Applied Satistics MS Exam-Lab

Shen Qu

Code: 34161

Contents

Preload data

```
## # A tibble: 50 x 9
##    country confirmed confirmed.double population land_area_skm pop_density
##    <chr>       <dbl>            <dbl>      <dbl>          <dbl>       <dbl>
##  1 San Ma~       501               21      33785             60       563.
##  2 Liecht~        81               33      37910            160       237.
##  3 Monaco         94               25      38682              2     19196
##  4 Andorra       723               24      77006            470       164.
##  5 Iceland      1789               27     352721         100250         3.53
##  6 Luxemb~      3665               26     607950           2430       250.
##  7 Monten~       316               21     622227          13450        46.3
##  8 Cyprus        795               20    1189265           9240       129.
##  9 Estonia      1592               22    1321977          43470        30.4
## 10 Latvia        778               23    1927174          62180        31.0
## # ... with 40 more rows, and 3 more variables: pop_largest_city <dbl>,
## #   life_expectancy <dbl>, gdp_capita <dbl>
```

Question 1 Linear Regression

Initial Analysis

Data Description

```
data1 <- read_xlsx("qe_lab/RegressionSpr16.xlsx")[-1,]
data1$weight <- round(as.numeric(data1$weight), 2)
data1$age <- as.integer(data1$age)
data1$height <- round(as.numeric(data1$height), 2)
data1$male <- factor(data1$male, labels=c("female","male"))
str(data1)
## tibble [30 x 4] (S3: tbl_df/tbl/data.frame)
##  $ weight: num [1:30] 250 110 243 118 249 ...
##  $ age   : int [1:30] 20 20 20 20 20 21 21 21 21 21 ...
##  $ height: num [1:30] 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 2 1 1 2 ...
```

The data includes 30 rows and 4 columns. Each row represents an observation. The column 'weight' is response variable. Column 'height', 'age' and 'male' are regressors.

'Age' is integer variable. 'Male' is binary variable. 'Weight' and 'Height' are numerical variables.

- Check Missing Values

```
sapply(data1,function(x)sum(is.na(x)))
## weight    age height   male
##      0      0      0      0
# data1[!complete.cases(data1), ]
```

There is no missing values in the data.

- Sample distribution

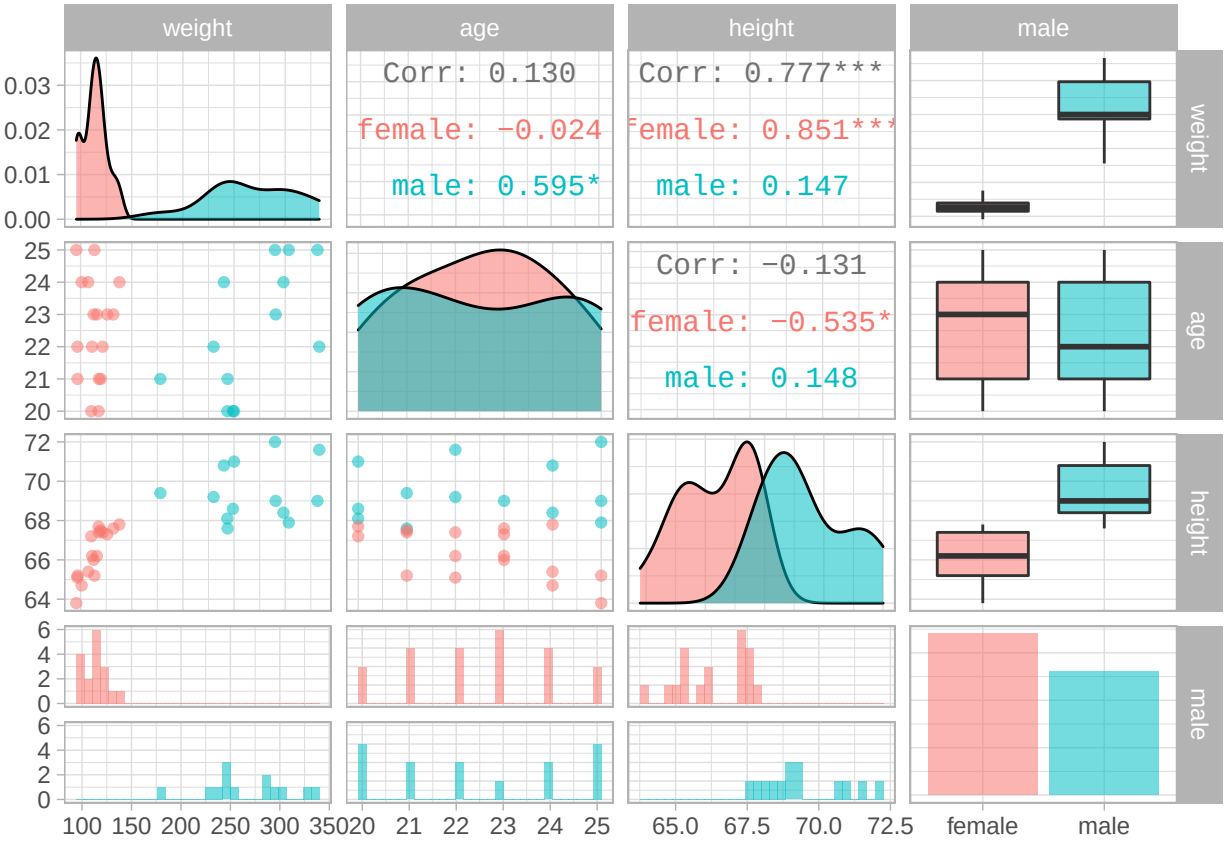
```
summary(data1)
##      weight            age           height          male
##  Min.   : 95.76   Min.   :20.0   Min.   :63.80   female:17
##  1st Qu.:112.93   1st Qu.:21.0   1st Qu.:66.20   male  :13
##  Median :129.01   Median :22.5   Median :67.60
##  Mean   :180.52   Mean   :22.5   Mean   :67.68
##  3rd Qu.:247.60   3rd Qu.:24.0   3rd Qu.:68.90
##  Max.   :333.27   Max.   :25.0   Max.   :72.00
```

For the 30 observations. the values of weight are from 95.76 to 333.27 and mean value is 180.52.

The age are from 20 to 25. It is a ordinal variable.

The height are from 63.80 to 72.00 and mean value is 67.68.

The number of female is 17, which is greater than that of male. The two groups are not balanced.



The scatter plots show a strong(medium) positive(negative) curved relationship between height and weight (Correlation is 0.777), a weak correlation among the age and weight (0.133), and a weak correlation between age and height (0.131).

The plot shows that, the relationships of variables are very different between male and female.

For male group, the correlation of weight and age is 0.214. The correlation of weight and height is 0.472. The correlation between age and height is 0.56.

For female group, the correlation of weight and age is 0.581. The correlation of weight and height is 0.0931. The correlation between age and height is 0.0711.

In the plot of weight vs. height, the range of the sample points in male and female groups are not overlap. The points of male are dispersed with greater values of weight and height. The points of female are tight with lessr values of weight and height.

- t test

```
t.test(weight~male,alternative="less",data1) # "two.sided", "less", "greater"
##
##  Welch Two Sample t-test
##
## data:  weight by male
## t = -12.164, df = 13.347, p-value = 6.667e-09
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -131.6118
## sample estimates:
## mean in group female   mean in group male
##             113.7941             267.7800
t.test(height~male,alternative="less",data1)
##
##  Welch Two Sample t-test
##
## data:  height by male
## t = -6.1357, df = 23.786, p-value = 1.269e-06
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -2.232019
## sample estimates:
## mean in group female   mean in group male
##             66.33529             69.43077
```

The t-test shows that the mean weight of male is greater than female at 0.05 significant level. The same goes with height.

- Kruskal-Wallis test

It is a unbalanced data with more female. The male group and female group may be independent if they come from unrelated populations and the samples do not affect each other.

Using the Kruskal-Wallis Test, we can decide whether the population distributions are identical without assuming them to follow the normal distribution.

```
kruskal.test(weight ~ male, data = data1)
##
##  Kruskal-Wallis rank sum test
##
## data:  weight by male
## Kruskal-Wallis chi-squared = 21.387, df = 1, p-value = 3.753e-06
kruskal.test(weight ~ age, data = data1)
##
##  Kruskal-Wallis rank sum test
##
## data:  weight by age
## Kruskal-Wallis chi-squared = 1.0671, df = 5, p-value = 0.957
```

At .05 significance level, we conclude that the male's weight and female's weight are non-identical populations. While the weight from different age groups are identical population.

- Multiple pairwise-comparison between groups

```r
pairwise.wilcox.test(data1$weight, data1$age,p.adjust.method = "BH")
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
##
## data:  data1$weight and data1$age
##
##    20 21 22 23 24
## 21 1  -  -  -  -
## 22 1  1  -  -  -
## 23 1  1  1  -  -
## 24 1  1  1  1  -
## 25 1  1  1  1  1
##
## P value adjustment method: BH
# c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none")
```

At .05 significance level, we conclude that different age groups are identical populations.

Fit a basic linear model

- Multiple linear regression

Fit a basic linear model with the first-order terms:

$$Weight = \beta_0 + \beta_1 Height + \beta_2 Age + (\beta_3 + \beta_4 Height + \beta_5 Age)male + \varepsilon$$

for $i = 1, 2, .., 30$. ε_i is random error. $\varepsilon_i \sim iidN(0, \sigma^2)$

```r
model_basic <- lm(weight ~male+height+age+(height+age):male,data1)
summary(model_basic)
##
## Call:
## lm.default(formula = weight ~ male + height + age + (height +
##     age):male, data = data1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -70.547  -1.408   0.300   4.639  67.558
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -735.206    460.301  -1.597   0.1233
## malemale         580.624    577.948   1.005   0.3251
## height            11.246      5.951   1.890   0.0709 .
## age                4.573      4.704   0.972   0.3407
## malemale:height   -9.407      7.817  -1.203   0.2405
## malemale:age       8.548      5.990   1.427   0.1665
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.21 on 24 degrees of freedom
## Multiple R-squared:  0.924,  Adjusted R-squared:  0.9082
## F-statistic: 58.36 on 5 and 24 DF,  p-value: 1.207e-12
```

The fitted model is statistically significant at 5% significance level (p-value=0.000). None the regression coefficients are significant at 5% significance level. The adjusted R^2 is high (0.9082). 90.82% of the variance in the dependent variable is predictable from the independent variable.

Multicollinearity Diagnostics

Multicollinearity implies two or more variables are near perfect linear combinations of one another. Variance inflation factors measure the inflation in the variances of the parameter estimates due to collinearities that exist among the predictors. In the presence of multicollinearity, regression estimates are unstable and have high standard errors.

A VIF of 1 means that there is no correlation among the k^{th} predictor and the remaining predictor variables, and hence the variance of β_k is not inflated at all. VIFs exceeding 4 warrant further investigation, while VIFs exceeding 10 are signs of serious multicollinearity requiring correction.

```
car::vif(model_basic)
##        male      height         age male:height    male:age
## 3872.712617    6.745032    3.047161 3417.566300  212.551570
```

All the VIF are less than 10. The test shows there isn't significant problem of collinearity.

Check Homogenity of Variances among groups

Bartlett's test is used to test if variances across samples is equal. It is sensitive to departures from normality.

```
bartlett.test(weight~male,data1)
##
##  Bartlett test of homogeneity of variances
##
## data:  weight by male
## Bartlett's K-squared = 20.013, df = 1, p-value = 7.693e-06
bartlett.test(data1$weight,data1$male)
##
##  Bartlett test of homogeneity of variances
##
## data:  data1$weight and data1$male
## Bartlett's K-squared = 20.013, df = 1, p-value = 7.693e-06
# olsrr::ols_test_bartlett(data1,'weight', group_var = 'male')
```

Bartlett tests show that variances are not equal across male and female groups

- Levene's test

The Levene test is an alternative test that is less sensitive to departures from normality.

```
leveneTest(weight~male, data1) # male*age
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value   Pr(>F)
## group  1  11.875 0.001813 **
##       28
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Fligner-Killeen test

The Fligner-Killeen test is one of the many tests for homogeneity of variances which is most robust against departures from normality.

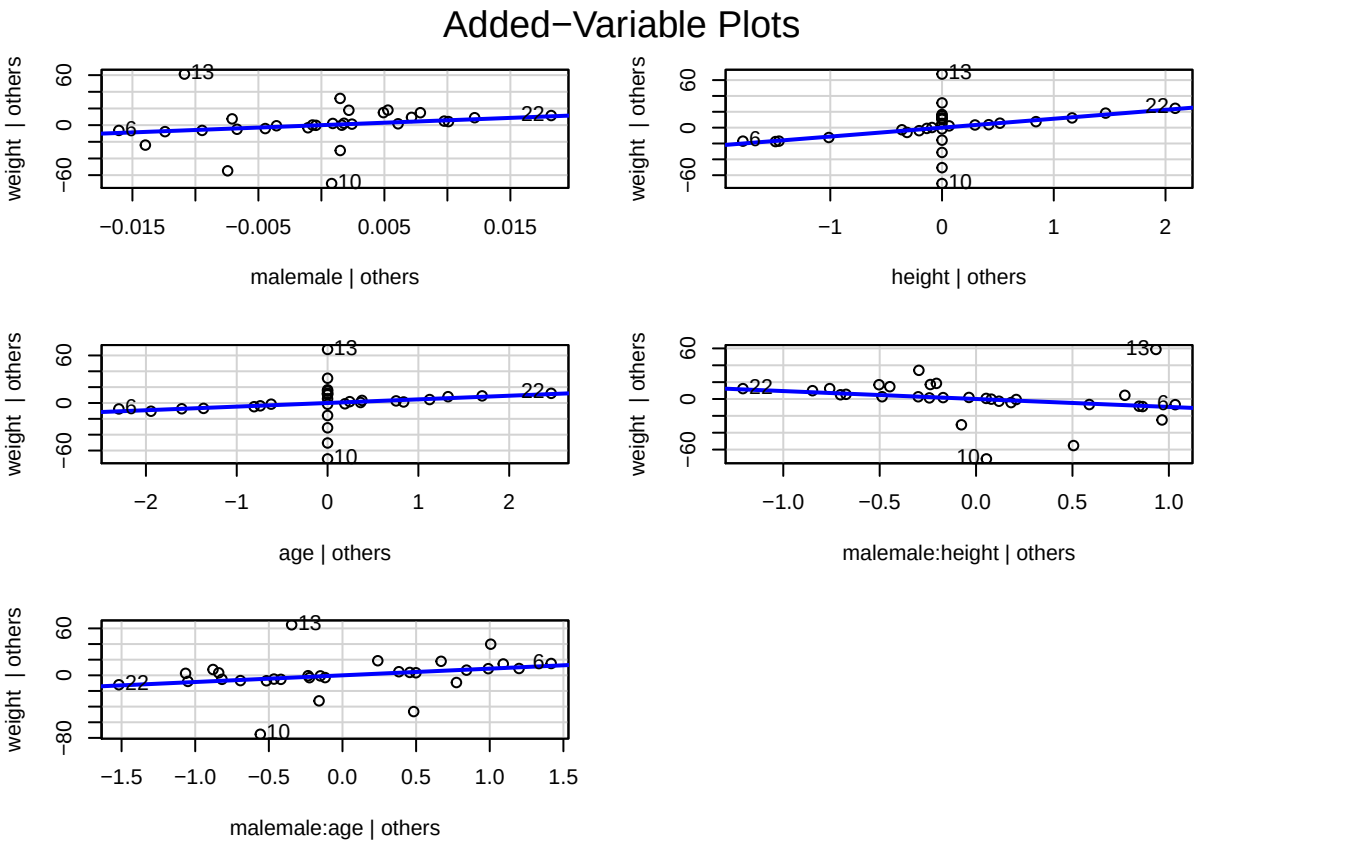
```
fligner.test(weight~male, data1)
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  weight by male
## Fligner-Killeen:med chi-squared = 6.903, df = 1, p-value = 0.008605
```

From the output above we can see that the p-value is less than the significance level of 0.05. The evidences suggest that the variance across groups is statistically significantly different. Therefore, we cannot assume the homogeneity of variances in the different groups.

Added Variable Plot

Remove 1 predictor from the model. Run the reduced model and obtain the residuals. Run a regression of the removed predictor on the remaining predictors and obtain the residuals.

```
# ols_plot_added_variable(model_basic)
avPlots(model_basic)
```



Added variable plot provides information about the marginal importance of a predictor variable X_k, given the other predictor variables already in the model. It shows the marginal importance of the variable in reducing the residual variability.

A strong linear relationship in the added 'male' variable plot indicates the increased importance of the contribution of 'male' to the model already containing the other predictors. The slope of the line fitted to the points in the added variable plot is equal to the regression coefficient when 'weight' is regressed on all variables including 'male'.

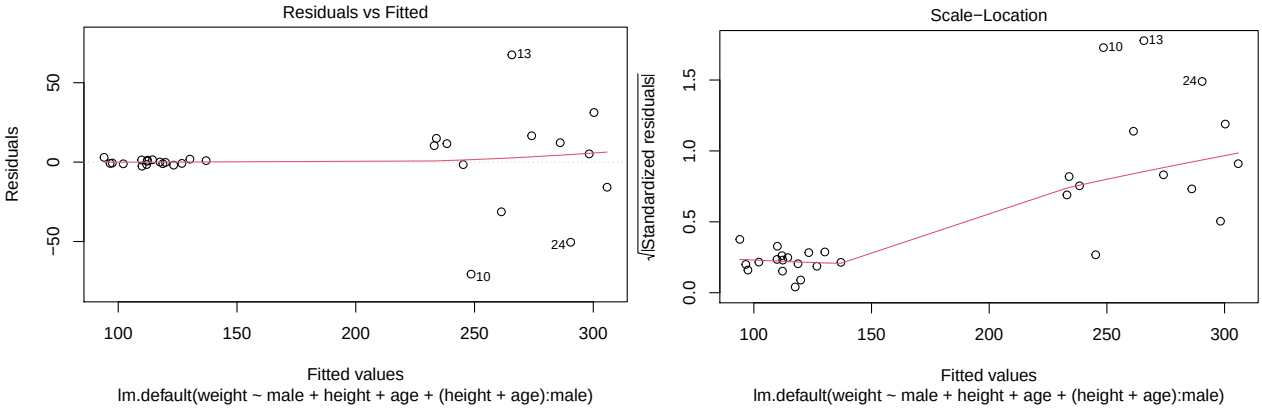
Residual vs. fitted values

If the error variances are unequal, try "stabilizing the variance" by transforming y, and stay within the linear regression framework Transforming the y values corrects problems with the error terms (and may help the non-linearity). Transforming the x values primarily corrects the non-linearity. A higher-order model may be needed.

If the response y is a **Poisson count**, the variances of the error terms are not constant but rather depend on the value of the predictor. A common recommendation is to transform the response using the "square root transformation"$y^* = \sqrt{y}$

If the response y is a **binomial proportion**, the variances of the error terms are not constant but rather depend on the value of the predictor. Another common recommendation is to transform the response using the "arcsine transformation," $\hat{p}^* = sin^{-1}\left(\sqrt{\hat{p}}\right)$

If the response y isn't anything special, but the error variances are unequal, a common recommendation is to try the natural log transformation or the "reciprocal transformation" $y^* = \frac{1}{y}$.



A residuals vs. fitted plot is better for determining linearity, and A scale-location plot is good for determining heteroskedasticity.

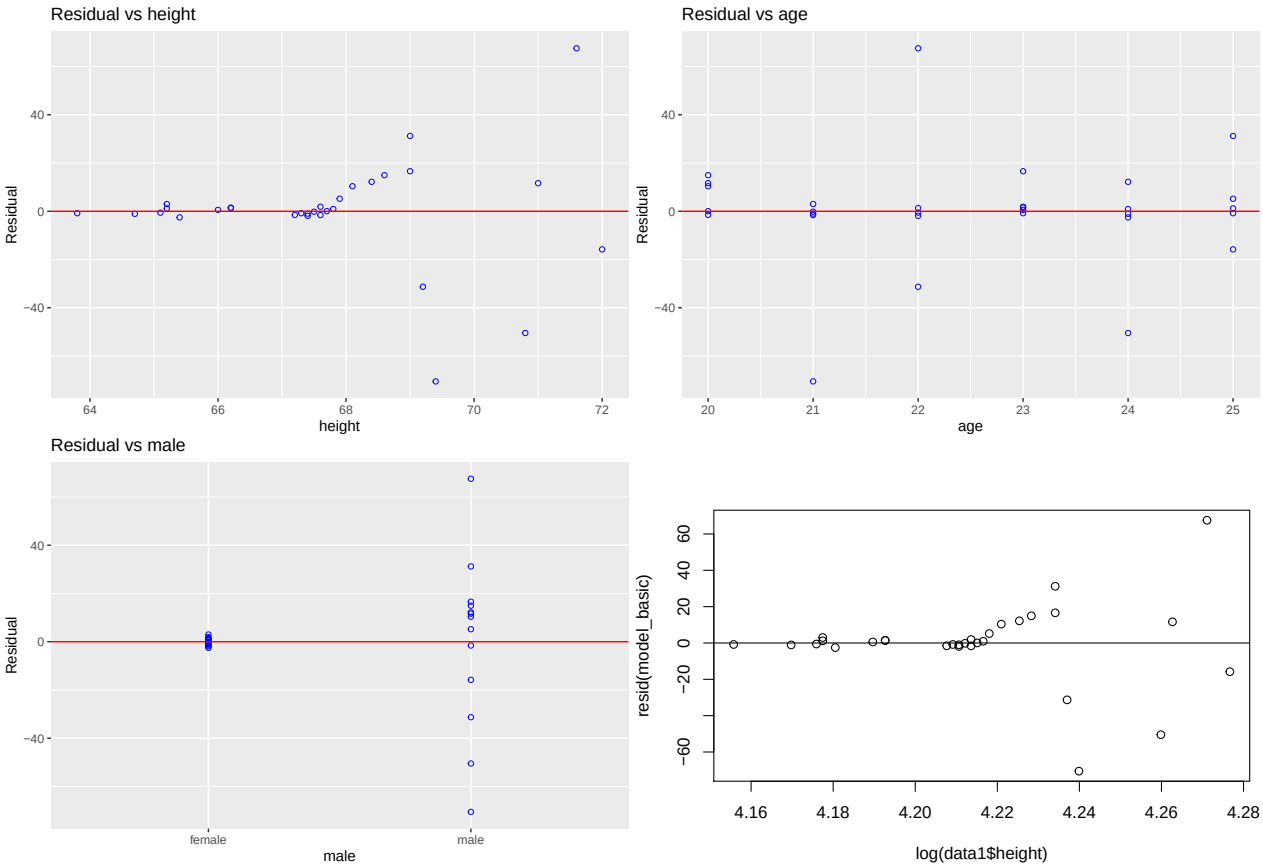
The plot of residuals v.s. predicted values detects non-linearity. there is discernible (a funnel) pattern or change in envelope. there is curvature and it doesn't show uniform randomness.

In this case, the residual are not constant spread. There is a little curved pattern on this plot. The curved red line and spread of points indicates that variance is not constant. Evidence against linearity assumption. We can transform response or add quadratic and interaction terms in the model.

Residual vs. explanatory values

Remove 1 predictor from the model. Run the reduced model and obtain the residuals. Plot the residuals vs. the removed predictor. If I see a pattern, then X_i should not be removed from the model. If I see no pattern, this is evidence that X_i could be removed.

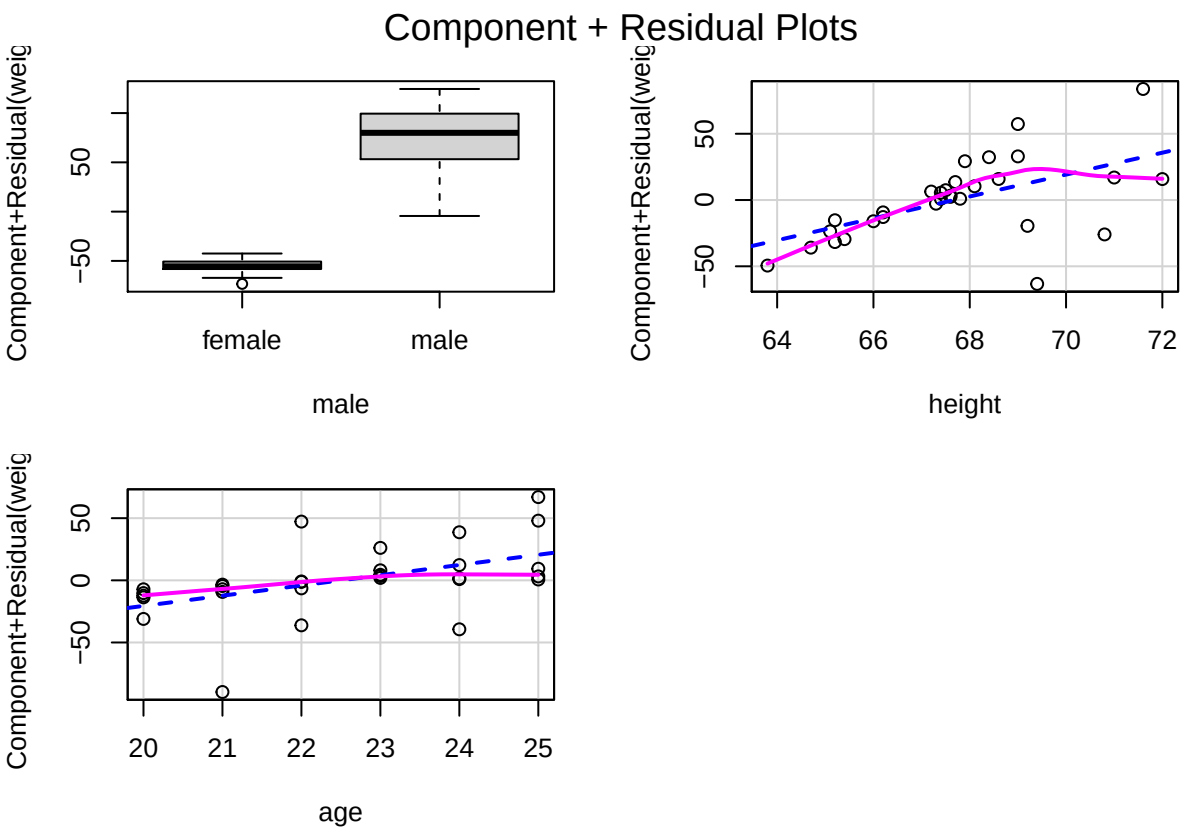
Graph to determine whether we should add a new predictor to the model already containing other predictors. The residuals from the model is regressed on the new predictor and if the plot shows non random pattern, you should consider adding the new predictor to the model.



If there is a curve pattern on the plot, the problem with nonlinearity is probably due to this X_i. This variable doesn't enter the model in the right way. We should consider higher order term or other transformation on X_i.

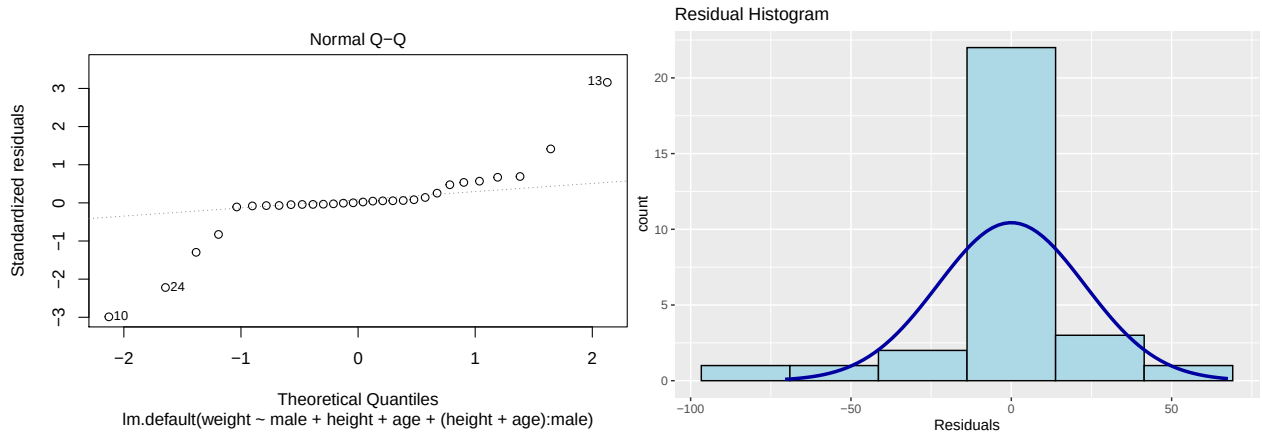
Partial Residual Plot

```
crPlots(lm(weight ~male+height+age,data1))
# ols_plot_comp_plus_resid(model_basic)
```



A partial residual plot essentially attempts to model the residuals of one predictor against the dependent variable. A component residual plot adds a line indicating where the line of best fit lies. The Y residuals represent the part of Y not explained by all the variables other than X. The X residuals represent the part of X not explained by other variables. The residual plus component plot indicates whether any non-linearity is present in the relationship between Y and X and can suggest possible transformations for linearizing the data.

Check residual Normality



Normality is more important for small sample size.

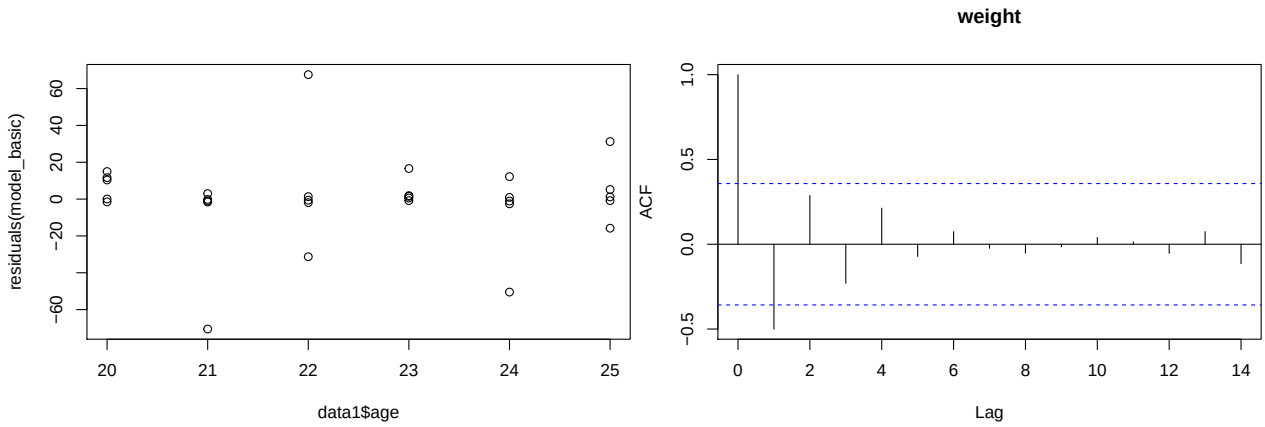
The Q-Q plot does not show that the residuals are normal distributed. Some points deviated from the straight line on the two tail sides. The assumption of normal distribution about errors is violated.

The histogram of the residuals shows a roughly normal distribution.

```
olsrr::ols_test_normality(model_basic)
## -----------------------------------------------
##        Test             Statistic       pvalue
## -----------------------------------------------
## Shapiro-Wilk              0.7838         0.0000
## Kolmogorov-Smirnov        0.3225         0.0028
## Cramer-von Mises          2.1767         0.0000
## Anderson-Darling          2.9956         0.0000
## -----------------------------------------------
```

None of four methods reject that the residuals are normal distributed.

Check residual independence




```
lmtest::dwtest(model_basic)
##
##  Durbin-Watson test
##
## data:  model_basic
## DW = 2.0475, p-value = 0.5456
## alternative hypothesis: true autocorrelation is greater than 0
```

If the randomness assumption is not valid, then a different model needs to be used. This will typically be either a time series model or a non-linear model (with age as the independent variable).

The serial autocorrelations should all be 0. Durbin-Watson test fails to reject that the model have zero autocorrelation.

Brief Summary

- The above analysis indicate that there are some violations of the model assumption. The basic linear model is not good for the data.

- A Box-Cox transformations can stabilize response variance, make the distribution of the response variable closer to the normal distribution, and improve the fit of the model to the data.
- Since the randomness assumption is not valid, I will consider a polynomial model.
- The male's weight and female's weight are not identical populations. We can consider analysis the models for the subgroups.

The Polynomial Model

The full Model

Fit a full model with quadratic and pairwise interaction terms:

$$Weight = \beta_0 + \beta_1 Height + \beta_2 Age + \beta_3 Height * Age + \beta_4 Height^2 + \beta_5 Age^2 +$$

$$(\beta_6 + \beta_7 Height + \beta_8 Age + \beta_9 Height * Age + \beta_{10} Height^2 + \beta_{11} Age^2) Male + \varepsilon_i$$

for $i = 1, 2, .., 30$. ε_i is random error. $\varepsilon_{ijk} \sim iidN(0, \sigma^2)$

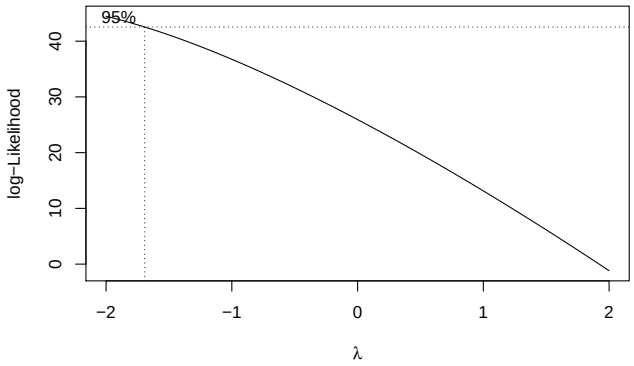
```
model1 <- lm(weight ~  male+height+age+age:height+I(height^2)+I(age^2)+
                        (height+age+age:height+I(height^2)+I(age^2)):male
                        , data1)
summary(model1)
##
## Call:
## lm.default(formula = weight ~ male + height + age + age:height +
##     I(height^2) + I(age^2) + (height + age + age:height + I(height^2) +
##     I(age^2)):male, data = data1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -54.708  -2.246   0.095   1.458  44.393
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           552.1050 30041.1685   0.018    0.986
## malemale            39721.3304 37487.4047   1.060    0.303
## height                 -9.7787   845.8085  -0.012    0.991
## age                   -47.1116   464.7650  -0.101    0.920
## I(height^2)             0.0539     6.0408   0.009    0.993
## I(age^2)                0.2566     3.9729   0.065    0.949
## height:age              0.6050     5.0065   0.121    0.905
## malemale:height     -1270.1455  1077.2463  -1.179    0.254
## malemale:age          439.5832   517.8404   0.849    0.407
## malemale:I(height^2)   10.0717     7.7958   1.292    0.213
## malemale:I(age^2)       0.1972     4.8798   0.040    0.968
## malemale:height:age    -6.3855     5.6965  -1.121    0.277
##
## Residual standard error: 24.86 on 18 degrees of freedom
## Multiple R-squared:  0.9446, Adjusted R-squared:  0.9107
## F-statistic: 27.87 on 11 and 18 DF,  p-value: 4.786e-09
```

The summary table shows that the full model is statistically significant at 5% significance level (p-value=0.000). The adjusted R^2 is higher (0.9107). 91.07% of the variance in the dependent variable is predictable from the independent variable.

None the predictors have significant effects on the average value of response at 0.05 significance level (all the coefficients have p-value > 0.2). We need to remove some terms.

Box-cox analysis

```
bc<- MASS::boxcox(model1)
bc$x[which.max(bc$y)]
## [1] -2
# bc$x[bc$y > max(bc$y) - 1/2 * qchisq(.95,1)]
```

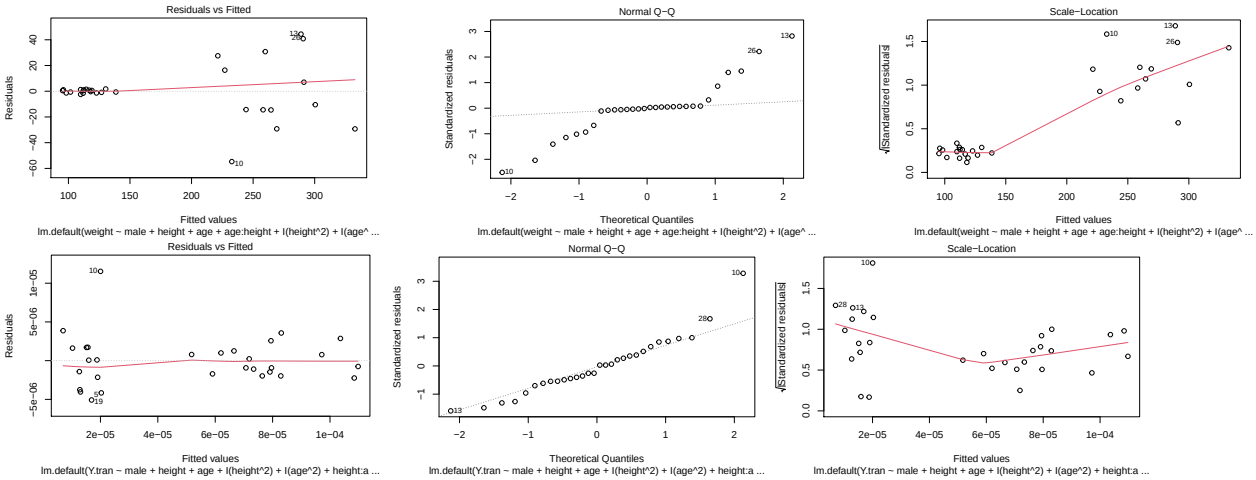


If the variances are unequal and/or error terms are not normal, we can use Box-Cox Transformation, a procedure for estimating an appropriate value for λ.

λ is given a "meaningful" number between -1 and 2, such as -1, -0.5, 0, 0.5, (1), 1.5, and 2. Then, we transform the response by taking it to a power λ. That is $y^\star = y^\lambda$. When $\lambda = 0$, the transformation is taken to be the natural log transformation. That is $y^\star = \ln(y)$. The result shows we can let $\lambda = -2$. We need to transform response variable. We refit the model with Y^{-2}

```
data1$Y.tran <- data1$weight^(-2)
model2 <- update(model1,Y.tran ~ .)
summary(model2)
##
## Call:
## lm.default(formula = Y.tran ~ male + height + age + I(height^2) +
##     I(age^2) + height:age + male:height + male:age + male:I(height^2) +
##     male:I(age^2) + male:height:age, data = data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -5.078e-06 -1.906e-06 -3.446e-07  1.527e-06  1.153e-05
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.242e-02  4.844e-03   2.565  0.01949 *
## malemale             -1.882e-02  6.045e-03  -3.114  0.00600 **
## height               -3.378e-04  1.364e-04  -2.477  0.02340 *
## age                  -4.982e-05  7.494e-05  -0.665  0.51467
## I(height^2)           2.304e-06  9.741e-07   2.365  0.02945 *
## I(age^2)             -1.350e-07  6.406e-07  -0.211  0.83550
## height:age            7.466e-07  8.073e-07   0.925  0.36729
## malemale:height       5.343e-04  1.737e-04   3.076  0.00651 **
## malemale:age          1.252e-05  8.350e-05   0.150  0.88249
## malemale:I(height^2) -3.813e-06  1.257e-06  -3.033  0.00715 **
## malemale:I(age^2)    -1.385e-08  7.869e-07  -0.018  0.98615
## malemale:height:age  -1.320e-07  9.186e-07  -0.144  0.88731
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.009e-06 on 18 degrees of freedom
## Multiple R-squared:  0.9919, Adjusted R-squared:  0.9869
## F-statistic: 199.9 on 11 and 18 DF,  p-value: < 2.2e-16
```

The adjusted R^2 is higher than before (.9869). A part of the coefficients are significant.



The transformations appear to have rectified the original problem (above) with the model since the fitted line of residual plot (below) now looks better.

Variable selection

- Stepwise Method

Build regression model from a set of candidate predictor variables by removing predictors based on Akaike Information Criteria, in a stepwise manner until there is no variable left to remove any more.

```
aic1 <- MASS::stepAIC(model2,direction = "both")
aic2 <- MASS::stepAIC(model2,direction = "backward")
aic3 <- MASS::stepAIC(model2,direction = "forward")
```

```
summary(aic1)
##
## Call:
## lm.default(formula = Y.tran ~ male + height + age + I(height^2) +
##     height:age + male:height + male:age + male:I(height^2), data = data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -4.733e-06 -1.902e-06 -4.065e-07  1.429e-06  1.156e-05
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.164e-02  3.676e-03   3.166  0.00465 **
## malemale            -1.840e-02  5.135e-03  -3.584  0.00175 **
## height              -3.134e-04  1.074e-04  -2.918  0.00823 **
## age                 -5.224e-05  2.237e-05  -2.336  0.02952 *
## I(height^2)          2.130e-06  7.883e-07   2.701  0.01337 *
## height:age           6.927e-07  3.357e-07   2.064  0.05162 .
## malemale:height      5.243e-04  1.515e-04   3.462  0.00233 **
## malemale:age         2.834e-06  1.272e-06   2.228  0.03692 *
## malemale:I(height^2) -3.754e-06  1.116e-06  -3.363  0.00295 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.737e-06 on 21 degrees of freedom
## Multiple R-squared:  0.9918, Adjusted R-squared:  0.9886
## F-statistic: 316.3 on 8 and 21 DF,  p-value: < 2.2e-16
```

Using “both sides”, “backward”, and “forward” AIC selection, there are still 9 coefficients. We can try other package.

```
ols_step_both_aic(model2)
##
##
##                                  Stepwise Summary
## -------------------------------------------------------------------------------------
## Variable              Method       AIC        RSS      Sum Sq      R-Sq       Adj. R-Sq
## -------------------------------------------------------------------------------------
## male:height          addition    -594.409    0.000      0.000     0.90624       0.89929
## male                 addition    -615.933    0.000      0.000     0.95720       0.95226
## male:age             addition    -648.182    0.000      0.000     0.98721       0.98455
## male:I(height^2)     addition    -651.860    0.000      0.000     0.99010       0.98695
## height:age           addition    -655.399    0.000      0.000     0.99177       0.98863
## -------------------------------------------------------------------------------------
```

Another AIC selection suggests the model contains six term: male, height, interaction of height and age, interaction of male and height, age, and height^2.

```
ols_step_both_p(model2)
```

- All possible subsets

```
p
##                                Best Subsets Regression
## ---------------------------------------------------------------------------------------
## Model Index    Predictors
## ---------------------------------------------------------------------------------------
##      1         male:height
##      2         male:height male:I(height^2)
##      3         male male:height male:age
##      4         male male:height male:age male:I(height^2)
##      5         male I(height^2) male:height male:age male:I(height^2)
##      6         male age I(height^2) male:height male:age male:I(height^2)
##      7         male height age height:age male:height male:age male:I(height^2)
##      8         male height age I(height^2) height:age male:height male:age male:I(height^2)
## ---------------------------------------------------------------------------------------
##
##                                        Subsets Regression Summary
## ---------------------------------------------------------------------------------------------------------------------------------
##                        Adj.        Pred
## Model    R-Square    R-Square    R-Square      C(p)         AIC          SBIC          SBC         MSEP        FPE         HSP         APC
## ---------------------------------------------------------------------------------------------------------------------------------
##   1        0.9062      0.8993      0.8813    213.2553    -594.4091     -679.7991    -588.8043    0.0000      0.0000      0.0000      0.1072
##   2        0.9576      0.9508      0.9433     84.2768    -614.1943     -701.3952    -605.7871    0.0000      0.0000      0.0000      0.0519
##   3        0.9872      0.9846      0.9808     10.6253    -648.1824     -893.3083    -638.3740    0.0000      0.0000      0.0000      0.0167
##   4        0.9901      0.9870      0.9792      5.2588    -651.8598    -1021.9069    -639.2490    0.0000      0.0000      0.0000      0.0139
##   5        0.9901      0.9870      0.9792      7.2588    -651.8598     -990.7652    -639.2490    0.0000      0.0000      0.0000      0.0148
##   6        0.9901      0.9870      0.9792      9.2588    -651.8598     -959.6235    -639.2490    0.0000      0.0000      0.0000      0.0159
##   7        0.9918      0.9886       0.983      7.0000    -655.3994    -1087.4867    -641.3874    0.0000      0.0000      0.0000      0.0142
##   8        0.9918      0.9886       0.983      9.0000    -655.3994    -1050.0295    -641.3874    0.0000      0.0000      0.0000      0.0153
## ---------------------------------------------------------------------------------------------------------------------------------
## AIC: Akaike Information Criteria
##  SBIC: Sawa's Bayesian Information Criteria
##  SBC: Schwarz Bayesian Criteria
##  MSEP: Estimated error of prediction, assuming multivariate normality
##  FPE: Final Prediction Error
##  HSP: Hocking's Sp
##  APC: Amemiya Prediction Criteria
```

We should choose model 4, the subset of predictors that do the best at meeting some well-defined objective criterion, such as having the largest R^2_{adj} value or the smallest Mallow's C_p and AIC. The reduced model is indicator male, interaction terms between male and height, age, and age, and height^2.

```
model3 <- update(model2,.~male+height+age+I(height^2)+(height+age+I(height^2)):male)
```

- Partial Regression

```r
car::Anova(model1)
```

The ANOVA table with Type II tests shows the partial sum of squares explained by X1 is , given that all the other regression coefficients are in the model. It shows that the effects of male are more Important.

Brief Summary

```r
# ols_mallows_cp(model_2020s1_p3,model_2020s1_p1)
summary(model3)
##
## Call:
## lm.default(formula = Y.tran ~ male + height + age + I(height^2) +
##     male:height + male:age + male:I(height^2), data = data1)
##
## Residuals:
##         Min          1Q      Median          3Q         Max
## -5.358e-06 -2.160e-06 -4.780e-07  1.544e-06  1.207e-05
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           7.918e-03  3.432e-03   2.307   0.0309 *
## malemale             -1.348e-02  4.872e-03  -2.766   0.0113 *
## height               -2.172e-04  1.037e-04  -2.094   0.0480 *
## age                  -6.103e-06  7.651e-07  -7.977 6.19e-08 ***
## I(height^2)           1.523e-06  7.838e-07   1.943   0.0649 .
## malemale:height       3.779e-04  1.434e-04   2.636   0.0151 *
## malemale:age          4.682e-06  9.682e-07   4.835 7.85e-05 ***
## malemale:I(height^2) -2.675e-06  1.057e-06  -2.531   0.0190 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.004e-06 on 22 degrees of freedom
## Multiple R-squared:  0.9901, Adjusted R-squared:  0.987
## F-statistic: 314.4 on 7 and 22 DF,  p-value: < 2.2e-16
# ols_regress(model3)
# olsrr::ols_press(model3)
# ols_pred_rsq(model3)
```

Considering several criteria for variable selection such as MSE, Mallow's C_p or AIC, the recommended model using stepwise and all possible subset selection is:

$$Weight^{-2} = \beta_0 + \beta_1 Height + \beta_2 Age + \beta_3 Height^2 +$$

$$(\beta_4 + \beta_5 Height + \beta_6 Age + \beta_7 Height^2) Male + \varepsilon_i$$

- The comparison of residual plot shows transform of response is necessary.
- The fitted overall model is statistically significant at 5% significance level (p-value=2.2×10^{-16}).
- The F-test or the equivalent t-test test the null hypothesis $H_0 : \beta_k = 0$. When the P-value <0.05. There is significant evidence at the 0.05 level to conclude that there is a linear association between the X_k and the natural Y.

- Most of the coefficients are significant. The coefficients in the latest model suggests how many units the average Y^{-2} increases by when the X_i increases by 1 inch and other variables are constants.
- Although some model can give higher R^2, we would like to choose the model with less explanatory variables.

Check Adequacy

Independence Test

- Durbin-Watson test

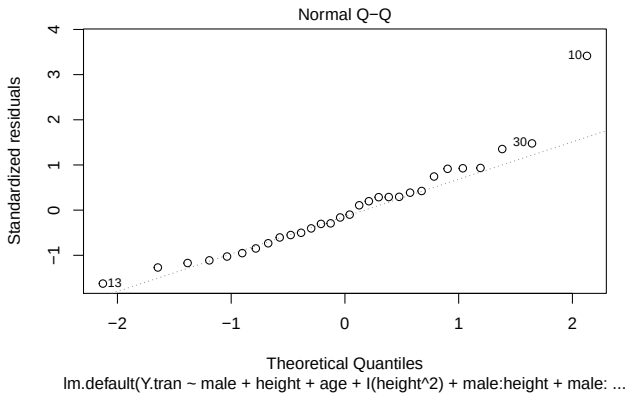
```
lmtest::dwtest(model3)
##
##  Durbin-Watson test
##
## data:  model3
## DW = 1.8058, p-value = 0.2743
## alternative hypothesis: true autocorrelation is greater than 0
```

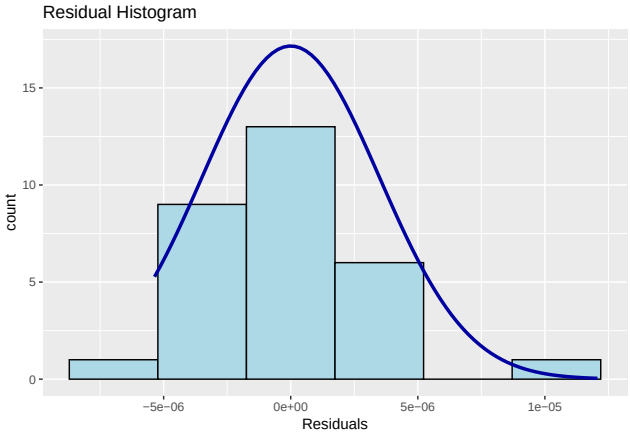
Durbin-Watson test fails to reject H_0. The model may have zero autocorrelation. The independent assumption is not violated.

Normality Test

- QQ plot

In the QQ plot (the residuals vs a set of hypothetical normal residuals), most of points follow approximately straight line. The plot seems to deviate from a straight line and curves up at the extreme percentiles. The right tail is longer than normal and the left tail is shorter. It shows a little violation of normal distribution assumption of residuals.





- Histogram

The histogram of the residuals shows a normal distribution or detects violation of normality assumption.

The histogram appears a litter right-skewed and does not show the ideal bell-shape for normality.

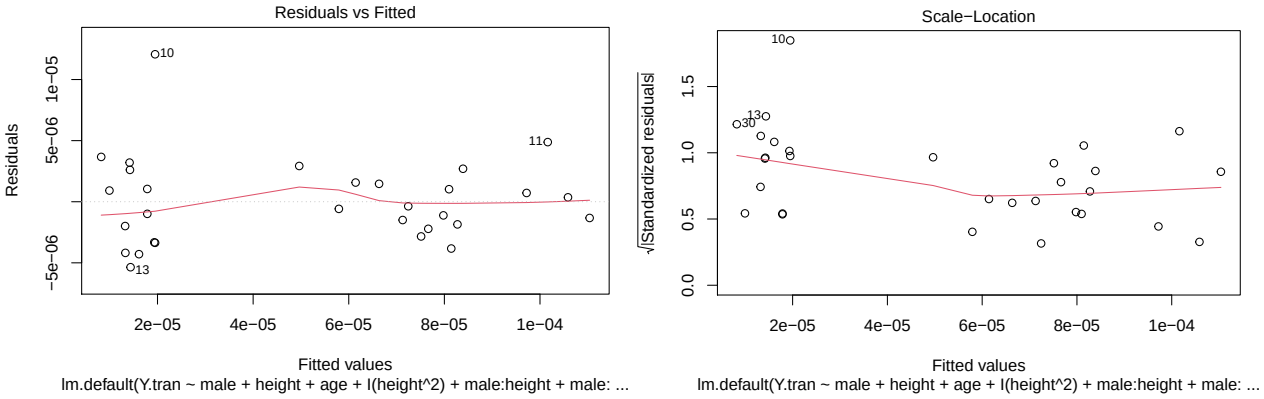
- Other normality tests

```
ols_test_normality(model3)
## -----------------------------------------------
##        Test             Statistic       pvalue
## -----------------------------------------------
## Shapiro-Wilk              0.9124         0.0171
## Kolmogorov-Smirnov        0.093          0.9365
## Cramer-von Mises          9.9999         0.0000
## Anderson-Darling          0.4767         0.2211
## -----------------------------------------------
# nortest::ad.test(residuals(model3))
```

Two out of four methods cannot reject the residuals are normal distributed.

Therefore, the transformation improved the assumption of normality. Due to small sample size, it is acceptable.

Heteroskedasticity Test



In the plots of residuals versus predicted value, there is a roughly even width band centered around zero. A horizontal red line with equally spread out points indicates constant variance. There isn't serious violation of the "equal variance" and "zero mean" assumption.

- Breusch Pagan Test:

It is used to test for herteroskedasticity (non-constant error variance). It tests whether the variance of the errors from a regression is dependent on the values of the independent variables. It is a χ^2 test.

```
ols_test_breusch_pagan(model3, rhs = F,multiple = T, p.adj ='sidak') # 'none','bonferroni', 'holm'
##
##  Breusch Pagan Test for Heteroskedasticity
##  -----------------------------------------
##  Ho: the variance is constant
##  Ha: the variance is not constant
##
##                Data
##  ---------------------------------
##  Response : Y.tran
##  Variables: fitted values of Y.tran
##
##          Test Summary
##  -----------------------------
##  DF            =    1
##  Chi2          =    4.663527
##  Prob > Chi2   =    0.03080985
bptest(model3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model3
## BP = 7.4323, df = 7, p-value = 0.3853
```

- Score Test

Test for heteroskedasticity under the assumption that the errors are independent and identically distributed (i.i.d.).

```
ols_test_score(model3, rhs = TRUE)
##
##  Score Test for Heteroskedasticity
##  ---------------------------------
##  Ho: Variance is homogenous
##  Ha: Variance is not homogenous
##
##  Variables: malemale height age I(height^2) malemale:height malemale:age malemale:I(height^2)
##
##         Test Summary
##  ----------------------------
##  DF             =    7
##  Chi2           =    7.432252
##  Prob > Chi2    =    0.3853014
ncvTest(model3)
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 4.663527, Df = 1, p = 0.03081
```

- F Test

F Test for heteroskedasticity under the assumption that the errors are independent and identically distributed (i.i.d.).

```
ols_test_f(model3, rhs = TRUE)
##
##  F Test for Heteroskedasticity
##  -----------------------------
##  Ho: Variance is homogenous
##  Ha: Variance is not homogenous
##
##  Variables: malemale height age I(height^2) malemale:height malemale:age malemale:I(height^2)
##
##       Test Summary
##  -------------------------
##  Num DF     =    7
##  Den DF     =    22
##  F          =    1.035039
##  Prob > F   =    0.4355911
```

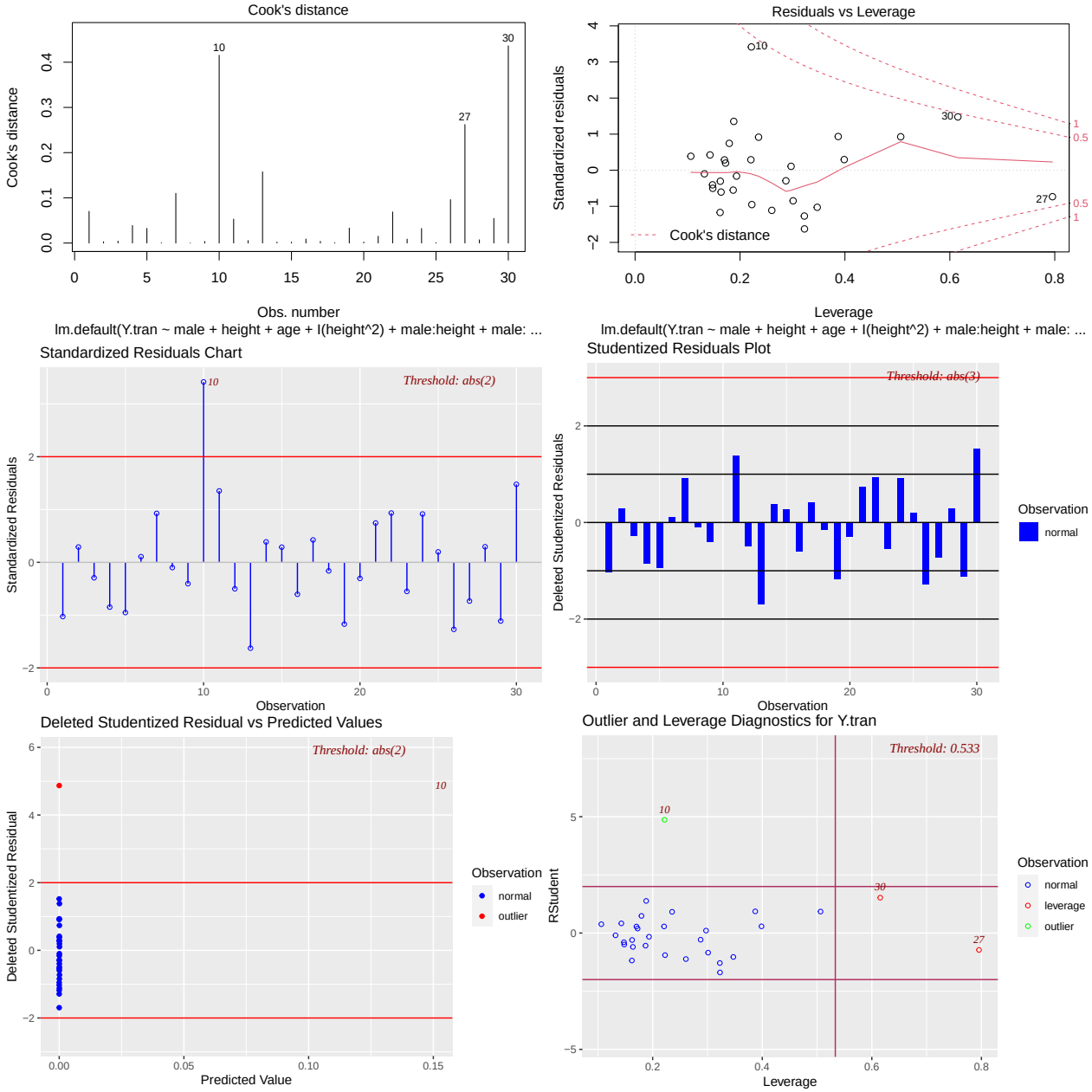
The above tests cannot reject that the variance is constant if controlling the variable of male.

- Weighted Least Squares

```
regRes <- lm(abs(model3$residuals)~model3$fitted.values)
weights = 1/regRes$fitted.values^2
wls <-  update(model3, weights=weights)
plot(model3)
plot(wls)
```

Regress the abs values of OLS residuals versus OLS fitted values. Store the fitted values from this regression. These fitted values are estimates of the error standard deviations.

Outlier and leverage points



- Cook's D Chart

Chart of Cook's distance to detect three observations (#10,#27,#30) that strongly influence fitted values of the model. It depends on both the residual and leverage i.e it takes it account both the x value and y value of the observation.

- Studentized Residuals vs Leverage Plot

Studentized deleted residuals is the deleted residual divided by its estimated standard deviation. Studentized residuals are going to be more effective for detecting outlying Y observations than standardized residuals. If an observation has an externally studentized residual that is larger than 2 (in absolute value) we can call it an outlier.

There are one outlier (#10) and two leverage points (#27,#13). These points can severely affect normality and homogeneity of variance. It can be useful to remove outliers to meet the test assumptions.

It's not really okay to remove some data points just to make the model work better. Keeping or removing them should be context-dependent.

Check Multicollinearity again

```
car::vif(model3)
##             male           height              age      I(height^2)
##     1.090457e+07     8.115123e+04     3.194781e+00     8.567607e+04
##      male:height         male:age male:I(height^2)
##     4.556772e+07     2.200713e+02     1.197413e+07
```

The quadratic and interaction terms will produce high VIF. We can fix it by fitting an orthogonal polynomial model using the argument of 'raw=T'

```
model4 <- lm(weight^(-2) ~ male+poly(height,2,raw=T)+age+age:male+poly(height,2,raw=T):male, data1)
```

```
summary(model3)$coefficient
##                           Estimate   Std. Error   t value     Pr(>|t|)
## (Intercept)           7.917619e-03 3.432477e-03  2.306678 3.086950e-02
## malemale             -1.347677e-02 4.871631e-03 -2.766377 1.126187e-02
## height               -2.171557e-04 1.036898e-04 -2.094283 4.797231e-02
## age                  -6.103441e-06 7.651067e-07 -7.977242 6.188941e-08
## I(height^2)           1.523147e-06 7.837640e-07  1.943375 6.487391e-02
## malemale:height       3.779364e-04 1.433812e-04  2.635886 1.509360e-02
## malemale:age          4.681672e-06 9.682112e-07  4.835383 7.854662e-05
## malemale:I(height^2) -2.675404e-06 1.057033e-06 -2.531050 1.902892e-02
summary(model4)$coefficient
##                                       Estimate   Std. Error   t value
## (Intercept)                       7.917619e-03 3.432477e-03  2.306678
## malemale                         -1.347677e-02 4.871631e-03 -2.766377
## poly(height, 2, raw = T)1        -2.171557e-04 1.036898e-04 -2.094283
## poly(height, 2, raw = T)2         1.523147e-06 7.837640e-07  1.943375
## age                              -6.103441e-06 7.651067e-07 -7.977242
## malemale:age                      4.681672e-06 9.682112e-07  4.835383
## malemale:poly(height, 2, raw = T)1  3.779364e-04 1.433812e-04  2.635886
## malemale:poly(height, 2, raw = T)2 -2.675404e-06 1.057033e-06 -2.531050
##                                       Pr(>|t|)
## (Intercept)                       3.086950e-02
## malemale                          1.126187e-02
## poly(height, 2, raw = T)1         4.797231e-02
## poly(height, 2, raw = T)2         6.487391e-02
## age                               6.188941e-08
## malemale:age                      7.854662e-05
## malemale:poly(height, 2, raw = T)1 1.509360e-02
## malemale:poly(height, 2, raw = T)2 1.902892e-02
```

The model summaries show that the new model can give the same values of the raw coefficients.

```
car::vif(model4)
##                                      GVIF Df GVIF^(1/(2*Df))
## male                         1.090457e+07  1     3302.207473
## poly(height, 2, raw = T)     1.981995e+02  2        3.752111
## age                          3.194781e+00  1        1.787395
## male:age                     2.200713e+02  1       14.834801
## male:poly(height, 2, raw = T) 3.152362e+08  2      133.247488
```

The new vif of the orthogonal polynomial model are not as serious as before.

Brief Summary

- It appears that the updated model with weight^{-2} as the response performs better.
- The relationship appears to be linear and the error terms appear independent and normally distributed with equal variances.
- Further information is needed for discussing the outlier and leverage points.
- The male and female groups are not identical, this model is the best one for the whole data.

Estmation and Prediction

```
pred <- (predict(model4, data.frame(age=26,height=70,male="male"),
             interval="predict", level=0.95 ))^(-0.5) # "confidence"
confint(model4, level=1-(0.05/6)) #
##                                                0.417 %      99.583 %
## (Intercept)                             -2.031543e-03  1.786678e-02
## malemale                                -2.759737e-02  6.438349e-04
## poly(height, 2, raw = T)1               -5.177042e-04  8.339292e-05
## poly(height, 2, raw = T)2               -7.486219e-07  3.794916e-06
## age                                     -8.321132e-06 -3.885751e-06
## malemale:age                             1.875276e-06  7.488068e-06
## malemale:poly(height, 2, raw = T)1 -3.765917e-05  7.935320e-04
## malemale:poly(height, 2, raw = T)2 -5.739254e-06  3.884462e-07
confint(model4,adjust.method = "bonferroni")
##                                                2.5 %        97.5 %
## (Intercept)                              7.990969e-04  1.503614e-02
## malemale                                -2.357991e-02 -3.373623e-03
## poly(height, 2, raw = T)1               -4.321950e-04 -2.116271e-06
## poly(height, 2, raw = T)2               -1.022801e-07  3.148574e-06
## age                                     -7.690176e-06 -4.516707e-06
## malemale:age                             2.673725e-06  6.689620e-06
## malemale:poly(height, 2, raw = T)1  8.058210e-05  6.752908e-04
## malemale:poly(height, 2, raw = T)2 -4.867557e-06 -4.832510e-07
```

Using the latest model, when age=26,height=70,male="male", the prediction interval is (209.7266614, 670.1934693) with fitted value of 283.0620907 .

Note:

We should be careful with the extrapolation problems. If using fitted model to predict response 'weight' at a predictor 'age' value out of observed range of age, the results are less reliable.

Each variable should have a reasonable range. In this case the upper limits is not reasonable.

Male group

```
data_male <- data1[data1$male=="male",]
data_female <- data1[data1$male=="female",]
```

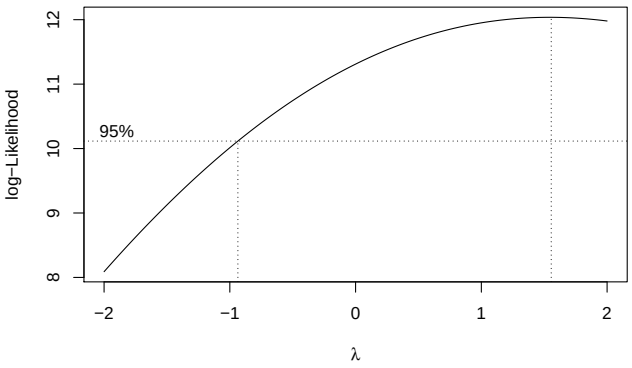
```
model_m <- lm(weight ~ height+I(height^2)+age+I(age^2)+age:height,  data_male)
```

```
summary(model_m)
##
## Call:
## lm.default(formula = weight ~ height + I(height^2) + age + I(age^2) +
##     age:height, data = data_male)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -54.71 -14.55 -10.48  27.55  44.39
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 40273.4353 35918.2293   1.121    0.299
## height      -1279.9242  1068.6013  -1.198    0.270
## I(height^2)    10.1256     7.8931   1.283    0.240
## age           392.4716   365.7958   1.073    0.319
## I(age^2)        0.4538     4.5385   0.100    0.923
## height:age     -5.7804     4.3529  -1.328    0.226
##
## Residual standard error: 39.82 on 7 degrees of freedom
## Multiple R-squared:  0.5311, Adjusted R-squared:  0.1962
## F-statistic: 1.586 on 5 and 7 DF,  p-value: 0.2791
```

Very low R^2 values

- Box-cox analysis

```
bc_m <- boxcox(model_m)
bc_m$x[which.max(bc_m$y)]
## [1] 1.555556
```



The result shows $\lambda = 1.5$. We can transform response variable as Weight$^{1.5}$.

```
model_m1 <- update(model_m,weight^(3/2) ~.)
summary(model_m1)
##
## Call:
## lm.default(formula = weight^(3/2) ~ height + I(height^2) + age +
```

```
##     I(age^2) + height:age, data = data_male)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1211.8  -405.7  -254.3   649.8  1130.5
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 935891.57  870003.61   1.076    0.318
## height      -30054.98   25883.43  -1.161    0.284
## I(height^2)    239.10     191.18   1.251    0.251
## age           9888.25    8860.23   1.116    0.301
## I(age^2)         8.14     109.93   0.074    0.943
## height:age    -143.62     105.44  -1.362    0.215
##
## Residual standard error: 964.6 on 7 degrees of freedom
## Multiple R-squared:  0.5345, Adjusted R-squared:  0.2021
## F-statistic: 1.608 on 5 and 7 DF,  p-value: 0.2737
```

- Stepwis Selcetion

```
summary(aic_m)
##
## Call:
## lm.default(formula = weight^(3/2) ~ height + I(height^2) + age +
##     height:age, data = data_male)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1216.4  -406.8  -239.5   672.6  1101.1
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 948643.78  798023.72   1.189    0.269
## height      -30538.37   23438.23  -1.303    0.229
## I(height^2)    242.56     173.46   1.398    0.200
## age           10259.59    6835.47   1.501    0.172
## height:age     -143.69      98.66  -1.456    0.183
##
## Residual standard error: 902.7 on 8 degrees of freedom
## Multiple R-squared:  0.5342, Adjusted R-squared:  0.3013
## F-statistic: 2.294 on 4 and 8 DF,  p-value: 0.1477
```

Using stepwise selection, the recommended model includes height, height^2, age, and interaction of age and height.

- All possible subsets

```
p
##                 Best Subsets Regression
## ---------------------------------------------------
## Model Index    Predictors
## ---------------------------------------------------
##      1         I(age^2)
##      2         I(height^2) I(age^2)
##      3         I(height^2) age height:age
##      4         height I(height^2) age height:age
##      5         height I(height^2) age I(age^2) height:age
## ---------------------------------------------------
##
##                                                    Subsets Regression Summary
## ----------------------------------------------------------------------------------------------------------------------------------------
##                        Adj.        Pred
## Model    R-Square    R-Square    R-Square     C(p)       AIC         SBIC        SBC          MSEP             FPE            HSP          APC
## ----------------------------------------------------------------------------------------------------------------------------------------
##   1        0.3611      0.3030      0.1799    0.6092   217.6281    182.2152   219.3230   10591658.6590   937868.2402   81281.9141   0.8713
##   2        0.3657      0.2388     -0.0369    2.5393   219.5332    187.1392   221.7930   11682924.6789  1092429.3206   98622.0914   1.0149
##   3        0.4353      0.2471     -0.0088    3.4921   220.0215    191.0782   222.8463   11700435.5744  1148094.6884  109744.3452   1.0666
##   4        0.5342      0.3013     -0.3168    4.0055   219.5198    195.1209   222.9095   11031092.8545  1128179.9510  116399.5188   1.0481
##   5        0.5345      0.2021     -1.0009    6.0000   221.5096    202.0029   225.4643   12859534.9494  1359913.7145  155077.8797   1.2634
## ----------------------------------------------------------------------------------------------------------------------------------------
## AIC: Akaike Information Criteria
##  SBIC: Sawa's Bayesian Information Criteria
##  SBC: Schwarz Bayesian Criteria
##  MSEP: Estimated error of prediction, assuming multivariate normality
##  FPE: Final Prediction Error
##  HSP: Hocking's Sp
##  APC: Amemiya Prediction Criteria
```

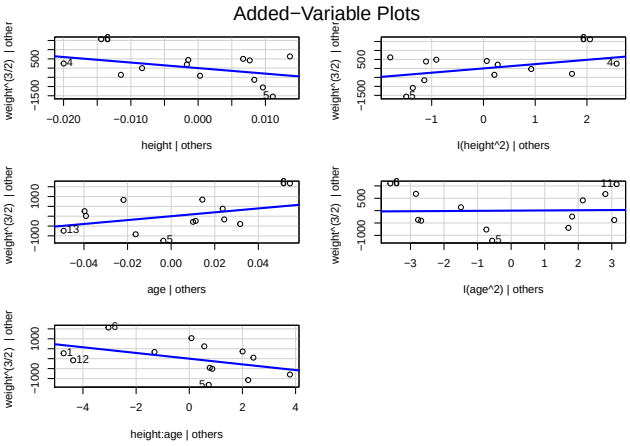
We should choose model 1, a subset with maximum R^2_{adj} and minimum Mallows's C_p and AIC.

The reduced model is $Weight^2 = \beta_0 age + \beta_1 age^2 + \varepsilon$.

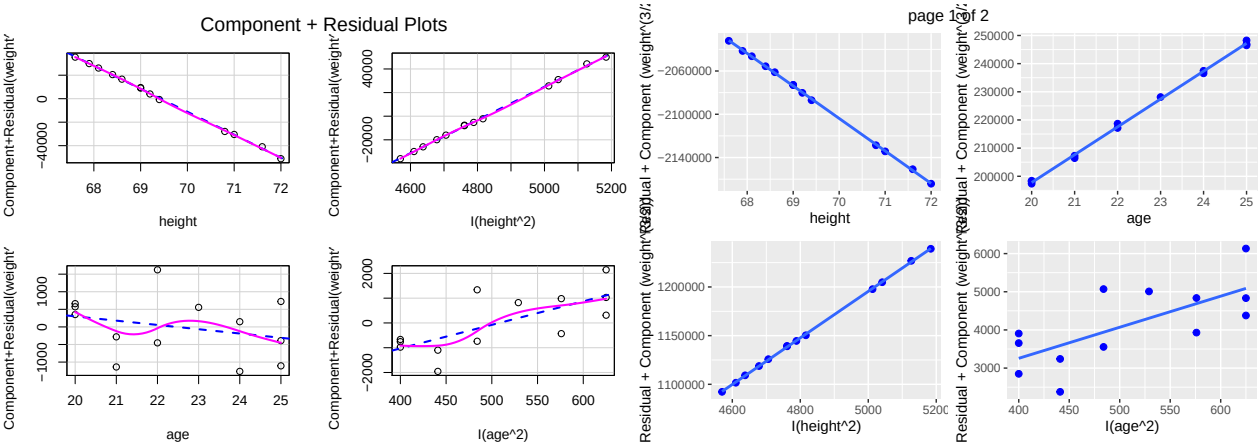
- Partial Regression

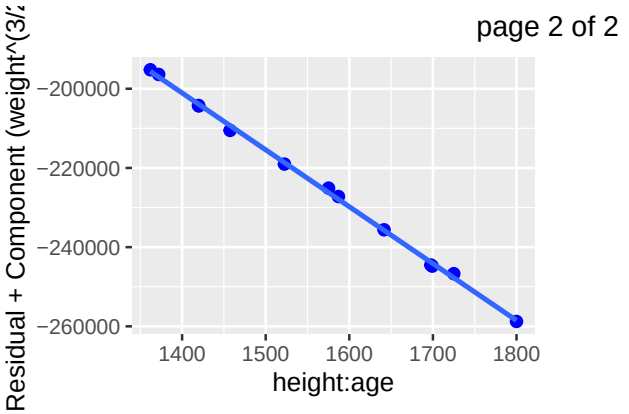
```
car::Anova(model_m1)
## Anova Table (Type II tests)
##
## Response: weight^(3/2)
##                Sum Sq Df F value Pr(>F)
## height         584069  1  0.6277 0.4542
## I(height^2) 1455266  1  1.5640 0.2513
## age               564  1  0.0006 0.9811
## I(age^2)         5102  1  0.0055 0.9430
## height:age    1726405  1  1.8554 0.2154
## Residuals    6513271  7
```

The ANOVA table with Type II tests shows that none the effects are significant.



The partial-regression plots shows age² are relatively less important.



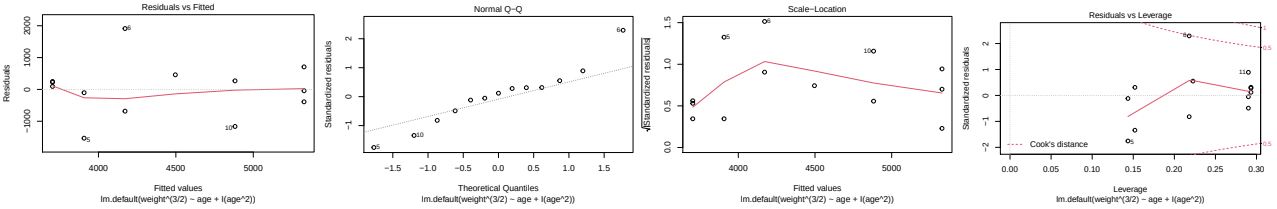


Component+Residual (Partial Residual) Plots

A partial residual plot essentially attempts to model the residuals of one predictor against the dependent variable. A component residual plot adds a line indicating where the line of best fit lies. The Y residuals represent the part of Y not explained by all the variables other than X. The X residuals represent the part of X not explained by other variables.

```
model_m2 <- update(model_m1,.~age+I(age^2))
summary(model_m2)
##
## Call:
## lm.default(formula = weight^(3/2) ~ age + I(age^2), data = data_male)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1532.45  -389.58    94.18   269.19  1912.41
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12339.70   52207.76   0.236    0.818
## age         -1037.22    4664.30  -0.222    0.828
## I(age^2)       30.27     103.52   0.292    0.776
##
## Residual standard error: 943.2 on 10 degrees of freedom
## Multiple R-squared:  0.3642, Adjusted R-squared:  0.237
## F-statistic: 2.864 on 2 and 10 DF,  p-value: 0.1039
```

- Residual Diagnosis



The residual plot doesn't show serious violation of assumption.

- Prediction

```
pred.m <- predict(model_m2,data.frame(age=26,height=70,male="male"),interval = "prediction",level=0.95)^(2/3)
```

```
compare <- rbind(pred[c(1,3,2)],pred.m)
rownames(compare) <- c("whole data","male group")
compare
##                  fit      lwr      upr
## whole data 283.0621 209.7267 670.1935
## male group 324.0987 194.3925 431.3605
```

Using the male model, when age=26,height=70, the prediction interval is (194.3925003, 431.3604931) with fitted value of 324.0987468 .

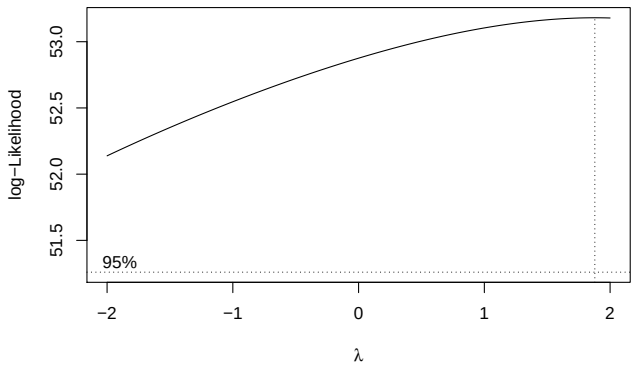
Although the model for male group has a low R^2 value, the prediction interval is more reasonable than the whole data model.

Female group

```
model_f <- lm(weight ~ height+I(height^2)+age+I(age^2)+age:height,  data_female)
```

- Box-cox analysis

```
bc_f <- boxcox(model_f)
bc_f$x[which.max(bc_f$y)]
## [1] 1.878788
```



The result shows $\lambda = 2$. We can transform response variable as Weight^2.

```
model_f1 <- update(model_f,weight^(2) ~.)
```

- Stepwis Selcetion

```
summary(aic_f)
##
## Call:
## lm.default(formula = weight^(2) ~ height + I(height^2) + age +
##     I(age^2) + height:age, data = data_female)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -546.5 -241.8  101.7  199.7  439.9
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 837089.90  408927.03   2.047  0.06531 .
## height      -21644.29   11513.33  -1.880  0.08685 .
## I(height^2)    142.42      82.23   1.732  0.11119
## age         -17915.41    6326.48  -2.832  0.01632 *
## I(age^2)        76.67      54.08   1.418  0.18400
## height:age     233.95      68.15   3.433  0.00559 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 338.4 on 11 degrees of freedom
## Multiple R-squared:  0.9897, Adjusted R-squared:  0.985
## F-statistic: 211.1 on 5 and 11 DF,  p-value: 1.542e-10
```

Using stepwise selection, the recommended model includes all the terms.

- All possible subsets

```
p
##                  Best Subsets Regression
## ------------------------------------------------------------
## Model Index    Predictors
## ------------------------------------------------------------
##      1         I(height^2)
##      2         age height:age
##      3         age I(age^2) height:age
##      4         height I(height^2) age height:age
##      5         height I(height^2) age I(age^2) height:age
## ------------------------------------------------------------
##
##                                                    Subsets Regression Summary
## ------------------------------------------------------------------------------------------------------------------------------------------------------
##                       Adj.        Pred
## Model     R-Square   R-Square    R-Square     C(p)        AIC         SBIC         SBC          MSEP             FPE           HSP          APC
## ------------------------------------------------------------------------------------------------------------------------------------------------------
##   1         0.7026     0.6827     0.6125    304.1572    302.0160     252.0276    304.5156    41210016.5614    2706633.1865    172980.3164    0.3768
##   2         0.9840     0.9818     0.9748      6.0096    254.2805      38.3507    257.6133     2380167.4773     163715.2233     10704.4569    0.0228
##   3         0.9854     0.9820     0.9748      6.6162    254.8274      21.3397    258.9935     2367280.3608     169958.5900     11465.4604    0.0237
##   4         0.9878     0.9837     0.9743      6.0097    253.7230     -65.2431    258.7223     2151447.9917     160694.5722     11288.4617    0.0224
##   5         0.9897     0.9850     0.9726      6.0000    252.8704    -181.4284    258.7029     2001008.3995     154960.2464     11453.5834    0.0216
## ------------------------------------------------------------------------------------------------------------------------------------------------------
## AIC: Akaike Information Criteria
##  SBIC: Sawa's Bayesian Information Criteria
##  SBC: Schwarz Bayesian Criteria
##  MSEP: Estimated error of prediction, assuming multivariate normality
##  FPE: Final Prediction Error
##  HSP: Hocking's Sp
##  APC: Amemiya Prediction Criteria
```

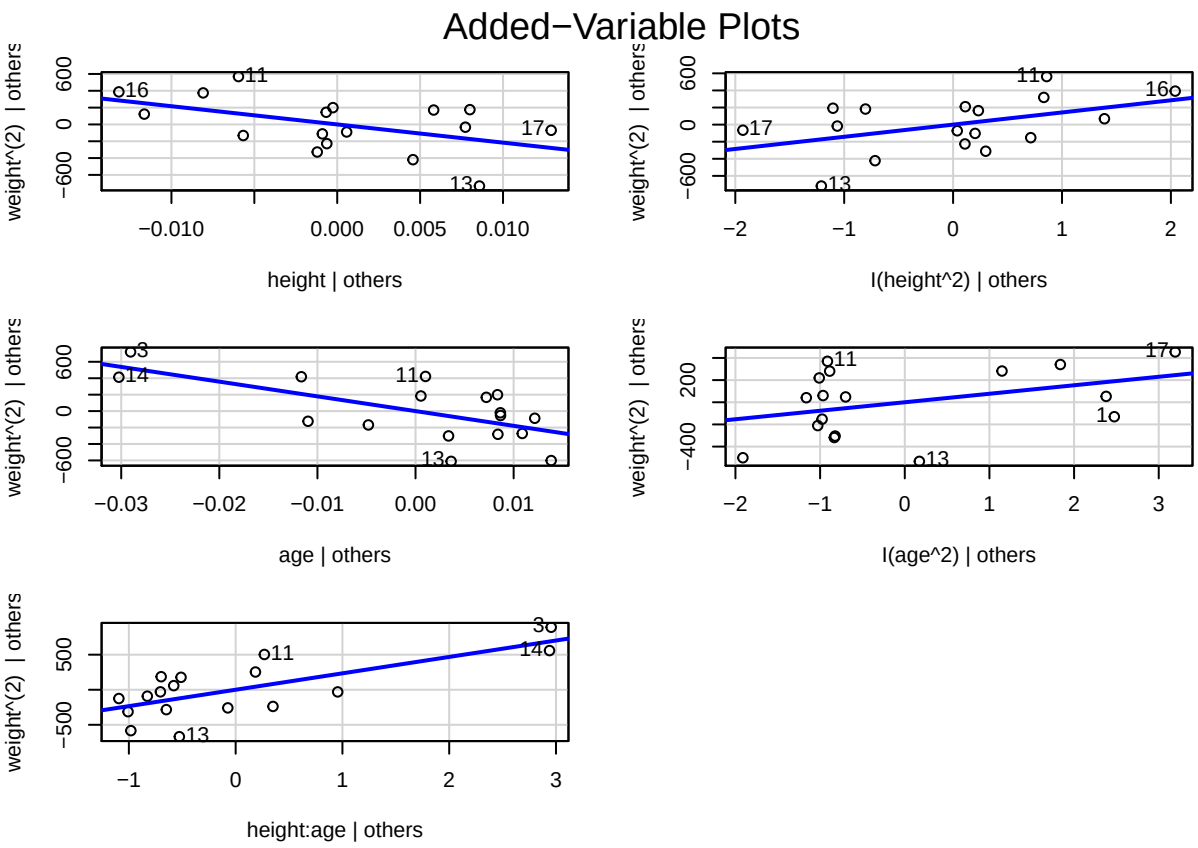
We should choose model 2, a subset with maximum R^2_{adj} and minimum Mallows's C_p and AIC.

The reduced model includes age and interaction of height and age.

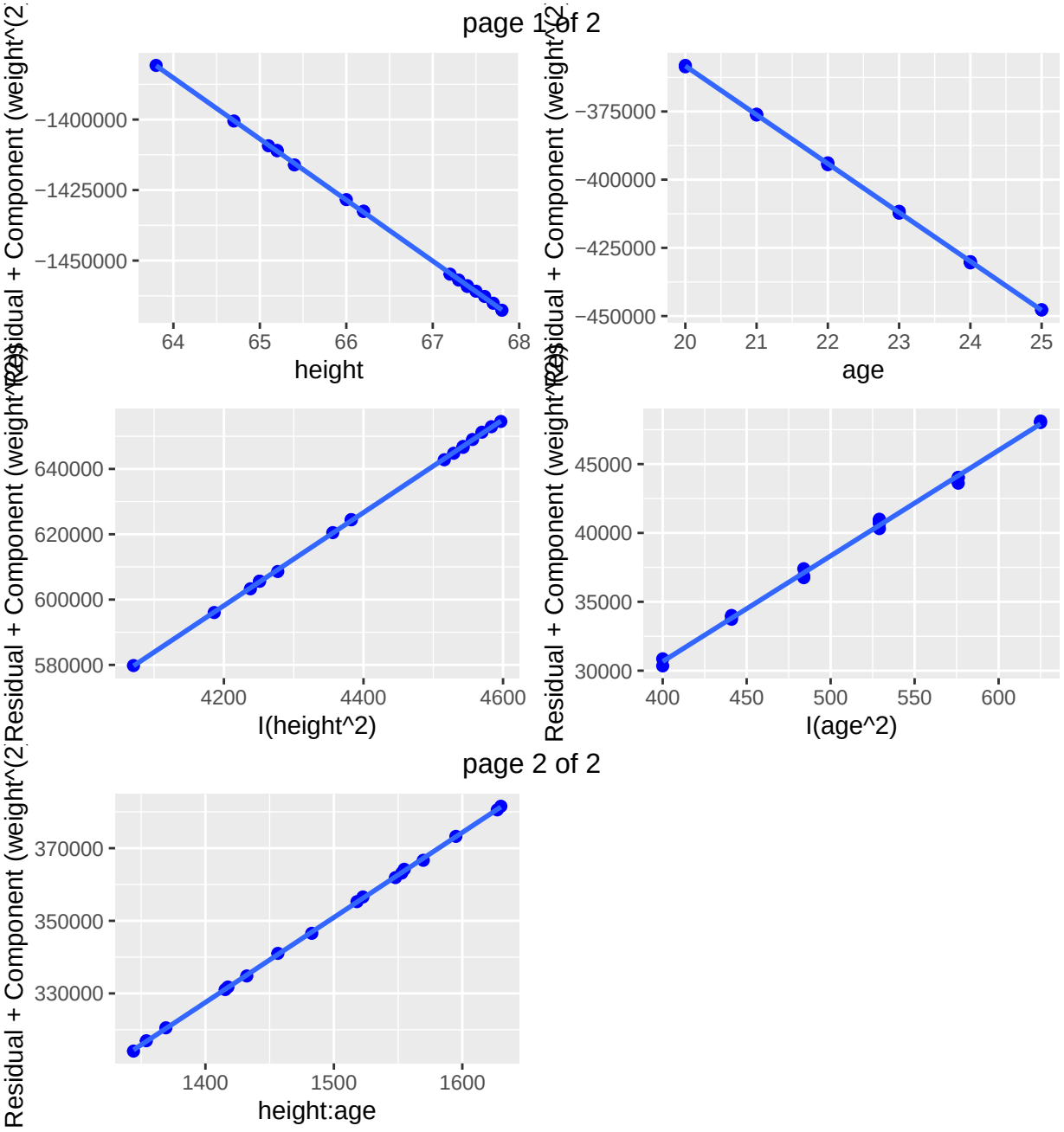
- Partial Regression

```
car::Anova(model_f1)
## Anova Table (Type II tests)
##
## Response: weight^(2)
##              Sum Sq Df F value   Pr(>F)
## height        14129  1  0.1234 0.732054
## I(height^2)  343564  1  2.9996 0.111193
## age          254506  1  2.2221 0.164158
## I(age^2)     230183  1  2.0097 0.183995
## height:age  1349838  1 11.7853 0.005594 **
## Residuals   1259894 11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA table with Type II tests shows that the interaction effects are significant.



The partial-regression plots shows these terms are equally important.



Component+Residual (Partial Residual) Plots

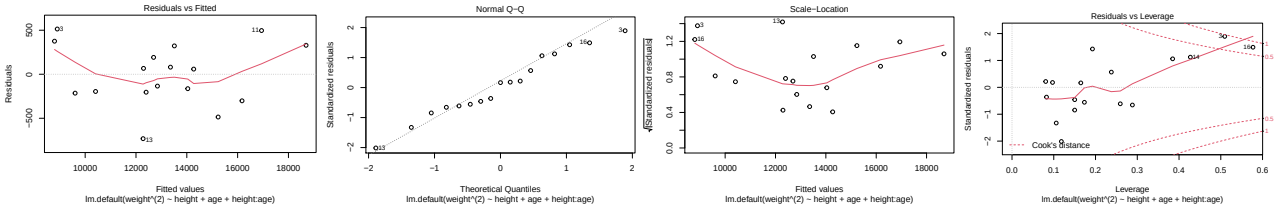
A partial residual plot essentially attempts to model the residuals of one predictor against the dependent variable. A component residual plot adds a line indicating where the line of best fit lies. The Y residuals represent the part of Y not explained by all the variables other than X. The X residuals represent the part of X not explained by other variables.

```
model_f2 <- update(model_f1,.~height+age+height:age)
summary(model_f2)
##
## Call:
## lm.default(formula = weight^(2) ~ height + age + height:age,
##     data = data_female)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -731.86 -204.04   58.42  321.90  513.62
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -9401.82   77159.85  -0.122   0.9049
## height        -24.60    1159.72  -0.021   0.9834
## age         -6378.53    3313.68  -1.925   0.0764 .
## height:age    112.37      49.88   2.253   0.0422 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 387.1 on 13 degrees of freedom
## Multiple R-squared:  0.984,  Adjusted R-squared:  0.9804
## F-statistic: 267.3 on 3 and 13 DF,  p-value: 6.279e-12
```

The female model has a very high R^2 value (0.9804). The female model is statistically significant at 5% significance level. The the coefficients of age and interaction of age vs. height are significant.

- Residual Diagnosis



The residual plots show some violation of assumption. Due to the small sample size, it is acceptable.

Conclusion

- There are many possible models with different predictor combinations. It's better to give up some model fit than to lose clear interpretations.
- Comparing to the old model, the redueced polynomial model has a higher adjusted R square and higher prediction R-square, which means it shows stronger predictive capability. All the coefficients in new model are statistically significant higher than 95% significance level. The transformation on response is necessary. Overall, the polynomial model is acceptable.
- The male group and female group are not identical. Fitting two models for the subgroup respectively is necessary.
- Although the male model has a bad performance on goodness-of-fit, its prediction interval is more reasonable than the polynomial model.
- Notice that the given new value of age (26) is outside the original range of age in the data. The prediction is less reliable.
- The female model shows strong predictive capability.

Question 2 Factorial Design

Preload Data

Data Description

- Experiment Type

```
Design <- readxl::read_excel("qe_lab/DesignFall17.xlsx")
data2 <- gather(Design[c(2:4,6:8),c(2:4,6:8,10:12)])
names(data2) <- c("machine","y")
data2 <- data2[c("y","machine")]
data2$machine <- as.factor(c(rep("machine1",18),rep("machine2",18),rep("machine3",18)) )
data2$station <- as.factor(rep(c(rep("station1",6),rep("station2",6),rep("station3",6)),3) )
data2$power <- as.factor(rep(c(rep("power1",3),rep("power2",3)),9) )
# data2$rep <- as.factor(rep(c("rep1","rep2","rep3"),18))
str(data2)
## tibble [54 x 4] (S3: tbl_df/tbl/data.frame)
##  $ y      : num [1:54] 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 ...
```

```
# xtabs(~machine + station + power, data = data2)
ftable(machine + station ~ power, data = data2)
##         machine machine1                   machine2                   machine3
##         station station1 station2 station3 station1 station2 station3 station1 station2 station3
## power
## power1               3        3        3        3        3        3        3        3        3
## power2               3        3        3        3        3        3        3        3        3
```

This experiment includes three factors: machine, power setting, and station. Three same stations apply on all fixed power settings in three specific machines. The levels of power settings are similar but not identical for different machines. The power factor is nested in machines. Machine and station, power and station are crossed factors. This is a nested and crossed design.

The number of observations taken within each factor are the same. The design is balanced. All the factor combinations have one replication. The design is complete.

$$y_{ijk} = \mu + \tau_i + \beta_{j(i)} + \gamma_k + (\tau\gamma)_{ik} + (\beta\gamma)_{kj(i)} + \varepsilon_{ijk}$$

for $i = 1, 2, 3;\ j = 1, 2;\ k = 1, 2, 3$

μ is the overall true mean response;

τ_i is the fixed main effect of i^{th} machine;

$\beta_{j(i)}$ is the fixed effect of j^{th} level of power nested in i^{th} machine;

γ_k is the main fixed effect of k^{th} station;

$(\tau\gamma)_{ik}$ is the interaction effect of i^{th} machine and k^{th} station;

$(\beta\gamma)_{j(i)k}$ is the interaction fixed effect of k^{th} station and j^{th} level of power nested in i^{th} machine.

y_{ijk} is response value (for the l^{th} replication) for j^{th} level of power nested in i^{th} machine when k^{th} station is applied;

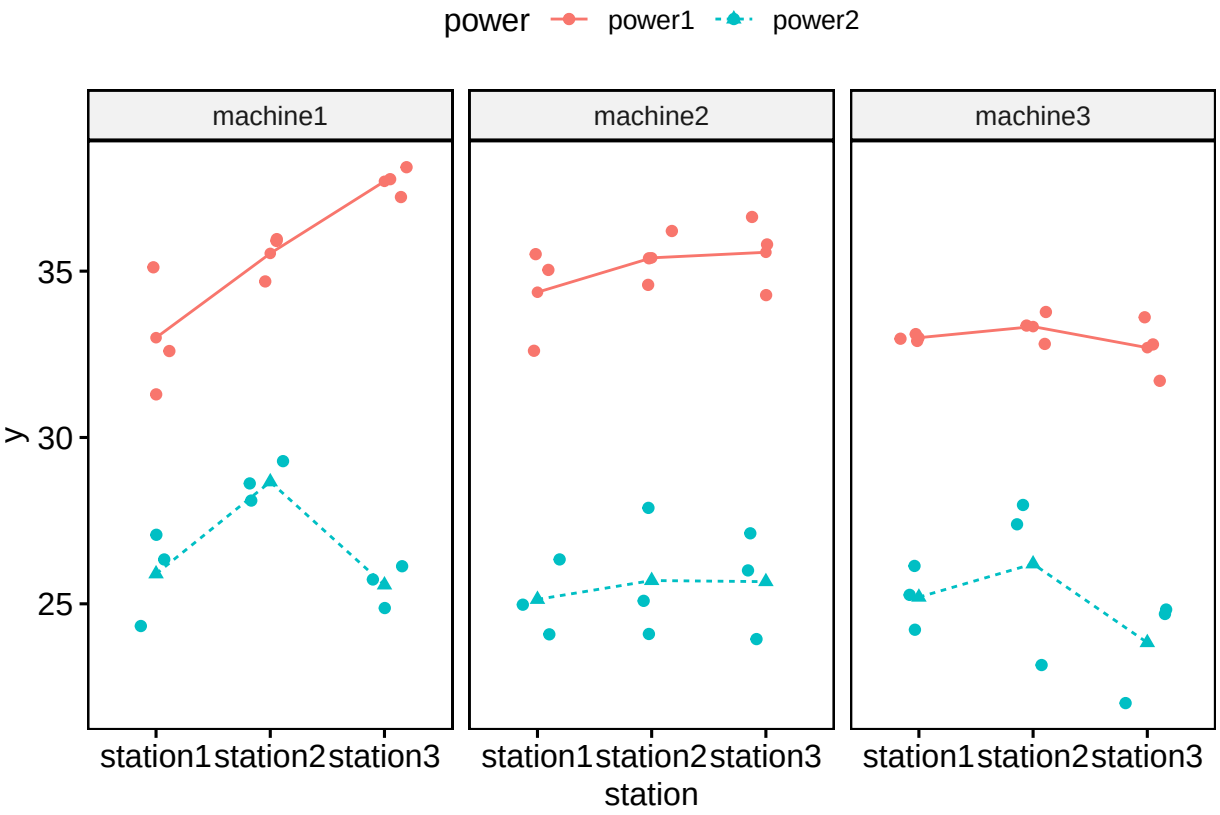
$\varepsilon_{(ijk)l}$ is random error (for the l^{th} replication) for j^{th} level of power nested in i^{th} machine when k^{th} station is applied.

Assumptions: $\varepsilon_{(ijk)l}$, $\beta_{j(i)}$, and $(\beta\gamma)_{j(i)k}$ are independent.

$\varepsilon_{ijk} \sim iidN(0, \sigma^2)$; $\sum_{i=1}^{2} \tau_i = 0$; $\sum_{k=1}^{3} \gamma_k = 0$; $\beta_{j(i)} \sim iidN(0, \sigma_\beta^2)$

$\sum_{i=1}^{2} (\tau\gamma)_{ik} = 0$; $\sum_{k=1}^{3} (\tau\gamma)_{ik} = 0$; $\sum_{i=1}^{2} (\beta\gamma)_{j(i)k} = 0$; $(\beta\gamma)_{j(i)k} \sim iidN(0, \frac{2-1}{2}\sigma_{\beta\gamma}^2)$

```
##    machine  min     Q1 median     Q3  max     mean       sd  n missing
## 1 machine1 24.3 26.500  30.30 35.700 38.1 31.06111 4.885409 18       0
## 2 machine2 23.9 25.325  30.25 35.300 36.6 30.30556 5.105530 18       0
## 3 machine3 22.0 24.925  29.85 32.975 33.8 29.04444 4.305840 18       0
##    power  min    Q1 median    Q3  max     mean       sd  n missing
## 1 power1 31.3 32.95   34.6 35.85 38.1 34.51111 1.831421 27       0
## 2 power2 22.0 24.50   25.7 27.10 29.3 25.76296 1.765731 27       0
##    station  min     Q1 median     Q3  max     mean       sd  n missing
## 1 station1 24.1 25.500  29.20 32.975 35.5 29.43333 4.316453 18       0
## 2 station2 23.2 27.925  31.05 34.675 36.2 30.80556 4.406609 18       0
## 3 station3 22.0 25.100  29.40 35.425 38.1 30.17222 5.623460 18       0
```



The above tables and plots show that: There is not much difference in the average yield from different machines. The average yield are very different between the two levels of power. There isn't clear pattern among the three stations.

Model Analysis

If someone think power is not nested in station. the ANOVA table is:

```
model.1 <- aov(y~machine+station+power+power:machine+
                      machine:station+power:station+
                      power:machine:station, data2)
anova(model.1) # only when all fixed factors.
## Analysis of Variance Table
##
## Response: y
##                        Df  Sum Sq Mean Sq  F value  Pr(>F)
## machine                 2   37.37   18.68  10.8913 0.00020 ***
## station                 2   16.98    8.49   4.9489 0.01262 *
## power                   1 1033.16 1033.16 602.2284 < 2e-16 ***
## machine:power           2    6.35    3.17   1.8505 0.17180
## machine:station         4   16.60    4.15   2.4195 0.06625 .
## station:power           2   16.30    8.15   4.7514 0.01475 *
## machine:station:power   4   12.91    3.23   1.8806 0.13507
## Residuals              36   61.76    1.72
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

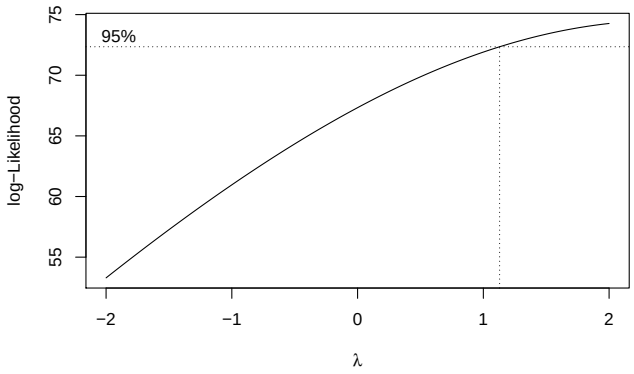
The corrected ANOVA table should combined the terms of 'power' and 'power:machine', combine the terms of 'station:power' and 'machine:power:station'.

```
data2$machine_f <- as.fixed(data2$machine)
data2$station_f <- as.fixed(data2$station)
data2$power_f <- as.fixed(data2$power)
# data2$rep_r <- as.random(data2$rep)
model.2 <- aov(y~power_f%in%machine_f+machine_f*station_f+power_f%in%machine_f:station_f, data2)
# gad(model.2)
anova(model.2)
## Analysis of Variance Table
##
## Response: y
##                               Df  Sum Sq Mean Sq  F value  Pr(>F)
## machine_f                      2   37.37   18.68  10.8913 0.00020 ***
## station_f                      2   16.98    8.49   4.9489 0.01262 *
## power_f:machine_f              3 1039.51  346.50 201.9765 < 2e-16 ***
## machine_f:station_f            4   16.60    4.15   2.4195 0.06625 .
## power_f:machine_f:station_f    6   29.21    4.87   2.8375 0.02292 *
## Residuals                     36   61.76    1.72
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA table show that all the main effects and the interaction effect are significant at 0.05 significance level except the interaction effect between machine and station (P-value=0.06625).

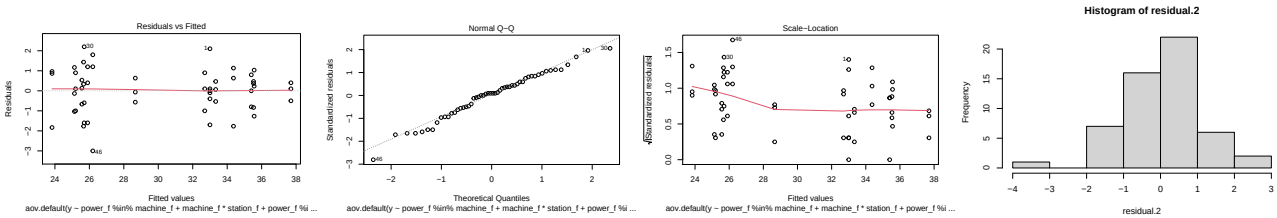
Transformation and Elimination

```
bc.2 <- boxcox(model.2)
model.3 <- update(model.2,y^2~.)
anova(model.3)
## Analysis of Variance Table
##
## Response: y^2
##                               Df  Sum Sq Mean Sq  F value    Pr(>F)
## machine_f                      2  147694   73847  13.2563 4.853e-05 ***
## station_f                      2   65167   32584   5.8491  0.006315 **
## machine_f:power_f              3 3792829 1264276 226.9511 < 2.2e-16 ***
## machine_f:station_f            4   66309   16577   2.9758  0.031995 *
## machine_f:station_f:power_f    6  121479   20246   3.6345  0.006378 **
## Residuals                     36  200545    5571
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

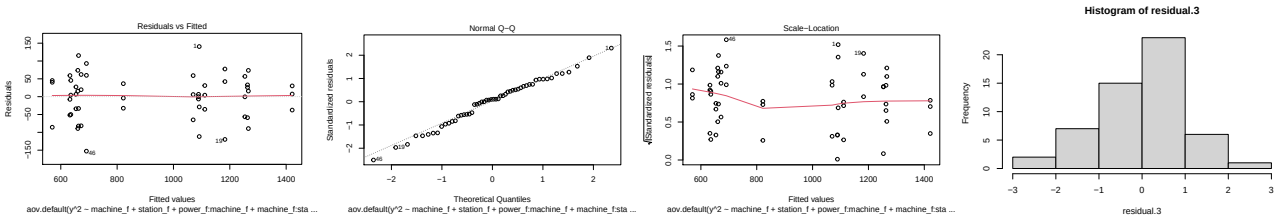


Using Box-cox transformation, we refit the model with Y^2. After a variance-stability transformation of the response, all the terms have significant effects at 5% significant level.

Check Adequacy




```
plot(model.3,1:3)
residual.3 <- rstudent(model.3)
hist(residual.3)
```



In the plots of residuals versus predicted value, there is no significant pattern on this plot. Therefore, the model is good enough to describe the effects of machine, power setting, station, and their interactions.

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 and the histogram of residuals is about normal. There is not serious violation of normal distribution assumption of residuals.

On account of the small sample size, the problems in the plots are not severe enough to have a dramatic impact on the analysis and conclusions.

Comparison

The Tables below show the summary of all those simple effect comparison tests.

- Tukey multiple pairwise-comparisons

The multiple pairwise-comparison between the means of groups.

```
# test(lsmeans(model.2,~machine_f,adjust=c("tukey")))
pairs(emmeans::lsmeans(model.3,~power_f,adjust=c("tukey")))
##  contrast                          estimate   SE df t.ratio p.value
##  power1 machine1 - power2 machine1    542.6 35.2 36  15.421 <.0001
##  power1 machine1 - power1 machine2     24.6 35.2 36   0.699 0.9809
##  power1 machine1 - power2 machine2    606.6 35.2 36  17.240 <.0001
##  power1 machine1 - power1 machine3    168.6 35.2 36   4.791 0.0004
##  power1 machine1 - power2 machine3    626.5 35.2 36  17.807 <.0001
##  power2 machine1 - power1 machine2   -518.0 35.2 36 -14.723 <.0001
##  power2 machine1 - power2 machine2     64.0 35.2 36   1.819 0.4669
##  power2 machine1 - power1 machine3   -374.0 35.2 36 -10.630 <.0001
##  power2 machine1 - power2 machine3     83.9 35.2 36   2.385 0.1885
##  power1 machine2 - power2 machine2    582.0 35.2 36  16.542 <.0001
##  power1 machine2 - power1 machine3    144.0 35.2 36   4.092 0.0029
##  power1 machine2 - power2 machine3    601.9 35.2 36  17.108 <.0001
##  power2 machine2 - power1 machine3   -438.0 35.2 36 -12.449 <.0001
##  power2 machine2 - power2 machine3     19.9 35.2 36   0.566 0.9926
##  power1 machine3 - power2 machine3    457.9 35.2 36  13.016 <.0001
##
## Results are averaged over the levels of: station_f
## P value adjustment: tukey method for comparing a family of 6 estimates
TukeyHSD(model.3,"machine_f:power_f",conf.level = 0.95)
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov.default(formula = y^2 ~ machine_f + station_f + power_f:machine_f + machine_f:station_f + power_f:machi
##
## $`machine_f:power_f`
##                                        diff       lwr        upr     p adj
## machine2:power1-machine1:power1  -24.57667 -130.4312   81.27785 0.9809332
## machine3:power1-machine1:power1 -168.56667 -274.4212  -62.71215 0.0003828
## machine1:power2-machine1:power1 -542.58778 -648.4423 -436.73326 0.0000000
## machine2:power2-machine1:power1 -606.58889 -712.4434 -500.73437 0.0000000
## machine3:power2-machine1:power1 -626.50889 -732.3634 -520.65437 0.0000000
## machine3:power1-machine2:power1 -143.99000 -249.8445  -38.13549 0.0029228
## machine1:power2-machine2:power1 -518.01111 -623.8656 -412.15660 0.0000000
## machine2:power2-machine2:power1 -582.01222 -687.8667 -476.15771 0.0000000
## machine3:power2-machine2:power1 -601.93222 -707.7867 -496.07771 0.0000000
## machine1:power2-machine3:power1 -374.02111 -479.8756 -268.16660 0.0000000
## machine2:power2-machine3:power1 -438.02222 -543.8767 -332.16771 0.0000000
## machine3:power2-machine3:power1 -457.94222 -563.7967 -352.08771 0.0000000
## machine2:power2-machine1:power2  -64.00111 -169.8556   41.85340 0.4668528
## machine3:power2-machine1:power2  -83.92111 -189.7756   21.93340 0.1884853
## machine3:power2-machine2:power2  -19.92000 -125.7745   85.93451 0.9926201
# agricolae::LSD.test(model.3,"machine_f", console=T)
```

```
library(emmeans)
(ma_po <- pairs(lsmeans(model.3,~ machine_f|power_f))) # machine_f|power_f
ma_po_st <- pairs(lsmeans(model.3,~ station_f|power_f|machine_f)) #machine_f|power_f:station_f
# test(rbind(ma_po,ma_po_st),adjust="tukey")
```

Tukey Multiple comparisons of means for 95% family-wise confidence level shows that most of the power setting nested in specific machine have a different effects on field at 5% significant level with a p-value of 0.000. That is, the combinations of machine and power affect the experiment significantly.

The exceptions include power 1 in machine 1 vs. power 1 in machine 2 (p-value=0.9964); power 2 in machine 1 vs. power 2 in machine 2 (p-value=0.3835); power 2 in machine 1 vs. power 2 in machine 3 (p-value=0.1123); power 2 in machine 2 vs. power 2 in machine 3 (p-value=0.1123).

The 3-way interaction presents a similar results.

Therefore, under power setting 2, the differences among the machines tend to disappear.

Summary

Appendix

Split-Plot design

This is a Split-Plot design model (fat is whole-plot factor and temperature is split-plot factor)

$$y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \gamma_k + (\tau\gamma)_{ik} + (\beta\gamma)_{jk} + (\tau\beta\gamma)_{ijk} + \varepsilon_{ijk}$$

for $i = 1, 2, 3, 4;\ j = 1, 2, 3;\ k = 1, 2, 3, 4$

μ is the overall true mean response;

τ_i is the effect of i^{th} replication of days;

β_j is the main effect of j^{th} level of temperature (effect of split-plot factor);

$(\tau\beta)_{ij}$ is the interaction effect of i^{th} replication and j^{th} level of temperature;

γ_k is the main effect of k^{th} level of fat (effect of whole-plot factor);

$(\tau\gamma)_{ik}$ is the interaction effect of i^{th} replicatin and k^{th} level of fat(whole-plot error);

$(\beta\gamma)_{jk}$ is the interaction effect of j^{th} level of temperature and k^{th} level of fat;

$(\tau\beta\gamma)_{ijk}$ is the interaction effect of i^{th} replicatin, j^{th} level of temperature and k^{th} level of fat (sub-plot error);

y_{ijk} is response value for the i^{th} replication when j^{th} level of temperature and k^{th} level of fat are applied;

ε_{ijk} is random error for the i^{th} replication when j^{th} level of temperature and k^{th} level of fat are applied.

Assumptions: For an experienced baker, he/she will try to let the recipe and temperature are accurate in each day. the covariance between two observations from the same level of the random factor can be either positive or negative. Thus, we assume this is a **restricted model**.

$\varepsilon_{ijk} \sim iidN(0, \sigma^2);\ \tau_i \sim iidN(0, \sigma^2_\tau)$

$\sum_{j=1}^{3} \beta_j = 0;\ \sum_{j=1}^{3} (\tau\beta)_{ij} = 0;\ (\tau\beta)_{ij} \sim iidN(0, \frac{3-1}{3}\sigma^2_{\tau\beta})$

$\sum_{k=1}^{4}\gamma_k = 0;\ \sum_{k=1}^{4}(\tau\gamma)_{ik} = 0;\ (\tau\gamma)_{ik} \sim iidN(0, \frac{4-1}{4}\sigma^2_{\tau\gamma})$

$\sum_{j=1}^{3}(\beta\gamma)_{jk} = 0;\ \sum_{k=1}^{4}(\beta\gamma)_{jk} = 0$

$\sum_{j=1}^{3}(\tau\beta\gamma)_{ijk} = 0;\ \sum_{k=1}^{4}(\tau\beta\gamma)_{ijk} = 0;\ (\tau\beta\gamma)_{ijk} \sim iidN(0, \frac{(3-1)(4-1)}{3\times4}\sigma^2_{\tau\beta\gamma})$

ε_{ijk}, τ_i, $(\tau\beta)_{ij}$, $(\tau\gamma)_{ik}$, $(\beta\gamma)_{jk}$, and $(\tau\beta\gamma)_{ijk}$ are independent.

```
table2019s2$Run_r <- as.random(table2019s2$Run)
table2019s2$Trt_f <- as.fixed(table2019s2$Trt)
table2019s2$Rev_f <- as.fixed(table2019s2$Rev)
model_2019s2_1<-aov(Shrink ~ Run_r+Trt_f + Trt_f%in%Run_r+Rev_f +Rev_f%in%Run_r + Trt_f:Rev_f,table2019s2)
gad(model_2019s2_1)
## Analysis of Variance Table
##
## Response: Shrink
##              Df  Sum Sq Mean Sq  F value    Pr(>F)
## Run_r         3   124.3   41.43  36.4670 5.099e-13 ***
## Trt_f         3  3012.5 1004.18  78.8364 8.810e-07 ***
## Rev_f         6 11051.8 1841.96 876.8437 < 2.2e-16 ***
## Run_r:Trt_f   9   114.6   12.74  11.2120 1.218e-09 ***
## Run_r:Rev_f  18    37.8    2.10   1.8491   0.04245 *
## Trt_f:Rev_f  18   269.5   14.97  13.1796 8.477e-14 ***
## Residual     54    61.3    1.14
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
model2019s22 <- lmer(Shrink ~ (1|Run) + Trt + (1|Run:Trt) + Rev + (1|Run:Rev) + Trt:Rev,table2019s2 , REML=T)
```

```
table2018s2$Run.r <- as.random(table2018s2$Run)
table2018s2$Method.f <- as.fixed(table2018s2$Method)
table2018s2$Storage.f <- as.fixed(table2018s2$Storage)
model2018s23 <- aov(Y~Run.r+Method.f+Method.f%in%Run.r+Storage.f+Storage.f%in%Run.r+Method.f:Storage.f+(Method.f:St
gad(model2018s23)
## Analysis of Variance Table
##
## Response: Y
##                            Df Sum Sq Mean Sq  F value    Pr(>F)
## Run.r                       2 2483.3 1241.65  69.2787 1.235e-15 ***
## Method.f                    2  156.3   78.16   0.3053    0.7527
## Storage.f                   1  703.2  703.19   0.2342    0.6762
## Run.r:Method.f              4 1024.0  256.00  14.2835 5.121e-08 ***
## Run.r:Storage.f             2 6004.3 3002.13 167.5054 < 2.2e-16 ***
## Method.f:Storage.f          2    3.8    1.91   0.0686    0.9348
## Run.r:Method.f:Storage.f    4  111.5   27.87   1.5552    0.1995
## Residual                   54  967.8   17.92
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
estimates(model2018s23)
## $tm
##                            Run.r Method.f Storage.f n
## Run.r                          1        3         2 4
## Method.f                       3        0         2 4
## Storage.f                      3        3         0 4
## Run.r:Method.f                 1        0         2 4
## Run.r:Storage.f                1        3         0 4
## Method.f:Storage.f             3        0         0 4
## Run.r:Method.f:Storage.f       1        0         0 4
## Res                            1        1         1 1
##
## $mse
##                            Mean square estimates
## Run.r                      "Res + Run.r"
## Method.f                   "Res + Run.r:Method.f + Method.f"
## Storage.f                  "Res + Run.r:Storage.f + Storage.f"
## Run.r:Method.f             "Res + Run.r:Method.f"
## Run.r:Storage.f            "Res + Run.r:Storage.f"
## Method.f:Storage.f         "Res + Run.r:Method.f:Storage.f + Method.f:Storage.f"
## Run.r:Method.f:Storage.f   "Res + Run.r:Method.f:Storage.f"
## Residual                   "Res"
##
## $f.versus
##                            F-ratio versus
## Run.r                      "Residual"
## Method.f                   "Run.r:Method.f"
```

```
## Storage.f                 "Run.r:Storage.f"
## Run.r:Method.f            "Residual"
## Run.r:Storage.f           "Residual"
## Method.f:Storage.f        "Run.r:Method.f:Storage.f"
## Run.r:Method.f:Storage.f "Residual"
```

- Take average of 4 crates

```
table2018s2.bar<- table2018s2 %>%
group_by(Run, Method,Storage) %>%
summarise(Y.bar = mean(Y))
```

The ANOVA table shows that only A have significant effects on the average Y at 0.05 significance level (p-value=).

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

Half-Normal Plot

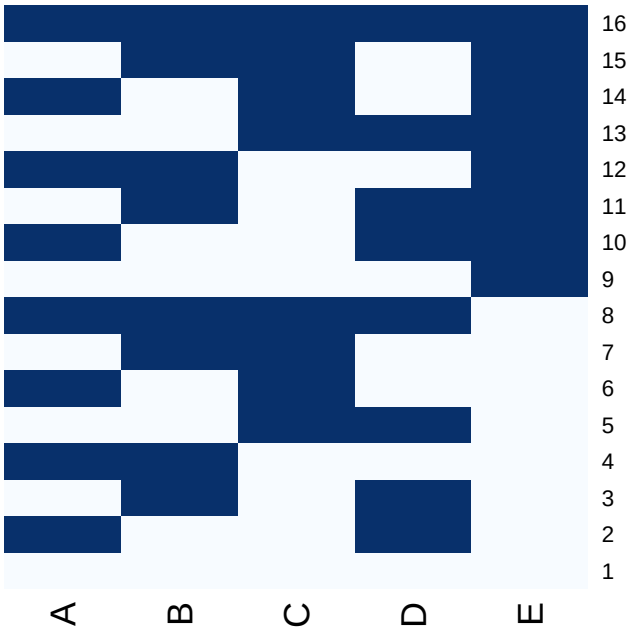
```
daewr::halfnorm(coef(model_2015f2_1)[2:16],alpha=0.10)
gghalfnorm::gghalfnorm(coef(model_2015f2_1)[2:16],labs = names(coef(model_2015f2_1)[2:16]),nlab = 4)
FrF2::DanielPlot(model_2015f2_1, half =T,alpha = 0.05)
```

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

2^k

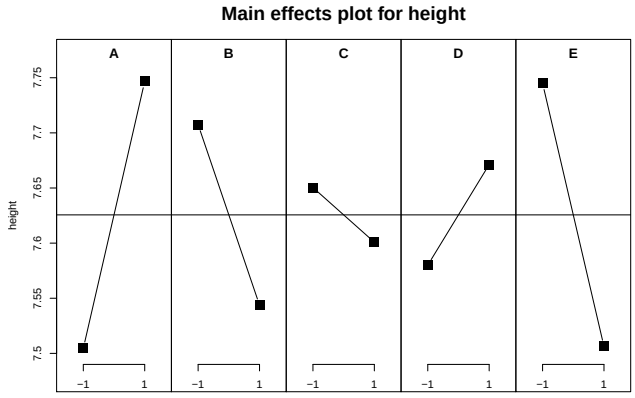
```
plan <- FrF2(16,6,generators = c("ABC","BCD"))
design.info(plan)
```

```
print(catlg, nfactors=6, nruns=16)
splitpick(6, catlg$`6-2.1`$gen, k.WP=2, nfac.WP=2)
```




```
##
##   A = B:C:D
##   B = A:C:D
##   C = A:B:D
##   D = A:B:C
##   A:B = C:D
##   A:C = B:D
##   A:D = B:C
##   A:B:E = C:D:E
##   A:C:E = B:D:E
##   A:D:E = B:C:E
```

```
MEPlot(model_2015f2_1)
# IAPlot(ff_2015f2)
```



The selected model is:

$y_{ijklm} = \mu + \tau_i + \beta_j + \gamma_k + \beta\gamma_{jk} + \delta_l + \varepsilon_{ijklm}$ i,j,k,l=1,2;m=1,2,3; N=48

```
model_2015f2_2 <- aov(height~A+B*E+D, table_2015f2)
# anova(model_2015f2_2)
```

Simulate a 2^{4-1} design

```
plan <- FrF2(16,nfactors = 4,alias.block.2fis=TRUE,randomize=F,default.levels = c(0, 1),replications = 3)
plan <- apply(plan, 2,function(x) as.numeric(x))
plan<-plan%>%transform(ABC=(A+B+C)%%2)
plan16<- plan[which(plan$ABC==plan$D),]
plan16<- plan16[order(plan16$Blocks,plan16$C,plan16$B,plan16$A),]
plan16 <- apply(plan16, 2,function(x) as.factor(x))%>%as.data.frame()
plan16<- cbind(plan16,y=table_2018f2[,5])
model16 <- aov(Heights~(A+B+C+D)^3, plan16)
aliases(model16)
##
##   A = B:C:D
##   B = A:C:D
##   C = A:B:D
##   D = A:B:C
##   A:B = C:D
##   A:C = B:D
##   A:D = B:C
```

Simulate a 2^{5-1} design

```
plan <- FrF2(32,nfactors = 5,alias.block.2fis=TRUE,randomize=F,default.levels = c(0, 1),replications = 3)
plan <- apply(plan, 2,function(x) as.numeric(x))
plan<-plan%>%transform(ABC=(A+B+C)%%2)
plan16<- plan[which(plan$ABC==plan$D),]
plan16<- plan16[order(plan16$Blocks,plan16$E,plan16$C,plan16$B,plan16$A),]
plan16 <- apply(plan16, 2,function(x) as.factor(x))%>%as.data.frame()
plan16<- cbind(plan16,y=table_2015f2[,6])
```

Simulate 2^{6-2} design

ABC=E,BCD=F;I=ABCE=BCDF=ADEF

```
#plan <- FrF2(16,generators=c("ABC","BCD"),randomize=F)#,alias.block.2fis=TRUE,blocks=4
#plan <- FrF2(16,gen=c(7,11),alias.block.2fis=TRUE,randomize=F)# ,blocks=4
plan <- FrF2(16,gen=c(7,13),alias.block.2fis=TRUE,randomize=F)#,blocks=4
design.info(plan)
## $type
## [1] "FrF2.generators"
##
## $nruns
## [1] 16
##
## $nfactors
## [1] 6
##
## $factor.names
## $factor.names$A
## [1] -1  1
##
## $factor.names$B
## [1] -1  1
##
## $factor.names$C
## [1] -1  1
##
## $factor.names$D
## [1] -1  1
##
## $factor.names$E
## [1] -1  1
##
```

```
## $factor.names$F
## [1] -1  1
##
##
## $generators
## [1] "E=ABC" "F=ACD"
##
## $aliased
## $aliased$legend
## [1] "A=A" "B=B" "C=C" "D=D" "E=E" "F=F"
##
## $aliased$main
## character(0)
##
## $aliased$fi2
## [1] "AB=CE"    "AC=BE=DF" "AD=CF"    "AE=BC"    "AF=CD"    "BD=EF"    "BF=DE"
##
##
## $FrF2.version
## [1] "2.2-2"
##
## $replications
## [1] 1
##
## $repeat.only
## [1] FALSE
##
## $randomize
## [1] FALSE
##
## $seed
## NULL
##
## $creator
## FrF2(16, gen = c(7, 13), alias.block.2fis = TRUE, randomize = F)
plan<- add.response(plan,rnorm(16))
model_ff <- lm(rnorm.16.~(A+B+C+D+E+F)^3, data = plan)
aliases(model_ff)
##
##  A = B:C:E = C:D:F
##  B = A:C:E = D:E:F
##  C = A:D:F = A:B:E
##  D = A:C:F = B:E:F
##  E = B:D:F = A:B:C
##  F = A:C:D = B:D:E
##  A:B = C:E
##  A:C = B:E = D:F
##  A:D = C:F
##  A:E = B:C
##  A:F = C:D
##  B:D = E:F
##  B:F = D:E
##  A:B:D = A:E:F = B:C:F = C:D:E
##  A:B:F = A:D:E = B:C:D = C:E:F
```

ABC=D,ABE=F;I=ABCD=ABEF=CDEF

```
plan <- fac.design(nlevels = 2,nfactors = 6,randomize = F,blocks = 4,block.gen = NULL)
plan<- add.response(plan,rnorm(64))
plan1 <- plan[plan$Blocks==1,]
model_ff <- lm(rnorm.64.~(A+B+C+D+E+F)^3, data = plan1)
aliases(model_ff)
##
##  A = B:C:D = B:E:F
##  B = A:E:F = A:C:D
##  C = D:E:F = A:B:D
##  D = C:E:F = A:B:C
##  E = C:D:F = A:B:F
##  F = C:D:E = A:B:E
##  A:B = C:D = E:F
##  A:C = B:D
##  A:D = B:C
##  A:E = B:F
##  A:F = B:E
##  C:E = D:F
##  C:F = D:E
##  A:C:E = A:D:F = B:C:F = B:D:E
##  A:C:F = A:D:E = B:C:E = B:D:F
```

Manual selections

Generate a full 2^6 design and create ABC, BCD, and ABE columns.

```
plan <- FrF2(64,nfactors = 6,alias.block.2fis=TRUE,randomize=F,default.levels = c(0, 1))
plan <- apply(plan, 2,function(x) as.integer(x))
plan<-plan%>%transform(ABC=(A+B+C)%%2,BCD=(B+C+D)%%2,ABE=(A+B+E)%%2)
```

Choose the rows with ABC=E and BCD=F

```
plan.abce<- plan[which(plan$ABC==plan$E & plan$BCD==plan$F),]
plan.abce <- apply(plan.abce, 2,function(x) as.factor(x))%>%as.data.frame()
plan.abce$y <- rnorm(16)
model.abce <- lm(y~(A+B+C+D+E+F)^3, data = plan.abce)
aliases(model.abce)
##
##  A = B:C:E = D:E:F
##  B = A:C:E = C:D:F
##  C = B:D:F = A:B:E
##  D = A:E:F = B:C:F
##  E = A:D:F = A:B:C
##  F = A:D:E = B:C:D
##  A:B = C:E
##  A:C = B:E
##  A:D = E:F
##  A:E = B:C = D:F
##  A:F = D:E
##  B:D = C:F
##  B:F = C:D
##  A:B:D = A:C:F = B:E:F = C:D:E
##  A:B:F = A:C:D = B:D:E = C:E:F
```

Choose the rows with ABC=D and ABE=F

```
plan.abcd<- plan[which(plan$ABC==plan$D & plan$ABE==plan$F),]
plan.abcd <- apply(plan.abcd, 2,function(x) as.factor(x))%>%as.data.frame()
plan.abcd$y <- rnorm(16)
model.abcd <- lm(y~(A+B+C+D+E+F)^3, data = plan.abcd)
aliases(model.abcd)
##
##  A = B:C:D = B:E:F
##  B = A:E:F = A:C:D
##  C = D:E:F = A:B:D
##  D = C:E:F = A:B:C
##  E = C:D:F = A:B:F
##  F = C:D:E = A:B:E
##  A:B = C:D = E:F
##  A:C = B:D
##  A:D = B:C
##  A:E = B:F
##  A:F = B:E
##  C:E = D:F
##  C:F = D:E
##  A:C:E = A:D:F = B:C:F = B:D:E
##  A:C:F = A:D:E = B:C:E = B:D:F
```

Simulate 2^{7-2}

```
plan <- FrF2(32,7,generators = c("ABCD","ABDE"),alias.info=3,randomize=F,default.levels=c(0,1))
design.info(plan)$aliased$main
design.info(plan)$aliased$fi2
design.info(plan)$aliased$fi3
```

```
plan <- apply(plan, 2,function(x) as.numeric(x))
plan<-plan%>%transform(ACE=(A+C+E)%%2,ACG=(A+C+G)%%2)
plan1<- plan[which(plan$ACE==0&plan$ACG==0),]
plan2<- plan[which(plan$ACE==1&plan$ACG==0),]
plan3<- plan[which(plan$ACE==0&plan$ACG==1),]
plan4<- plan[which(plan$ACE==1&plan$ACG==1),]
```

3^(3-1) Factorial Design

The number of factor = 3

The alia structure is:

$I = ABC = A^2B^2C^2$; resolution=3

$A = A^2BC = B^2C^2 = A^2 = BC = AB^2C^2$

$B = AB^2C = A^2C^2 = B^2 = AC = A^2BC^2$

$C = ABC^2 = A^2B^2 = C^2 = AB = A^2B^2C$

$A^2B = B^2C = AC^2 = AB^2 = A^2C = BC^2$

```
table_2015s2 <- readxl::read_xlsx("qe_lab/Bottles.xlsx")
table_2015s2[9,3] <- 1
```

```
plan <- fac.design(nlevels = 3,nfactors = 3,randomize = F,block.gen = NULL)
plan <- apply(plan, 2,function(x) as.integer(x)-1)
plan<-plan%>%transform(AB=(A+B)%%3)
plan$y <- table_2015s2$Time
plan.abc<- plan[which(plan$AB==plan$C),]
heatmap(as.matrix(plan.abc[,1:4]),Colv = NA,Rowv = NA,scale = "column",col = blues9)
for(i in 1:3){plan.abc[,i]<-as.factor(plan.abc[,i])}
```

Choose the rows of AB=C, the selected runs are

```
plan.abc
```

The model only includes the main effects.

$y_{ijk} = \mu + \tau_i + \beta_j + \gamma_k + \varepsilon_{ijk}$ i,j,k=1,2,3

```
model.abc <- aov(y~(A+B+C), data = plan.abc)
anova(model.abc)
```

Choose the rows of ABC=B, the selected runs are

```
plan <- fac.design(nlevels = 3,nfactors = 3,randomize = F,block.gen = NULL)
plan <- apply(plan, 2,function(x) as.integer(x)-1)
plan<-plan%>%transform(ABB=(A+B+B)%%3)
plan$y <- rnorm(27)
plan.abc<- plan[which(plan$ABB==plan$C),]
plan.abc <- apply(plan.abc, 2,function(x) as.factor(x))%>%as.data.frame()
model.abc <- aov(y~(A+B+C)^3, data = plan.abc)
anova(model.abc)
```

- 3^{6-3}

```
plan <- fac.design(nlevels = 3,nfactors = 6,randomize = F,block.gen = NULL)
plan <- apply(plan, 2,function(x) as.integer(x)-1)
plan<-plan%>%transform(AB=(A+B)%%3,BC=(B+C)%%3,ABC=(A+B+C)%%3)
plan1<- plan[which(plan$AB==plan$D&plan$BC==plan$E&plan$ABC==plan$F),]
heatmap(as.matrix(plan1[,1:6]),Colv = NA,Rowv = NA,scale = "column",col = blues9)
plan1 <- apply(plan1, 2,function(x) as.factor(x))%>%as.data.frame()
```

```
plan1$y<- rnorm(27)
model_36 <- lm(y~(A+B+C+D+E+F)^3, data = plan1)
anova(model_36)
```

Latin squre

This is a Latin Square Design include 5 level treatments and 5 $\times$ 5 column and row.

The model is:

$$y_{ijk} = \mu + \tau_i + \beta_j + \gamma_k + \varepsilon_{jik}$$

for $i, j, k = 1, 2, .., 5$.

y_{ijk} is the value of potato weight when i^{th} level of treatment and j^{th} level column and k^{th} row are applied.

τ_i is fixed main effect of i^{th} level of Treatment ;

β_j is block effect of j^{th} level of columns;

γ_j is block effect of j^{th} level of rows;

ε_{jik} is random error when i^{th} level of treatment and j^{th} level column and k^{th} row are applied

μ is the overall true mean .

The model includes below assumptions:

$\varepsilon_{ijk} \sim iidN(0, \sigma^2)$; $\sum_{k=1}^{5} \tau_i = 0$; $\sum_{k=1}^{5} \beta_j = 0$; $\sum_{k=1}^{5} \gamma_k = 0$; and independent

The first plot shows that it is a 5 $\times$ 5 Latin Square Design.

The plot of value v.s. treatment shows some difference in the average value among 5 treatment levels.

Both of the plots of value v.s. column and row show an increasing trend from 1 to 5.

```
group_by(table_2019f1, Trt) %>%
  summarise(
    count = n(),
    mean = mean(Val, na.rm = TRUE),
    sd = sd(Val, na.rm = TRUE)
  )
## # A tibble: 5 x 4
##   Trt   count  mean    sd
##   <fct> <int> <dbl> <dbl>
```

```
## 1 P1        5  63.3  5.35
## 2 P2        5  68.1  4.74
## 3 P3        5  70.6  6.43
## 4 P4        5  71.0  4.50
## 5 P5        5  69.7  6.53
```

```
model_2019f1_1 <- aov(Val ~ Trt+Col+Row, table_2019f1)
# anova(model_2019f1_1)
```

The ANOVA table shows that all the treatment, columns and rows have significant effects on the average value of response at 0.05 significance level (p-value=0.01602, 0.001803, 0.03976 respectively).

Paired test

```
TukeyHSD(model_2019f1_1,"Trt",conf.level = 0.95)
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov.default(formula = Val ~ Trt + Col + Row, data = table_2019f1)
##
## $Trt
##         diff        lwr       upr     p adj
## P2-P1  4.76 -1.7426534 11.262653 0.2000378
## P3-P1  7.30  0.7973466 13.802653 0.0256163
## P4-P1  7.70  1.1973466 14.202653 0.0182881
## P5-P1  6.40 -0.1026534 12.902653 0.0544550
## P3-P2  2.54 -3.9626534  9.042653 0.7269894
## P4-P2  2.94 -3.5626534  9.442653 0.6152509
## P5-P2  1.64 -4.8626534  8.142653 0.9244612
## P4-P3  0.40 -6.1026534  6.902653 0.9996186
## P5-P3 -0.90 -7.4026534  5.602653 0.9911110
## P5-P4 -1.30 -7.8026534  5.202653 0.9658349
```

It can be seen from the output, that only the difference between trt3 and trt1 is significant with an adjusted p-value of 0.025.

The Tukey multiple comparisons show that there is significant different between treatment level 3 and 1 (p-value=0.02561634), and between level 4 and 1 (p-value=0.01828808) at a 5% significance level. The difference between other paired level of treatment are not significant at a 5% significance level. The analysis suggest fertilizer 1 is not as good as fertilizer 3 or 4.

The Latin square design is used to eliminate two nuisance sources of variability. It systematically allows blocking in two directions. Thus, the rows and columns actually represents two restrictions on randomization.

```
model_2019f1_2 <- lm(Val ~ Trt+Col, table_2019f1)
anova(model_2019f1_2)
```

Using Randomized Complete Block Design, the model cannot eliminate the nuisance sources in rows. The ANOVA table shows the treatment effects become not significant.

```
model_2019f1_3 <- lm(Val ~ Trt, table_2019f1)
anova(model_2019f1_3)
```

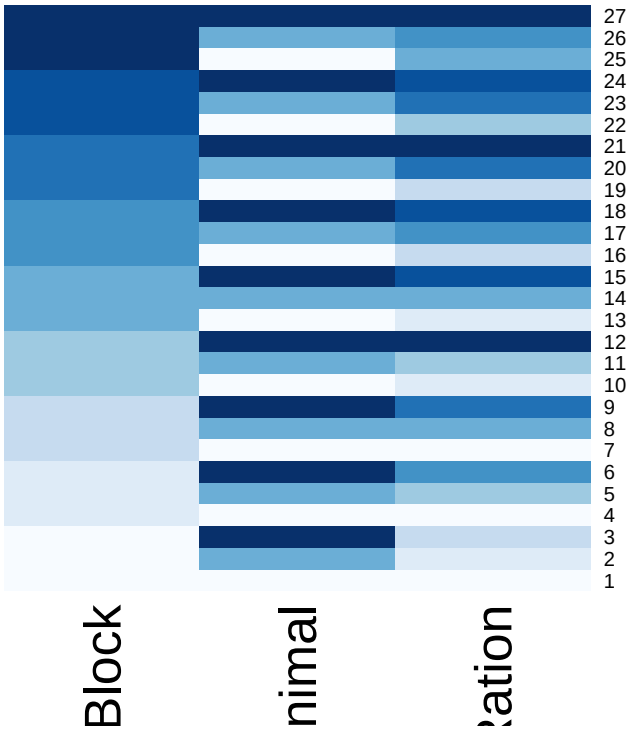
Using Completely Randomized Design, the model cannot eliminate the nuisance sources in both column and rows. The ANOVA table shows the p-value of treatment effects is larger.

- Graeco Latin Sq

```
T1<-c("a","b","c","d","e")
T2<-1:5
pander::pander(design.graeco(T1,T2,serie=1,randomization=F)$sketch)
```

BIBD

```
table_2017sd1 <- read_xlsx("qe_lab/NBalance.xlsx")
heatmap(as.matrix(table_2017sd1[,1:3]),Colv = NA,Rowv = NA,scale = "column",col = blues9)
xtabs(Animal~Ration+Block, data = table_2017sd1)
##         Block
## Ration 1 2 3 4 5 6 7 8 9
##      1 1 1 1 0 0 0 0 0 0
##      2 2 0 0 1 1 0 0 0 0
##      3 3 0 0 0 0 1 1 0 0
##      4 0 2 0 2 0 0 0 1 0
##      5 0 0 2 0 2 0 0 0 1
##      6 0 3 0 0 0 2 0 0 2
##      7 0 0 3 0 0 0 2 2 0
##      8 0 0 0 0 3 3 0 3 0
##      9 0 0 0 3 0 0 3 0 3
xtabs(Ration~Animal+Block, data = table_2017sd1)
##         Block
## Animal 1 2 3 4 5 6 7 8 9
##      1 1 1 1 2 2 3 3 4 5
##      2 2 4 5 4 5 6 7 7 6
##      3 3 6 7 9 8 8 9 8 9
#ftable(Block~ Animal+Ration, data = table_2017sd1)
table_2017sd1$Block <- factor(table_2017sd1$Block,
labels=c("Blk1","Blk2","Blk3","Blk4","Blk5","Blk6","Blk7","Blk8","Blk9"))
table_2017sd1$Animal <- factor(table_2017sd1$Animal,
                               labels = c("Ani1","Ani2","Ani3"))
table_2017sd1$Ration <- factor(table_2017sd1$Ration,
labels=c("Rat1","Rat2","Rat3","Rat4","Rat5","Rat6","Rat7","Rat8","Rat9"))
# str(table_2017sd1)
```




```
# t(apply(tab, 1, function(x) (1:4)[x != 0]))
```

A balanced incomplete block design (BIBD) is an incomplete block design where all pairs of treatments occur together in the same block equally often ($= \lambda$).

We use the following notation:

a=9: number of treatments (Rations) b=9: number of blocks k=3: number of units per block (k<a) (number of animals each block gets to see) r=3: number of replicates per treatment ("how often do we see a ration across all blocks?") N=ar=bk=27: total number of units

For every setting k<a we can find a BIBD by taking all possible subsets

An unreduced balanced incomplete block design have $\binom{a}{k}=\binom{9}{3}=84$ binomial coefficient.

```
combn(x = 9, m = 3)
```

However, $\frac{k-1}{a-1}r = \frac{3}{4}$ is not an integer. $\lambda = 1$ is the least number of times each pair of treatment appear in the same block.

Therefore, r=4, b=12 is needed.

```
ibd::bibd(v = 9, b = 12, r = 4, k = 3, lambda = 1)$design
```

- As 2-factor Factorial Design

Drop off the Block, the model is:

$y_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij}$ i=1,2,..,9;j=1,2,3; N=27

```
model_2017sd1_2f <- aov(Nitrogen~Animal+Ration, table_2017sd1)
# anova(model_2017sd1_2f)
```

The ANOVA table indicates that no Factors are significant at a 5% significance level, with p-values of and respectively.

- As RCBD

```
model_2017sd1_rcbd <- aov(Nitrogen~Animal+Block, table_2017sd1)
# anova(model_2017sd1_rcbd)
```

Imputing Missing Data

There are three types of missingness mechanisms:

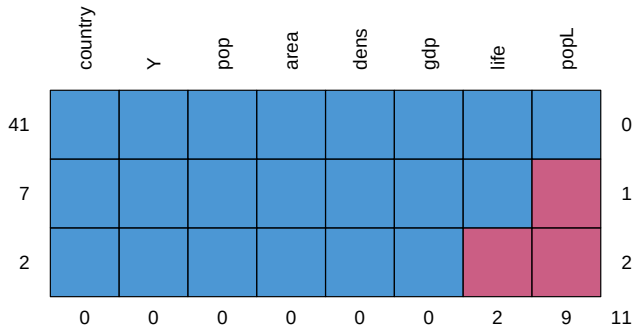
Missing completely at random (MCAR): when cases with missing values can be thought of as a random sample of all the cases; MCAR occurs rarely in practice.

Missing at random (MAR): when conditioned on all the data we have, any remaining missingness is completely random; that is, it does not depend on some missing variables. So missingness can be modelled using the observed data. Then, we can use specialised missing data analysis methods on the available data to correct for the effects of missingness.

Missing not at random (MNAR): when data is neither MCAR nor MAR. This is difficult to handle because it will require strong assumptions about the patterns of missingness.

One common way people try to deal with missing data is to delete all cases for which a value is missing. This method is called complete case analysis (CC). However, CC is valid only if data is MCAR. Another method is multiple imputation (MI), which is a monte carlo method that simulates multiple values to impute (fill-in) each missing value, then analyses each imputed dataset separately and finally pools the results together. We use MI as we work with the example dataset.

```
library(mice)
md.pattern(table_2020s1,rotate.names=T)
##    country Y pop area dens gdp life popL
## 41       1 1   1    1    1   1    1    1  0
## 7        1 1   1    1    1   1    1    0  1
## 2        1 1   1    1    1   1    0    0  2
##          0 0   0    0    0   0    2    9 11
```



This plot gives the frequencies for different combination of variables missing. Blue refers to observed data and red to the missing data.

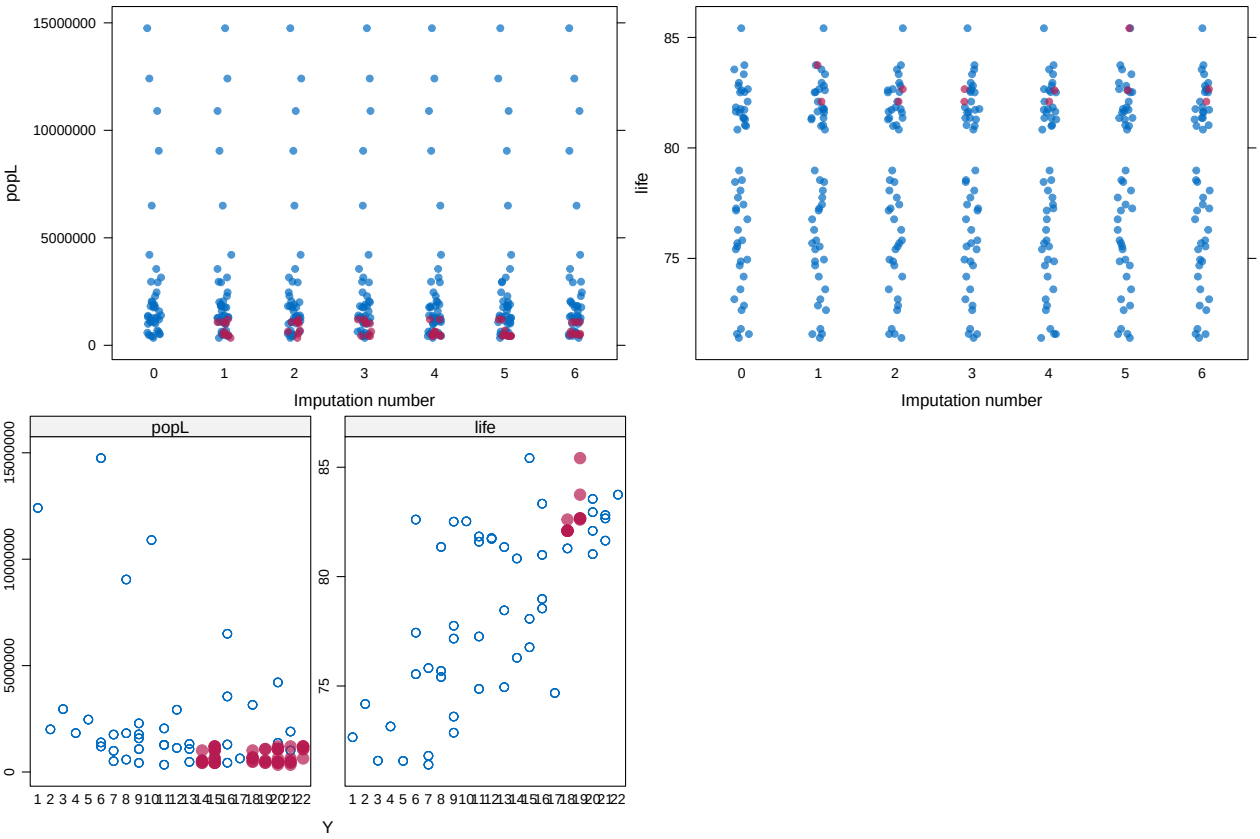
In this case, Missing proportion is 18%. 3 patterns observed from all possible patterns. We see that there are 7 cases where popL is missing whereas all the other variables are observed. There are 2 case where both of life and popL is missing.

```
# ?mice
imputed <- mice(table_2020s1[,-1],m=6,maxit=30,seed=500,method="cart",print=F) #"pmm",,"mean"
```

pmm stands for predictive mean matching, default method of mice() for imputation of continous incomplete variables; for each missing value, pmm finds a set of observed values with the closest predicted mean as the missing one and imputes the missing values by a random draw from that set. Therefore, pmm is restricted to the observed values, and might do fine even for categorical data (though not recommended).

- Inspect quality of imputations

```
imputed$imp$popL
##          1       2       3       4       5       6
## 2   509707  685587 1012225  637089  685587  475577
## 11  475577  536055 1012225  429920  437027  536055
## 25  342577  637089 1201426  475577  437027  509707
## 29 1201426 1077333  637089 1201426 1201426 1077333
## 31  475577  342577 1201426  437027  437027 1080324
## 33 1077333 1077333  437027  509707  437027  536055
## 36  637089 1201426  437027 1201426  437027 1077333
## 43 1012225 1012225  429920  437027 1201426  637089
## 46 1077333 1080324 1077333  637089  475577  475577
imputed$imp[[5]]
##          1       2       3       4       5       6
## 2   509707  685587 1012225  637089  685587  475577
## 11  475577  536055 1012225  429920  437027  536055
## 25  342577  637089 1201426  475577  437027  509707
## 29 1201426 1077333  637089 1201426 1201426 1077333
## 31  475577  342577 1201426  437027  437027 1080324
## 33 1077333 1077333  437027  509707  437027  536055
## 36  637089 1201426  437027 1201426  437027 1077333
## 43 1012225 1012225  429920  437027 1201426  637089
## 46 1077333 1080324 1077333  637089  475577  475577
stripplot(imputed, popL, pch = 19, xlab = "Imputation number")
stripplot(imputed, life, pch = 19, xlab = "Imputation number")
stripplot(imputed, popL+life~Y, cex=c(1,2), pch=c(1,20),jitter=FALSE,layout=c(2,1))
```



We can inspect the distributions of the original and the imputed data:

Blue represents the observed data and red shows the imputed data. Here, we expect the red points (imputed data) have almost the same shape as blue points (observed data). Blue points are constant across imputed datasets, but red points differ from each other, which represents our uncertainty about the true values of missing data.

```
imputed_2020s1 <- complete(imputed)
```

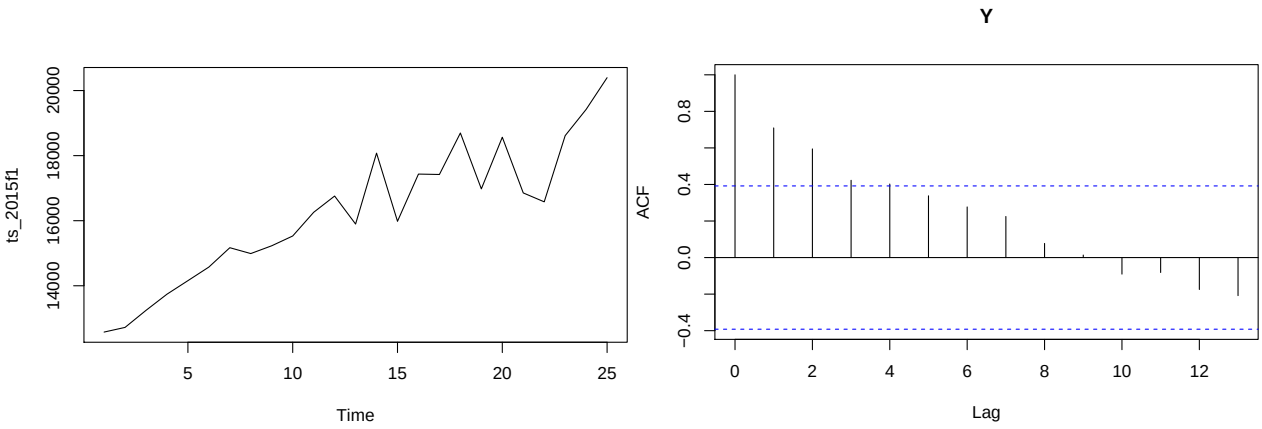
mice() imputes each missing value with a plausible value (simulates a value to fill-in the missing one) until all missing values are imputed and dataset is completed. Repeats the process for multiple times, say m times and stores all the m complete(d)/imputed datasets.

Time series Model

The first-order autoregression model of GDP growth can be estimated by computing OLS estimates in the regression of Y_t on Y_{t-1}

$$y_t = \beta_0 + \beta_1 y_{t-1} + \mu_t$$

- Check Autocorrelation



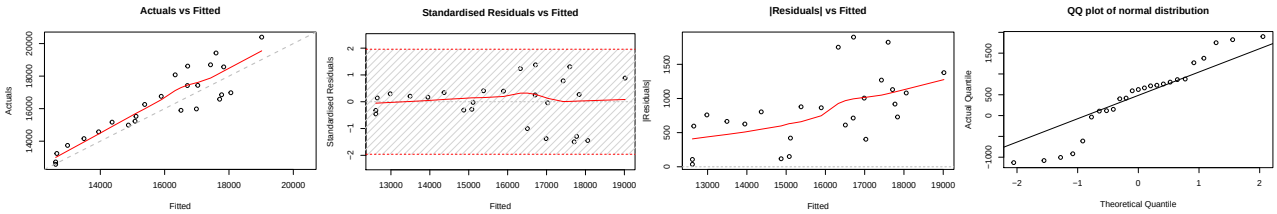
The time series plot of Y shows a strong autocorrelations.

The difference of Y_t and Y_{t-1} still shows some autocorrelations with lag=1.

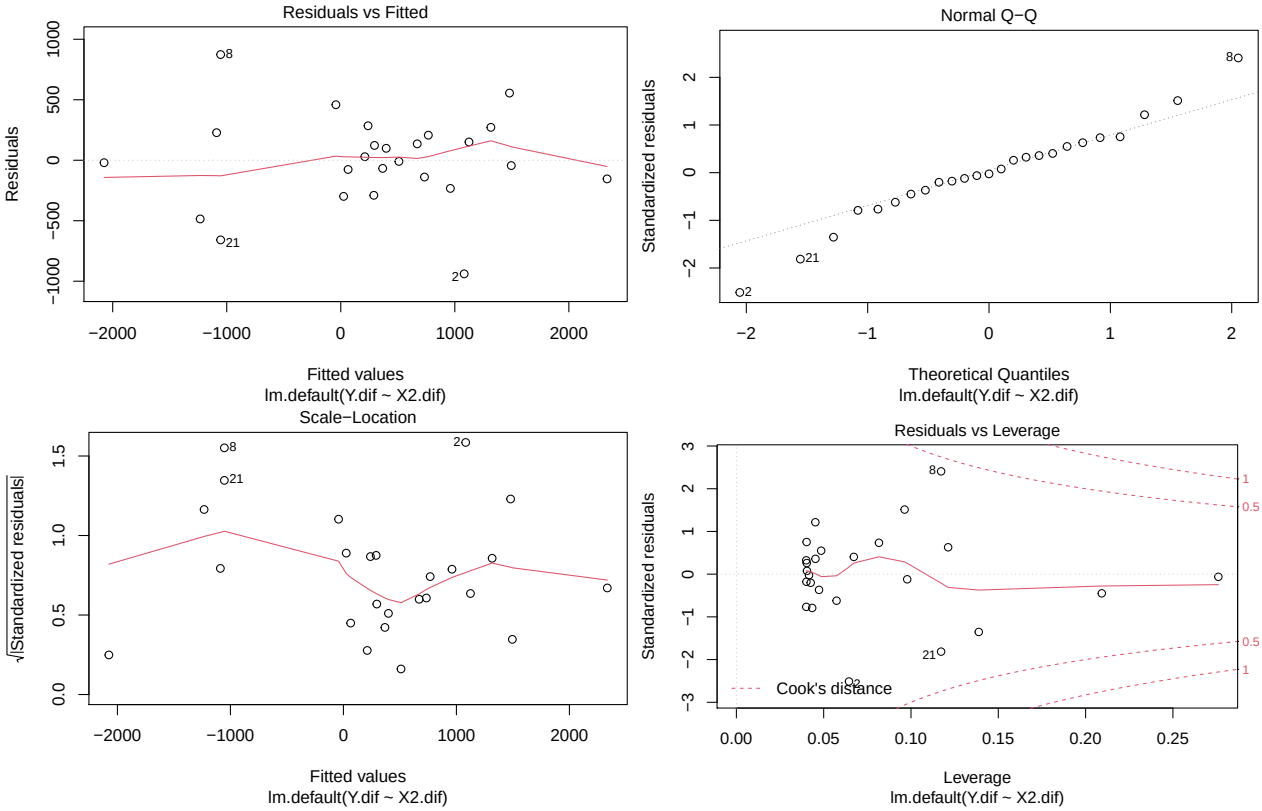
Simple Moving Average is a method of time series smoothing and is forecasting technique.

```
## Time elapsed: 0.03 seconds
## Model estimated: SMA(2)
## Initial values were produced using backcasting.
##
## Loss function type: MSE; Loss function value: 941154.385
## Error standard deviation: 1011.432
## Sample size: 25
## Number of estimated parameters: 2
## Number of degrees of freedom: 23
## Information criteria:
##       AIC      AICc       BIC      BICc
## 418.8185 419.3639 421.2562 422.1341
##
## 95% parametric prediction interval was constructed
## Time Series:
## Start = 1
## End = 25
## Frequency = 1
##       Series 1
##  [1,] 12612.75
##  [2,] 12612.75
##  [3,] 12648.50
##  [4,] 12982.00
##  [5,] 13492.50
##  [6,] 13949.00
##  [7,] 14366.00
##  [8,] 14872.50
##  [9,] 15080.00
## [10,] 15110.00
## [11,] 15380.00
## [12,] 15894.00
## [13,] 16508.00
## [14,] 16327.50
## [15,] 16987.50
## [16,] 17030.00
## [17,] 16707.50
## [18,] 17427.00
## [19,] 18058.50
## [20,] 17837.00
## [21,] 17772.00
## [22,] 17710.00
## [23,] 16717.00
## [24,] 17597.50
## [25,] 19018.00
```

Using the Time Series model, when x1=20 and x2=1900, the prediction interval is (1.6328949×10^4, 1.9345051×10^4), with fitted value of 1.7837×10^4.




```
table_2015f1$Y.dif <-c(0, diff(table_2015f1$Y))
table_2015f1$X2.dif <-c(0, diff(table_2015f1$X2))
diff_2015f1 <- lm(Y.dif~X2.dif,table_2015f1)
summary(diff_2015f1)
##
## Call:
## lm.default(formula = Y.dif ~ X2.dif, data = table_2015f1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -939.13 -154.36   -9.78  207.24  873.59
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 289.8662    77.2916    3.75  0.00104 **
## X2.dif       -1.4097     0.1108  -12.72  6.8e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 386.4 on 23 degrees of freedom
## Multiple R-squared:  0.8756, Adjusted R-squared:  0.8702
## F-statistic: 161.9 on 1 and 23 DF,  p-value: 6.8e-12
plot(diff_2015f1)
```




```
diff_2015f1$fitted.values[20]+table_2015f1$Y[19]
##       20
## 18294.14
```

```
predict(diff_2015f1,data.frame(X2.dif=table_2015f1$X2.dif[20]),interval = "predict")+table_2015f1$Y[19]
##        fit       lwr      upr
## 1 18294.14 17462.92 19125.36
```

When X1=20, the difference of X2 equals -728

```
regRes <- lm(abs(diff_2015f1$residuals)~diff_2015f1$fitted.values)
weights = 1/regRes$fitted.values^2
wls <-  lm(Y.dif~X2.dif,table_2015f1, weights=weights)
plot(wls)
```

Other non-linear model

- Loess model

```
loess_2015f1<- stats::loess(Y ~ X1+X2,table_2015f1)
# summary(loess_2015f1)
```

```
ggplot(table_2015f1,aes(X2,Y))+geom_point()+geom_smooth(method = "loess")
# ggplot(table_2015f1,aes(X2,Y))+geom_point()+geom_smooth(span =0.75)
```

- Spline model

```
sp_2015f1 <- lm(Y ~ splines::ns(X1, 2)+splines::ns(X2, 2),table_2015f1)
# summary(sp_2015f1)
```

- Gam model

```
gam_2015f1 <- mgcv::gam(Y ~ s(X1)+s(X2),data=table_2015f1)
# summary(gam_2015f1)
```

```
plot(gam_2015f1,all.terms=TRUE,pages=1)
```

```
fit = lm(Y ~ X1+X2,table_2015f1)
b=matrix(fit$coefficients) #betas
bT = t(b) #transpose betas for R
bhat = bT%*%b #this is our denominator
fit.anova = anova(fit) #need MSE
sigma.squared = tail(fit.anova$`Mean Sq`, n=1)
p = length(b)
my_k = (p*sigma.squared)/bhat
fit.ridge <-  lmridge::lmridge(Y ~ X1+X2, table_2015f1, K=my_k) #K is cap!!!
#compare ridge SSE & full SSE
ridgeSSE = sum((residuals.lmridge(fit.ridge)^2))
fullSSE=sum(fit$residuals^2)
```

```
prc$x <-  prcomp(table_2015f1[,1:2]) #13 rows in table
prc_2015f1lm(table_2015f1$Y ~ prc$x[,1]+prc$x[,2])
summary(prc_2015f1)
```

```
# plot(sp_2018f1)
sp.pred<- predict(sp_2015f1,data.frame(X1=20,X2=1900),interval = "prediction",level=0.95)^0.5
#predict(gam_2018f1,data.frame(X1=20,X2=1900),interval = "prediction",level=0.95)^0.5
# predict(loess_2018f1,data.frame(X1=20,X2=1900),interval = "prediction",level=0.95)^0.5
```

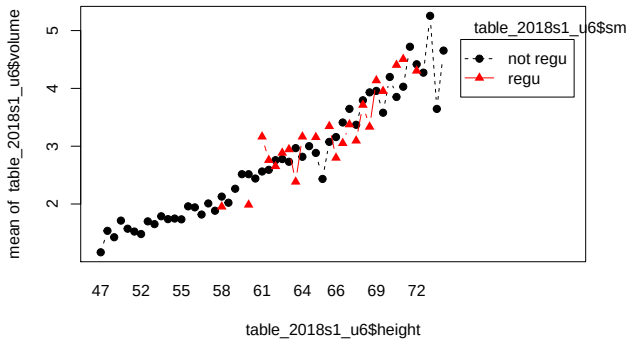
When x1=20 and x2=1900, the Spline model gives a fitted value of 136.4252602 . the prediction interval is (134.1991891, 138.6155869).

```
compare <- rbind(ts.pred,sp.pred) # po.pred,
rownames(compare) <- c("Time.serious","Spline")# "polynomial",
compare
##                        fit        lwr        upr
## Time.serious 17837.0000 16328.9486 19345.0514
## Spline         136.4253   134.1992   138.6156
```

Plot example

```
ggplot(table_2015f1,aes(X2,Y, color=X1))+
  labs(x="adv",y="prof",color="month")+
  geom_point()+theme_light()
ggplot(table_2018s1_u6, aes(smoker,fill=male))+
  geom_bar()+facet_wrap(.~age,ncol = 7)+theme_light()
ggline(data1,"height","weight",add=c("mean","jitter"),color="age")
ggline(data1,"height","weight",add=c("mean","jitter"),color="male")
```

```
ggpubr::ggline(table2018s2,"Storage","Y",add=c("mean","jitter"),color="Shipping",
  shape="Shipping",linetype="Shipping",ylab="acceptability",facet.by="Run")
```




```
plotly::plot_ly(table_2015f1, x=~X1,y=~Y^2,type="scatter")%>% add_lines(x=~X1,y=fitted(model_2015f1_3))
```

The above plots show that:

Not all the lines are parallel in the interaction plot. Therefore, in the model, there is the interaction effect of source level and technicians nested in the lab.

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.

often used test

ANOVA test with no assumption of equal variances

```
oneway.test(oxygen ~ creek, data = table_2016f2)
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  oxygen and creek
## F = 118.57, num df = 2.000, denom df = 15.496, p-value = 4.046e-10
```

Pairwise t-tests with no assumption of equal variances

```
pairwise.t.test(table_2016f2$oxygen, table_2016f2$creek,
                p.adjust.method = "BH", pool.sd = FALSE)
##
##  Pairwise comparisons using t tests with non-pooled SD
##
## data:  table_2016f2$oxygen and table_2016f2$creek
##
##        creek1  creek2
## creek2 2.5e-05 -
## creek3 0.00016 2.1e-09
##
## P value adjustment method: BH
```

F test

```
var.test(y~machine, subset(data2,machine=="machine1"|machine=="machine2"), alternative = "two.sided")
var.test(y~machine, subset(data2,machine=="machine1"|machine=="machine3"), alternative = "two.sided")
var.test(y~machine, subset(data2,machine=="machine2"|machine=="machine3"), alternative = "two.sided")
```

The F test indicates that there is not enough evidence to reject the null hypothesis that the two variances of creek1 and creek2 are equal at the 0.05 significance level (p-value=0.4509). The same goes for creek1 and creek3 (p-value=0.05499).

There is significant difference between the two variances of creek1 and creek3. The p-value of F-test is p = 0.009893 which is greater than the significance level 0.05.

The Tables below show the summary of all those simple effect comparison tests.

Lack of Fit F Test

Assess how much of the error in prediction is due to lack of model fit. The residual sum of squares resulting from a regression can be decomposed into 2 components:

If most of the error is due to lack of fit and not just random error, the model should be discarded and a new model must be built. The lack of fit F test works only with simple linear regression. Moreover, it is important that the data contains repeat observations i.e. replicates

for at least one of the values of the predictor x. This test generally only applies to datasets with plenty of replicates.

```
ols_pure_error_anova(lm(Y~X1,table_2015f1))
```

LRT test

```
# lrtest (model_2015f1_1,model_2015f1_2)
logLikA <- -371.679
logLikB <- -382.403
1-pchisq(-2*(logLikA-logLikB), df = 2, lower.tail = FALSE)
## [1] 0
```

Compute table margins and relative frequency

```
spread(table_2018s1_u6,as.character(age),volume)
addmargins(table_2018s1_u6[,3:5])
marginSums(as.array(table_2018s1_u6),1, margin = NULL)
proportions(table_2018s1_u6[,2:3]) # , margin = NULL
```

Table margins correspond to the sums of counts along rows or columns of the table.

Relative frequencies express table entries as proportions of table margins (i.e., row or column totals).

Residual Fit Spread Plot

Plot to detect non-linearity, influential observations and outliers. Consists of side-by-side quantile plots of the centered fit and the residuals. It shows how much variation in the data is explained by the fit and how much remains in the residuals. For inappropriate models, the spread of the residuals in such a plot is often greater than the spread of the centered fit.

```
ols_plot_resid_fit_spread(model_2015f1_3)
```

Deleted Studentized Residual vs Fitted Values Plot

Graph for detecting outliers.

```
ols_plot_resid_stud_fit(model_2015f1_2)
```

PRESS and RMSE

```
model_2019s1_2 <- lm(table_2019s1_250,formula=log(y)~ x2+A+B)
model_2019s1_3 <- lm(table_2019s1_500,formula=log(y)~ x2+A+B)
Metrics::rmse(table_2019s1_500$y,exp(predict(model_2019s1_2,table_2019s1_500)))
ols_press(model_2019s1_3)
MPV::PRESS(model_2019s1_3)
sum((residuals(model_2019s1_3)/(1 - lm.influence(model_2019s1_3)$hat))^2)
ols_pred_rsq(model_2019s1_3)
# 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)
# by definition PRESS
sum((table_2019s1_500$y-exp(model_2019s1_2$fit))^2)
sum((table_2019s1_500$y-exp(predict(model_2019s1_2,table_2019s1_500)))^2)
# one method of RMSE
sqrt(mean(model_2019s1_3$residuals^2))
```

Lmer function

```
# When some factors are random
model_2017f2_3<-lmer(y~(1|machine)+station+power+
  (1|machine:station)+ (1|machine:station:power),table_2017f2,REML=TRUE)
summary(model_2017f2_3)$varcor
confint(model_2019s2_2)[1:4,1:2]
ranova(model_2017f2_3)
```

The results of variance components and condidence intervals show that none of the effects related with technician has significant variance on average value of purity at 0.05 significance level. The variance of interaction effect between sources and technicians nested in labs is zero with confidence intervals $(0, 1.539^2)$ at 0.05 significance level. The variance of technicians nested in labs is zero with confidence intervals $(0, 1.603^2)$ at 0.05 significance level.

anscombe's quartet

The first scatter plot appears to be a simple linear relationship, corresponding to two variables correlated where y could be modelled as gaussian with mean linearly dependent on x.

The second graph is not distributed normally; while a relationship between the two variables is obvious, it is not linear, and the Pearson correlation coefficient is not relevant. A more general regression and the corresponding coefficient of determination would be more appropriate.

In the third graph , the distribution is linear, but should have a different regression line (a robust regression would have been called for). The calculated regression is offset by the one outlier which exerts enough influence to lower the correlation coefficient from 1 to 0.816.

Finally, the fourth graph shows an example when one high-leverage point is enough to produce a high correlation coefficient, even though the other data points do not indicate any relationship between the variables.

The quartet is still often used to illustrate the importance of looking at a set of data graphically before starting to analyze according to a particular type of relationship, and the inadequacy of basic statistic properties for describing realistic datasets.