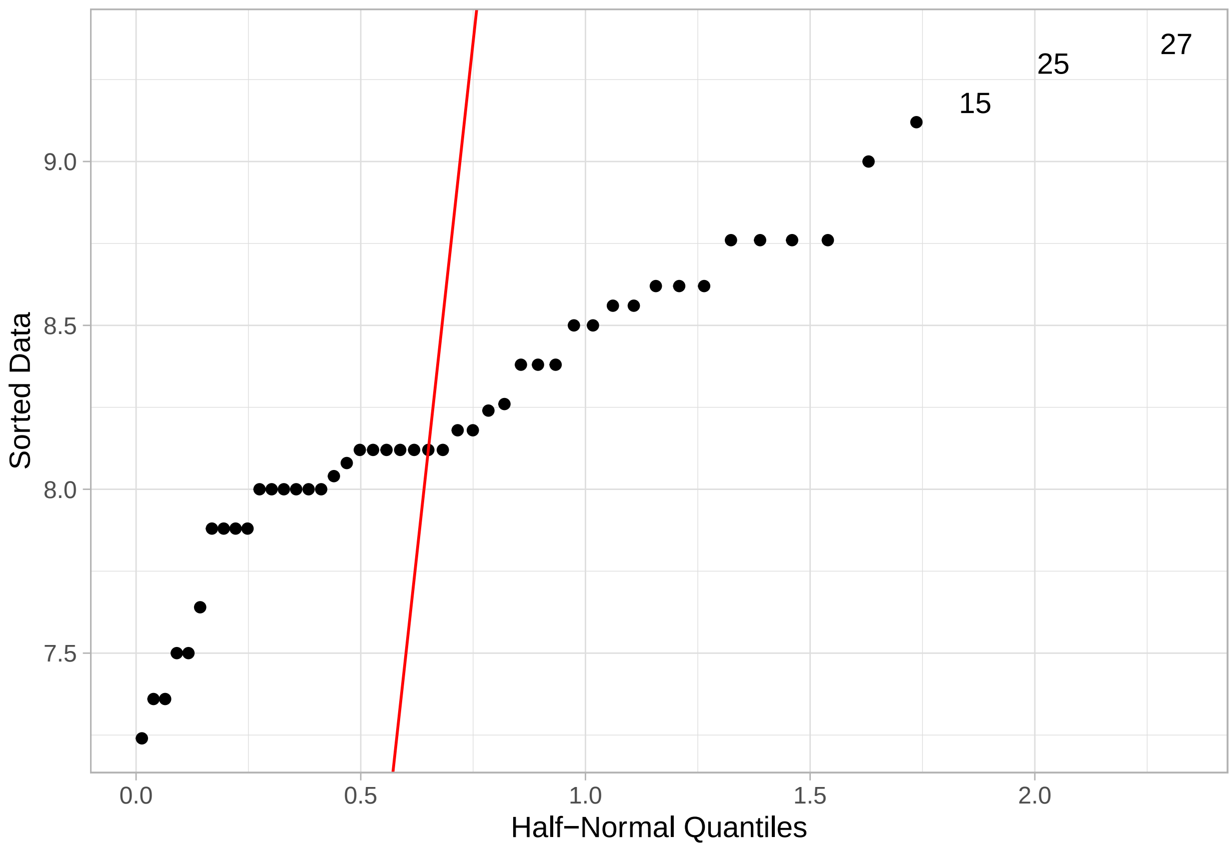
```r
summary(model_2018f2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)    
## A            1  2.813  2.8130  16.359 0.000232 ***
## B            1  1.287  1.2871   7.485 0.009232 ** 
## C            1  0.118  0.1180   0.686 0.412346    
## D            1  0.400  0.3997   2.324 0.135232    
## A:B          1  0.042  0.0420   0.244 0.623819    
## A:C          1  0.000  0.0001   0.000 0.983441    
## B:C          1  0.025  0.0252   0.147 0.703832    
## Residuals   40  6.878  0.1720                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
model_2018f2_2 <- aov(Heights~A+B, table_2018f2)
summary(model_2018f2_2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)    
## A            1  2.813  2.8130  16.962 0.000161 ***
## B            1  1.287  1.2871   7.761 0.007788 ** 
## Residuals   45  7.463  0.1658                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
library(gghalfnorm)
gghalfnorm(x = table_2018f2$Heights, nlab = 3)+ ggplot2::theme_light()
```



2019S

Robert Fountain*, Daniel Taylor-Rodriguez
Instructions:

1. Two 8.5" x 11" pages of notes (front and back) are allowed.
2. Perform the statistical analysis in your software of preference for the two problems below. The data sets for each problem are on the flash drive provided. Create a word or pdf document with your findings. Save the document to the flash drive provided with your name as the file name. You may use scratch paper during the exam, but everything you want considered for grading must be included in your document. Additionally, you must copy and paste the code used for the analysis at the end of the word/pdf document you submit.
3. For each question discuss all relevant aspects of your analysis (exploratory and modeling) supporting them with graphical and numerical summaries that are important for communicating results. It should also include a discussion of diagnostics and model adequacy, and rationale for any transformations or other key modeling decisions. The report should include interpretations of the output, written so that a statistically literate person can understand and apply the findings in each case.

2019S1

[4.2.1 PRESS residuals]
The goal of this exercise is to find the best model for predicting (out-of-sample) Y based on the continuous variables x1, x2, x3, and on the binary variables A and B. The data set is in the dataset "ModelBuildingData.xlsx". Consider possible transformations of Y, and for the linear predictor consider 2-way interactions and quadratic terms. Include all appropriate diagnostics, and make any necessary adjustments to the data so model assumptions are met.
Use only the first 250 observations for model training model (i.e., selection, fitting and diagnostics). With your top model, obtain predictions for all 250 remaining observations (the hold-out samples), and their corresponding 95% predictive intervals. Finally, calculate and interpret (in term of the model predictive ability) the Prediction Root Mean Square Error (PRMSE), as follows:

```
table_2019s1 <- readxl::read_xlsx("qe_lab/ModelBuildingData.xlsx")
str(table_2019s1)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    500 obs. of  6 variables:
##  $ y : num  0.858 1.07 0.782 1.195 1.065 ...
##  $ x1: num  -0.469 0.679 -1.802 -0.386 0.576 ...
##  $ x2: num  -1.903 -0.615 0.64 -1.82 -0.401 ...
##  $ x3: num  0.595 -1.845 -0.521 0.55 -1.775 ...
##  $ A : chr  "a1" "a1" "a1" "a1" ...
##  $ B : chr  "b1" "b1" "b1" "b2" ...
```

```
dplyr::glimpse(table_2019s1)
```

```
## Observations: 500
## Variables: 6
## $ y  <dbl> 0.8575377, 1.0700376, 0.7816973, 1.1954874, 1.0648021, 0.75...
## $ x1 <dbl> -0.4686792, 0.6794381, -1.8021745, -0.3855567, 0.5755118, -...
## $ x2 <dbl> -1.9030527, -0.6147600, 0.6401392, -1.8199357, -0.4007813, ...
## $ x3 <dbl> 0.5945833, -1.8454583, -0.5208516, 0.5502561, -1.7749336, -...
## $ A  <chr> "a1", "a1", "a1", "a1", "a1", "a1", "a1", "a1", "a2", "a1",...
```

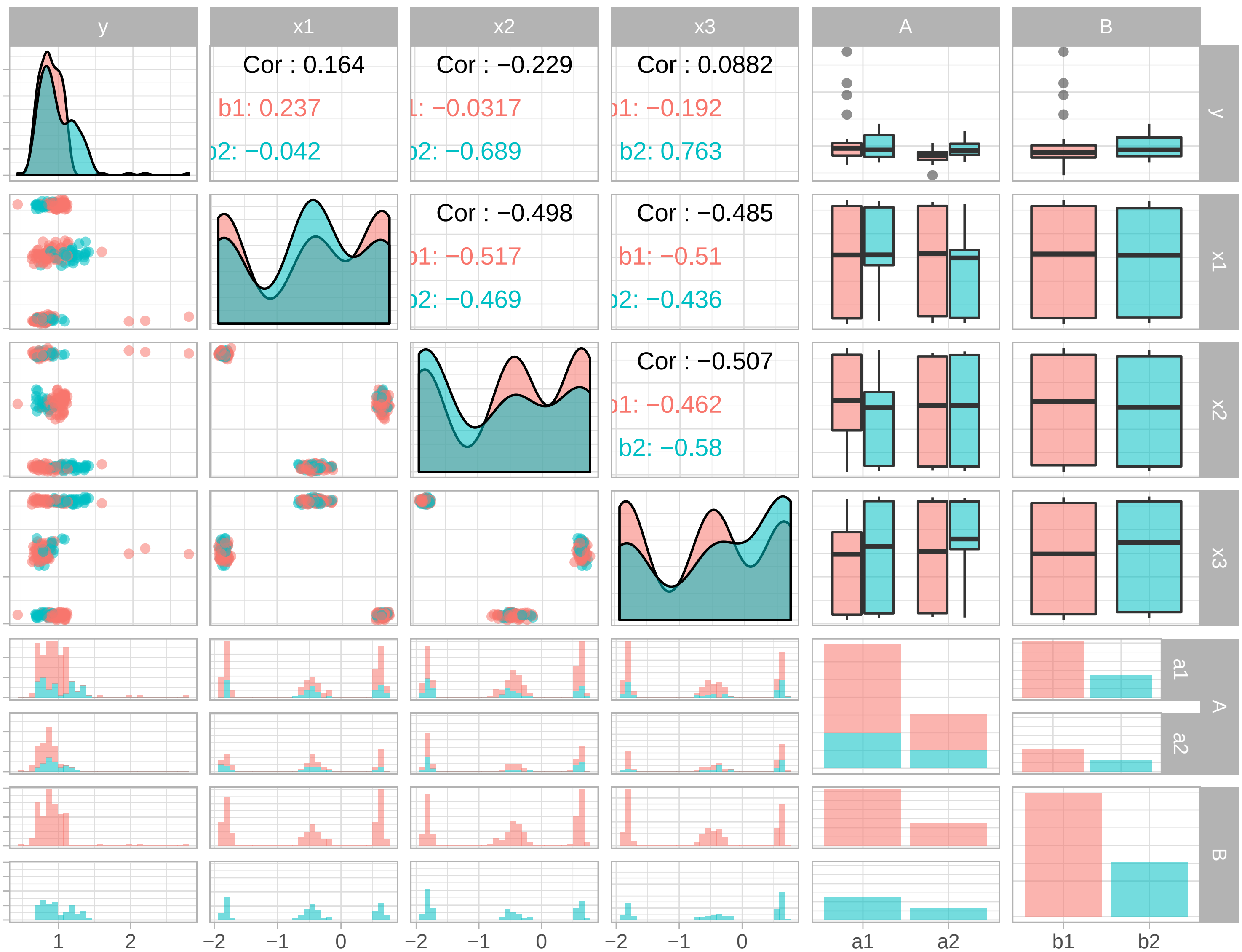
```
## $ B  <chr> "b1", "b1", "b1", "b2", "b1", "b1", "b2", "b2", "b2", "b2",...
```

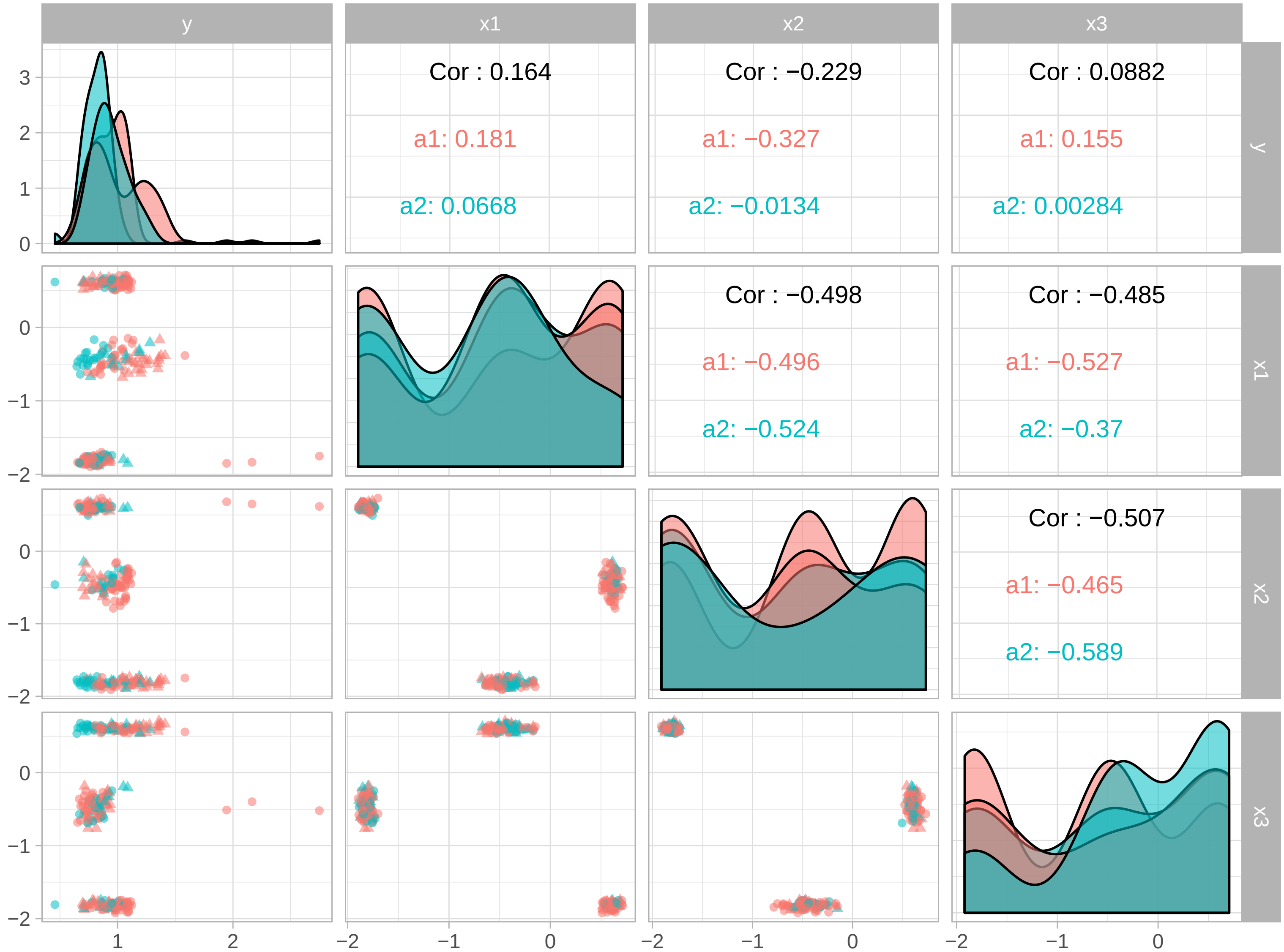
```r
table_2019s1_250 <- table_2019s1[1:250,]
table_2019s1_500 <- table_2019s1[251:500,]
str(table_2019s1_250)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    250 obs. of  6 variables:
##  $ y : num  0.858 1.07 0.782 1.195 1.065 ...
##  $ x1: num  -0.469 0.679 -1.802 -0.386 0.576 ...
##  $ x2: num  -1.903 -0.615 0.64 -1.82 -0.401 ...
##  $ x3: num  0.595 -1.845 -0.521 0.55 -1.775 ...
##  $ A : chr  "a1" "a1" "a1" "a1" ...
##  $ B : chr  "b1" "b1" "b1" "b2" ...
```

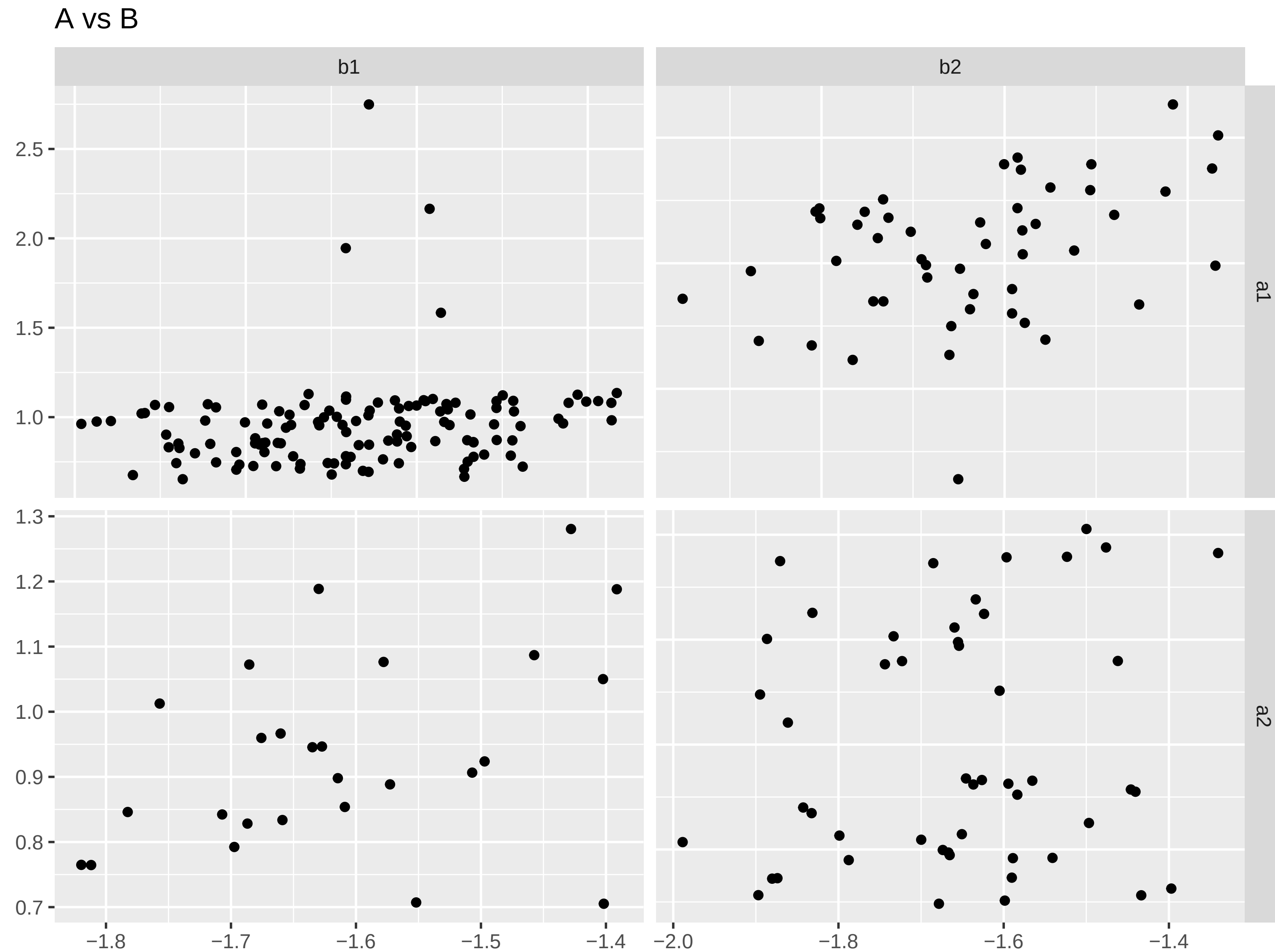
```r
str(table_2019s1_500)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    250 obs. of  6 variables:
##  $ y : num  0.86 0.91 0.998 0.979 0.803 ...
##  $ x1: num  0.635 -1.764 -0.472 0.585 -1.755 ...
##  $ x2: num  -0.588 0.552 -1.822 -0.352 0.599 ...
##  $ x3: num  -1.809 -0.377 0.578 -1.861 -0.602 ...
##  $ A : chr  "a1" "a1" "a1" "a2" ...
##  $ B : chr  "b1" "b2" "b1" "b1" ...
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

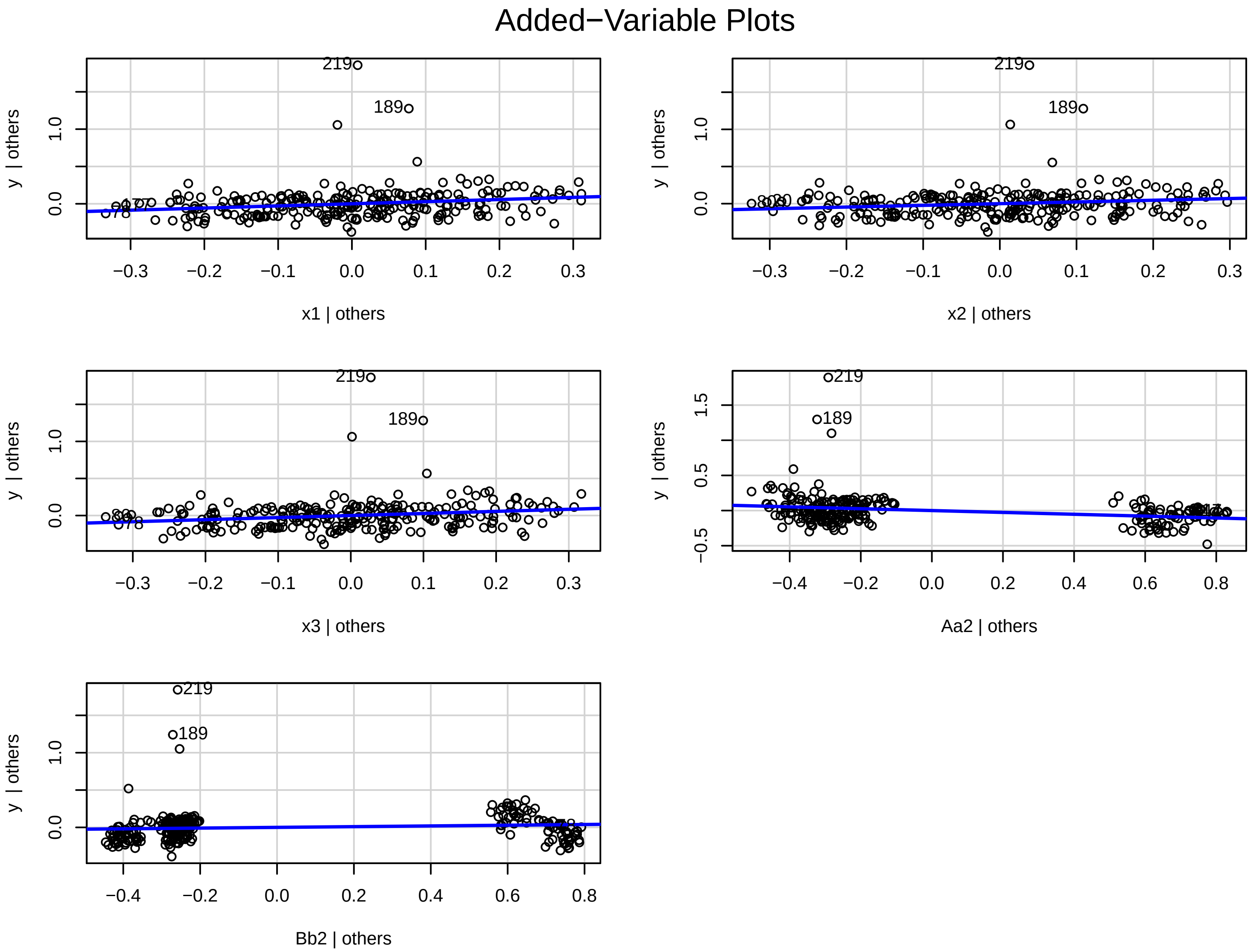


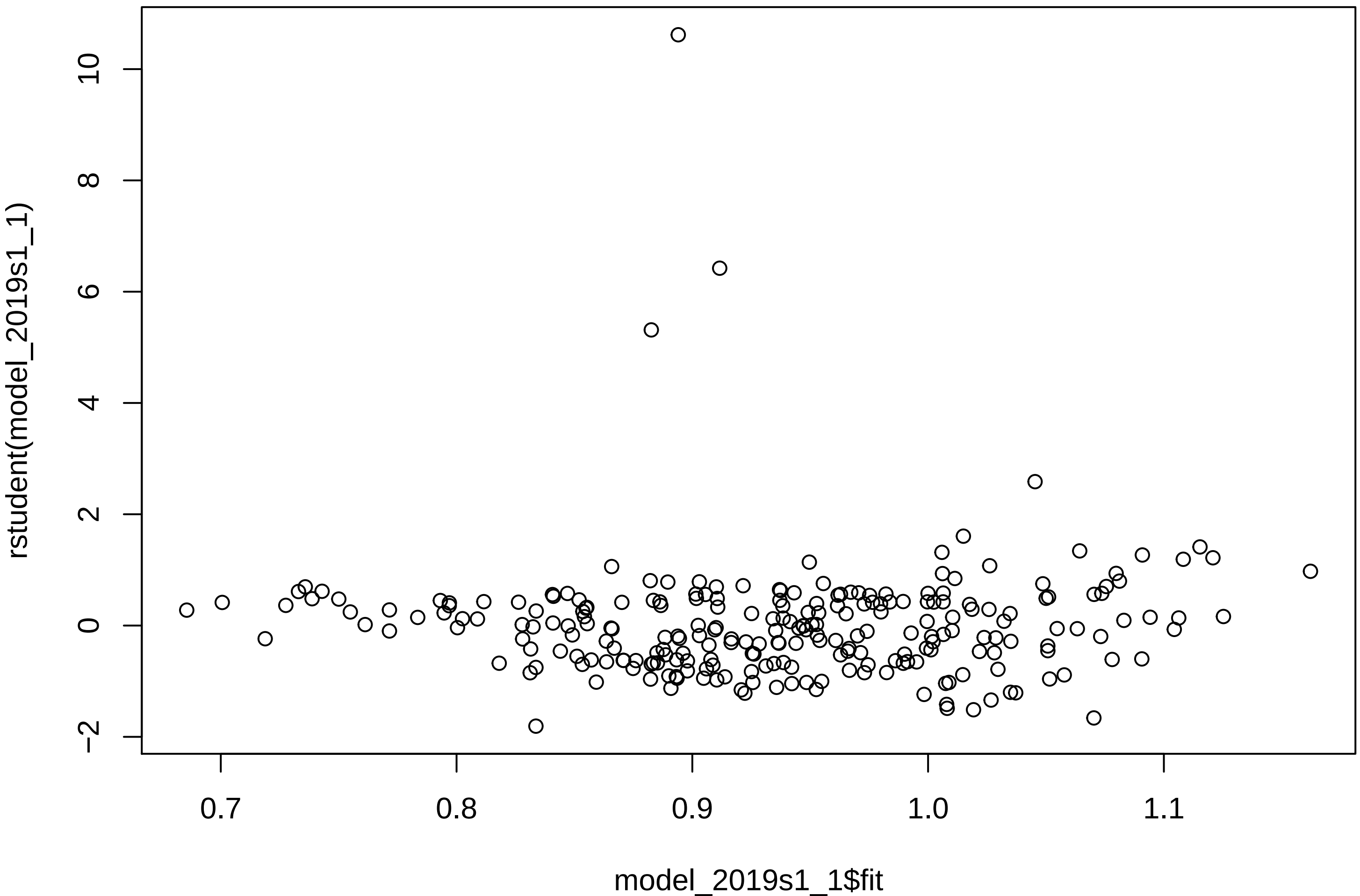


y
x1
x2
x3
Cor : 0.164
a1: 0.181
a2: 0.0668
Cor : −0.229
a1: −0.327
a2: −0.0134
Cor : 0.0882
a1: 0.155
a2: 0.00284
Cor : −0.498
a1: −0.496
a2: −0.524
Cor : −0.485
a1: −0.527
a2: −0.37
Cor : −0.507
a1: −0.465
a2: −0.589

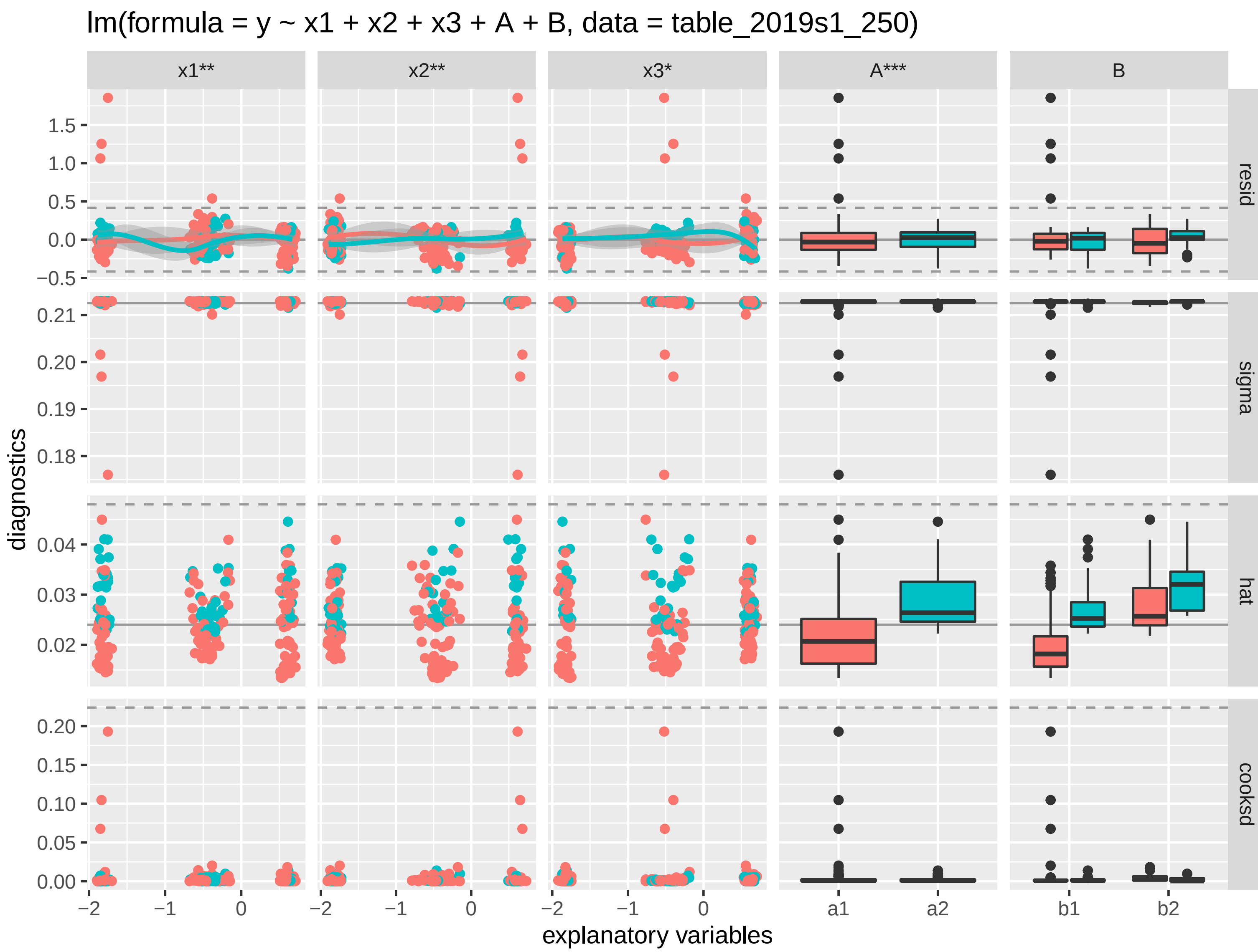



```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + A + B, data = table_2019s1_250)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37764 -0.12857 -0.01063  0.08906  1.85501
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.39962    0.15936   8.783 2.87e-16 ***
## x1           0.28627    0.09490   3.017  0.00283 **
## x2           0.22936    0.09579   2.394  0.01740 *
## x3           0.27945    0.09504   2.940  0.00359 **
## Aa2         -0.13200    0.02971  -4.443 1.35e-05 ***
## Bb2          0.04808    0.02956   1.627  0.10513
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2125 on 244 degrees of freedom
## Multiple R-squared:  0.1533, Adjusted R-squared:  0.1359
## F-statistic: 8.835 on 5 and 244 DF,  p-value: 1.007e-07
## Anova Table (Type II tests)
##
## Response: y
##            Sum Sq  Df F value    Pr(>F)
## x1         0.4110   1  9.0998  0.002827 **
## x2         0.2590   1  5.7330  0.017405 *
## x3         0.3905   1  8.6459  0.003593 **
## A          0.8916   1 19.7399 1.347e-05 ***
## B          0.1195   1  2.6456  0.105129
## Residuals 11.0213 244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##        x1        x2        x3         A         B
## 48.920081 50.351228 49.831957  1.033647  1.023466
```


Added−Variable Plots
219
189
y | others
x1 | others
x2 | others
x3 | others
Aa2 | others
Bb2 | others
−0.3 −0.2 −0.1 0.0 0.1 0.2 0.3
0.0 1.0
−0.5 0.5 1.5
−0.4 −0.2 0.0 0.2 0.4 0.6 0.8



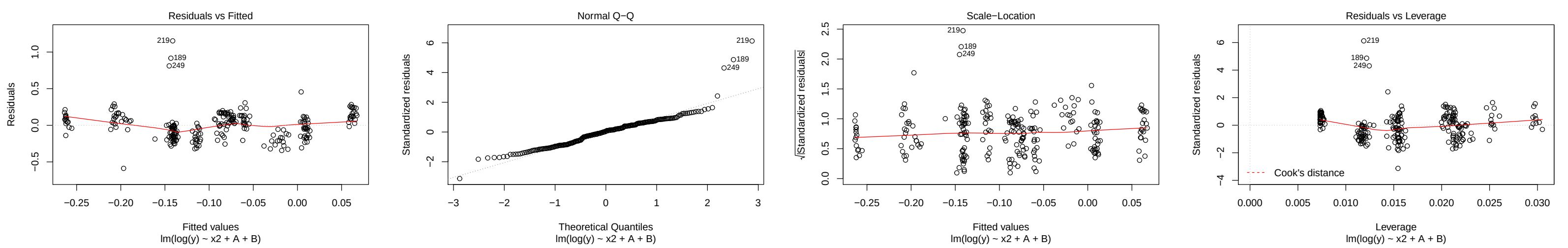

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

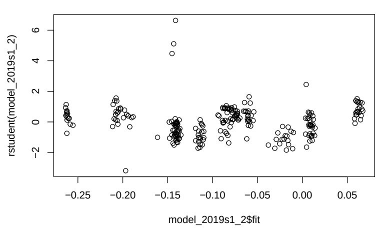


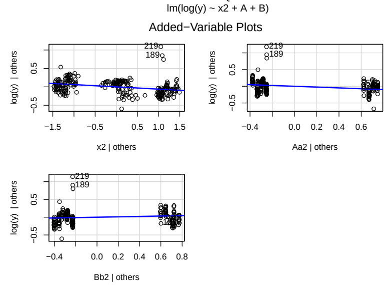

```
library(olsrr)
# ols_plot_diagnostics(model_2019s1_1)
ols_step_both_aic(model_2019s1_1)
ols_step_both_p(lm(table_2019s1_250,formula=log(y)~ x1+x2+x3+A+B))
```

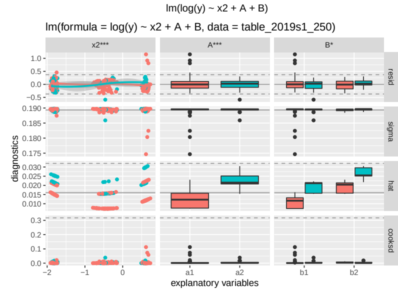
```
##
## Call:
## lm(formula = log(y) ~ x2 + A + B, data = table_2019s1_250)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.58842 -0.13906  0.00879  0.11250  1.15248
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.10330    0.01704  -6.064 4.97e-09 ***
## x2          -0.06149    0.01214  -5.066 7.96e-07 ***
## Aa2         -0.12183    0.02614  -4.661 5.15e-06 ***
## Bb2          0.05454    0.02624   2.079   0.0387 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1893 on 246 degrees of freedom
## Multiple R-squared:  0.1695, Adjusted R-squared:  0.1594
## F-statistic: 16.74 on 3 and 246 DF,  p-value: 6.312e-10
## Anova Table (Type II tests)
##
## Response: log(y)
##           Sum Sq  Df F value    Pr(>F)
## x2        0.9202   1 25.6682 7.965e-07 ***
## A         0.7789   1 21.7278 5.154e-06 ***
## B         0.1549   1  4.3212   0.03868 *
## Residuals 8.8191 246
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##       x2        A        B
## 1.018554 1.007957 1.015740
##                   48.8 %      51.2 %
## (Intercept) -0.10383914 -0.10277031
## x2          -0.06187532 -0.06111380
## Aa2         -0.12265219 -0.12101237
```

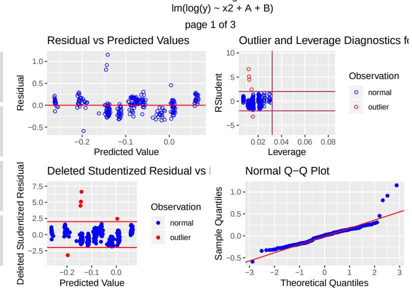
```
## Bb2          0.05371814  0.05536428
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## Analysis of Variance Table
##
## Response: log(y)
##            Df Sum Sq Mean Sq F value    Pr(>F)
## x2          1 0.8964 0.89643 25.0051 1.088e-06 ***
## A           1 0.7486 0.74864 20.8825 7.733e-06 ***
## B           1 0.1549 0.15491  4.3212   0.03868 *
## Residuals 246 8.8191 0.03585
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

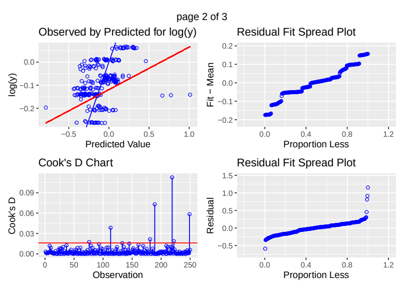


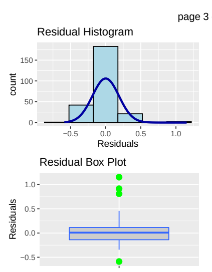













```r
model_2019s1_3 <- lm(table_2019s1_500,formula=log(y)~ x2+A+B)
summary(model_2019s1_3)
```

```
##
## Call:
## lm(formula = log(y) ~ x2 + A + B, data = table_2019s1_500)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34814 -0.10156 -0.00912  0.09695  0.32199
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.137812   0.012796 -10.770  < 2e-16 ***
## x2          -0.074053   0.008689  -8.523 1.59e-15 ***
## Aa2         -0.098649   0.019104  -5.164 5.00e-07 ***
## Bb2          0.055665   0.018112   3.073  0.00235 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1357 on 246 degrees of freedom
## Multiple R-squared:  0.3167, Adjusted R-squared:  0.3084
## F-statistic: 38.01 on 3 and 246 DF,  p-value: < 2.2e-16
```

```r
ols_regress(log(y)~ x2+A+B, data = table_2019s1_500)
```

```
##                         Model Summary
## ---------------------------------------------------------------
## R                       0.563       RMSE                 0.136
## R-Squared               0.317       Coef. Var         -129.304
## Adj. R-Squared          0.308       MSE                  0.018
## Pred R-Squared          0.294       MAE                  0.111
## ---------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                               ANOVA
## -------------------------------------------------------------------
##                 Sum of
##                Squares        DF    Mean Square      F         Sig.
## -------------------------------------------------------------------
## Regression       2.099         3          0.700    38.007    0.0000
## Residual         4.528       246          0.018
## Total            6.627       249
```

```
## -----------------------------------------------------------------
##
##                              Parameter Estimates
## -------------------------------------------------------------------------------------------
##       model      Beta    Std. Error    Std. Beta        t        Sig      lower     upper
## -------------------------------------------------------------------------------------------
## (Intercept)    -0.138        0.013                  -10.770    0.000    -0.163    -0.113
##          x2    -0.074        0.009       -0.452      -8.523    0.000    -0.091    -0.057
##         Aa2    -0.099        0.019       -0.273      -5.164    0.000    -0.136    -0.061
##         Bb2     0.056        0.018        0.163       3.073    0.002     0.020     0.091
## -------------------------------------------------------------------------------------------
```

```
library(Metrics)
Metrics::rmse(table_2019s1_500$y,exp(predict(model_2019s1_2,table_2019s1_500)))
```

```
## [1] 0.1285634
```

```
ols_press(model_2019s1_3)
```

```
## [1] 4.681989
```

```
MPV::PRESS(model_2019s1_3)
```

```
## [1] 4.681989
```

```
sum((residuals(model_2019s1_3)/(1 - lm.influence(model_2019s1_3)$hat))^2)
```

```
## [1] 4.681989
```

```
ols_pred_rsq(model_2019s1_3)
```

```
## [1] 0.2935096
```

```
# str(model_2019s1_3)
# From 564-lab caculate prediction power
deviation <- table_2019s1_500$y-mean(table_2019s1_500$y)
SST <- deviation%*%deviation
1-(MPV::PRESS(model_2019s1_3)/SST)
```

```
##           [,1]
## [1,] 0.2378794
```

```
# by definition PRESS
sum((table_2019s1_500$y-exp(model_2019s1_2$fit))^2)
```

```
## [1] 8.358063
```

```
sum((table_2019s1_500$y-exp(predict(model_2019s1_2,table_2019s1_500)))^2)
```

```
## [1] 4.13214
```

```
# one method of RMSE
sqrt(mean(model_2019s1_3$residuals^2))
```

```
## [1] 0.1345847
```

```
# remove outlier
table_2019s1_250[c(189,219,249),]
table_2019s1_250_noouter <- table_2019s1_250[-c(189,219,249), ]
table_2019s1_250_noouter <- table_2019s1_250[-c(113,189,219,249), ]
model_2019s1_noouter <- lm(y ~ sqrt(!is.na(x1))+x2+x3+A+B, data = table_2019s1_250_noouter)
summary(model_2019s1_noouter)
plot(model_2019s1_noouter)
```

a. calculate for each observation the square of the prediction errors,
b. obtain the square root of the average of all squared prediction errors.

https://blog.minitab.com/blog/adventures-in-statistics-2/multiple-regession-analysis-use-adjusted-r-squared-and-predicted-r-squared-to

2019S2

[14.4] [566-fe-4] [Example 8.4]
An experiment was conducted to compare 4 wool fiber treatments (Trt) at 7 dry cycle revolutions (Rev) over 4 experimental runs (Run) (i.e., the blocks). The outcome measured from this experiment was the top shrinkage (Shrink) of the fiber. A restriction on the randomization: within each experimental run (blocks), wool fiber treatments were randomized to whole plots, and within each whole plot, measurements were obtained for all of 7 dry cycle revolutions (split plot treatments). In other words, the experiment was set as a **split-plot** design with:

a. whole plot (wool fiber treatment) treatments: untreated, alcoholic potash 15 Sec, alcoholic potash 4Min, and alcoholic potash 15Min;
b. subplot treatments: dry cycle revolutions (200 to 1400 by 200); and
c. blocks: 4 experimental runs (possibly different days).

Do a full analysis and report your findings for the experiment above (data in "Wool-Shrink.xlsx"), using a split plot design where both Trt and Rev are treated as categorical variables.

```
table_2019s2 <- readxl::read_xlsx("~/qushen26/stat2019_website/static/stat566/qe_lab/WoolShrink.xlsx")
str(table_2019s2)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    112 obs. of  4 variables:
##  $ Run   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Trt   : num  1 1 1 1 1 1 1 2 2 2 ...
##  $ Rev   : num  200 400 600 800 1000 1200 1400 200 400 600 ...
##  $ Shrink: num  8 18.5 29 34.3 37.5 40.2 43.2 10.8 13.2 21 ...
```

```
library(dplyr)
dplyr::glimpse(table_2019s2)
```

```
## Observations: 112
## Variables: 4
## $ Run    <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Trt    <dbl> 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3...
## $ Rev    <dbl> 200, 400, 600, 800, 1000, 1200, 1400, 200, 400, 600, 80...
```

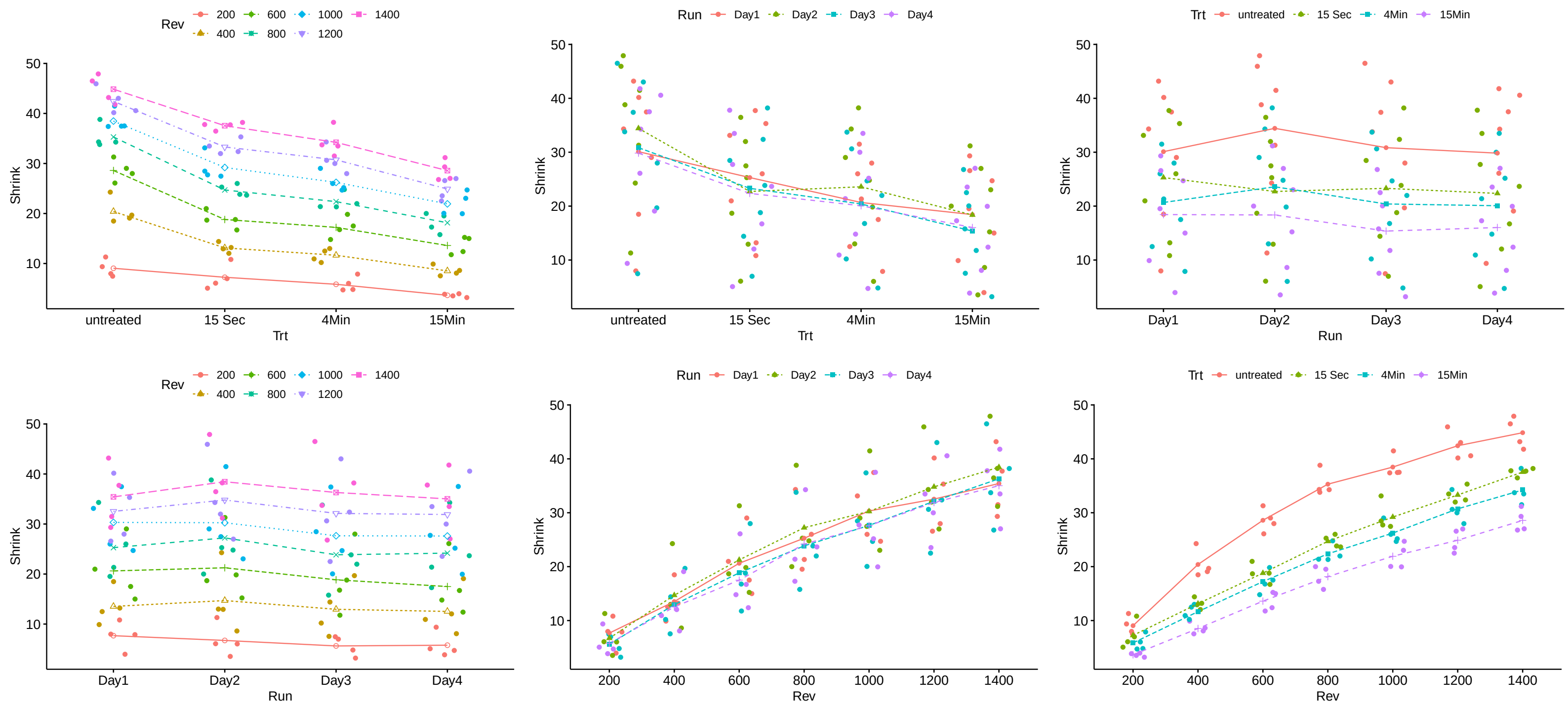
```
## $ Shrink <dbl> 8.0, 18.5, 29.0, 34.3, 37.5, 40.2, 43.2, 10.8, 13.2, 21...
```

```
table_2019s2$Run <- factor(table_2019s2$Run,labels=c("Day1","Day2","Day3","Day4"))
table_2019s2$Trt <- factor(table_2019s2$Trt,labels=c("untreated","15 Sec","4Min","15Min"))
table_2019s2$Rev <- as.factor(table_2019s2$Rev)
str(table_2019s2)
```

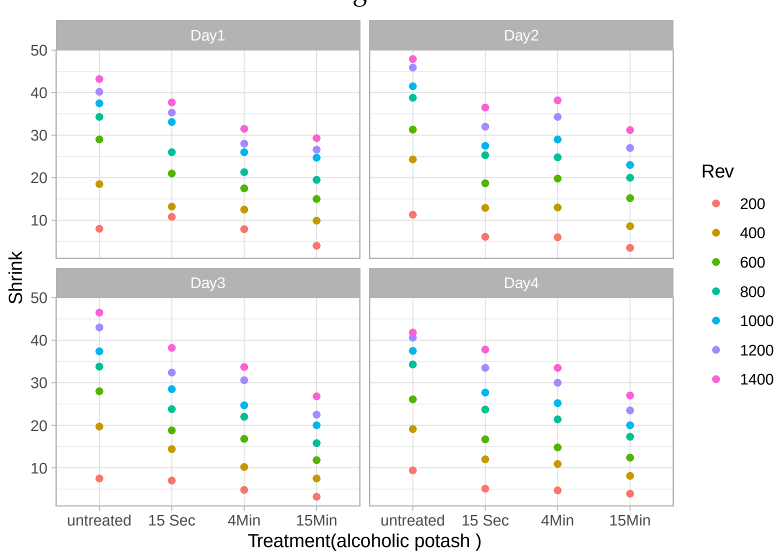
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    112 obs. of  4 variables:
##  $ Run   : Factor w/ 4 levels "Day1","Day2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Trt   : Factor w/ 4 levels "untreated","15 Sec",..: 1 1 1 1 1 1 1 2 2 2 ...
##  $ Rev   : Factor w/ 7 levels "200","400","600",..: 1 2 3 4 5 6 7 1 2 3 ...
##  $ Shrink: num  8 18.5 29 34.3 37.5 40.2 43.2 10.8 13.2 21 ...
```

```
dplyr::glimpse(table_2019s2)
```

```
## Observations: 112
## Variables: 4
## $ Run    <fct> Day1, Day1, Day1, Day1, Day1, Day1, Day1, Day1, Day1, D...
## $ Trt    <fct> untreated, untreated, untreated, untreated, untreated, ...
## $ Rev    <fct> 200, 400, 600, 800, 1000, 1200, 1400, 200, 400, 600, 80...
## $ Shrink <dbl> 8.0, 18.5, 29.0, 34.3, 37.5, 40.2, 43.2, 10.8, 13.2, 21...
```



The above plots show that: There is not much difference in the average shrink from different days. The average shrink are lower when the treatment is longer. The average shrink are higher when the revolutions are faster.
The tables show the same thing with the numerical summaries for each factor level and their combinations.




```
library(GAD)
table_2019s2$Run_r <- as.random(table_2019s2$Run)
table_2019s2$Trt_f <- as.fixed(table_2019s2$Trt)
table_2019s2$Rev_f <- as.fixed(table_2019s2$Rev)
model_2019s2_1 <- aov(formula = Shrink ~ Run_r+Trt_f + Trt_f%in%Run_r+ Rev_f%in%Run_r + Rev_f + Trt_f:Rev_f, data=ta
pander::pander(gad(model_2019s2_1))
```

Table 2: Analysis of Variance Table

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Run_r	3	124.3	41.43	36.47	5.099e-13
Trt_f	3	3013	1004	78.84	8.81e-07
Rev_f	6	11052	1842	876.8	3.405e-21
Run_r:Trt_f	9	114.6	12.74	11.21	1.218e-09
Run_r:Rev_f	18	37.81	2.101	1.849	0.04245
Trt_f:Rev_f	18	269.5	14.97	13.18	8.477e-14
Residual	54	61.35	1.136	NA	NA

The results show all the main effects and the interaction effect of Runs and Recolutions are significant at 0.05 significance level (P-value=0.5082).

```
library("lme4")
model_2019s2_2 <- lmer(formula = Shrink ~ (1|Run) + Trt + (1|Run:Trt) + Rev + (1|Run:Rev) + Trt:Rev, data=table_2019
summary(model_2019s2_2)$varcor
##  Groups   Name        Std.Dev.
##  Run:Rev  (Intercept) 0.49104
##  Run:Trt  (Intercept) 1.28736
##  Run      (Intercept) 0.99516
##  Residual             1.06587
pander::pander(confint(model_2019s2_2)[1:4,1:2])
```

Computing profile confidence intervals ...

	2.5 %	97.5 %
.sig01	0	0.726
.sig02	0.7415	1.82
.sig03	0	2.512
.sigma	0.7906	1.097

The results of variance components show the variance of interaction term of Runs and revolutions is negligible and hence dropping interaction term of them.

```
model_2019s2_3 <- aov(formula = Shrink ~ Run_r+Trt_f + Trt_f%in%Run_r+ Rev_f + Trt_f:Rev_f, data=table_2019s2)
pander::pander(gad(model_2019s2_3))
```

Table 4: Analysis of Variance Table

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Run_r	3	124.3	41.43	30.08	1.024e-12
Trt_f	3	3013	1004	78.84	8.81e-07
Rev_f	6	11052	1842	1337	1.01e-71
Run_r:Trt_f	9	114.6	12.74	9.249	3.546e-09
Trt_f:Rev_f	18	269.5	14.97	10.87	4.62e-14
Residual	72	99.16	1.377	NA	NA

```
model_2019s2_4<- lmer(formula = Shrink ~ (1|Run)+Trt+Rev+(1|Run:Trt)+Rev*Trt, data=table_2019s2, REML = TRUE)
```

The ANOVA table of new model shows that the interaction effects are significant. This means that the effects of day v.s.revolutions and treatment v.s.revolutions on the shrink are not independent. Hence, the simple effects must be tested.

When the day2, the mean shrinks between the 15-Sec and 4-Min treatment don't have significant difference. For all the rest of days, the mean shrinks are significantly different between any different treatment.

The changes of days for a given treatment don't give consistent results.

For untreated cases, the mean shrinks are not significantly different between 1200 and 1400 revolutions. For all the rest of treatments, the mean shrinks are significantly different between any different revolutions.

For a given revolution, 15-Sec and 4-Min treatment don't have significant differece on the mean shrinks.

Run_r	Trt_f	contrast	estimate	SE	df	t.ratio	p.value
Day1	.	untreated - 15 Sec	4.8000000	0.6272872	72	7.6519974	0.0000000
Day1	.	untreated - 4Min	9.4285714	0.6272872	72	15.0307091	0.0000000
Day1	.	untreated - 15Min	11.6714286	0.6272872	72	18.6061960	0.0000000
Day1	.	15 Sec - 4Min	4.6285714	0.6272872	72	7.3787118	0.0000000
Day1	.	15 Sec - 15Min	6.8714286	0.6272872	72	10.9541986	0.0000000
Day1	.	4Min - 15Min	2.2428571	0.6272872	72	3.5754869	0.0219854
Day2	.	untreated - 15 Sec	11.7142857	0.6272872	72	18.6745174	0.0000000
Day2	.	untreated - 4Min	10.8428571	0.6272872	72	17.2853155	0.0000000
Day2	.	untreated - 15Min	16.0714286	0.6272872	72	25.6205270	0.0000000
Day2	**.**	**15 Sec - 4Min**	**-0.8714286**	**0.6272872**	**72**	**-1.3892019**	**0.9337766**
Day2	.	15 Sec - 15Min	4.3571429	0.6272872	72	6.9460095	0.0000001
Day2	.	4Min - 15Min	5.2285714	0.6272872	72	8.3352114	0.0000000
Day3	.	untreated - 15 Sec	7.5428571	0.6272872	72	12.0245673	0.0000000
Day3	.	untreated - 4Min	10.4428571	0.6272872	72	16.6476491	0.0000000
Day3	.	untreated - 15Min	15.4714286	0.6272872	72	24.6640273	0.0000000
Day3	.	15 Sec - 4Min	2.9000000	0.6272872	72	4.6230818	0.0006908
Day3	.	15 Sec - 15Min	7.9285714	0.6272872	72	12.6394600	0.0000000
Day3	.	4Min - 15Min	5.0285714	0.6272872	72	8.0163782	0.0000000
Day4	.	untreated - 15 Sec	7.4714286	0.6272872	72	11.9106983	0.0000000
Day4	.	untreated - 4Min	9.7571429	0.6272872	72	15.5545066	0.0000000
Day4	.	untreated - 15Min	13.8000000	0.6272872	72	21.9994925	0.0000000
Day4	.	15 Sec - 4Min	2.2857143	0.6272872	72	3.6438083	0.0179467
Day4	.	15 Sec - 15Min	6.3285714	0.6272872	72	10.0887942	0.0000000
Day4	.	4Min - 15Min	4.0428571	0.6272872	72	6.4449859	0.0000005
.	untreated	Day1 - Day2	-4.3285714	0.6272872	72	-6.9004619	0.0000001
.	**untreated**	**Day1 - Day3**	**-0.7428571**	**0.6272872**	**72**	**-1.1842377**	**0.9760367**
.	**untreated**	**Day1 - Day4**	**0.2714286**	**0.6272872**	**72**	**0.4327022**	**0.9999934**
.	untreated	Day2 - Day3	3.5857143	0.6272872	72	5.7162242	0.0000106
.	untreated	Day2 - Day4	4.6000000	0.6272872	72	7.3331642	0.0000000
.	**untreated**	**Day3 - Day4**	**1.0142857**	**0.6272872**	**72**	**1.6169399**	**0.8470570**
.	15 Sec	Day1 - Day2	2.5857143	0.6272872	72	4.1220581	0.0039382
.	**15 Sec**	**Day1 - Day3**	**2.0000000**	**0.6272872**	**72**	**3.1883322**	**0.0644078**
.	15 Sec	Day1 - Day4	2.9428571	0.6272872	72	4.6914032	0.0005393
.	**15 Sec**	**Day2 - Day3**	**-0.5857143**	**0.6272872**	**72**	**-0.9337259**	**0.9956923**
.	**15 Sec**	**Day2 - Day4**	**0.3571429**	**0.6272872**	**72**	**0.5693450**	**0.9999261**
.	**15 Sec**	**Day3 - Day4**	**0.9428571**	**0.6272872**	**72**	**1.5030709**	**0.8959656**
.	4Min	Day1 - Day2	-2.9142857	0.6272872	72	-4.6458556	0.0006362
.	**4Min**	**Day1 - Day3**	**0.2714286**	**0.6272872**	**72**	**0.4327022**	**0.9999934**
.	**4Min**	**Day1 - Day4**	**0.6000000**	**0.6272872**	**72**	**0.9564997**	**0.9948267**
.	4Min	Day2 - Day3	3.1857143	0.6272872	72	5.0785578	0.0001277
.	4Min	Day2 - Day4	3.5142857	0.6272872	72	5.6023552	0.0000166
.	**4Min**	**Day3 - Day4**	**0.3285714**	**0.6272872**	**72**	**0.5237974**	**0.9999643**
.	**15Min**	**Day1 - Day2**	**0.0714286**	**0.6272872**	**72**	**0.1138690**	**1.0000000**
.	15Min	Day1 - Day3	3.0571429	0.6272872	72	4.8735936	0.0002759
.	15Min	Day1 - Day4	2.4000000	0.6272872	72	3.8259987	0.0102635
.	15Min	Day2 - Day3	2.9857143	0.6272872	72	4.7597246	0.0004202
.	15Min	Day2 - Day4	2.3285714	0.6272872	72	3.7121297	0.0145964
.	**15Min**	**Day3 - Day4**	**-0.6571429**	**0.6272872**	**72**	**-1.0475949**	**0.9898501**

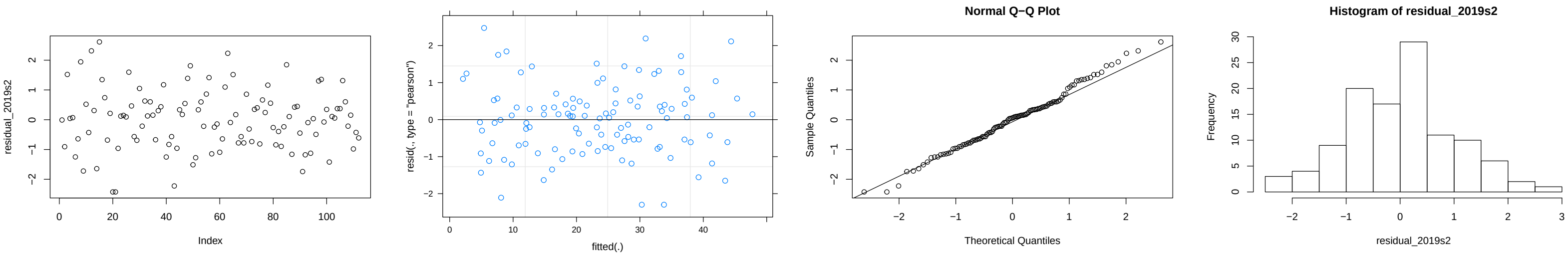
Trt	Rev	contrast	estimate	SE	df	t.ratio	p.value
untreated	.	200 - 400	-11.350	0.8298202	72.00000	-13.677662	0.0000000
untreated	.	200 - 600	-19.550	0.8298202	72.00000	-23.559322	0.0000000
untreated	.	200 - 800	-26.250	0.8298202	72.00000	-31.633360	0.0000000
untreated	.	200 - 1000	-29.425	0.8298202	72.00000	-35.459490	0.0000000
untreated	.	200 - 1200	-33.375	0.8298202	72.00000	-40.219558	0.0000000
untreated	.	200 - 1400	-35.800	0.8298202	72.00000	-43.141878	0.0000000
untreated	.	400 - 600	-8.200	0.8298202	72.00000	-9.881659	0.0000000
untreated	.	400 - 800	-14.900	0.8298202	72.00000	-17.955698	0.0000000
untreated	.	400 - 1000	-18.075	0.8298202	72.00000	-21.781828	0.0000000
untreated	.	400 - 1200	-22.025	0.8298202	72.00000	-26.541896	0.0000000
untreated	.	400 - 1400	-24.450	0.8298202	72.00000	-29.464216	0.0000000
untreated	.	600 - 800	-6.700	0.8298202	72.00000	-8.074039	0.0000000
untreated	.	600 - 1000	-9.875	0.8298202	72.00000	-11.900169	0.0000000
untreated	.	600 - 1200	-13.825	0.8298202	72.00000	-16.660236	0.0000000
untreated	.	600 - 1400	-16.250	0.8298202	72.00000	-19.582556	0.0000000
untreated	.	800 - 1000	-3.175	0.8298202	72.00000	-3.826130	0.0233929
untreated	.	800 - 1200	-7.125	0.8298202	72.00000	-8.586198	0.0000000
untreated	.	800 - 1400	-9.550	0.8298202	72.00000	-11.508518	0.0000000
untreated	.	1000 - 1200	-3.950	0.8298202	72.00000	-4.760067	0.0010353
untreated	.	1000 - 1400	-6.375	0.8298202	72.00000	-7.682387	0.0000000
untreated	**.**	**1200 - 1400**	**-2.425**	**0.8298202**	**72.00000**	**-2.922320**	**0.2371347**
15 Sec	.	200 - 400	-5.875	0.8298202	72.00000	-7.079847	0.0000001
15 Sec	.	200 - 600	-11.550	0.8298202	72.00000	-13.918679	0.0000000
15 Sec	.	200 - 800	-17.450	0.8298202	72.00000	-21.028653	0.0000000
15 Sec	.	200 - 1000	-21.950	0.8298202	72.00000	-26.451514	0.0000000
15 Sec	.	200 - 1200	-26.050	0.8298202	72.00000	-31.392344	0.0000000
15 Sec	.	200 - 1400	-30.300	0.8298202	72.00000	-36.513936	0.0000000
15 Sec	.	400 - 600	-5.675	0.8298202	72.00000	-6.838831	0.0000003
15 Sec	.	400 - 800	-11.575	0.8298202	72.00000	-13.948806	0.0000000
15 Sec	.	400 - 1000	-16.075	0.8298202	72.00000	-19.371667	0.0000000
15 Sec	.	400 - 1200	-20.175	0.8298202	72.00000	-24.312497	0.0000000
15 Sec	.	400 - 1400	-24.425	0.8298202	72.00000	-29.434089	0.0000000
15 Sec	.	600 - 800	-5.900	0.8298202	72.00000	-7.109974	0.0000001
15 Sec	.	600 - 1000	-10.400	0.8298202	72.00000	-12.532836	0.0000000
15 Sec	.	600 - 1200	-14.500	0.8298202	72.00000	-17.473666	0.0000000
15 Sec	.	600 - 1400	-18.750	0.8298202	72.00000	-22.595257	0.0000000
15 Sec	.	800 - 1000	-4.500	0.8298202	72.00000	-5.422862	0.0000856
15 Sec	.	800 - 1200	-8.600	0.8298202	72.00000	-10.363691	0.0000000
15 Sec	.	800 - 1400	-12.850	0.8298202	72.00000	-15.485283	0.0000000
15 Sec	.	1000 - 1200	-4.100	0.8298202	72.00000	-4.940830	0.0005344
15 Sec	.	1000 - 1400	-8.350	0.8298202	72.00000	-10.062421	0.0000000
15 Sec	.	1200 - 1400	-4.250	0.8298202	72.00000	-5.121592	0.0002718
4Min	.	200 - 400	-5.800	0.8298202	72.00000	-6.989466	0.0000001
4Min	.	200 - 600	-11.375	0.8298202	72.00000	-13.707789	0.0000000
4Min	.	200 - 800	-16.525	0.8298202	72.00000	-19.913953	0.0000000
4Min	.	200 - 1000	-20.375	0.8298202	72.00000	-24.553513	0.0000000
4Min	.	200 - 1200	-24.875	0.8298202	72.00000	-29.976375	0.0000000
4Min	.	200 - 1400	-28.375	0.8298202	72.00000	-34.194156	0.0000000
4Min	.	400 - 600	-5.575	0.8298202	72.00000	-6.718323	0.0000004
4Min	.	400 - 800	-10.725	0.8298202	72.00000	-12.924487	0.0000000
4Min	.	400 - 1000	-14.575	0.8298202	72.00000	-17.564047	0.0000000
4Min	.	400 - 1200	-19.075	0.8298202	72.00000	-22.986908	0.0000000
4Min	.	400 - 1400	-22.575	0.8298202	72.00000	-27.204690	0.0000000
4Min	.	600 - 800	-5.150	0.8298202	72.00000	-6.206164	0.0000037
4Min	.	600 - 1000	-9.000	0.8298202	72.00000	-10.845724	0.0000000
4Min	.	600 - 1200	-13.500	0.8298202	72.00000	-16.268585	0.0000000
4Min	.	600 - 1400	-17.000	0.8298202	72.00000	-20.486367	0.0000000
4Min	.	800 - 1000	-3.850	0.8298202	72.00000	-4.639559	0.0015948
4Min	.	800 - 1200	-8.350	0.8298202	72.00000	-10.062421	0.0000000
4Min	.	800 - 1400	-11.850	0.8298202	72.00000	-14.280203	0.0000000
4Min	.	1000 - 1200	-4.500	0.8298202	72.00000	-5.422862	0.0000856
4Min	.	1000 - 1400	-8.000	0.8298202	72.00000	-9.640643	0.0000000
4Min	.	1200 - 1400	-3.500	0.8298202	72.00000	-4.217781	0.0067840
15Min	.	200 - 400	-4.875	0.8298202	72.00000	-5.874767	0.0000143
15Min	.	200 - 600	-9.950	0.8298202	72.00000	-11.990550	0.0000000
15Min	.	200 - 800	-14.500	0.8298202	72.00000	-17.473666	0.0000000
15Min	.	200 - 1000	-18.275	0.8298202	72.00000	-22.022844	0.0000000
15Min	.	200 - 1200	-21.250	0.8298202	72.00000	-25.607958	0.0000000
15Min	.	200 - 1400	-24.925	0.8298202	72.00000	-30.036629	0.0000000
15Min	.	400 - 600	-5.075	0.8298202	72.00000	-6.115783	0.0000054
15Min	.	400 - 800	-9.625	0.8298202	72.00000	-11.598899	0.0000000
15Min	.	400 - 1000	-13.400	0.8298202	72.00000	-16.148077	0.0000000
15Min	.	400 - 1200	-16.375	0.8298202	72.00000	-19.733191	0.0000000
15Min	.	400 - 1400	-20.050	0.8298202	72.00000	-24.161862	0.0000000
15Min	.	600 - 800	-4.550	0.8298202	72.00000	-5.483116	0.0000677
15Min	.	600 - 1000	-8.325	0.8298202	72.00000	-10.032294	0.0000000
15Min	.	600 - 1200	-11.300	0.8298202	72.00000	-13.617408	0.0000000
15Min	.	600 - 1400	-14.975	0.8298202	72.00000	-18.046079	0.0000000
15Min	.	800 - 1000	-3.775	0.8298202	72.00000	-4.549179	0.0021940
15Min	.	800 - 1200	-6.750	0.8298202	72.00000	-8.134293	0.0000000
15Min	.	800 - 1400	-10.425	0.8298202	72.00000	-12.562963	0.0000000
15Min	.	1000 - 1200	-2.975	0.8298202	72.00000	-3.585114	0.0471045
15Min	.	1000 - 1400	-6.650	0.8298202	72.00000	-8.013785	0.0000000
15Min	.	1200 - 1400	-3.675	0.8298202	72.00000	-4.428670	0.0033332
.	**200**	**untreated - 15 Sec**	**1.800**	**1.2247951**	**23.24089**	**1.469634**	**0.9779005**
.	**200**	**untreated - 4Min**	**3.200**	**1.2247951**	**23.24089**	**2.612682**	**0.4439814**
.	200	untreated - 15Min	5.400	1.2247951	23.24089	4.408901	0.0141095
.	**200**	**15 Sec - 4Min**	**1.400**	**1.2247951**	**23.24089**	**1.143048**	**0.9980039**
.	**200**	**15 Sec - 15Min**	**3.600**	**1.2247951**	**23.24089**	**2.939267**	**0.2741847**
.	**200**	**4Min - 15Min**	**2.200**	**1.2247951**	**23.24089**	**1.796219**	**0.9003941**
.	400	untreated - 15 Sec	7.275	1.2247951	23.24089	5.939769	0.0004029
.	400	untreated - 4Min	8.750	1.2247951	23.24089	7.144052	0.0000259
.	400	untreated - 15Min	11.875	1.2247951	23.24089	9.695499	0.0000001
.	**400**	**15 Sec - 4Min**	**1.475**	**1.2247951**	**23.24089**	**1.204283**	**0.9965681**
.	400	15 Sec - 15Min	4.600	1.2247951	23.24089	3.755730	0.0589409
.	**400**	**4Min - 15Min**	**3.125**	**1.2247951**	**23.24089**	**2.551447**	**0.4805075**
.	600	untreated - 15 Sec	9.800	1.2247951	23.24089	8.001338	0.0000040
.	600	untreated - 4Min	11.375	1.2247951	23.24089	9.287268	0.0000003
.	600	untreated - 15Min	15.000	1.2247951	23.24089	12.246946	0.0000000
.	**600**	**15 Sec - 4Min**	**1.575**	**1.2247951**	**23.24089**	**1.285929**	**0.9934208**
.	600	15 Sec - 15Min	5.200	1.2247951	23.24089	4.245608	0.0203740
.	**600**	**4Min - 15Min**	**3.625**	**1.2247951**	**23.24089**	**2.959679**	**0.2652318**
.	800	untreated - 15 Sec	10.600	1.2247951	23.24089	8.654509	0.0000010
.	800	untreated - 4Min	12.925	1.2247951	23.24089	10.552786	0.0000000
.	800	untreated - 15Min	17.150	1.2247951	23.24089	14.002342	0.0000000
.	**800**	**15 Sec - 4Min**	**2.325**	**1.2247951**	**23.24089**	**1.898277**	**0.8595571**
.	800	15 Sec - 15Min	6.550	1.2247951	23.24089	5.347833	0.0016047
.	800	4Min - 15Min	4.225	1.2247951	23.24089	3.449557	0.1095576
.	1000	untreated - 15 Sec	9.275	1.2247951	23.24089	7.572695	0.0000101
.	1000	untreated - 4Min	12.250	1.2247951	23.24089	10.001673	0.0000001
.	1000	untreated - 15Min	16.550	1.2247951	23.24089	13.512464	0.0000000
.	**1000**	**15 Sec - 4Min**	**2.975**	**1.2247951**	**23.24089**	**2.428978**	**0.5561910**
.	1000	15 Sec - 15Min	7.275	1.2247951	23.24089	5.939769	0.0004029
.	1000	4Min - 15Min	4.300	1.2247951	23.24089	3.510791	0.0971176
.	1200	untreated - 15 Sec	9.125	1.2247951	23.24089	7.450226	0.0000132
.	1200	untreated - 4Min	11.700	1.2247951	23.24089	9.552618	0.0000002
.	1200	untreated - 15Min	17.525	1.2247951	23.24089	14.308516	0.0000000
.	**1200**	**15 Sec - 4Min**	**2.575**	**1.2247951**	**23.24089**	**2.102393**	**0.7561134**
.	1200	15 Sec - 15Min	8.400	1.2247951	23.24089	6.858290	0.0000491
.	1200	4Min - 15Min	5.825	1.2247951	23.24089	4.755898	0.0063729
.	1400	untreated - 15 Sec	7.300	1.2247951	23.24089	5.960181	0.0003842
.	1400	untreated - 4Min	10.625	1.2247951	23.24089	8.674920	0.0000010
.	1400	untreated - 15Min	16.275	1.2247951	23.24089	13.287937	0.0000000

- Conclusion

Choosing a higher revolution for a given treatment can get a larger shrink.

In most of the cases, longter alcoholic potash have less shrink. This effect will be more significant when higher revolution.

- Model Adequacy Checking



In the plots of residuals versus predicted value of shrink, there is no significant pattern on this plot. Therefore, the fitted model is good enough to describe the relationship between the mean value of shrink and the days, revolutions, and treatment.

The residuals in this plot are almost symmetrically distributed about zero and hence zero mean assumption is not violated. Further, the vertical deviation of the residuals from zero is about same for each predicted value and hence the constant variance assumption is not violated.

The points are along the straight line in the normal qq plot shown at bottom left and the histogram of residuals shown at the top right is about normal. These plots show no violation of normal distribution assumption of residuals.