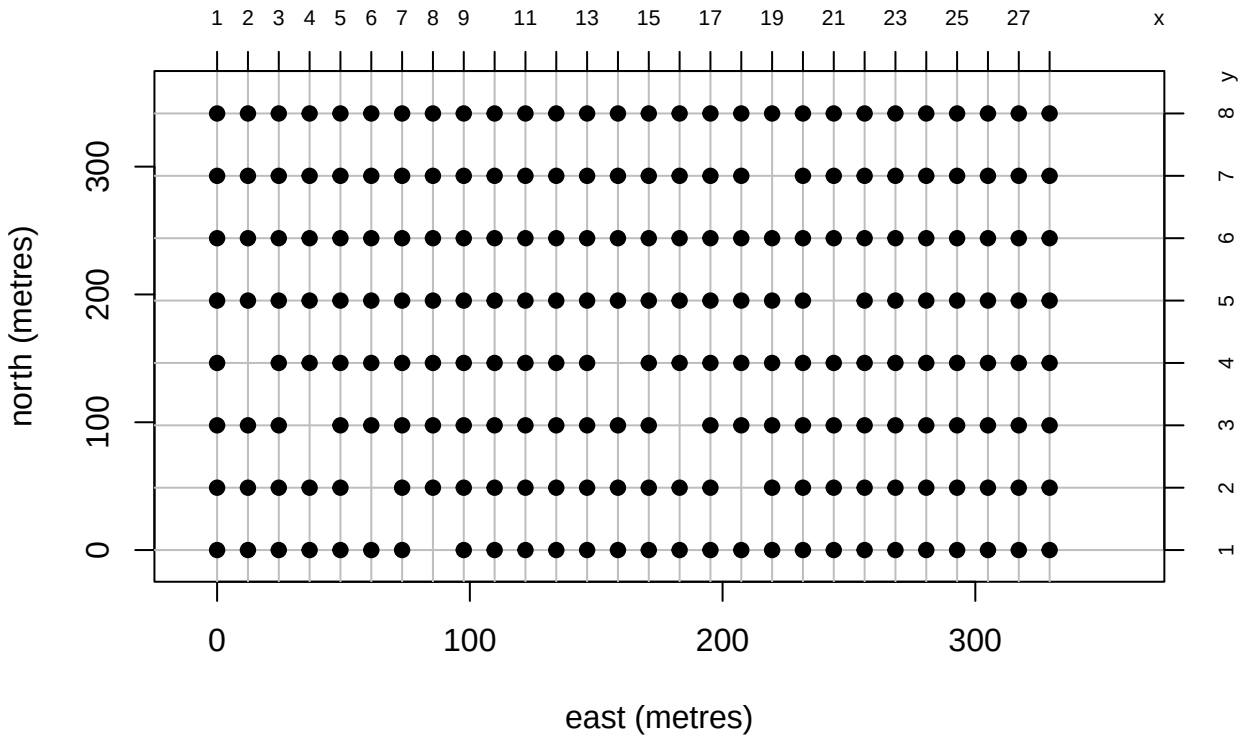
STAT6020_PredictiveAnalytics_Assignment1

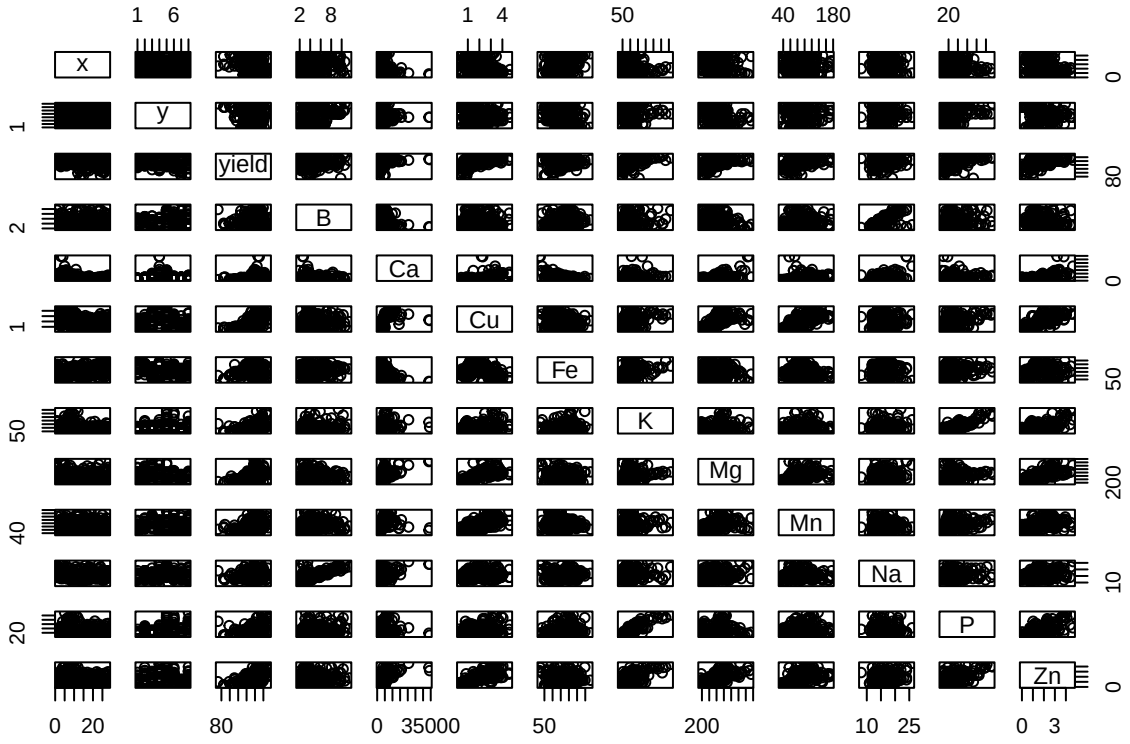
Prepared by Tim Virga

19/08/2021

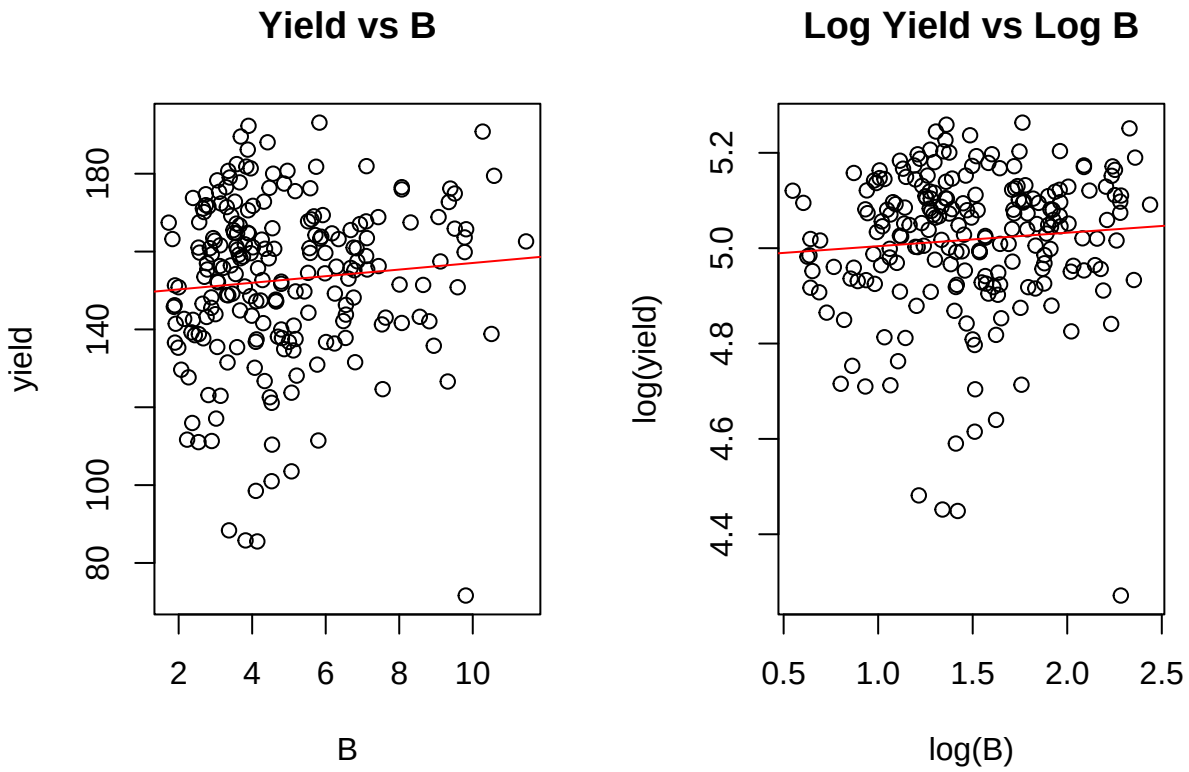


Preliminary exploration of the data

```
head(gy) # understanding what individual nutrients consist of the data set.
summary(gy) # understanding the quartile spread of each nutrient
pairs(gy) # Observing bivariate relationships at a glance
```



```r
par(mfrow=c(1,2))
plot(yield~B, main="Yield vs B", data=gy)# standard scatter plot
abline(lm(yield~B, data=gy), col="red")
plot(log(yield)~log(B), main="Log Yield vs Log B", data=gy) # standard log scatter plot
abline(lm(log(yield)~log(B), data=gy), col="red")
```



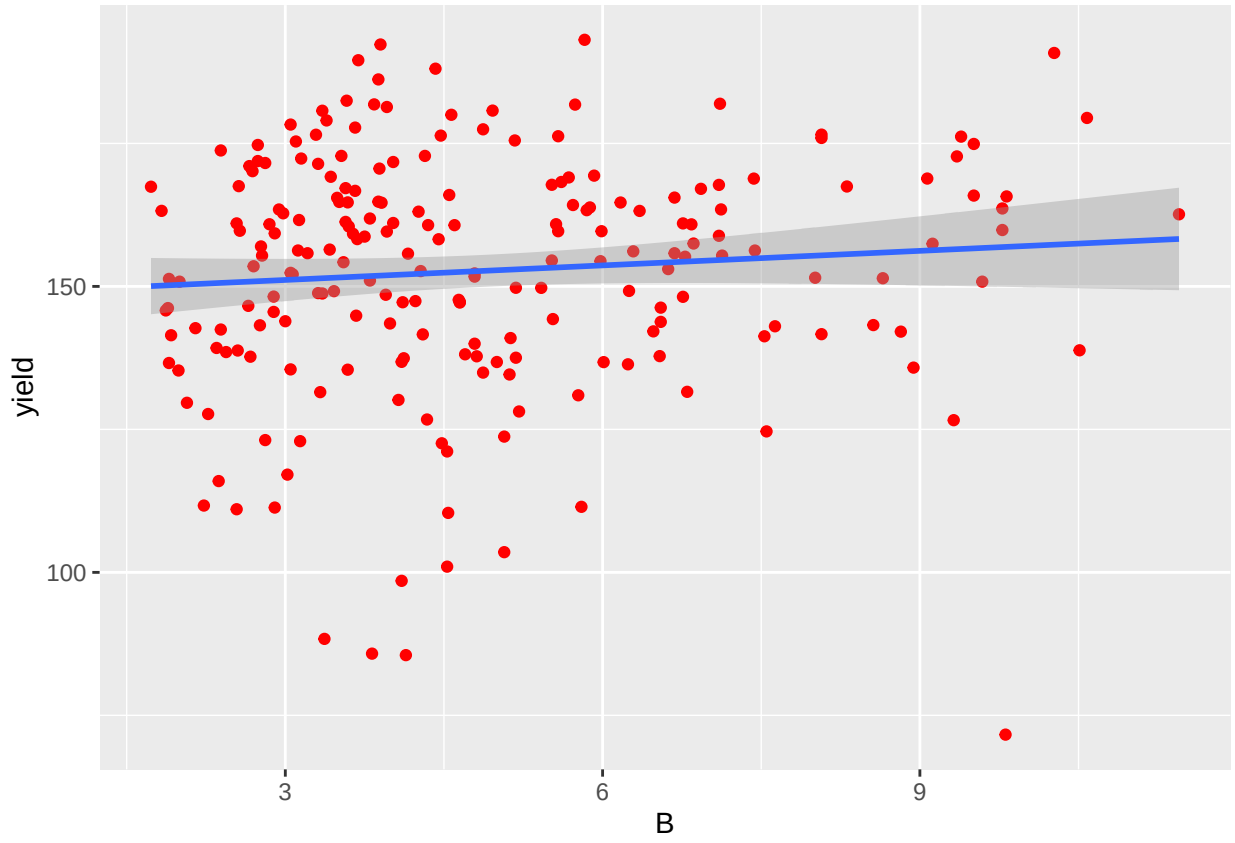
Plotting the same data Yield~B with ggplot:

```
library(ggplot2)    # using ggplot to produce the same plot
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
ggplot(data=gy, mapping=aes(x=B, y=yield))+
geom_point(col="red")+
geom_smooth(method="lm", se=TRUE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

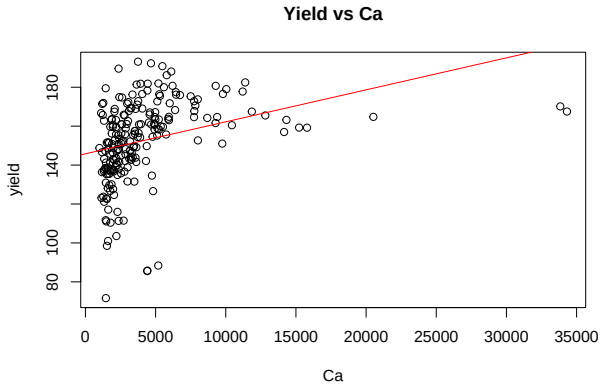


1.

Comment 1

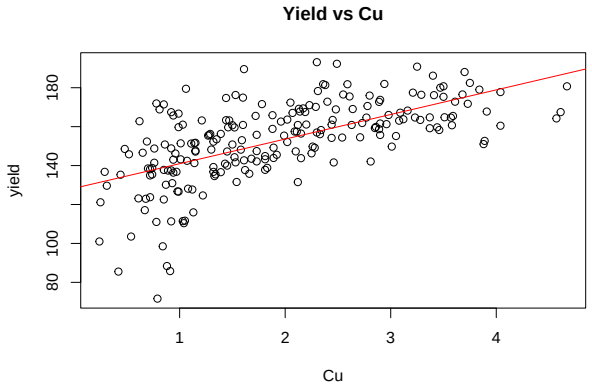
Both the standard scatter plot and log scatter show that there is a relatively insignificant relationship between nutrient B and Yield, however, slightly positive. B would not form a high quality predictor of the response variable.

Analysis and comments on remaining nutrients



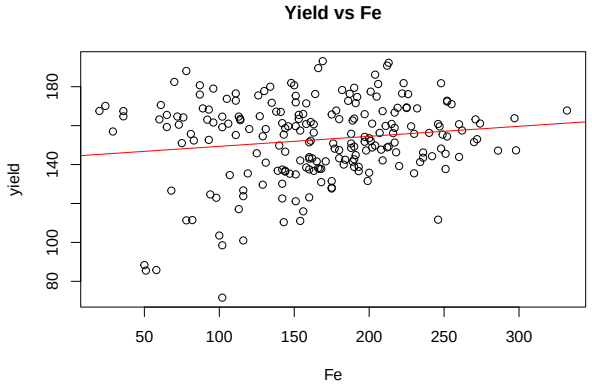
Comment 2

The scatter plot of Yield~Ca shows that there is a relationship between nutrient Ca and Yield, although limited by data given only a small number of high Ca yields. Ca would likely form a considered predictor variable of the response variable.



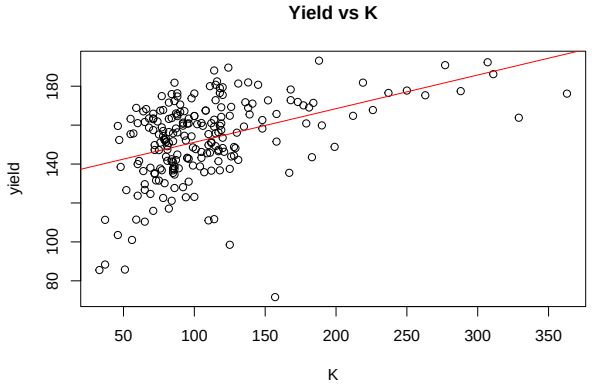
Comment 3

The scatter plot of Yield~Cu shows that there is a relatively significant linear relationship between nutrient Cu and Yield. We can observe a steady increase in yield for increased in the value of Cu. Cu would likely form an important predictor variable of the response variable.



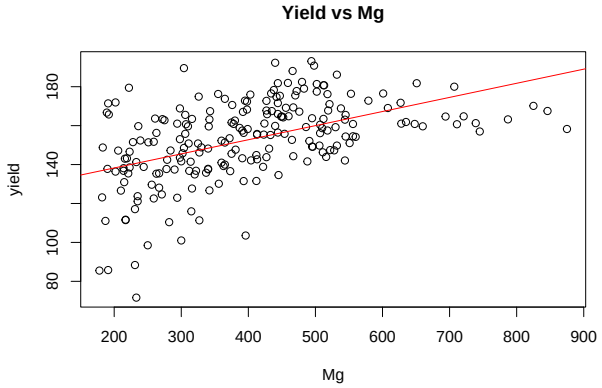
Comment 4

The standard scatter plot for Yield~Fe shows that there is a relatively insignificant relationship between nutrient Fe and yield, however, in dense areas of the plot there are clear suggestions of a linear relationship. Fe may present as a reasonable predictor of the response variable.



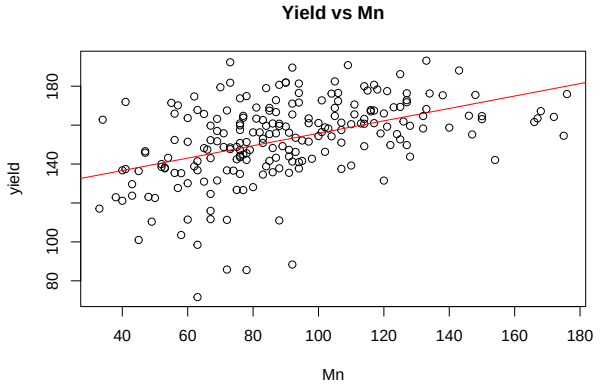
Comment 5

The scatter plot of Yield~K shows that there is a weak linear relationship between nutrient K and Yield. The strength of the relationship may not be strong as we observe a high density of data points within a short distance of eachother, mostly disorganised.



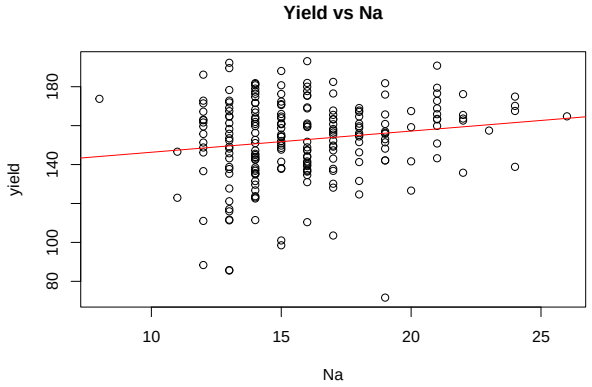
Comment 6

The scatter plot of Yield~Mg shows that there is a linear relationship between nutrient Mg and Yield. Mg would likely form an important predictor variable of the response variable as we can see the higher values of Mg relating to some of the larger yields, and the lower values of Mg relating to some of the lowest yields.



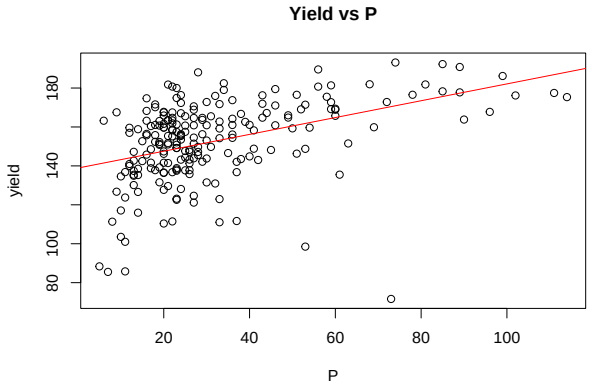
Comment 7

Almost identical in relation to Mg, The scatter plot abline of Yield~Mn shows that there is a similar relationship between nutrient Mn and Yield, although more variance in the data than Yield~Mg. Mn would likely form a considered predictor variable of the response variable.



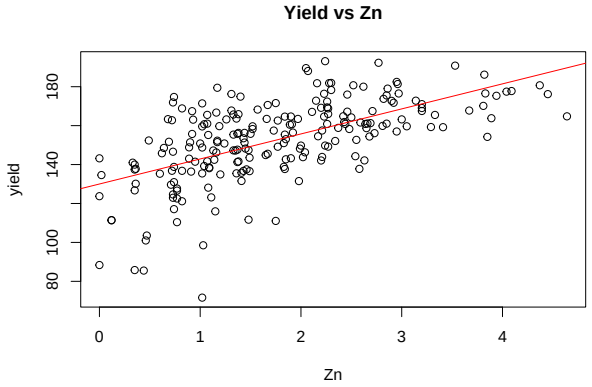
Comment 8

The standard scatter plot for Yield~Na shows that there is a relatively insignificant relationship between nutrient Na and yield, however, slightly positive. Na would not form a high quality predictor of the response variable.



Comment 9

The standard scatter plot for Yield~P shows that there is a relationship between nutrient P and Yield however, we observe a high density of data points in the low P and mid to high yield range which may result in P not representing an ideal predictor of the response variable.

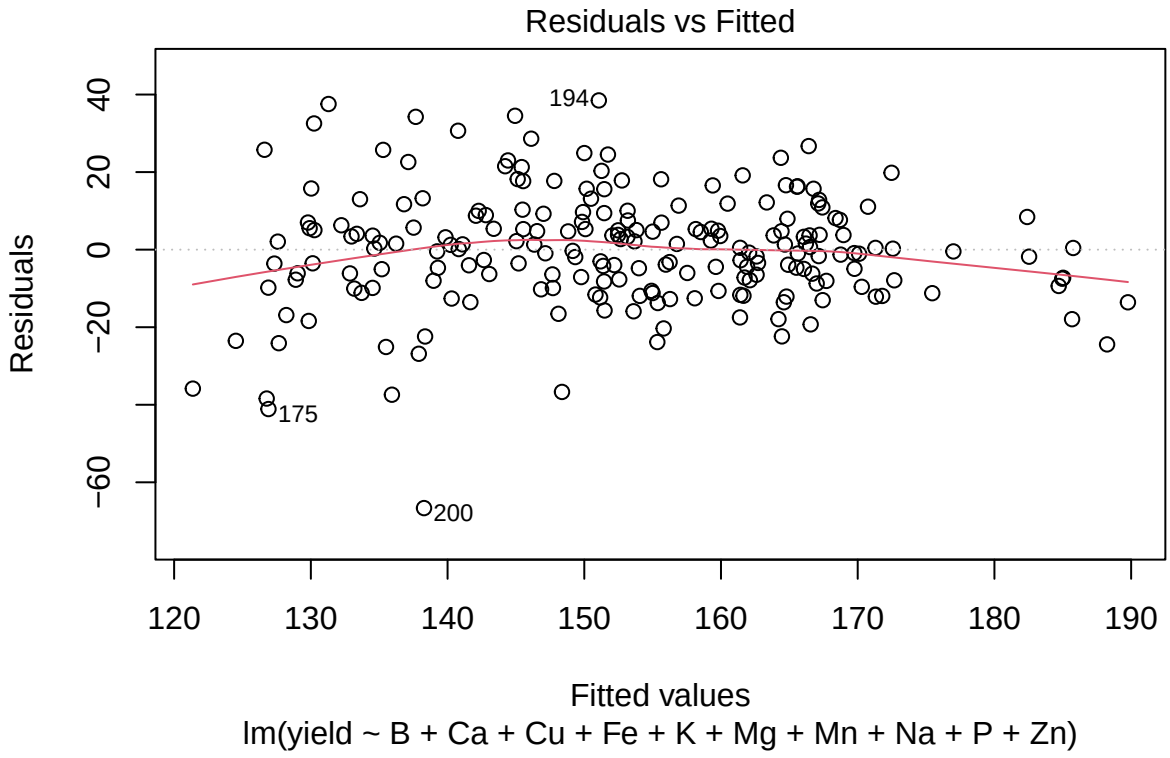


Comment 10

The standard scatter plot for Yield~Zn shows a strong linear relationship between nutrient Zn and Yield. There is a consistent theme of increase Yield for an increase in Zn. Zn would form a high quality predictor of the response variable. head(gy)

Multiple Linear Regression

```
gy.lm = lm(yield~B+Ca+Cu+Fe+K+Mg+Mn+Na+P+Zn, data=gy)
plot(gy.lm, which=1)
```




```
summary(gy.lm)
```

```
##
## Call:
## lm(formula = yield ~ B + Ca + Cu + Fe + K + Mg + Mn + Na + P +
##     Zn, data = gy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66.647  -8.132   0.289   8.051  38.485
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 99.9477940  9.0365559  11.060  < 2e-16 ***
## B           -0.2203443  0.9570173  -0.230 0.818136
## Ca           0.0006444  0.0005269   1.223 0.222758
## Cu           8.4303066  2.2915733   3.679 0.000300 ***
## Fe           0.0908001  0.0251081   3.616 0.000377 ***
## K            0.0250436  0.0442193   0.566 0.571779
```

```
## Mg           0.0092341  0.0129092   0.715 0.475237
## Mn           0.0616807  0.0537394   1.148 0.252406
## Na           0.2425047  0.7193104   0.337 0.736362
## P            0.1231888  0.1181029   1.043 0.298154
## Zn           0.0605328  2.4457687   0.025 0.980279
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.14 on 204 degrees of freedom
## Multiple R-squared:  0.4945, Adjusted R-squared:  0.4697
## F-statistic: 19.95 on 10 and 204 DF,  p-value: < 2.2e-16
```

2.

We can observe in a standard lm for Yield with all nutrients that there are 2 predictor variables Cu & Fe that share a low P statistic value (P=<.10 as per significance level guidance), implying evidence against the H_0 hypothesis. We can therefore investigate the significance of the relationship between the highlight predictor variables and the response variable.

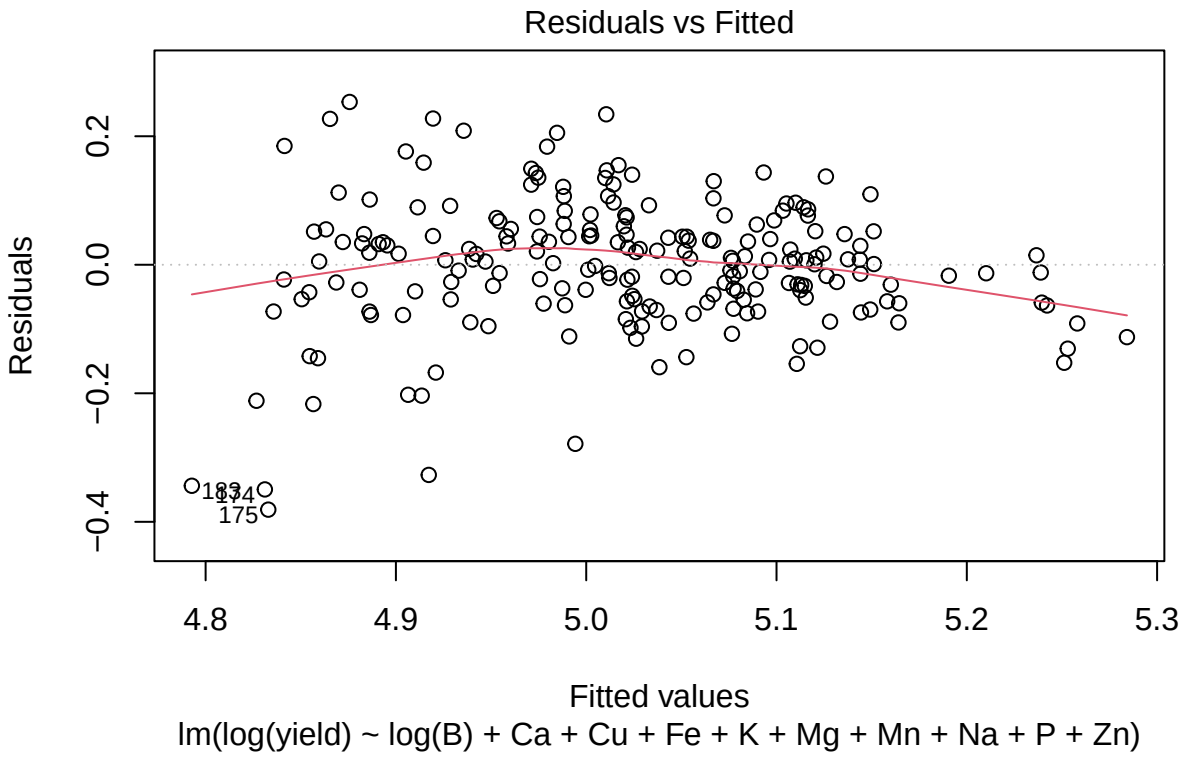
3.

One method of filtering out observations from a data frame is to exclude a concatenation of the row.

```
gy2 = gy[-c(200,0),] # excludes observation 200
print(gy2[c(200,0),]) # prints observations 200 which should now be gone from the df
```

```
##      x y  yield    B   Ca   Cu Fe  K  Mg Mn Na  P   Zn
## 201 14 8 122.94 3.14 1487 0.68 98 94 294 38 11 33 0.73
```

```
gy.lm3 = lm(log(yield)~log(B)+Ca+Cu+Fe+K+Mg+Mn+Na+P+Zn, data=gy2)
plot(gy.lm3, which=1)
```




```
summary(gy.lm3)
```

```
##
## Call:
## lm(formula = log(yield) ~ log(B) + Ca + Cu + Fe + K + Mg + Mn +
##     Na + P + Zn, data = gy2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38102 -0.05431  0.00651  0.05368  0.25333
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.625e+00  5.550e-02  83.339  < 2e-16 ***
## log(B)      -2.347e-02  2.942e-02  -0.798 0.426012
## Ca           2.848e-06  3.548e-06   0.803 0.423092
## Cu           5.735e-02  1.590e-02   3.606 0.000391 ***
## Fe           6.113e-04  1.732e-04   3.531 0.000513 ***
## K            8.571e-05  3.046e-04   0.281 0.778707
## Mg           9.041e-05  8.811e-05   1.026 0.306052
## Mn           4.481e-04  3.675e-04   1.219 0.224146
## Na           5.574e-03  4.377e-03   1.274 0.204278
## P            1.338e-03  8.226e-04   1.626 0.105421
## Zn          -5.398e-03  1.683e-02  -0.321 0.748729
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1037 on 203 degrees of freedom
## Multiple R-squared:  0.4957, Adjusted R-squared:  0.4708
## F-statistic: 19.95 on 10 and 203 DF,  p-value: < 2.2e-16
```

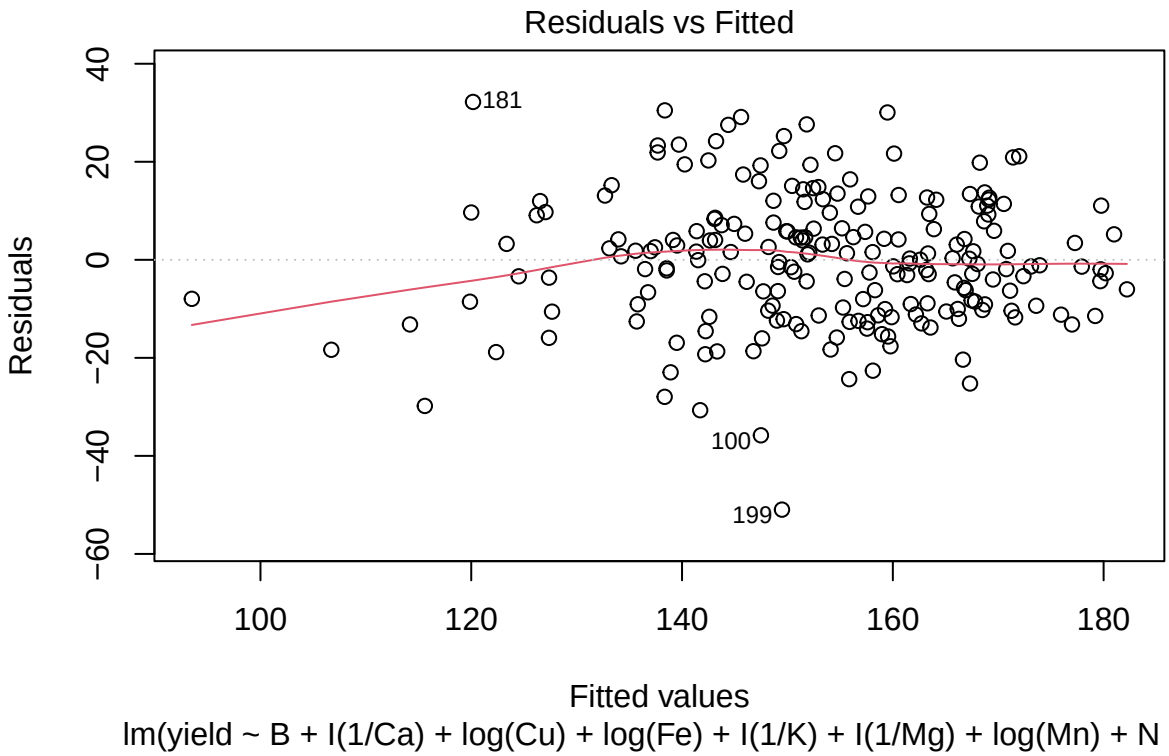
By removing the outlier observation 200 we observe a slight increase in R^2 statistic and lower P values for almost all variables, further supporting our findings on the previous model with a greater degree of confidence. The scale of the residuals axis has been adjusted and presents a better lm fit, therefore making the change reasonable.

4.

```
gy.loglm = lm(yield~B+I(1/Ca)+log(Cu)+log(Fe)+I(1/K)+I(1/Mg)+log(Mn)+Na+log(P)+log(Zn+1), data=gy2)
summary(gy.loglm)
```

```
##
## Call:
## lm(formula = yield ~ B + I(1/Ca) + log(Cu) + log(Fe) + I(1/K) +
##     I(1/Mg) + log(Mn) + Na + log(P) + log(Zn + 1), data = gy2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.963  -9.644  -0.029   8.956  32.202
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.285e+02  2.829e+01   4.540 9.63e-06 ***
## B            3.798e-01  7.856e-01   0.483 0.629289
## I(1/Ca)      2.396e+04  1.262e+04   1.899 0.058977 .
## log(Cu)      1.366e+01  3.810e+00   3.586 0.000421 ***
## log(Fe)     -2.082e+00  2.855e+00  -0.729 0.466769
## I(1/K)      -7.911e+02  4.140e+02  -1.911 0.057423 .
## I(1/Mg)     -4.346e+03  1.975e+03  -2.200 0.028938 *
## log(Mn)      3.194e+00  4.597e+00   0.695 0.487902
## Na          -9.069e-02  5.604e-01  -0.162 0.871598
## log(P)       5.652e+00  3.259e+00   1.734 0.084370 .
## log(Zn + 1)  7.425e+00  5.103e+00   1.455 0.147255
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.62 on 203 degrees of freedom
## Multiple R-squared:  0.5616, Adjusted R-squared:  0.5401
## F-statistic: 26.01 on 10 and 203 DF,  p-value: < 2.2e-16
```

```
plot(gy.loglm,which=1)
```



By transforming some of the predictor variables with log and standardization, we observe a significant increase in the R^2 statistic, suggesting an improved model fit accounting for more variance than previous models. With variable transformations there are caveats, such as the problem of multicollinearity due to similarities in the predictor variable data relationships. Regardless, our transformations have resulted in surfacing relationships across a number of variables with a P = <.10 significance level: Ca, Cu, K, Mg and P. This aligns well with earlier speculation on what variables would form good predictors by analysing the bivariate relationships.

Subset Selection

5.

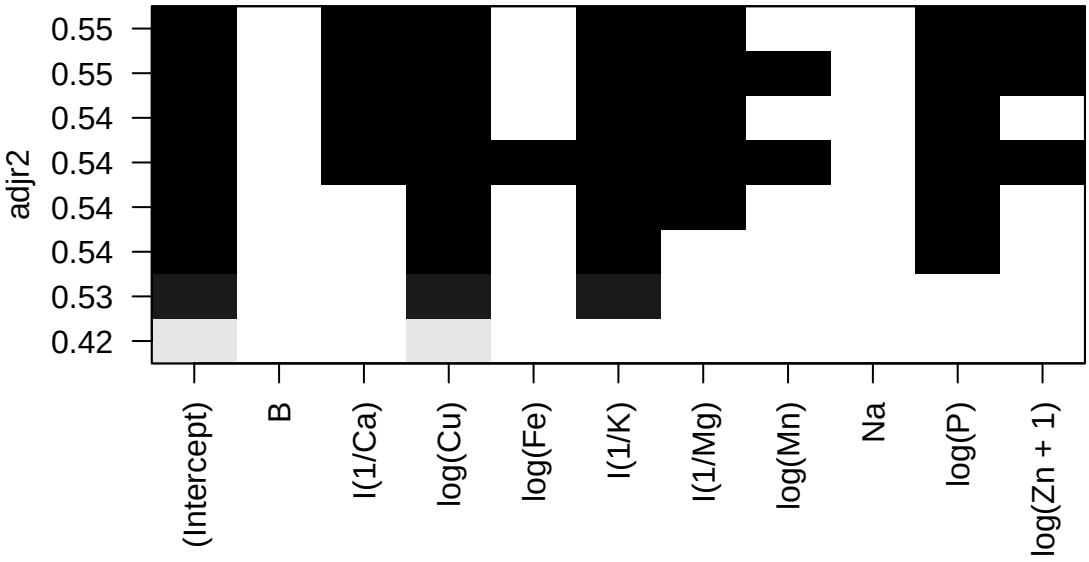
```
## Warning: package 'leaps' was built under R version 4.1.3
```

```
gy2.fsstest= regsubsets(yield~B+I(1/Ca)+log(Cu)+log(Fe)+I(1/K)+I(1/Mg)+log(Mn)+Na+log(P)+log(Zn+1),
                        data=gy2, method="forward")
summary(gy2.fsstest)
```

```
## Subset selection object
## Call: regsubsets.formula(yield ~ B + I(1/Ca) + log(Cu) + log(Fe) +
##     I(1/K) + I(1/Mg) + log(Mn) + Na + log(P) + log(Zn + 1), data = gy2,
##     method = "forward")
## 10 Variables  (and intercept)
##              Forced in Forced out
```

```
## B               FALSE      FALSE
## I(1/Ca)         FALSE      FALSE
## log(Cu)         FALSE      FALSE
## log(Fe)         FALSE      FALSE
## I(1/K)          FALSE      FALSE
## I(1/Mg)         FALSE      FALSE
## log(Mn)         FALSE      FALSE
## Na              FALSE      FALSE
## log(P)          FALSE      FALSE
## log(Zn + 1)     FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: forward
##          B   I(1/Ca) log(Cu) log(Fe) I(1/K) I(1/Mg) log(Mn) Na  log(P)
## 1  ( 1 ) " " " "     "*"     " "     " "    " "     " "     " " " "
## 2  ( 1 ) " " " "     "*"     " "     "*"    " "     " "     " " " "
## 3  ( 1 ) " " " "     "*"     " "     "*"    " "     " "     " " "*"
## 4  ( 1 ) " " " "     "*"     " "     "*"    "*"     " "     " " "*"
## 5  ( 1 ) " " "*"     "*"     " "     "*"    "*"     " "     " " "*"
## 6  ( 1 ) " " "*"     "*"     " "     "*"    "*"     " "     " " "*"
## 7  ( 1 ) " " "*"     "*"     " "     "*"    "*"     "*"     " " "*"
## 8  ( 1 ) " " "*"     "*"     "*"     "*"    "*"     "*"     " " "*"
##          log(Zn + 1)
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) " "
## 5  ( 1 ) " "
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
## 8  ( 1 ) "*"
```

```
plot(gy2.fsstest, scale="adjr2")
```



Noting from our previous observations of the transformed variables lm: Ca, Cu, K, Mg and P all exhibited a relationship with the response variable. We see this echoed again during our forward subset selection approach by observing the summary and plot (adjr2) for our variables. Across the evaluation of our predictor variables at each quantity of the # of variables, the most relational variables in order of confidence are: Cu, K, P, Mg and Ca. Our Forward step selection algorithm advises that the next most significant variable in a 6 step calculation would be Zn however, Zn does not meet our R^2 statistic requirements of P=<.10 and therefore may be considered for exclusion in fitting our model for concerns of over-fitting the model to training data.

Model Regularisation

6.

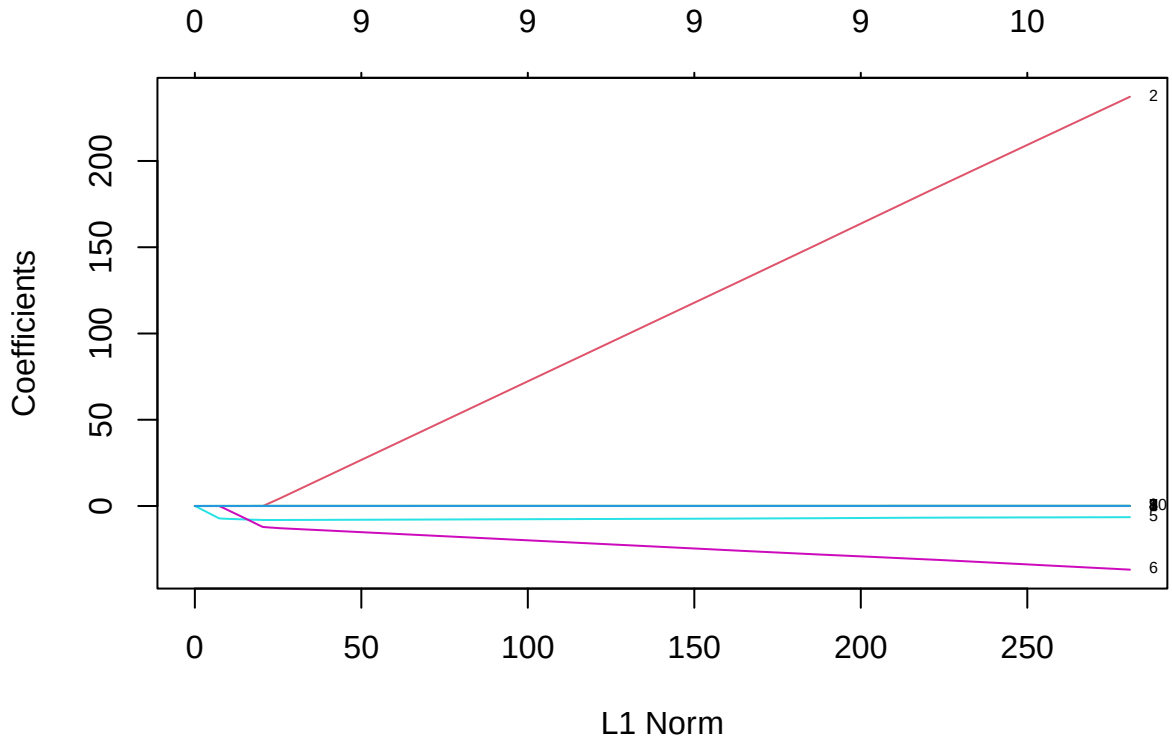
```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 4.1.3
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-4
```

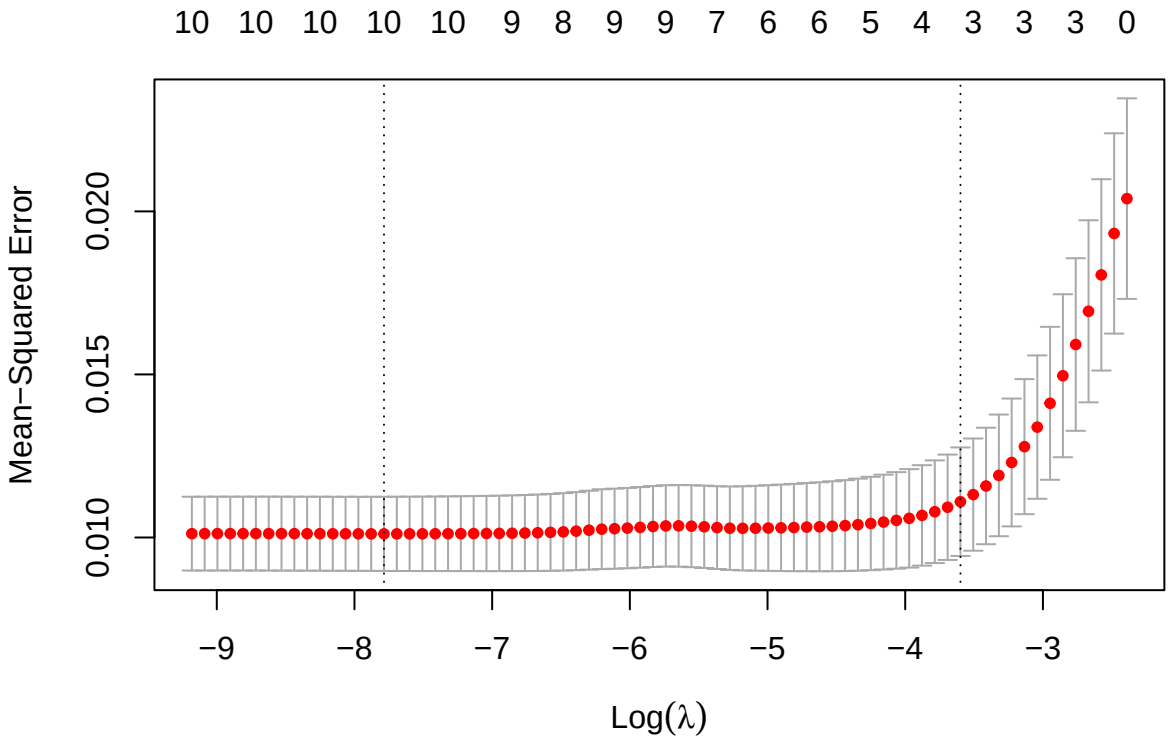
```
X = model.matrix(gy.loglm)[,-1]
Y = log(gy2$yield)
lasso.gy = glmnet(X, Y, alpha=1)
plot(lasso.gy, label=TRUE)
```



Here we have implemented LASSO regression, which utilises L1 norm for regularisation aka the sum of the coefficients absolute values. The LASSO method uses λ value increases as a penalty function to regularize the coefficients. The plot is interpreted from right to left where each of the lines represents a variable converging toward the MSE $= 0$ which eventually reduces to the RSS. We observe the LASSO method performing a degree of subset selection by noting the drop in variables from 10 to 9 to 0 as we converge on MSE $= 0$.

7.

```
library(glmnet)
cv_lasso.gy = cv.glmnet(X, Y, alpha=1) #cross-validation for lasso
plot(cv_lasso.gy)
```



We interpret this plot from left to right, with the MSE scaled on the Y axis and $log(\lambda)$ on the X axis. Note the Y scale starting at MSE = 0 and increases in value as the value of lambda increases; adding to bias. We observe the cross-validation model reducing the number of variables at a cost to the estimated standard error denoted by the vertical interval lines extending out from each red point. The leftmost dotted vertical line represents the model with the least amount of estimated MSE. The right-most dotted vertical line indicates to us the tradeoff we can consider where the model's estimated MSE is 1 standard error from the most accurate standard error model (left-most line). As the model increases in λ it performs subset selection with only a reasonable increase in MSE. To identify the lambda value as recommended by our cv model, we can perform the 1se command, log transformed:

```
cv_lasso.gy$lambda.1se # cross validation lasso model standard error
```

```
## [1] 0.02734946
```

```
log(cv_lasso.gy$lambda.1se)
```

```
## [1] -3.599059
```

-3.8 to -3.5 $log(\lambda)$ would offer a reasonable trade-off resulting in a more simplistic model at the cost of only a slight increase in estimated MSE. A range has been provided here as with each execution of this code, a different set of data is selected for the calculation, therefore producing slightly different results.

Our coefficients for the model are as follows:

```
coef(cv_lasso.gy) # cross-validation lasso coefficients
```

```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                        s1
## (Intercept)  5.012157248
## B            .
## I(1/Ca)      .
## log(Cu)      0.070263832
## log(Fe)      .
## I(1/K)      -6.448698268
## I(1/Mg)      .
## log(Mn)      .
## Na           .
## log(P)       0.000886324
## log(Zn + 1)  0.043093982
```

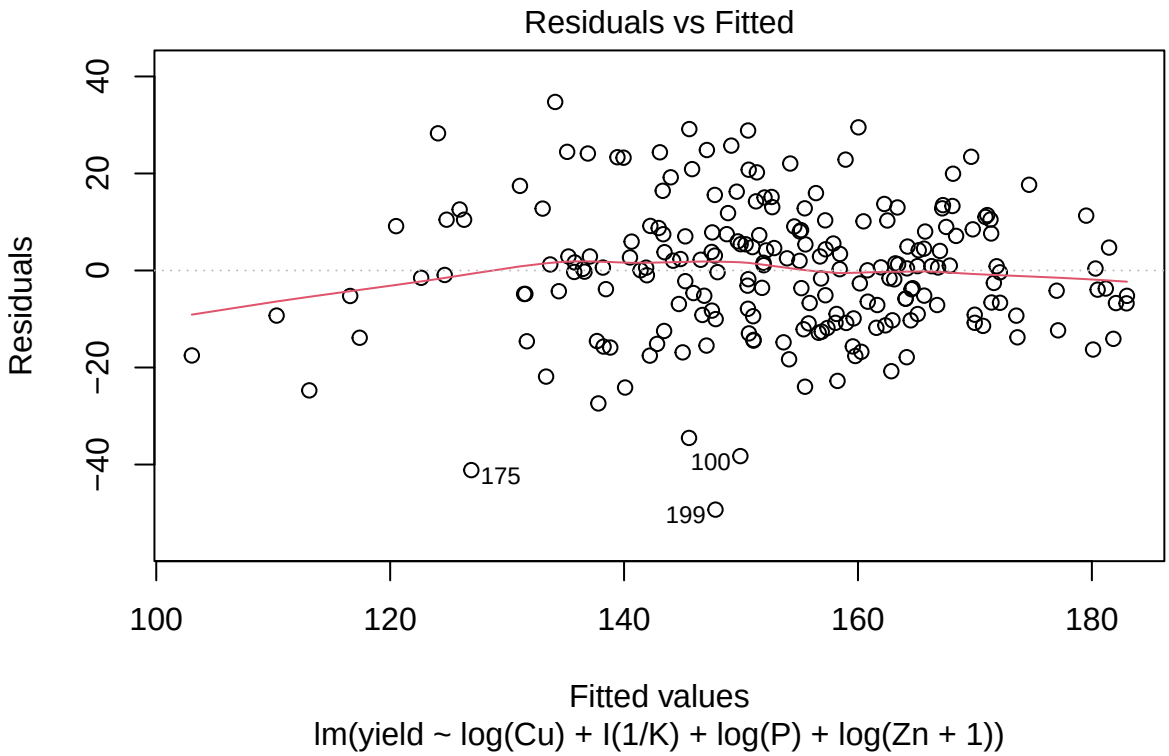
We observe these 4 coefficients (excl intercept) log(Cu), I(1/K), log(P) and log(Zn+1) as the selection performed by the cv model representing the most promising coefficients to consider in a trade off that reduces the complexity of the model at only a slight increase to estimated MSE.

8.

```
head(gy2) # check column names
```

```
##   x y  yield    B   Ca   Cu  Fe   K  Mg Mn Na  P   Zn
## 1 1 1 174.73 2.74 2608 1.44 192  88 442 62 14 16 0.74
## 2 2 1 147.43 4.23 3167 1.73 180  81 522 66 16 20 1.48
## 3 3 1 155.39 2.78 2359 1.27 249 110 413 85 15 24 1.33
## 4 4 1 141.46 1.92 1766 0.76 189 117 314 63 15 21 0.95
## 5 5 1 148.54 3.95 1647 0.48 188  99 304 73 15 17 0.64
## 6 6 1 135.30 1.99 1505 0.44 147  72 263 58 13 14 0.60
```

```
gy3 = subset(gy2, select= c("yield", "Cu", "K", "P", "Zn")) # select subset of variables
gy3.lm = lm(yield~log(Cu)+I(1/K)+log(P)+log(Zn+1), data=gy3) # transform as per previous item
plot(gy3.lm, which=1)
```



The variable coefficients supplied by the cv model were almost identical to those identified in the fss model, with the main difference being that the cv model suggested nutrient Zn over Ca and Mg which were more highly regarded in the fss model.

Recalling in the earlier log scatter plot analyses the respective P statistic for Zn was close to the P=<.10 significance level, but not quite there- the difference between Zn, Mg and Ca was relatively negligible- this was also observed in the adjr2 plot. The reason for the difference in coefficient suggestions here from cv models is a combination of the limitations of the fss model design and multicollinearity. The fss model is only evaluating a limited number of parameters through each 'step' iteration without an indepth consideration for error. The cv model was able to evaluate all predictors across MSE = 0 up to a reasonable tradeoff MSE of 1se inclusive of the lambda penalty to suggest a more accurate model, with more consideration for error.