M565 Exam2_Part2

Neeraj Namani

2024-03-15

```
Data <- read.csv("~/Downloads/PlantData.txt", sep = "",
                 stringsAsFactors = TRUE)
head(Data)
```

```
##    NR  Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1 269 21345    44.89  344 35.5   72 13275  13189       251
## 2 260 20170    41.41  219  5.5   19 12273  14575      9691
## 3 260 10590    42.86   51 12.5   63  4941  11952      2060
## 4 262 18134    43.16  142 25.4   38  9332  12397         0
## 5 257 25565    42.57   14 31.6   37  8565  14683      3988
## 6 263 21985    44.89   73 14.9   73 10066  13255         0
```

Exploratory Data Analysis:

```
# Data Frame Description

str(Data)
```

```
## 'data.frame':    137 obs. of  9 variables:
##  $ NR       : int  269 260 260 262 257 263 260 266 250 259 ...
##  $ Area     : int  21345 20170 10590 18134 25565 21985 16460 22887 3092 4757 ...
##  $ Latitude : num  44.9 41.4 42.9 43.2 42.6 ...
##  $ Elev     : int  344 219 51 142 14 73 118 429 32 126 ...
##  $ Dist     : num  35.5 5.5 12.5 25.4 31.6 14.9 39.8 36.8 6.2 41 ...
##  $ Soil     : int  72 19 63 38 37 73 48 55 17 54 ...
##  $ Years    : int  13275 12273 4941 9332 8565 10066 7213 12233 6186 7499 ...
##  $ Deglac   : int  13189 14575 11952 12397 14683 13255 14206 13746 13129 13850 ...
##  $ Human.pop: int  251 9691 2060 0 3988 0 4961 1173 0 0 ...
```

```
# Displaying Summary Statistics for all variables

summary(Data)
```

```
##        NR             Area          Latitude          Elev      
##  Min.   :246.0   Min.   :  288   Min.   :41.08   Min.   :  6.0  
##  1st Qu.:257.0   1st Qu.: 5636   1st Qu.:41.86   1st Qu.:109.0  
##  Median :259.0   Median :12975   Median :42.86   Median :191.0  
##  Mean   :259.3   Mean   :12718   Mean   :42.91   Mean   :220.5  
##  3rd Qu.:263.0   3rd Qu.:19378   3rd Qu.:43.85   3rd Qu.:347.0  
##  Max.   :269.0   Max.   :26525   Max.   :44.94   Max.   :465.0  
##       Dist            Soil           Years           Deglac     
##  Min.   : 0.40   Min.   : 1.00   Min.   : 3834   Min.   :11732  
##  1st Qu.: 8.90   1st Qu.:18.00   1st Qu.: 6087   1st Qu.:12535  
##  Median :22.60   Median :34.00   Median : 9060   Median :13225  
##  Mean   :21.53   Mean   :35.54   Mean   : 8918   Mean   :13337  
##  3rd Qu.:32.60   3rd Qu.:55.00   3rd Qu.:11588   3rd Qu.:14136
##  Max.   :42.50   Max.   :73.00   Max.   :13996   Max.   :14998
##    Human.pop
##  Min.   :    0
##  1st Qu.:    0
##  Median :    0
##  Mean   : 2054
##  3rd Qu.: 3515
##  Max.   :10695
```

```
sapply(Data, is.factor)
```

```
##        NR      Area  Latitude      Elev      Dist      Soil     Years    Deglac
##     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE
## Human.pop
##     FALSE
```

Here, we can see that all columns are “FALSE”, it means that there are no categorical variables.

```
# Check for any missing values

missing_values <- colSums(is.na(Data))
missing_values
```

```
##        NR      Area  Latitude      Elev      Dist      Soil     Years    Deglac
##         0         0         0         0         0         0         0         0
## Human.pop
##         0
```

Here, there are no missing values in the data.

```
# Find the unique value count for all the columns in the data to know whether there are any categorical

unique_counts <- sapply(Data, function(x) length(unique(x)))
unique_counts
```

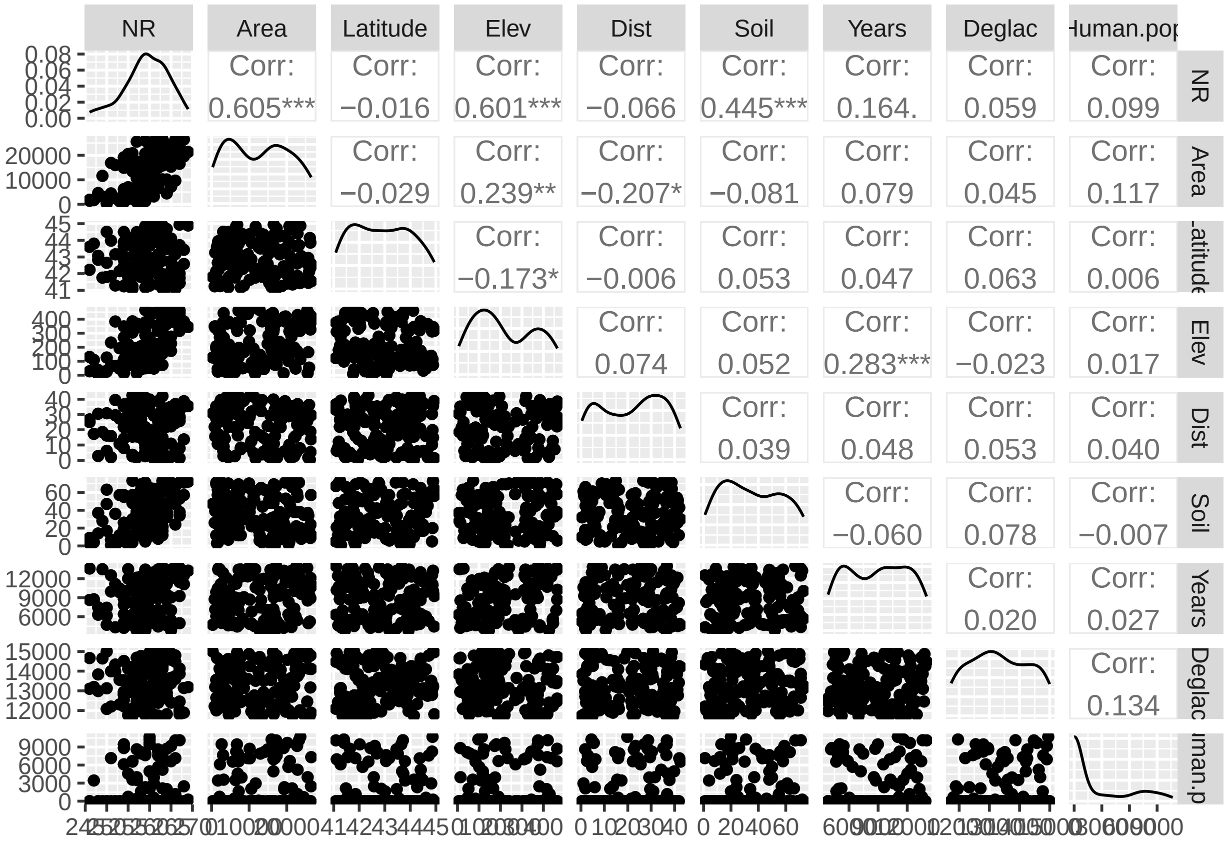
```
##        NR      Area  Latitude      Elev      Dist      Soil     Years    Deglac
##        24       133       118       120       117        66       137       135
## Human.pop
##        51
```

```
library(GGally)
```

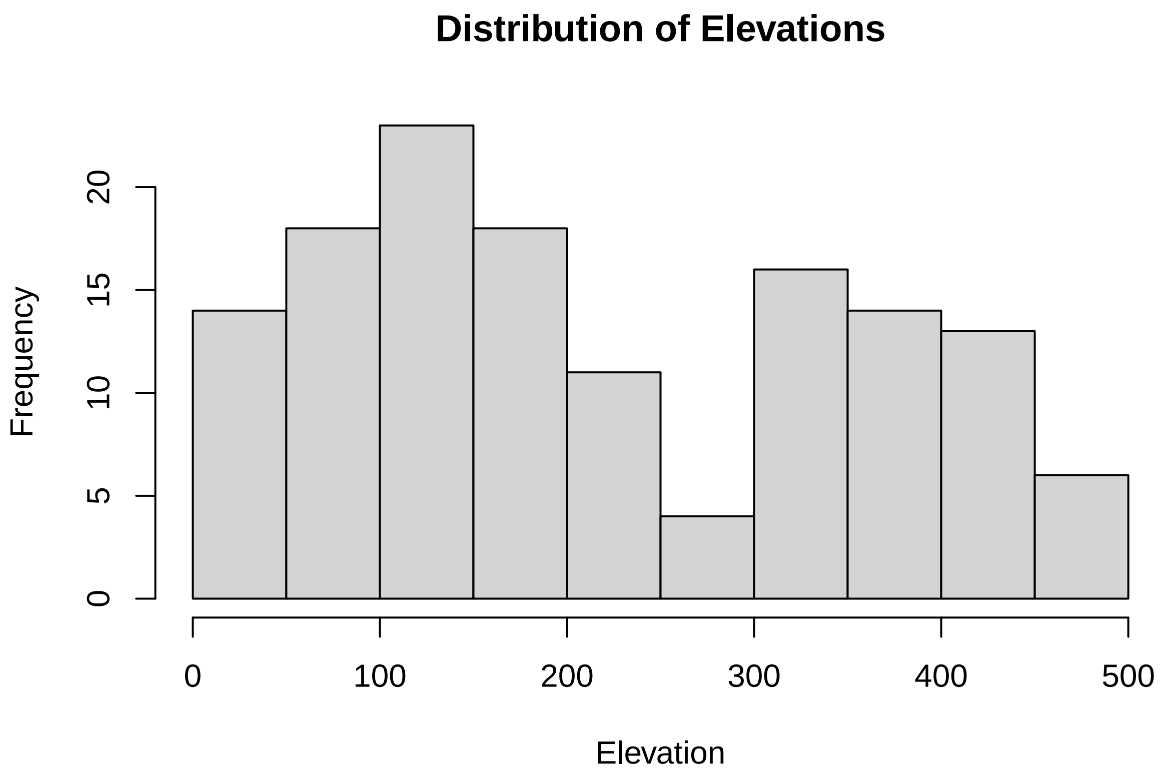
```
## Loading required package: ggplot2

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(ggplot2)
```

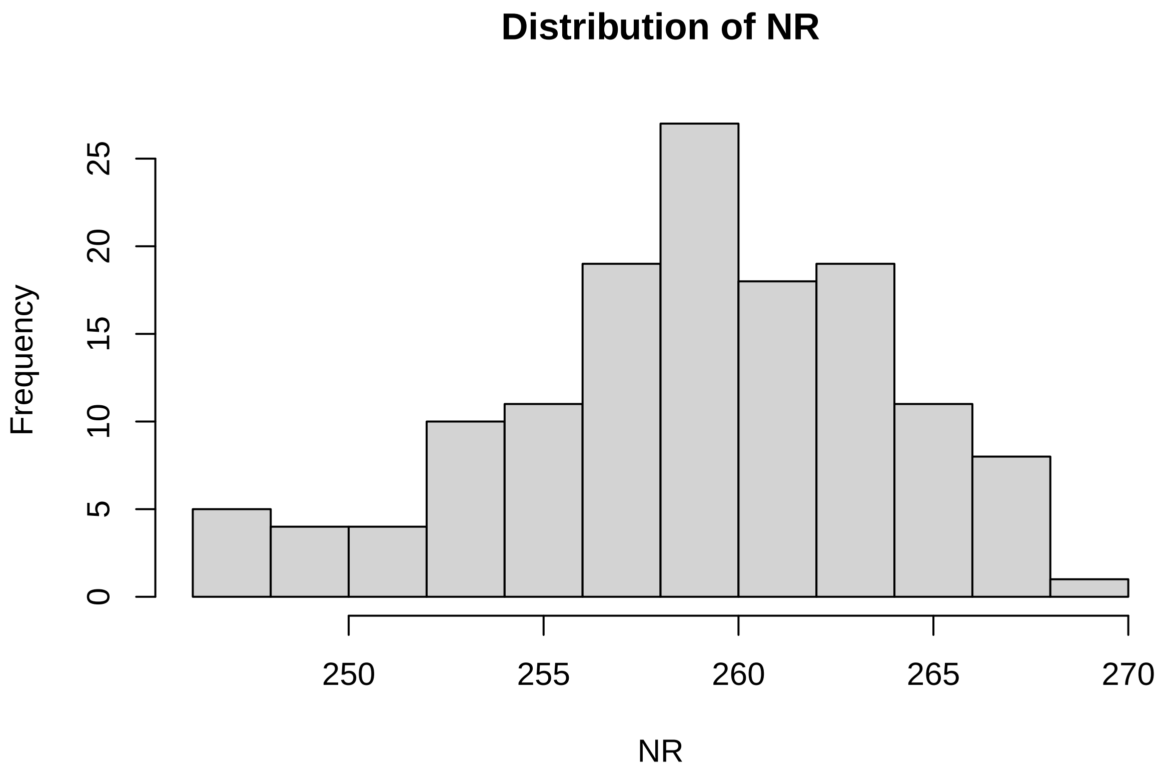
```
# Graphical display of data

ggpairs(Data)
```



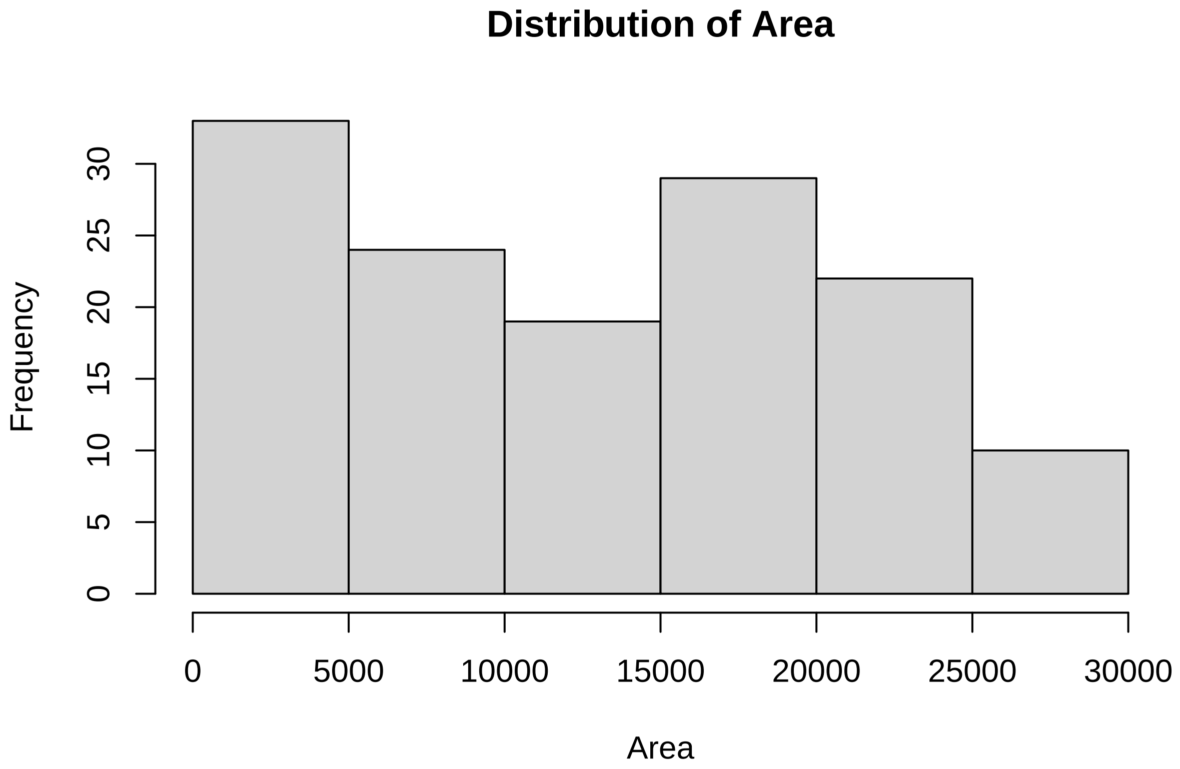
```
# Distribution of Numeric Variables
# Histogram for all numeric variables


hist(Data$Elev, main = "Distribution of Elevations", xlab = "Elevation")
```



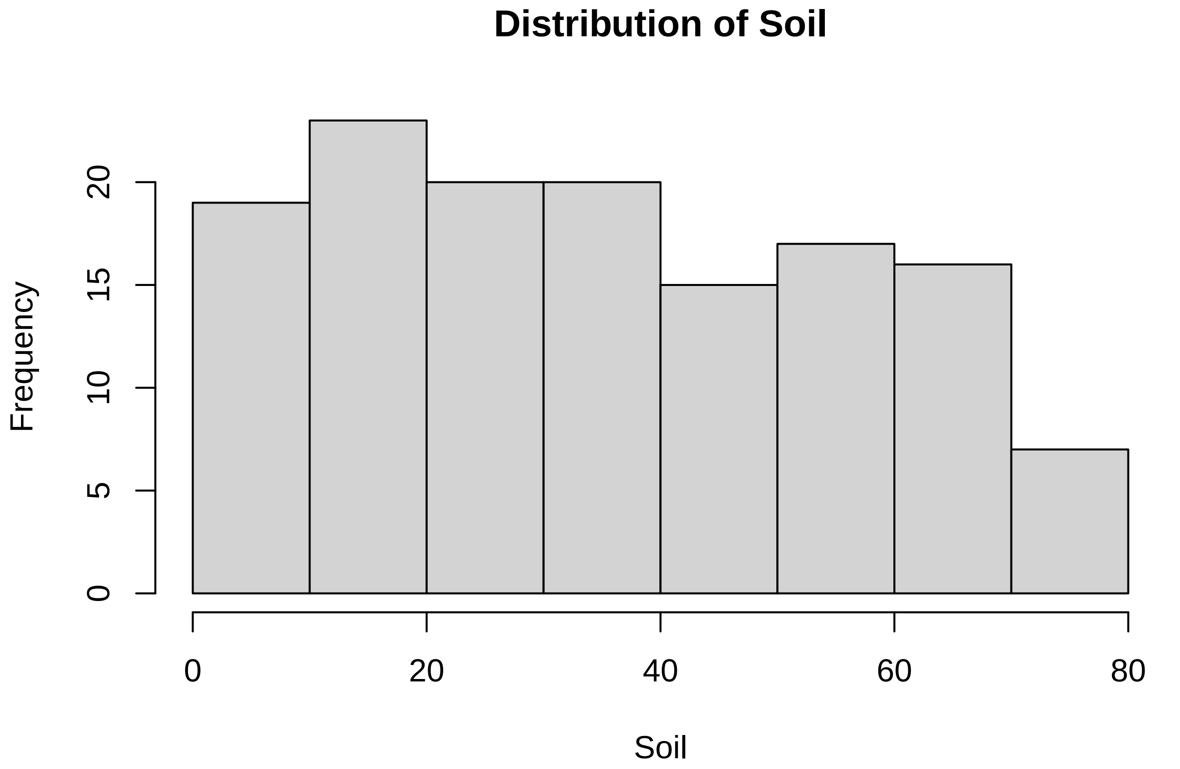
```
hist(Data$NR, main = "Distribution of NR", xlab = "NR")
```



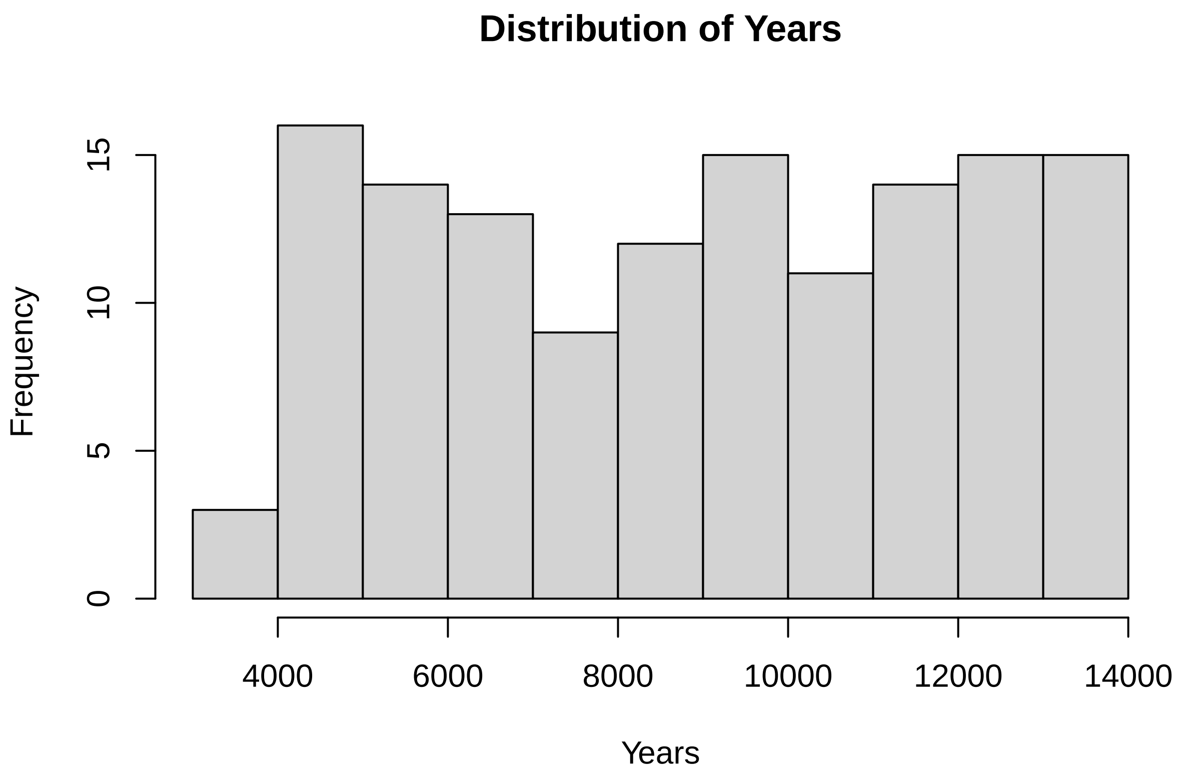
```
hist(Data$Area, main = "Distribution of Area", xlab = "Area")
```



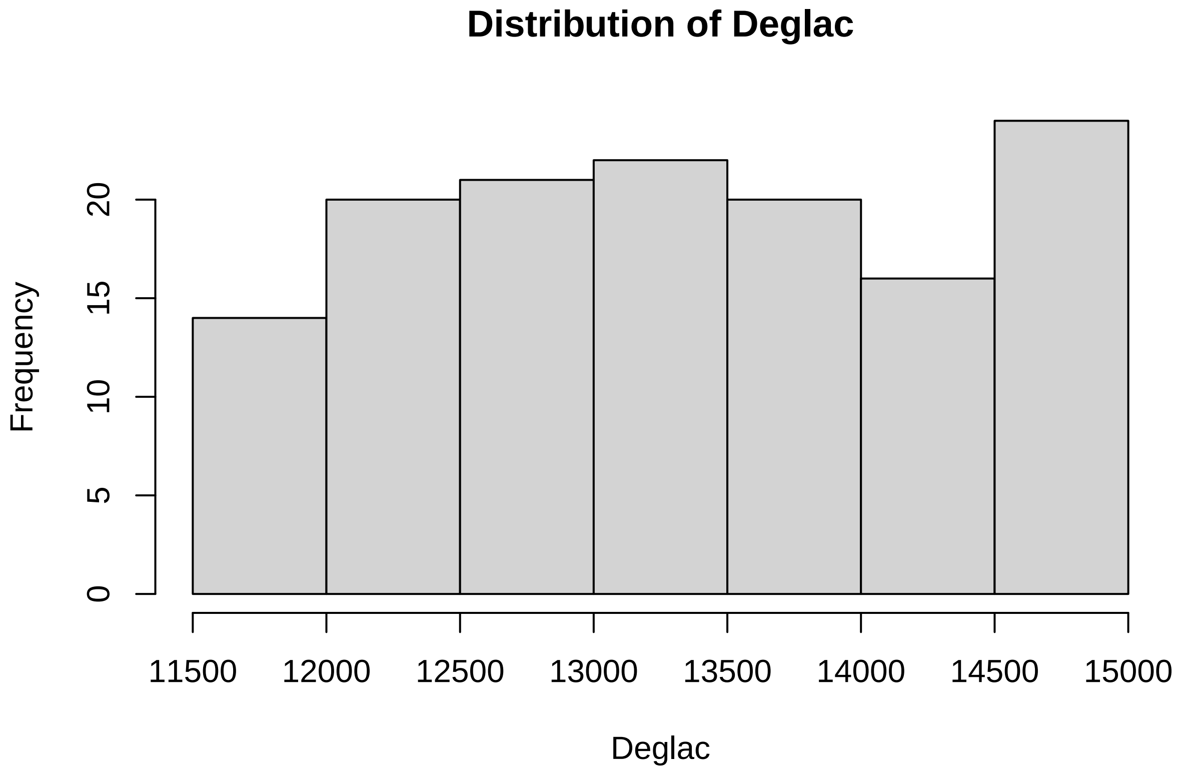
```r
hist(Data$Soil, main = "Distribution of Soil", xlab = "Soil")
```



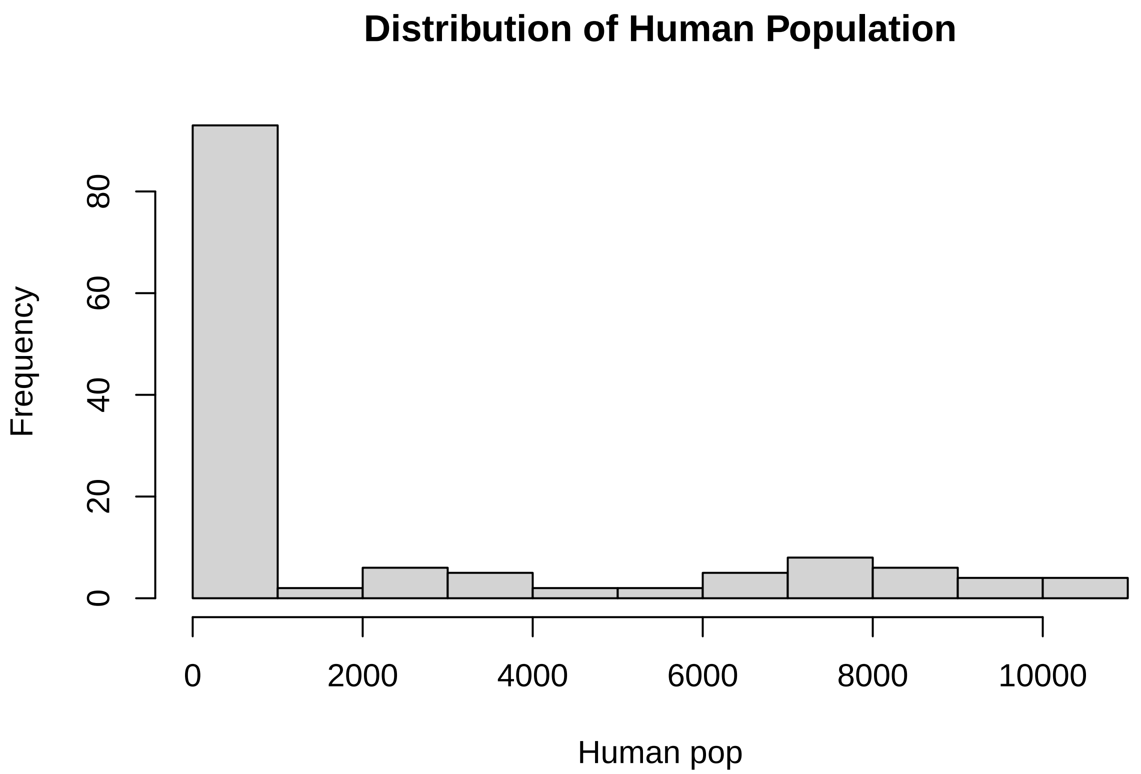
```r
hist(Data$Years, main = "Distribution of Years", xlab = "Years")
```



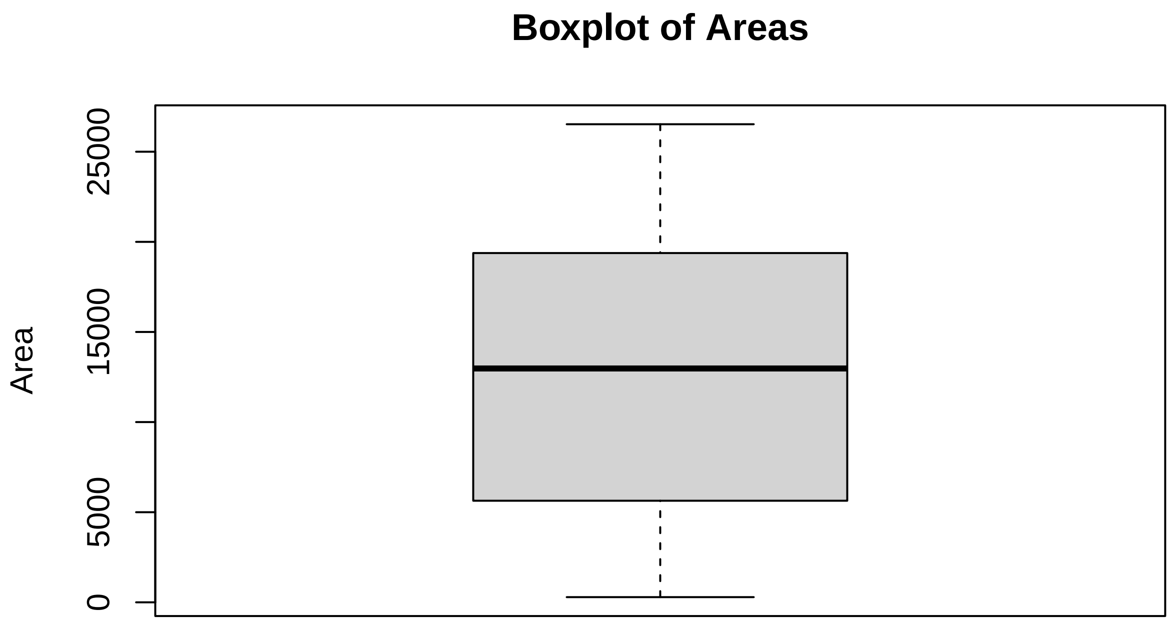
```r
hist(Data$Deglac, main = "Distribution of Deglac", xlab = "Deglac")
```



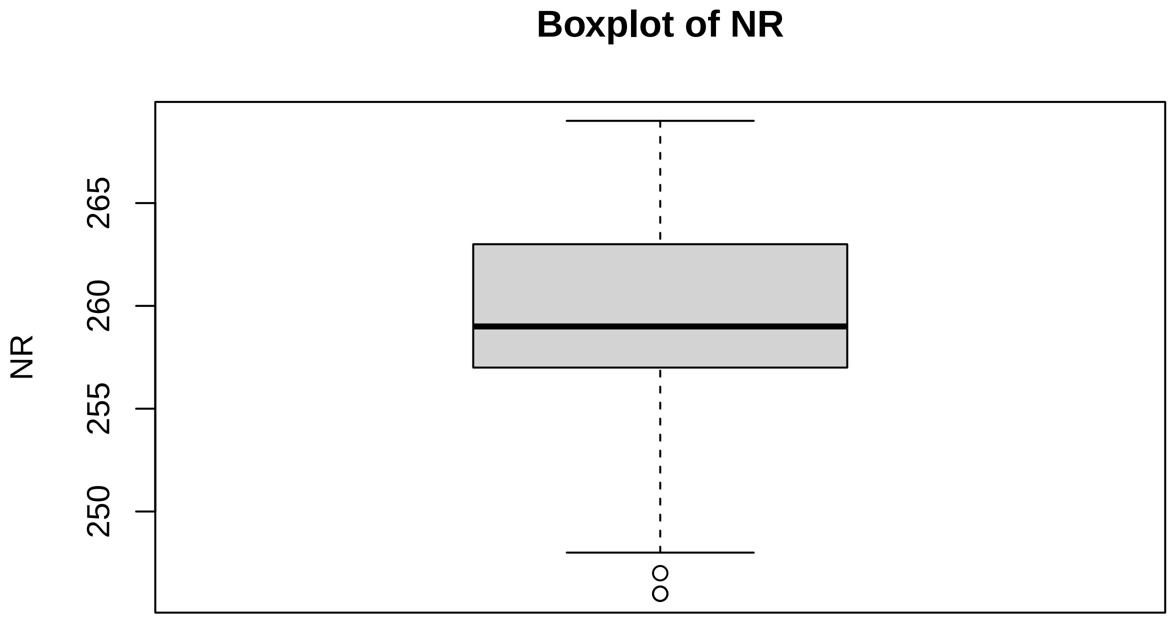
```r
hist(Data$Human.pop, main = "Distribution of Human Population", xlab = "Human pop")
```



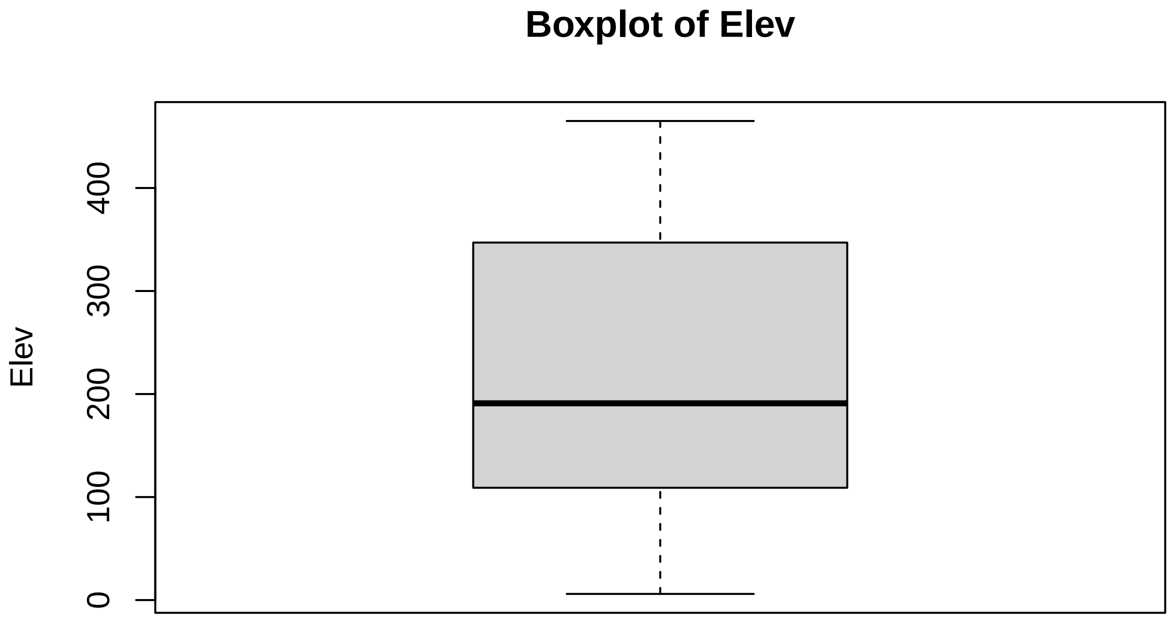
```
# Boxplot for all numeric variables

boxplot(Data$Area, main = "Boxplot of Areas", ylab = "Area")
```



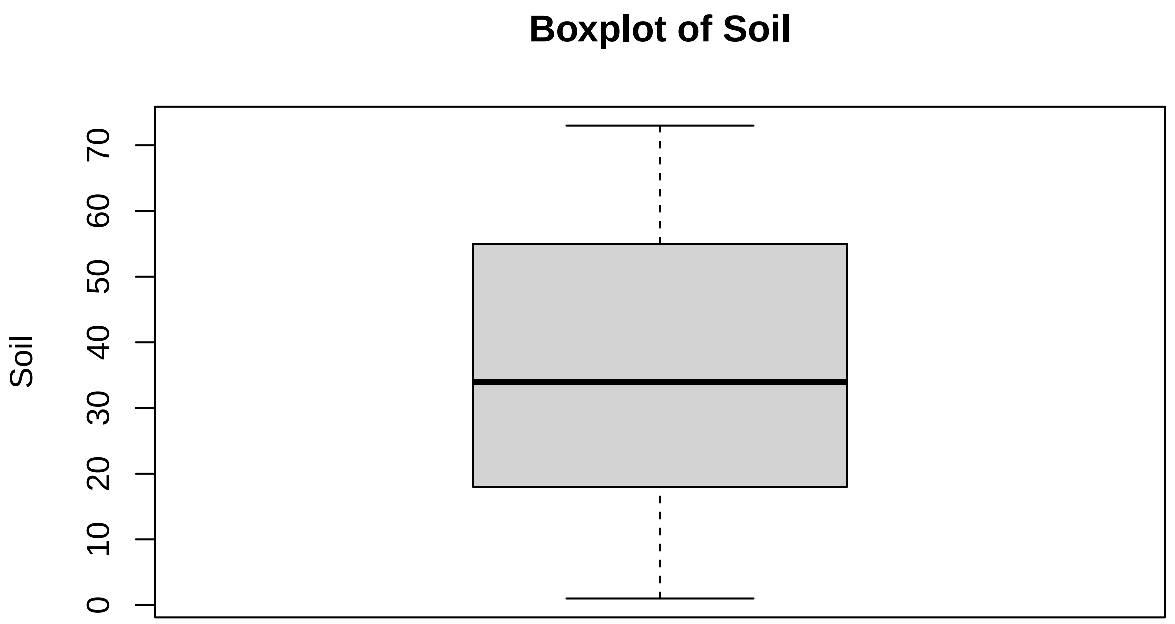
```
boxplot(Data$NR, main = "Boxplot of NR", ylab = "NR")
```



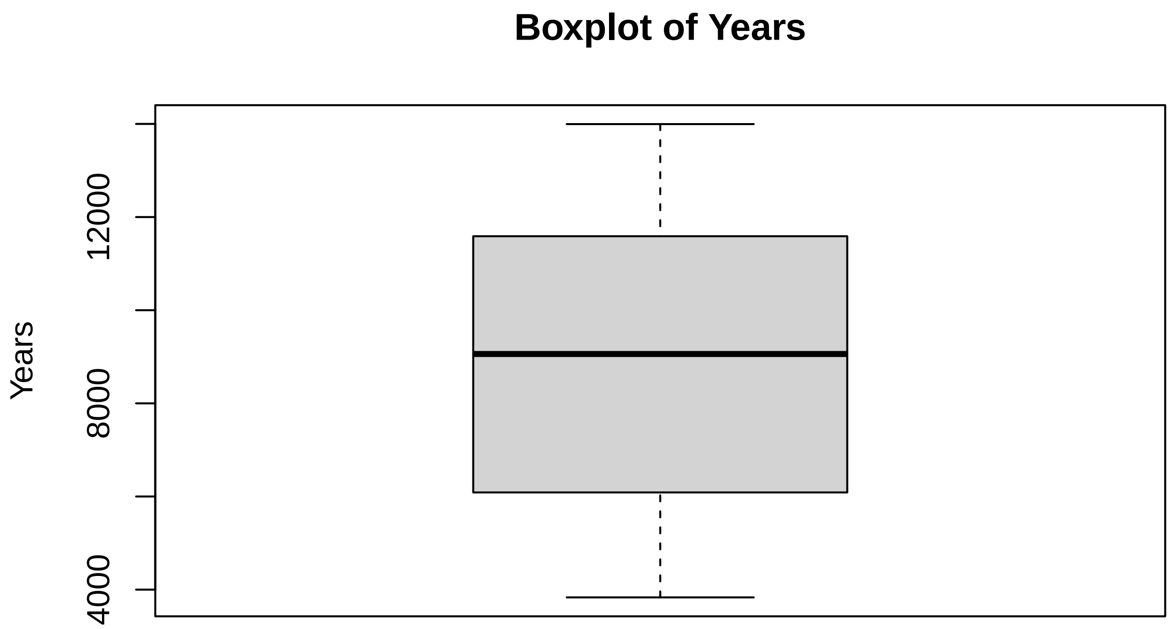
```
boxplot(Data$Elev, main = "Boxplot of Elev", ylab = "Elev")
```



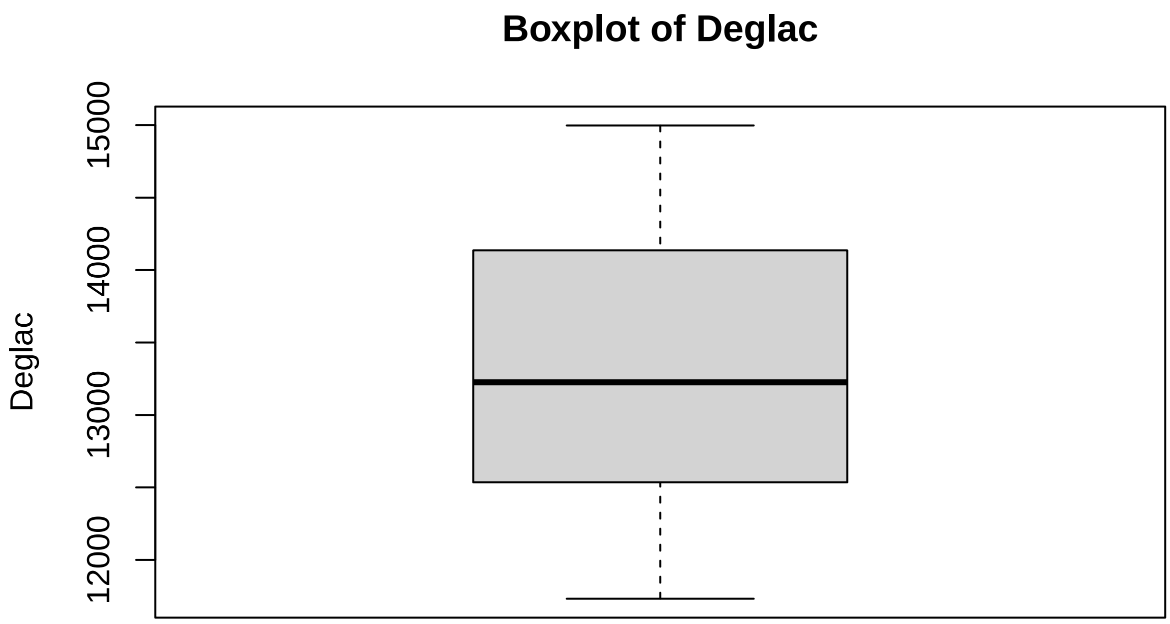
```
boxplot(Data$Soil, main = "Boxplot of Soil", ylab = "Soil")
```



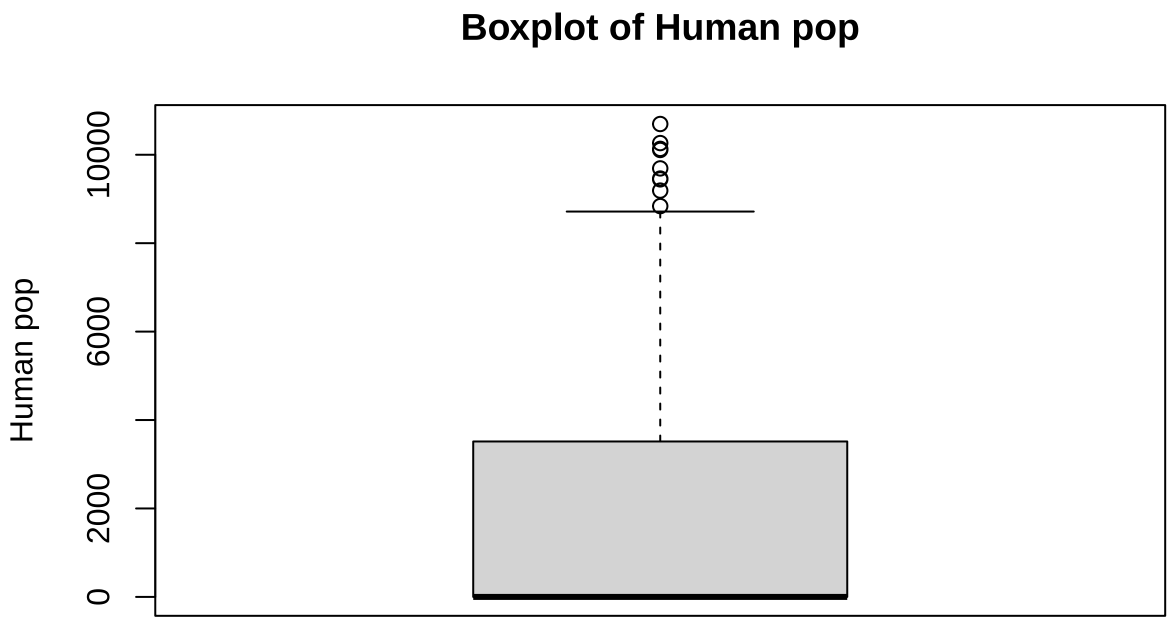
```r
boxplot(Data$Years, main = "Boxplot of Years", ylab = "Years")
```



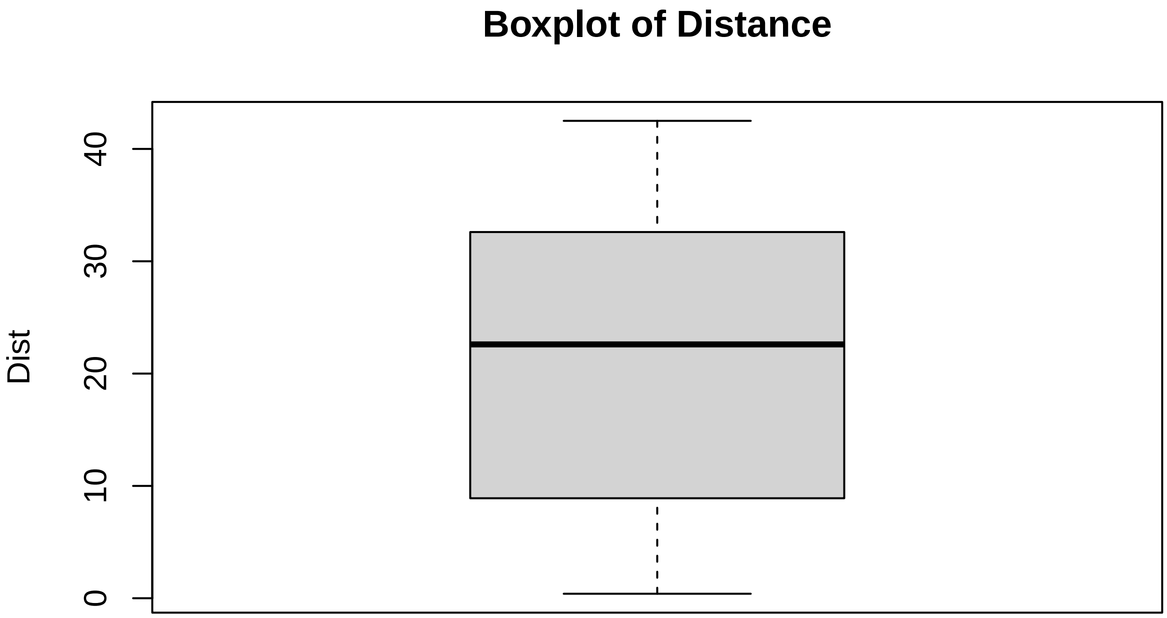
```r
boxplot(Data$Deglac, main = "Boxplot of Deglac", ylab = "Deglac")
```



```
boxplot(Data$Human.pop, main = "Boxplot of Human pop", ylab = "Human pop")
```



```
# Boxplot for Distance in Data
boxplot(Data$Dist, main = "Boxplot of Distance", ylab = "Dist")
```



Let us construct the Correlation matrix for all numeric variables present in the dataset.

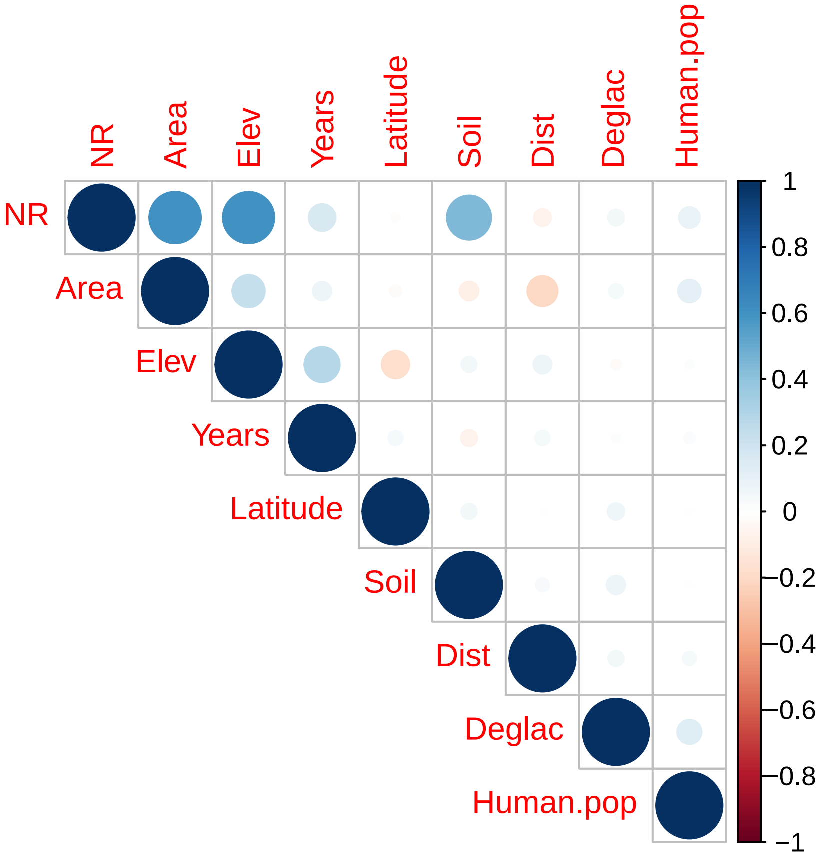
```r
# Correlation matrix for numeric variables
cor_matrix <- cor(Data[,sapply(Data, is.numeric)])
print(cor_matrix)
```

```
##                   NR        Area     Latitude        Elev         Dist
## NR         1.00000000  0.60468048 -0.016223652  0.60101050 -0.066034481
## Area       0.60468048  1.00000000 -0.029495273  0.23932063 -0.206994524
## Latitude  -0.01622365 -0.02949527  1.000000000 -0.17263287 -0.006414827
## Elev       0.60101050  0.23932063 -0.172632866  1.00000000  0.074213753
## Dist      -0.06603448 -0.20699452 -0.006414827  0.07421375  1.000000000
## Soil       0.44457900 -0.08070415  0.052769736  0.05231514  0.039389016
## Years      0.16425651  0.07850543  0.047116069  0.28255288  0.048288775
## Deglac     0.05863866  0.04531215  0.063103498 -0.02272465  0.053333883
## Human.pop  0.09875745  0.11680390  0.005987895  0.01658915  0.040495312
##                 Soil       Years      Deglac    Human.pop
## NR         0.44457900  0.16425651  0.05863866  0.098757449
## Area      -0.08070415  0.07850543  0.04531215  0.116803901
## Latitude   0.05276974  0.04711607  0.06310350  0.005987895
## Elev       0.05231514  0.28255288 -0.02272465  0.016589152
## Dist       0.03938902  0.04828877  0.05333388  0.040495312
## Soil       1.00000000 -0.06034206  0.07844720 -0.006764540
## Years     -0.06034206  1.00000000  0.01998947  0.026764018
## Deglac     0.07844720  0.01998947  1.00000000  0.133680189
## Human.pop -0.00676454  0.02676402  0.13368019  1.000000000
```

```r
# Plot the correlation matrix
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```r
corrplot(cor_matrix, method = "circle", type = "upper", order = "hclust")
```



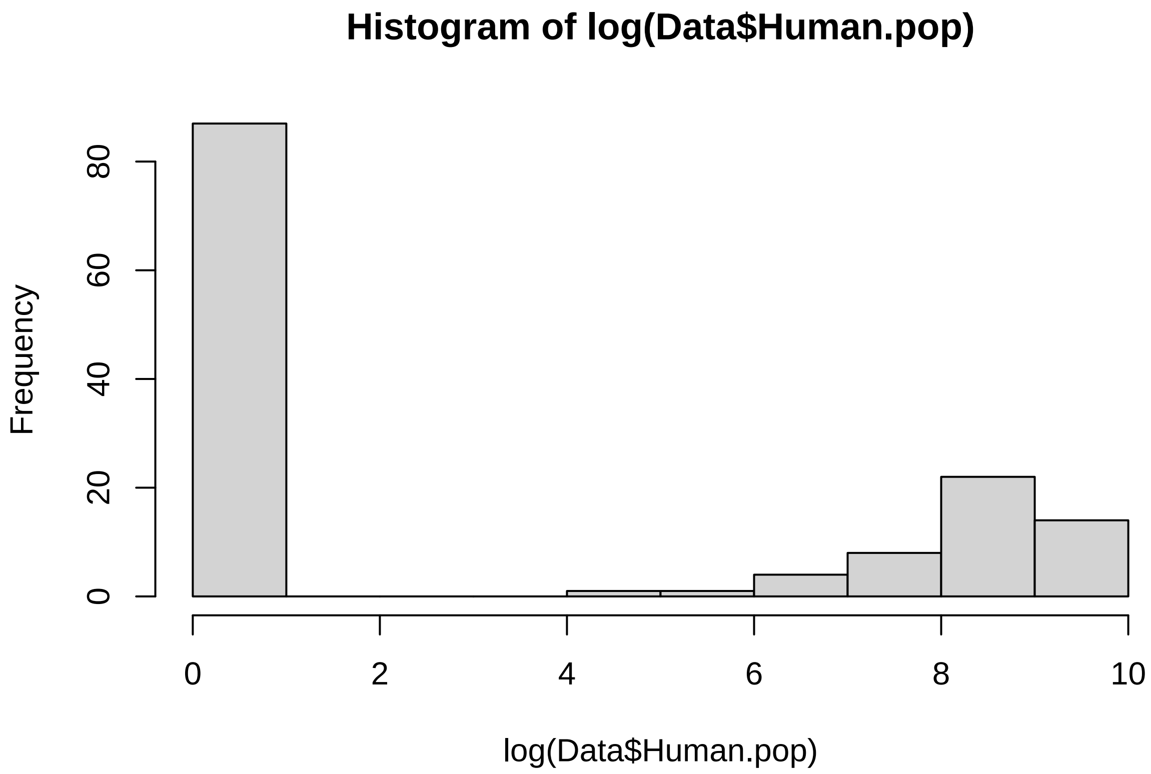
```
w <- 1 # constant weight
Data$Human.pop <- Data$Human.pop + w
head(Data)
```

```
##    NR  Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1 269 21345    44.89  344 35.5   72 13275  13189       252
## 2 260 20170    41.41  219  5.5   19 12273  14575      9692
## 3 260 10590    42.86   51 12.5   63  4941  11952      2061
## 4 262 18134    43.16  142 25.4   38  9332  12397         1
## 5 257 25565    42.57   14 31.6   37  8565  14683      3989
## 6 263 21985    44.89   73 14.9   73 10066  13255         1
```

Diagnostics:

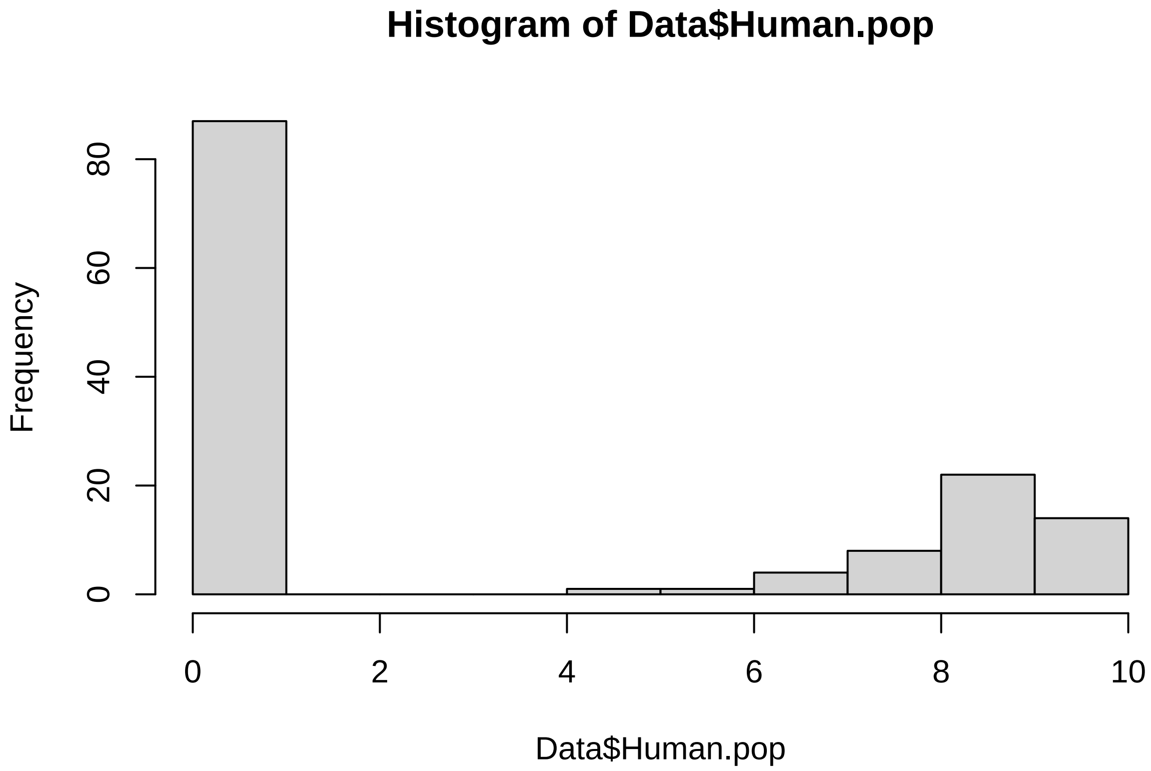
Human.pop is skewed to the right. Taking log may make it more symmetric. Let us look at the distribution of log(Human.pop)

```
hist(log(Data$Human.pop))
```



This does not make it symmetric. So, let us try to transform Human.pop with log(1+x).

```r
Data$Human.pop <- log1p(Data$Human.pop)
hist(Data$Human.pop)
```

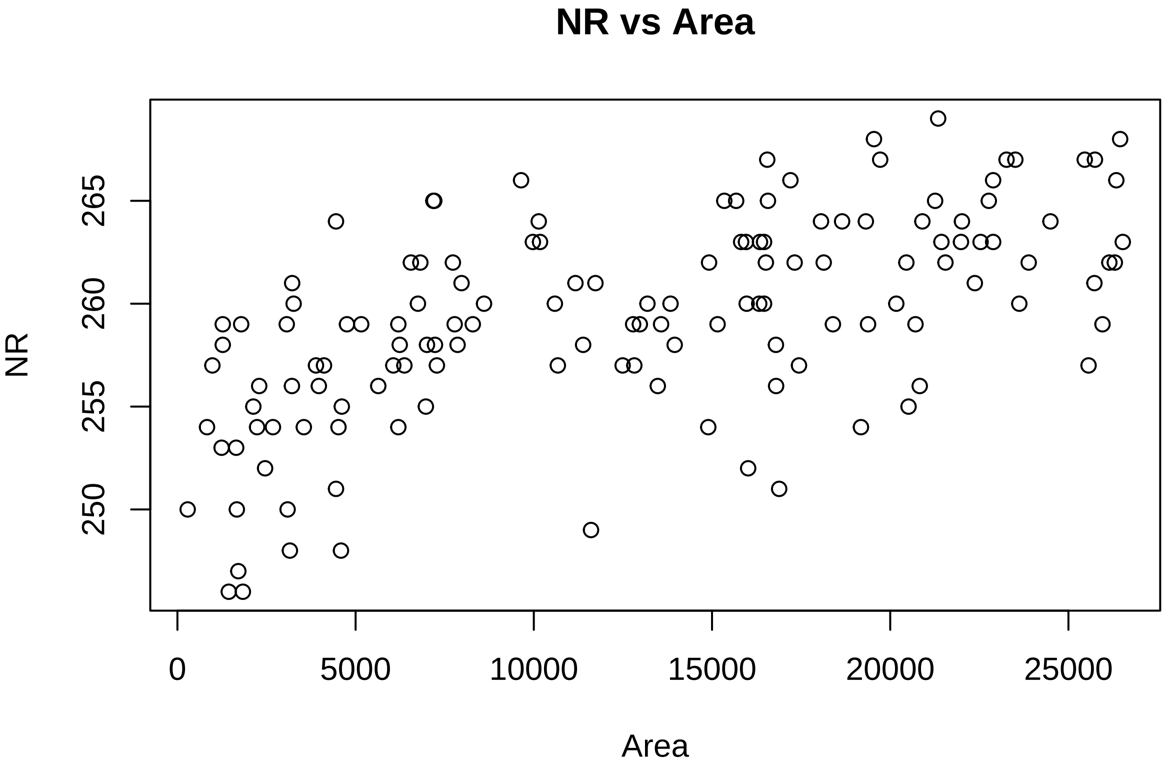


```
summary(Data$Human.pop)
```

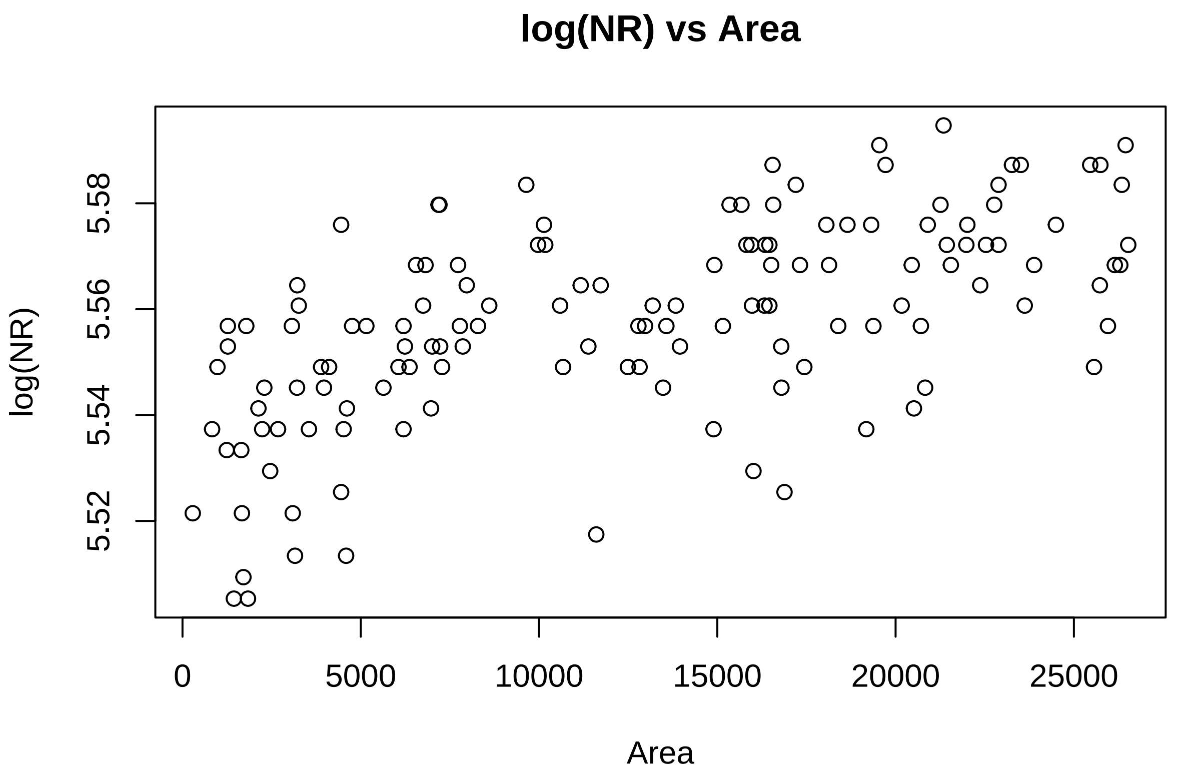
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.6931  0.6931  0.6931  3.4682  8.1654  9.2777
```

This seems like it worked better. So let us transform the Human.pop using log(1+x).

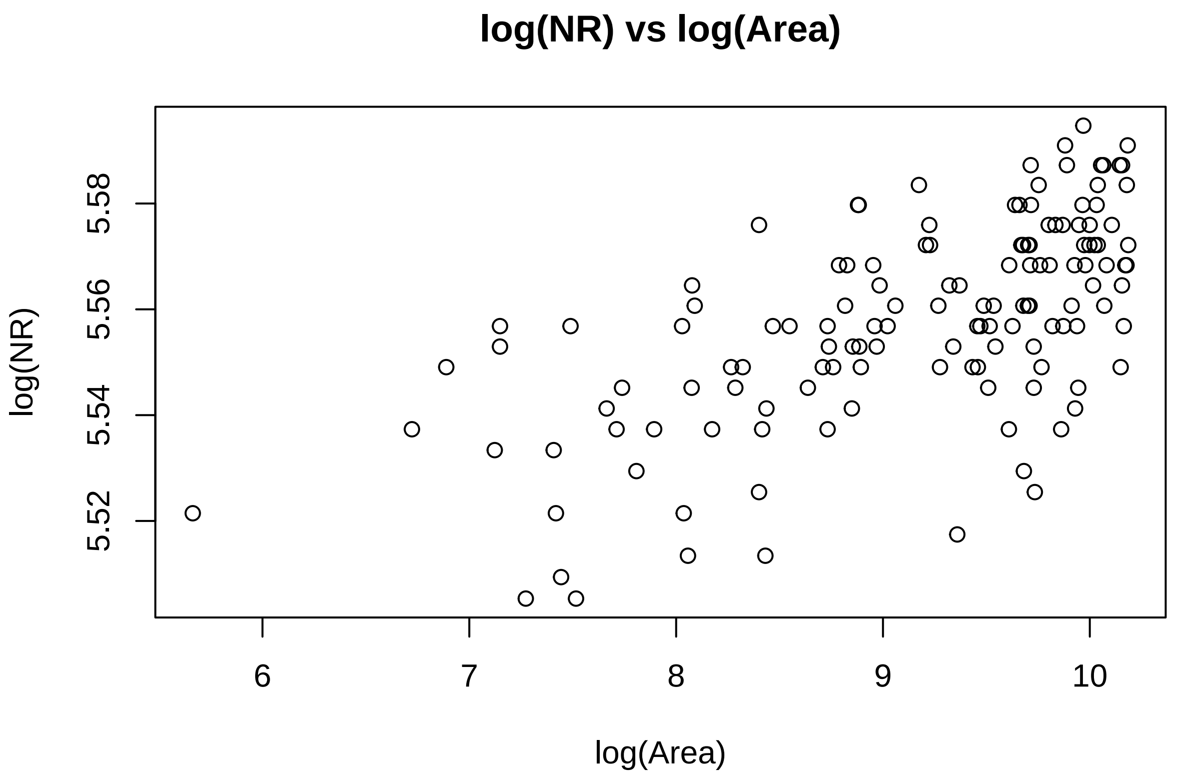
```
# Compare relationships of NR with Area because it has more correlation with NR



plot(NR~Area,Data, main = "NR vs Area")
```



```
plot(log(NR)~Area,Data, main = "log(NR) vs Area")
```



```
plot(log(NR)~log(Area),Data, main = "log(NR) vs log(Area)")
```



Modelling:

Full Model:

```
model <- lm(NR~Area + Latitude + Elev + Dist + Soil + Years + Deglac + Human.pop, data = Data)
summary(model)
```

```
##
## Call:
## lm(formula = NR ~ Area + Latitude + Elev + Dist + Soil + Years +
##     Deglac + Human.pop, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7164 -1.3241  0.3865  1.6539  4.0977
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.378e+02  7.879e+00  30.180   <2e-16 ***
## Area         3.297e-04  2.651e-05  12.435   <2e-16 ***
## Latitude     2.202e-01  1.755e-01   1.255    0.212
## Elev         1.634e-02  1.553e-03  10.517   <2e-16 ***
## Dist        -4.479e-03  1.547e-02  -0.290    0.773
## Soil         1.074e-01  9.265e-03  11.588   <2e-16 ***
## Years        2.847e-05  6.706e-05   0.425    0.672
## Deglac       1.151e-05  2.036e-04   0.057    0.955
## Human.pop    4.225e-02  5.348e-02   0.790    0.431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.264 on 128 degrees of freedom
## Multiple R-squared:  0.8037, Adjusted R-squared:  0.7915
## F-statistic: 65.52 on 8 and 128 DF,  p-value: < 2.2e-16
```

From this, we can say that Latitude, Dist, Years, Deglac and Human.pop are not significant due to their high p-value.

```
model1 <- lm(NR ~ Area + Elev + Soil, data = Data)
summary(model1)
```

```
##
## Call:
## lm(formula = NR ~ Area + Elev + Soil, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2943 -1.1000  0.4222  1.3946  3.7124
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.477e+02  5.471e-01  452.68   <2e-16 ***
## Area        3.352e-04  2.519e-05   13.31   <2e-16 ***
## Elev        1.607e-02  1.436e-03   11.19   <2e-16 ***
## Soil        1.081e-01  9.098e-03   11.88   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.245 on 133 degrees of freedom
## Multiple R-squared:  0.7995, Adjusted R-squared:  0.7949
## F-statistic: 176.7 on 3 and 133 DF,  p-value: < 2.2e-16
```

Let us include Human.pop

```
model2 <- lm(NR ~ Area + Elev + Soil + Human.pop, data = Data)
summary(model2)
```

```
##
## Call:
## lm(formula = NR ~ Area + Elev + Soil + Human.pop, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1749 -1.0179  0.3228  1.5240  3.8235
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.475e+02  5.645e-01 438.525   <2e-16 ***
## Area        3.318e-04  2.548e-05  13.019   <2e-16 ***
## Elev        1.614e-02  1.439e-03  11.216   <2e-16 ***
## Soil        1.078e-01  9.112e-03  11.829   <2e-16 ***
## Human.pop   4.746e-02  5.226e-02   0.908    0.366
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.247 on 132 degrees of freedom
## Multiple R-squared:  0.8007, Adjusted R-squared:  0.7947
## F-statistic: 132.6 on 4 and 132 DF,  p-value: < 2.2e-16
```

Let us try transforming Area with log.

```
model3 <- lm(NR ~ log(Area) + Elev + Soil + Human.pop, data = Data)
summary(model3)
```

```
##
## Call:
## lm(formula = NR ~ log(Area) + Elev + Soil + Human.pop, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9810 -0.9447  0.2965  1.3143  3.7564
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.244e+02  1.778e+00 126.199   <2e-16 ***
## log(Area)   2.994e+00  1.949e-01  15.363   <2e-16 ***
## Elev        1.689e-02  1.287e-03  13.129   <2e-16 ***
## Soil        1.056e-01  8.230e-03  12.832   <2e-16 ***
## Human.pop   1.745e-02  4.755e-02   0.367    0.714
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.034 on 132 degrees of freedom
## Multiple R-squared:  0.8367, Adjusted R-squared:  0.8318
## F-statistic: 169.1 on 4 and 132 DF,  p-value: < 2.2e-16
```

Let us try the log transformation on the full model with every predictor variables

```r
model4 <- lm(log(NR) ~ log(Area) + log(Latitude) + log(Elev) + log(Dist) + log(Soil) +log(Years) + log(
summary(model4)
```

```
##
## Call:
## lm(formula = log(NR) ~ log(Area) + log(Latitude) + log(Elev) +
##     log(Dist) + log(Soil) + log(Years) + log(Deglac) + log(Human.pop),
##     data = Data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0154854 -0.0026918  0.0005945  0.0026189  0.0096715
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     5.3422391  0.0694958  76.871   <2e-16 ***
## log(Area)       0.0119221  0.0004174  28.564   <2e-16 ***
## log(Latitude)   0.0063357  0.0140262   0.452    0.652
## log(Elev)       0.0118431  0.0004279  27.680   <2e-16 ***
## log(Dist)       0.0005139  0.0003715   1.383    0.169
## log(Soil)       0.0118182  0.0004145  28.510   <2e-16 ***
## log(Years)     -0.0001910  0.0010072  -0.190    0.850
## log(Deglac)    -0.0016847  0.0050899  -0.331    0.741
## log(Human.pop) -0.0003409  0.0003153  -1.081    0.282
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.004246 on 128 degrees of freedom
## Multiple R-squared:  0.9541, Adjusted R-squared:  0.9512
## F-statistic: 332.3 on 8 and 128 DF,  p-value: < 2.2e-16
```

Let us try some interactions

```r
model5 <- lm(NR ~ Area * Elev + Soil + Human.pop, data = Data)
summary(model5)
```

```
##
## Call:
## lm(formula = NR ~ Area * Elev + Soil + Human.pop, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1966 -1.1109  0.3786  1.4709  3.9196
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.466e+02  7.151e-01 344.907  < 2e-16 ***
## Area         4.167e-04  4.883e-05   8.535 3.04e-14 ***
## Elev         2.080e-02  2.700e-03   7.703 2.89e-12 ***
## Soil         1.063e-01  9.035e-03  11.763  < 2e-16 ***
## Human.pop    3.444e-02  5.205e-02   0.662   0.5093
## Area:Elev   -3.597e-07  1.771e-07  -2.031   0.0442 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.221 on 131 degrees of freedom
## Multiple R-squared:  0.8068, Adjusted R-squared:  0.7994
## F-statistic: 109.4 on 5 and 131 DF,  p-value: < 2.2e-16
```

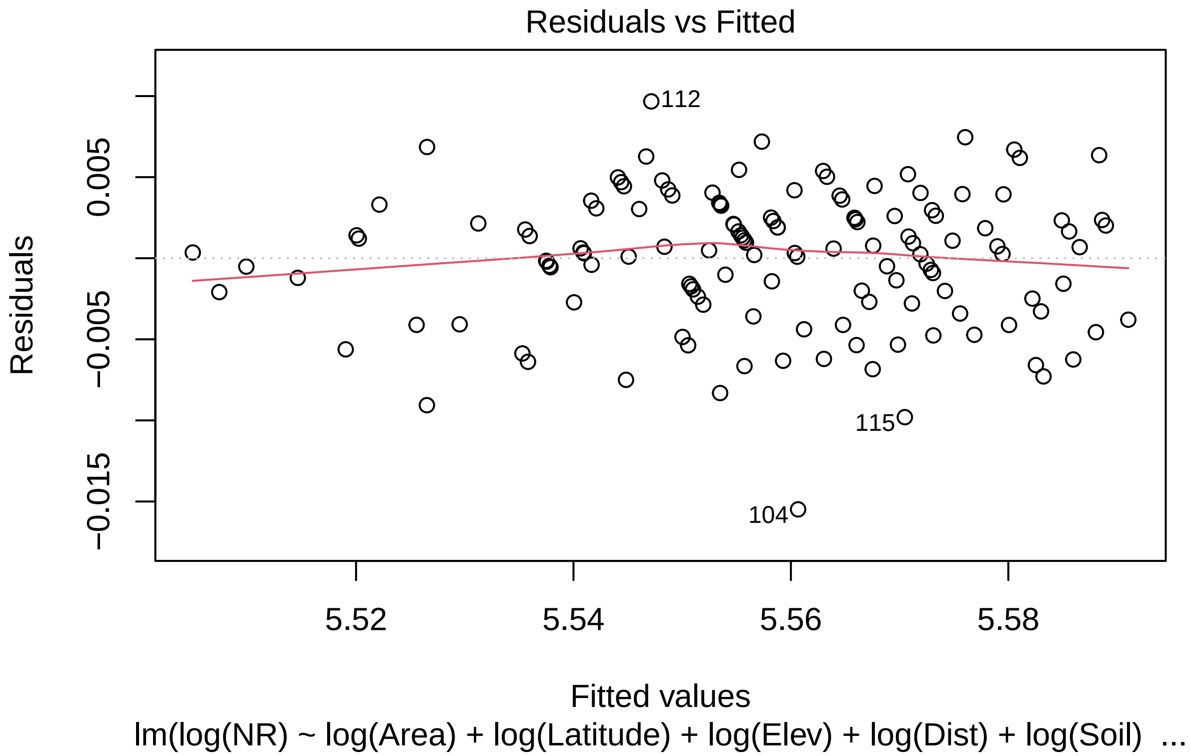
```
r_squared_model1 <- summary(model1)$r.squared
r_squared_model2 <- summary(model2)$r.squared
r_squared_model3 <- summary(model3)$r.squared
r_squared_model4 <- summary(model4)$r.squared
r_squared_model5 <- summary(model5)$r.squared

cat("R-Squared values\nModel1: ",r_squared_model1,"\nModel2: ",r_squared_model2,
    "\nModel3: ",r_squared_model3,"\nModel4: ",r_squared_model4,"\nModel5: ",r_squared_model5)
```

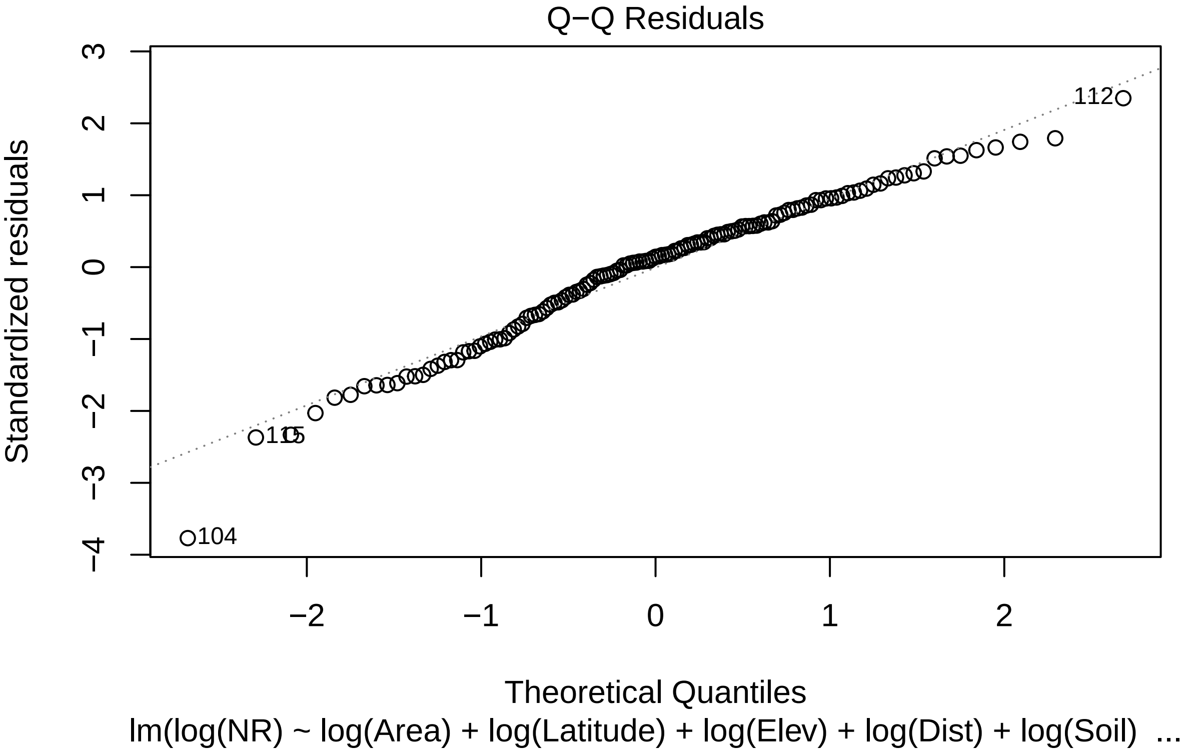
```
## R-Squared values
## Model1:  0.799456
## Model2:  0.800701
## Model3:  0.8367242
## Model4:  0.9540634
## Model5:  0.8067878
```

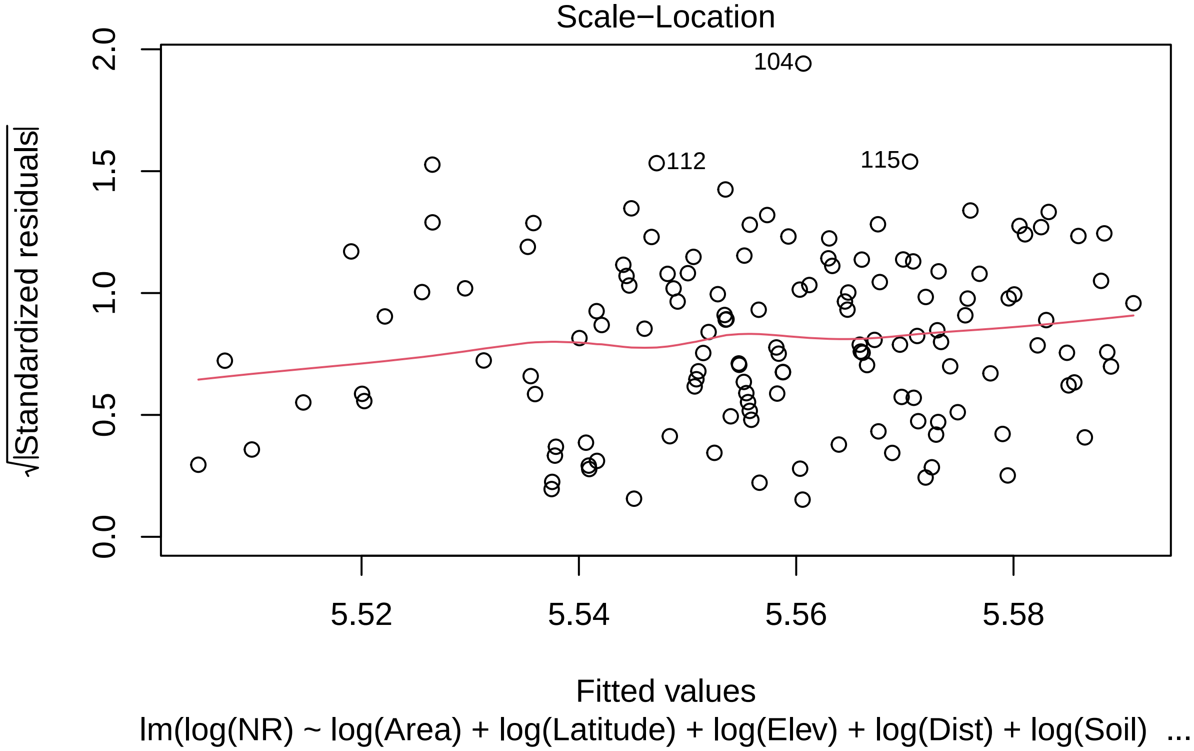
from all these models, our best model based on R2 value is model4 which is model4 <- lm(log(NR) ~ log(Area) + log(Latitude) + log(Elev) + log(Dist) + log(Soil) +log(Years) + log(Deglac) + log(Human.pop), data = Data)

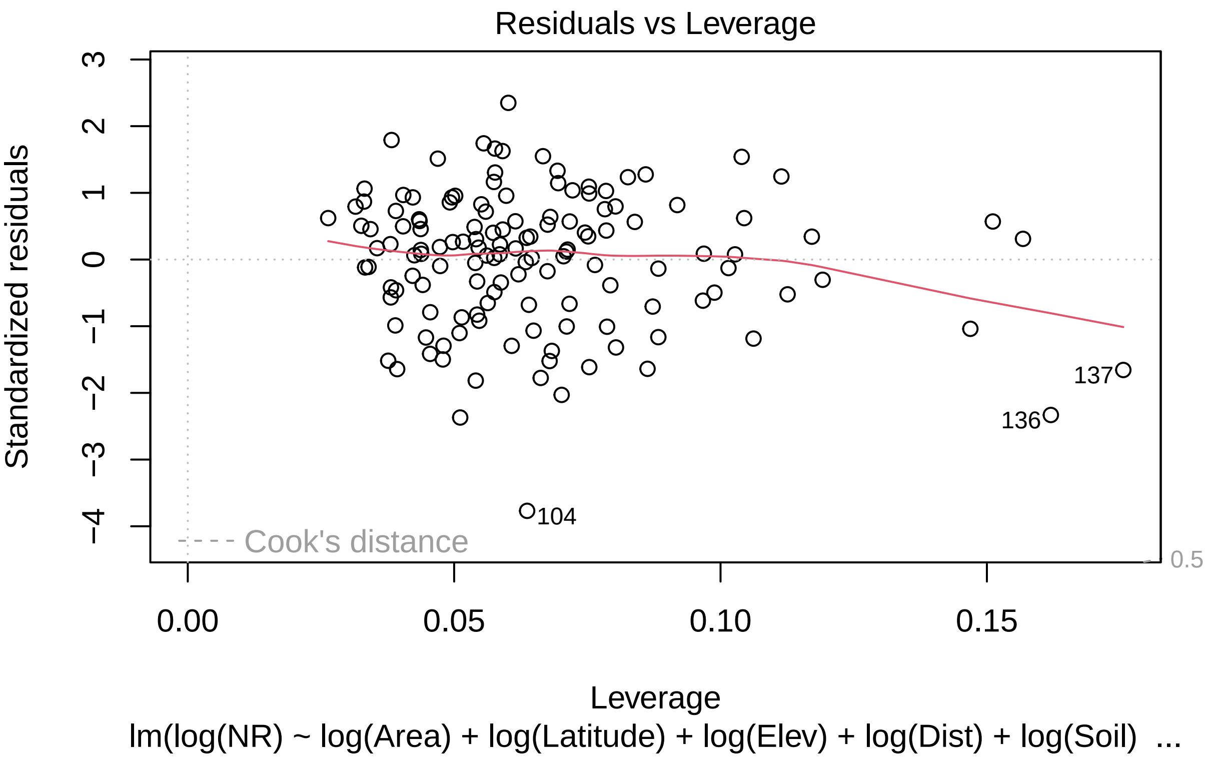
So we can say that this is our best model.

```
plot(model4)
```




Q−Q Residuals
Standardized residuals
3
2
1
0
−1
−2
−3
−4
112
115
104
−2
−1
0
1
2
Theoretical Quantiles
lm(log(NR) ~ log(Area) + log(Latitude) + log(Elev) + log(Dist) + log(Soil) ...


Scale−Location
√|Standardized residuals|
2.0
1.5
1.0
0.5
0.0
104
112
115
5.52
5.54
5.56
5.58
Fitted values
lm(log(NR) ~ log(Area) + log(Latitude) + log(Elev) + log(Dist) + log(Soil) ...



Let us try some other transformations

```
# polynomial transformation

model6 <- lm(NR ~ Area + Latitude + Elev + poly(Dist,2) + Soil + Years + Deglac + Human.pop, data = Dat
summary(model6)
```

```
##
## Call:
## lm(formula = NR ~ Area + Latitude + Elev + poly(Dist, 2) + Soil +
##     Years + Deglac + Human.pop, data = Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.3921 -1.3061  0.2181  1.5233  4.1151
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.387e+02  7.734e+00  30.859   <2e-16 ***
## Area            3.255e-04  2.605e-05  12.493   <2e-16 ***
## Latitude        1.962e-01  1.723e-01   1.138   0.2571
## Elev            1.658e-02  1.526e-03  10.864   <2e-16 ***
## poly(Dist, 2)1 -8.300e-01  2.306e+00  -0.360   0.7195
## poly(Dist, 2)2  5.591e+00  2.259e+00   2.475   0.0146 *
## Soil            1.058e-01  9.106e-03  11.621   <2e-16 ***
## Years           4.418e-05  6.607e-05   0.669   0.5049
## Deglac          7.084e-06  1.996e-04   0.035   0.9717
## Human.pop       5.299e-02  5.262e-02   1.007   0.3159
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.22 on 127 degrees of freedom
## Multiple R-squared:  0.8128, Adjusted R-squared:  0.7995
## F-statistic: 61.25 on 9 and 127 DF,  p-value: < 2.2e-16
```

```
#Square Root Transformation

model7 <- lm(NR ~ Area + Latitude + Elev + Dist + sqrt(Soil) + Years + Deglac + Human.pop, data = Data)
summary(model7)
```

```
##
## Call:
## lm(formula = NR ~ Area + Latitude + Elev + Dist + sqrt(Soil) +
##     Years + Deglac + Human.pop, data = Data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.664 -1.208  0.282  1.637  3.667
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.344e+02  7.180e+00  32.649   <2e-16 ***
## Area         3.324e-04  2.419e-05  13.744   <2e-16 ***
## Latitude     2.380e-01  1.599e-01   1.488    0.139
## Elev         1.677e-02  1.413e-03  11.870   <2e-16 ***
## Dist        -2.551e-03  1.411e-02  -0.181    0.857
## sqrt(Soil)   1.240e+00  9.064e-02  13.683   <2e-16 ***
## Years        1.268e-05  6.105e-05   0.208    0.836
## Deglac      -2.832e-05  1.858e-04  -0.152    0.879
## Human.pop    2.625e-02  4.883e-02   0.537    0.592
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.065 on 128 degrees of freedom
## Multiple R-squared:  0.8367, Adjusted R-squared:  0.8265
## F-statistic: 81.98 on 8 and 128 DF,  p-value: < 2.2e-16
```

```
# Cube Root transformation

model8 <- lm(NR ~ Area + Latitude + I(Elev^(1/3)) + Dist + Soil + Years + Deglac + Human.pop, data = Da
summary(model8)
```

```
##
## Call:
## lm(formula = NR ~ Area + Latitude + I(Elev^(1/3)) + Dist + Soil +
##     Years + Deglac + Human.pop, data = Data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.708 -1.125  0.278  1.489  3.468
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.329e+02  7.189e+00  32.399   <2e-16 ***
## Area          3.296e-04  2.386e-05  13.813   <2e-16 ***
## Latitude      1.956e-01  1.581e-01   1.237    0.218
## I(Elev^(1/3)) 1.663e+00  1.302e-01  12.768   <2e-16 ***
## Dist          2.985e-04  1.396e-02   0.021    0.983    
## Soil          1.104e-01  8.362e-03  13.199   <2e-16 ***
## Years         2.340e-05  6.039e-05   0.388    0.699    
## Deglac        4.150e-06  1.843e-04   0.023    0.982    
## Human.pop     3.909e-02  4.839e-02   0.808    0.421    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.05 on 128 degrees of freedom
## Multiple R-squared:  0.8391, Adjusted R-squared:  0.829 
## F-statistic: 83.42 on 8 and 128 DF,  p-value: < 2.2e-16
```

```r
# sine transformation

model9 <- lm(NR ~ Area + Latitude + Elev + Dist + Soil + sin(Years) + Deglac + Human.pop, data = Data)
summary(model9)
```

```
## 
## Call:
## lm(formula = NR ~ Area + Latitude + Elev + Dist + Soil + sin(Years) + 
##     Deglac + Human.pop, data = Data)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -6.750 -1.223  0.272  1.555  4.252 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  2.374e+02  7.873e+00  30.160   <2e-16 ***
## Area         3.297e-04  2.648e-05  12.449   <2e-16 ***
## Latitude     2.320e-01  1.745e-01   1.329    0.186    
## Elev         1.660e-02  1.488e-03  11.152   <2e-16 ***
## Dist        -4.792e-03  1.547e-02  -0.310    0.757    
## Soil         1.067e-01  9.231e-03  11.558   <2e-16 ***
## sin(Years)  -1.800e-01  2.696e-01  -0.667    0.506    
## Deglac       1.659e-05  2.034e-04   0.082    0.935    
## Human.pop    4.622e-02  5.337e-02   0.866    0.388    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.262 on 128 degrees of freedom
## Multiple R-squared:  0.8041, Adjusted R-squared:  0.7919 
## F-statistic: 65.69 on 8 and 128 DF,  p-value: < 2.2e-16
```

Model Selection:

Train Test Split:

```r
# Create an index for the training. and testing sets

split_index <- sample(1:nrow(Data), nrow(Data)*0.7) # Adjust the split ratio as needed

# Create training and testing datasets

train_data <- Data[split_index,]
test_data <- Data[-split_index,]
```

```
# Backwards Selection

direction = "backward"

bsel <- step(model, trace = 0) # By AIC
formula(bsel)
```

```
## NR ~ Area + Elev + Soil
```

```
# NR ~ Area + Elev + Soil

Criteria <- function(model){
  out <- data.frame(`p+1`=length(model$coef),
                    R2adj = summary(model)$adj,
                    AIC = AIC(model),
                    BIC = BIC(model))
  return(out)
}

rbind(
  bsel = Criteria(bsel)
)
```

```
##      p.1     R2adj     AIC      BIC
## bsel   4 0.7949325 616.355 630.9549
```

All Subsets Regression model

```
library(leaps)
model_sub <- regsubsets(NR ~ Area + Latitude + Elev + Dist + Soil + Years + Deglac + Human.pop, nbest =
summary(model_sub)
```

```
## Subset selection object
## Call: regsubsets.formula(NR ~ Area + Latitude + Elev + Dist + Soil +
##     Years + Deglac + Human.pop, nbest = 3, really.big = TRUE,
##     nvmax = 10, data = train_data)
## 8 Variables  (and intercept)
##           Forced in Forced out
## Area          FALSE      FALSE
## Latitude      FALSE      FALSE
## Elev          FALSE      FALSE
## Dist          FALSE      FALSE
## Soil          FALSE      FALSE
## Years         FALSE      FALSE
## Deglac        FALSE      FALSE
## Human.pop     FALSE      FALSE
## 3 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1  ( 1 ) "*"  " "      " "  " "  " "  " "   " "    " "
## 1  ( 2 ) " "  " "      "*"  " "  " "  " "   " "    " "
## 1  ( 3 ) " "  " "      " "  " "  "*"  " "   " "    " "
## 2  ( 1 ) "*"  " "      " "  " "  "*"  " "   " "    " "
## 2  ( 2 ) "*"  " "      "*"  " "  " "  " "   " "    " "
## 2  ( 3 ) " "  " "      "*"  " "  "*"  " "   " "    " "
## 3  ( 1 ) "*"  " "      "*"  " "  "*"  " "   " "    " "
## 3  ( 2 ) "*"  " "      " "  " "  "*"  "*"   " "    " "
## 3  ( 3 ) "*"  " "      " "  "*"  "*"  " "   " "    " "
## 4  ( 1 ) "*"  " "      "*"  " "  "*"  " "   " "    "*"
## 4  ( 2 ) "*"  "*"      "*"  " "  "*"  " "   " "    " "
## 4  ( 3 ) "*"  " "      "*"  " "  "*"  " "   "*"    " "
## 5  ( 1 ) "*"  "*"      "*"  " "  "*"  " "   " "    "*"
## 5  ( 2 ) "*"  " "      "*"  " "  "*"  "*"   " "    "*"
## 5  ( 3 ) "*"  " "      "*"  " "  "*"  " "   "*"    "*"
## 6  ( 1 ) "*"  "*"      "*"  " "  "*"  "*"   " "    "*"
## 6  ( 2 ) "*"  "*"      "*"  " "  "*"  " "   "*"    "*"
## 6  ( 3 ) "*"  "*"      "*"  "*"  "*"  " "   " "    "*"
## 7  ( 1 ) "*"  "*"      "*"  " "  "*"  "*"   "*"    "*"
## 7  ( 2 ) "*"  "*"      "*"  "*"  "*"  "*"   " "    "*"
## 7  ( 3 ) "*"  "*"      "*"  "*"  "*"  " "   "*"    "*"
## 8  ( 1 ) "*"  "*"      "*"  "*"  "*"  "*"   "*"    "*"
```

Check the results

```
sbest <- summary(model_sub)

names(sbest)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
cbind(sbest$which, sbest$cp)
```

```
##   (Intercept) Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1           1    1        0    0    0    0     0      0         0 168.234939
## 1           1    0        0    1    0    0     0      0         0 193.586418
## 1           1    0        0    0    0    1     0      0         0 279.388975
## 2           1    1        0    0    0    1     0      0         0  59.753307
## 2           1    1        0    1    0    0     0      0         0  99.919349
## 2           1    0        0    1    0    1     0      0         0 120.646877
## 3           1    1        0    1    0    1     0      0         0   2.171879
## 3           1    1        0    0    0    1     1      0         0  58.196483
## 3           1    1        0    0    1    1     0      0         0  58.889447
## 4           1    1        0    1    0    1     0      0         1   2.365048
## 4           1    1        1    1    0    1     0      0         0   3.411931
## 4           1    1        0    1    0    1     0      1         0   3.937158
## 5           1    1        1    1    0    1     0      0         1   3.622172
## 5           1    1        0    1    0    1     1      0         1   4.112016
## 5           1    1        0    1    0    1     0      1         1   4.200352
## 6           1    1        1    1    0    1     1      0         1   5.238693
## 6           1    1        1    1    0    1     0      1         1   5.436518
## 6           1    1        1    1    1    1     0      0         1   5.588650
## 7           1    1        1    1    0    1     1      1         1   7.044081
## 7           1    1        1    1    1    1     1      0         1   7.212405
## 7           1    1        1    1    1    1     0      1         1   7.383547
## 8           1    1        1    1    1    1     1      1         1   9.000000
```

```
cbind(sbest$which, sbest$adjr2)
```

```
##   (Intercept) Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1           1    1        0    0    0    0     0      0         0 0.4172641
## 1           1    0        0    1    0    0     0      0         0 0.3602763
## 1           1    0        0    0    0    1     0      0         0 0.1674002
## 2           1    1        0    0    0    1     0      0         0 0.6619819
## 2           1    1        0    1    0    0     0      0         0 0.5707109
## 2           1    0        0    1    0    1     0      0         0 0.5236109
## 3           1    1        0    1    0    1     0      0         0 0.7951445
## 3           1    1        0    0    0    1     1      0         0 0.6664385
## 3           1    1        0    0    1    1     0      0         0 0.6648466
## 4           1    1        0    1    0    1     0      0         1 0.7970653
## 4           1    1        1    1    0    1     0      0         0 0.7946335
## 4           1    1        0    1    0    1     0      1         0 0.7934135
## 5           1    1        1    1    0    1     0      0         1 0.7965301
## 5           1    1        0    1    0    1     1      0         1 0.7953794
## 5           1    1        0    1    0    1     0      1         1 0.7951719
## 6           1    1        1    1    0    1     1      0         1 0.7951289
## 6           1    1        1    1    0    1     0      1         1 0.7946589
## 6           1    1        1    1    1    1     0      0         1 0.7942975
## 7           1    1        1    1    0    1     1      1         1 0.7932417
## 7           1    1        1    1    1    1     1      0         1 0.7928372
## 7           1    1        1    1    1    1     0      1         1 0.7924260
## 8           1    1        1    1    1    1     1      1         1 0.7909447
```

```
cbind(sbest$which, sbest$bic)
```

```
##   (Intercept) Area Latitude Elev Dist Soil Years Deglac Human.pop
## 1           1    1        0    0    0    0     0      0         0  -43.210309
## 1           1    0        0    1    0    0     0      0         0  -34.346599
## 1           1    0        0    0    0    1     0      0         0   -9.312505
## 2           1    1        0    0    0    1     0      0         0  -91.423752
## 2           1    1        0    1    0    0     0      0         0  -68.715815
## 2           1    0        0    1    0    1     0      0         0  -58.825899
## 3           1    1        0    1    0    1     0      0         0 -135.483612
## 3           1    1        0    0    0    1     1      0         0  -89.169008
## 3           1    1        0    0    1    1     0      0         0  -88.716690
## 4           1    1        0    1    0    1     0      0         1 -132.874427
## 4           1    1        1    1    0    1     0      0         0 -131.742822
## 4           1    1        0    1    0    1     0      1         0 -131.180127
## 5           1    1        1    1    0    1     0      0         1 -129.131801
## 5           1    1        0    1    0    1     1      0         1 -128.596094
## 5           1    1        0    1    0    1     0      1         1 -128.499808
## 6           1    1        1    1    0    1     1      0         1 -124.999425
## 6           1    1        1    1    0    1     0      1         1 -124.781752
## 6           1    1        1    1    1    1     0      0         1 -124.614696
## 7           1    1        1    1    0    1     1      1         1 -120.660175
## 7           1    1        1    1    1    1     1      0         1 -120.474512
## 7           1    1        1    1    1    1     0      1         1 -120.286113
## 8           1    1        1    1    1    1     1      1         1 -116.154979
```

```
cbind(sbest$which, sbest$aic)
```

```
##   (Intercept)  Area Latitude  Elev  Dist  Soil Years Deglac Human.pop
## 1        TRUE  TRUE    FALSE FALSE FALSE FALSE FALSE  FALSE     FALSE
## 1        TRUE FALSE    FALSE  TRUE FALSE FALSE FALSE  FALSE     FALSE
## 1        TRUE FALSE    FALSE FALSE FALSE  TRUE FALSE  FALSE     FALSE
## 2        TRUE  TRUE    FALSE FALSE FALSE  TRUE FALSE  FALSE     FALSE
## 2        TRUE  TRUE    FALSE  TRUE FALSE FALSE FALSE  FALSE     FALSE
## 2        TRUE FALSE    FALSE  TRUE FALSE  TRUE FALSE  FALSE     FALSE
## 3        TRUE  TRUE    FALSE  TRUE FALSE  TRUE FALSE  FALSE     FALSE
## 3        TRUE  TRUE    FALSE FALSE FALSE  TRUE  TRUE  FALSE     FALSE
## 3        TRUE  TRUE    FALSE FALSE  TRUE  TRUE FALSE  FALSE     FALSE
## 4        TRUE  TRUE    FALSE  TRUE FALSE  TRUE FALSE  FALSE      TRUE
## 4        TRUE  TRUE     TRUE  TRUE FALSE  TRUE FALSE  FALSE     FALSE
## 4        TRUE  TRUE    FALSE  TRUE FALSE  TRUE FALSE   TRUE     FALSE
## 5        TRUE  TRUE     TRUE  TRUE FALSE  TRUE FALSE  FALSE      TRUE
## 5        TRUE  TRUE    FALSE  TRUE FALSE  TRUE  TRUE  FALSE      TRUE
## 5        TRUE  TRUE    FALSE  TRUE FALSE  TRUE FALSE   TRUE      TRUE
## 6        TRUE  TRUE     TRUE  TRUE FALSE  TRUE  TRUE  FALSE      TRUE
## 6        TRUE  TRUE     TRUE  TRUE FALSE  TRUE FALSE   TRUE      TRUE
## 6        TRUE  TRUE     TRUE  TRUE  TRUE  TRUE FALSE  FALSE      TRUE
## 7        TRUE  TRUE     TRUE  TRUE FALSE  TRUE  TRUE   TRUE      TRUE
## 7        TRUE  TRUE     TRUE  TRUE  TRUE  TRUE  TRUE  FALSE      TRUE
## 7        TRUE  TRUE     TRUE  TRUE  TRUE  TRUE FALSE   TRUE      TRUE
## 8        TRUE  TRUE     TRUE  TRUE  TRUE  TRUE  TRUE   TRUE      TRUE
```

Best model aic, bic,Cp

```
mybestmodel <- function(Xnames, Yname, dataset, p, crit = "bic"){
  if(crit == "Cp"){
    n <- dim(dataset)[1]

    fullMSE = summary(lm(as.formula(paste(Yname,"~.")), data = dataset))$sigma^2
  }
  varsel <- lapply(0:p, function(x) combn(p,x))

  modcrit <- numeric(p);
  form <- character(p)

  for(k in 1:p){
    s <- dim(varsel[[k+1]])[2]
    tempform <- character(s); tempcrit <- numeric(s)

    for(j in 1:s){
      temp <- Xnames[varsel[[k+1]][,j]]
      tempform[j] <- ifelse(length(temp)>1,
                            paste(temp, collapse = "+"), temp)
      tempform[j] <- paste(Yname, tempform[j], sep = '~')
      tempmod <- lm(as.formula(tempform[j]), data = dataset)
      if (crit == "aic"){
        tempcrit[j] <- AIC(tempmod)
      }
      if (crit == "bic"){
        tempcrit[j] <- BIC(tempmod)
      }
      if (crit == "r2"){
        tempcrit[j] <- summary(tempmod)$adj
      }
      if(crit=="Cp"){
        tempcrit[j] <- sum(tempmod$res^2)/fullMSE+2*(k+1)-n
      }
    }
    # best model of size k
    if(crit %in% c("aic", "bic")){
      best <- which.min(tempcrit)
    }
    if(crit == "r2"){
      best <- which.max(tempcrit)
    }
    if(crit == "Cp"){
      best <- which.min(abs(tempcrit[j]-(k+1)))
    }
    form[k] <- tempform[best]
    modcrit[k] <- tempcrit[best]
  }
  if(crit %in% c("aic","bic")){
    out <- form[which.min(modcrit)]
  }
  if(crit == "r2"){
    out <- form[which.max(modcrit)]
  }
  if(crit == "Cp"){
    out <- form[which.min(abs(modcrit[-p]-(2:p)))]
  }
  return(out)
}

p <- length(names(train_data))-1
Xnames <- names(train_data)[-1]
Yname <- "NR"
dataset <- train_data
form <- mybestmodel(Xnames, Yname, train_data, p, crit = "bic")
form
```

```
## [1] "NR~Area+Elev+Soil"
```

```
bicform <- mybestmodel(Xnames, Yname, train_data, p, crit = "bic")
modbic <- lm(as.formula(bicform), data = train_data)
aicform <- mybestmodel(Xnames, Yname, train_data, p, crit = "aic")
modaic <- lm(as.formula(aicform), data = train_data)
cpform <- mybestmodel(Xnames, Yname, train_data, p, crit = "Cp")
modCp <- lm(as.formula(cpform), data = train_data)
r2form <- mybestmodel(Xnames, Yname, train_data, p, crit = "r2")
modr2 <- lm(as.formula(r2form), data = train_data)
```

Now finding the best model using backward stepwise selection:

```
modback <- step(lm(model, data = train_data), trace = 0, direction = "backward")
```

Display the adjR2, BIC, Cp, AIC and Bsel for all the models

```
rbind(bicm = Criteria(modbic),
      aicm = Criteria(modaic),
      adjr2m = Criteria(modr2),
      cpm = Criteria(modCp),
      bsel = Criteria(modback)
      )
```

```
##        p.1     R2adj      AIC      BIC
## bicm     4 0.7951445 440.5731 453.3425
## aicm     4 0.7951445 440.5731 453.3425
## adjr2m   5 0.7970653 440.6284 455.9517
## cpm      7 0.7905466 445.4971 465.9281
## bsel     4 0.7951445 440.5731 453.3425
```

Now use cross validation to select the final model.

```
cv.lm <- function(data, formulae, nfolds = 5) {
  data <- na.omit(data) # remove missing values
  formulae <- sapply(formulae, as.formula)
  n <- nrow(data)
  fold.labels <- sample(rep(1:nfolds, length.out = n))
  mses <- matrix(NA, nrow = nfolds, ncol = length(formulae))
  colnames <- as.character(formulae)
  for(fold in 1:nfolds) {
    test.rows <- which(fold.labels == fold)
    train <- data[-test.rows,]
    test <- data[test.rows,]
    for(form in 1:length(formulae)) {
      current.model <- lm(formula = formulae[[form]], data = train)
      predictions <- predict(current.model, newdata = test)
      test.responses <- eval(formulae[[form]][[2]], envir = test)
      test.errors <- test.responses - predictions
      mses[fold, form] <- mean(test.errors^2)
    }
  }
  return(colMeans(mses))
}

#set.seed(10)
formulae <- c(formula(modbic),
              formula(modaic),
              formula(modr2),
              formula(modCp),
              formula(modback))

mse <- cv.lm(data = train_data, formulae, nfolds = 5)
print(mse)
```

```
## [1] 5.932144 5.932144 5.881048 6.589575 5.932144
```

In this seed, models modaic, modbic and modback metrics has the same least error value. Therefore, we select the model modbic is the final model.

Error values in order: modCp > modr2 > modbic = modaic = modback

```
summary(modbic)
```

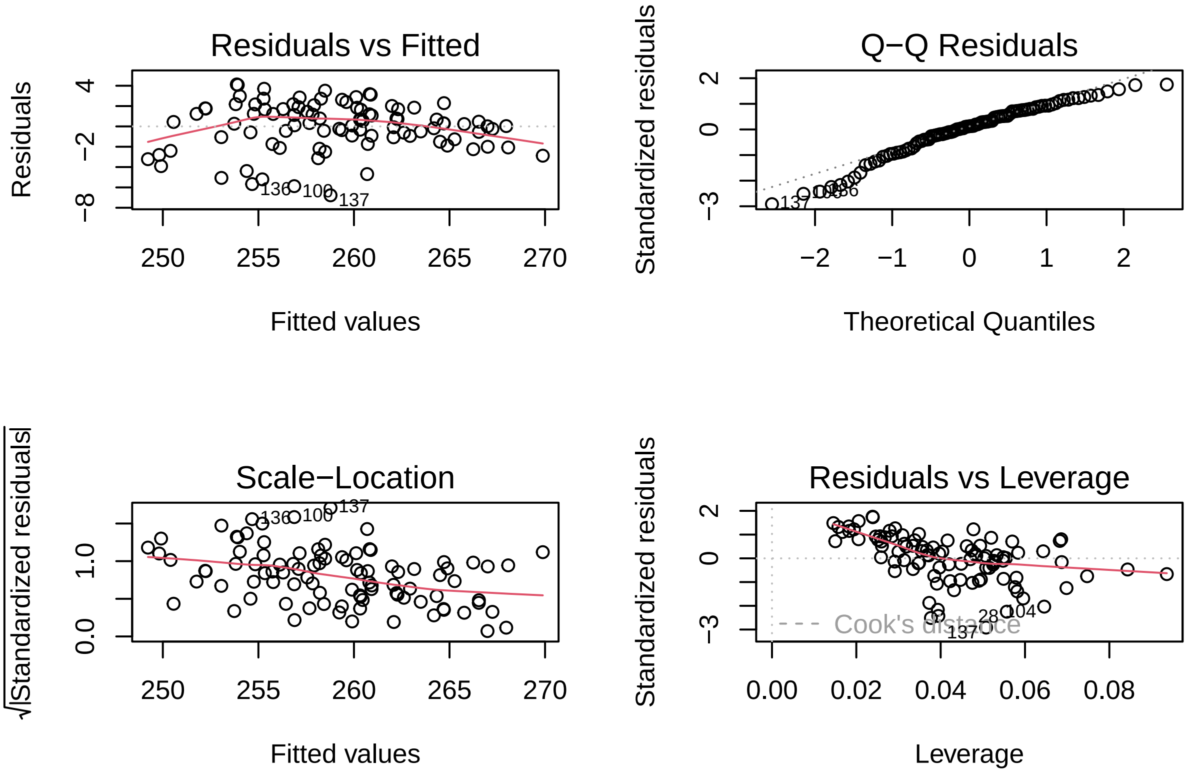
```
##
## Call:
## lm(formula = as.formula(bicform), data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7751 -1.1695  0.2922  1.7302  4.1282
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.471e+02  6.816e-01 362.551   <2e-16 ***
## Area        3.672e-04  3.312e-05  11.088   <2e-16 ***
## Elev        1.479e-02  1.897e-03   7.798    1e-11 ***
## Soil        1.128e-01  1.118e-02  10.089   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.384 on 91 degrees of freedom
## Multiple R-squared:  0.8017, Adjusted R-squared:  0.7951
## F-statistic: 122.6 on 3 and 91 DF,  p-value: < 2.2e-16
```

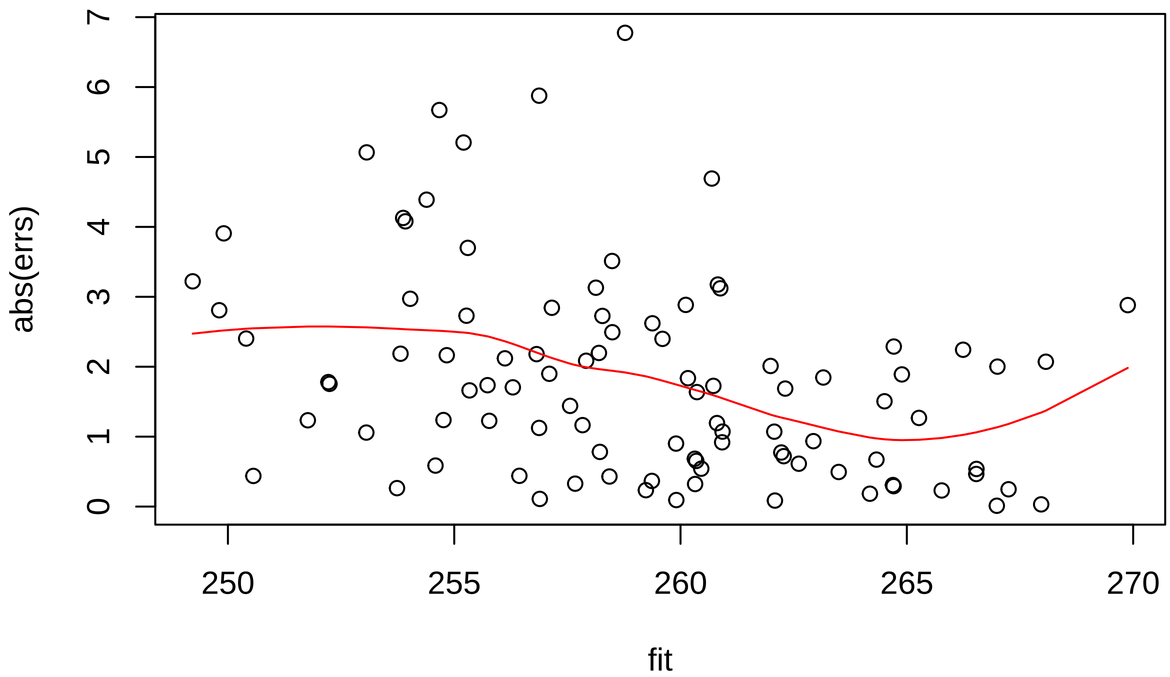
Weighted Regression

Now, lets perform weighted regression on modbic.

```
par(mfrow = c(2,2))
plot(modbic,1)
plot(modbic,2)
plot(modbic,3)
plot(modbic,5)
```



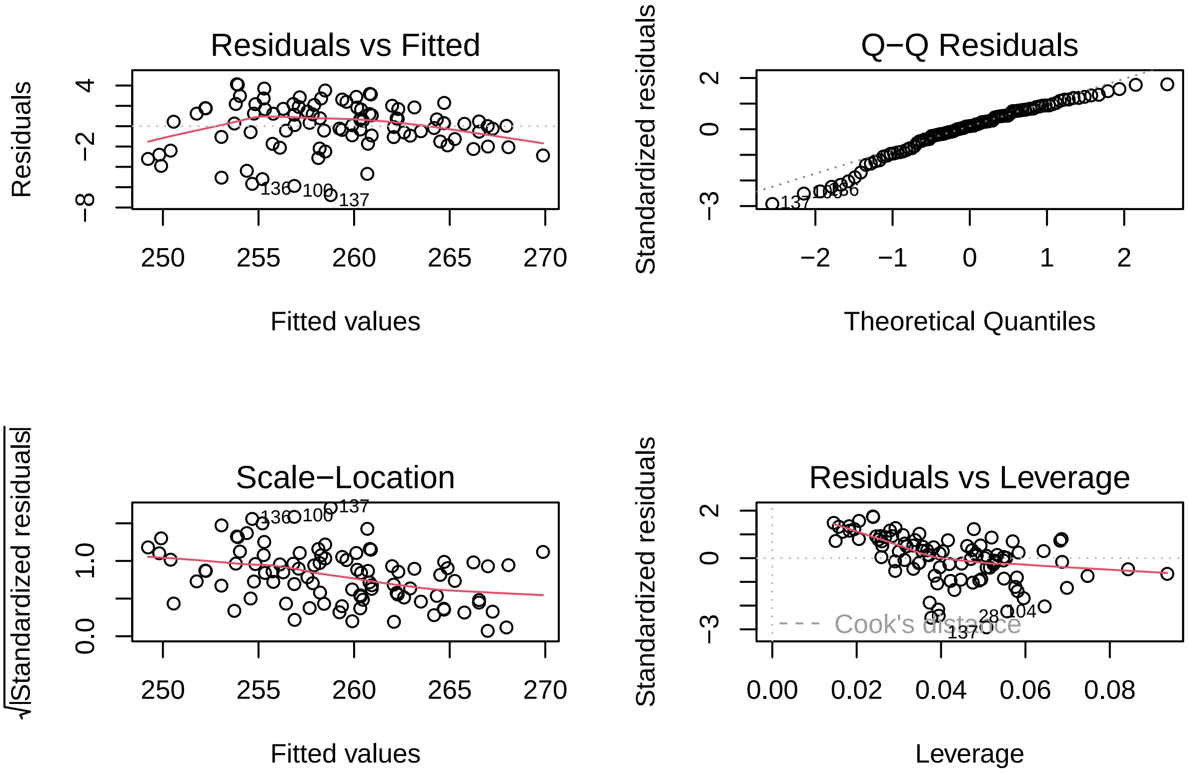
```
errs <- modbic$residuals
fit <- modbic$fitted.values
plot(abs(errs)~fit)
lomod<-loess(abs(errs)~fit)
inx<-order(fit)
lines(fit[inx],lomod$fitted[inx], col = "red")
```



```
ermod <- lm(abs(errs)~fit)
fit.as.sd <- abs(ermod$fitted.values)
w <- 1/fit.as.sd
wls_model <- lm(modbic, weights = w, data = train_data)
summary(wls_model)
```

```
##
## Call:
## lm(formula = modbic, data = train_data, weights = w)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7751 -1.1695  0.2922  1.7302  4.1282
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.471e+02  6.816e-01 362.551   <2e-16 ***
## Area        3.672e-04  3.312e-05  11.088   <2e-16 ***
## Elev        1.479e-02  1.897e-03   7.798    1e-11 ***
## Soil        1.128e-01  1.118e-02  10.089   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.384 on 91 degrees of freedom
## Multiple R-squared:  0.8017, Adjusted R-squared:  0.7951
## F-statistic: 122.6 on 3 and 91 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(wls_model,1)
plot(wls_model,2)
plot(wls_model,3)
plot(wls_model,5)
```



Prediction Error (Generalization Error)

```r
predictions_test <- predict(modback, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.116811
```

```r
predictions_test <- predict(modCp, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.145663
```

```r
predictions_test <- predict(modaic, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.116811
```

```r
predictions_test <- predict(modbic, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.116811
```

```r
predictions_test <- predict(modr2, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.406523
```

```r
predictions_test <- predict(wls_model, newdata = test_data)

generalization_error <- mean((test_data$NR - predictions_test)^2)

cat("Generalization Error:", generalization_error, "\n")
```

```
## Generalization Error: 4.116811
```

Final Model

from the above we can see that wls_model, modback, modaic and modback have the same generalization error.

Let us further investigate the coefficients of these four models.

```r
modbic$coefficients
```

```
##  (Intercept)         Area         Elev         Soil 
## 2.471174e+02 3.672412e-04 1.478823e-02 1.128147e-01
```

```r
modaic$coefficients
```

```
##  (Intercept)         Area         Elev         Soil 
## 2.471174e+02 3.672412e-04 1.478823e-02 1.128147e-01
```

```r
modback$coefficients
```

```
##  (Intercept)         Area         Elev         Soil 
## 2.471174e+02 3.672412e-04 1.478823e-02 1.128147e-01
```

```r
wls_model$coefficients
```

```
##  (Intercept)         Area         Elev         Soil 
## 2.471174e+02 3.672412e-04 1.478823e-02 1.128147e-01
```

We have noticed that the coefficients of all these models are same which indicated that these are the same models. So we can consider any of these as our final model. Let us say modbic is our final model due to less generalization error.

```r
# Let us print the model

bicform
```

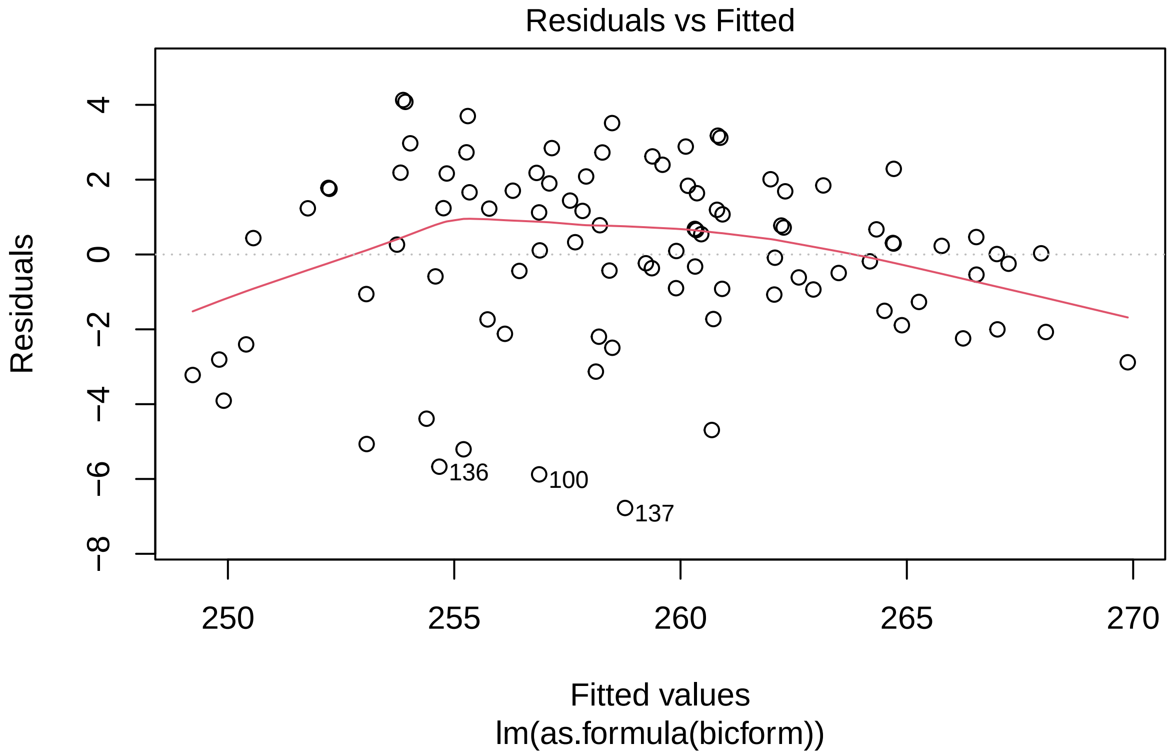
```
## [1] "NR~Area+Elev+Soil"
```

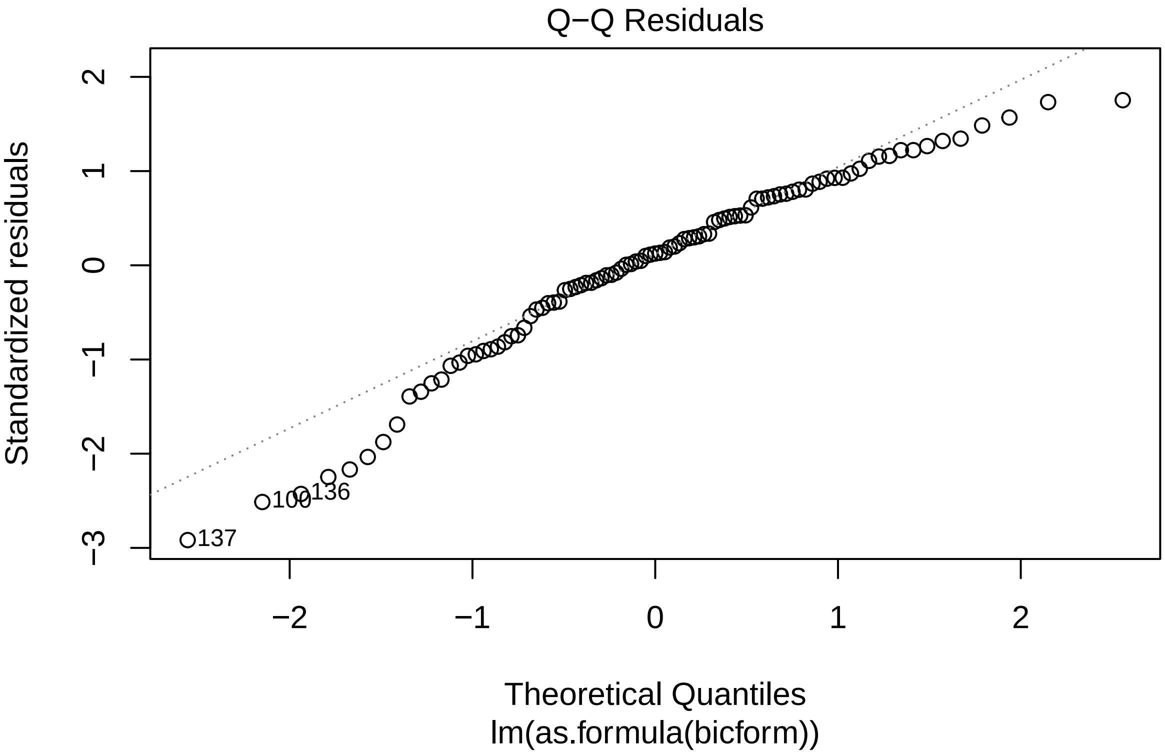
```r
# Let us print the summary of the model

summary(modbic)
```

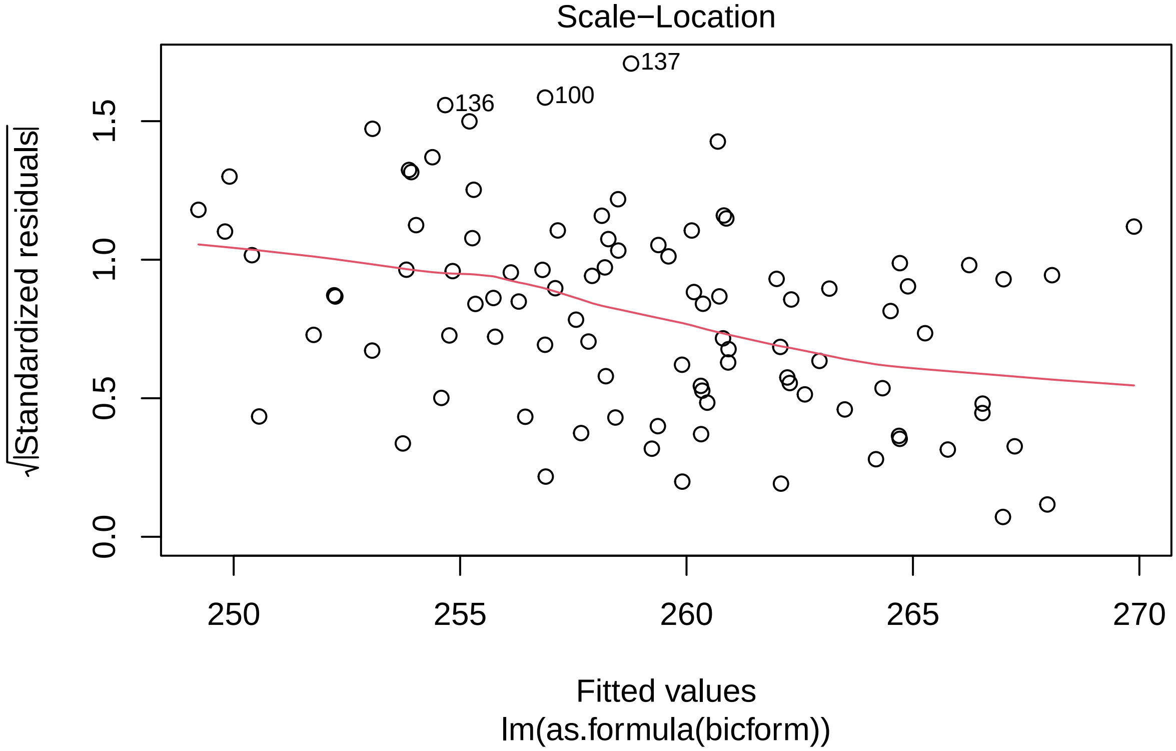
```
## 
## Call:
## lm(formula = as.formula(bicform), data = train_data)
## 
```

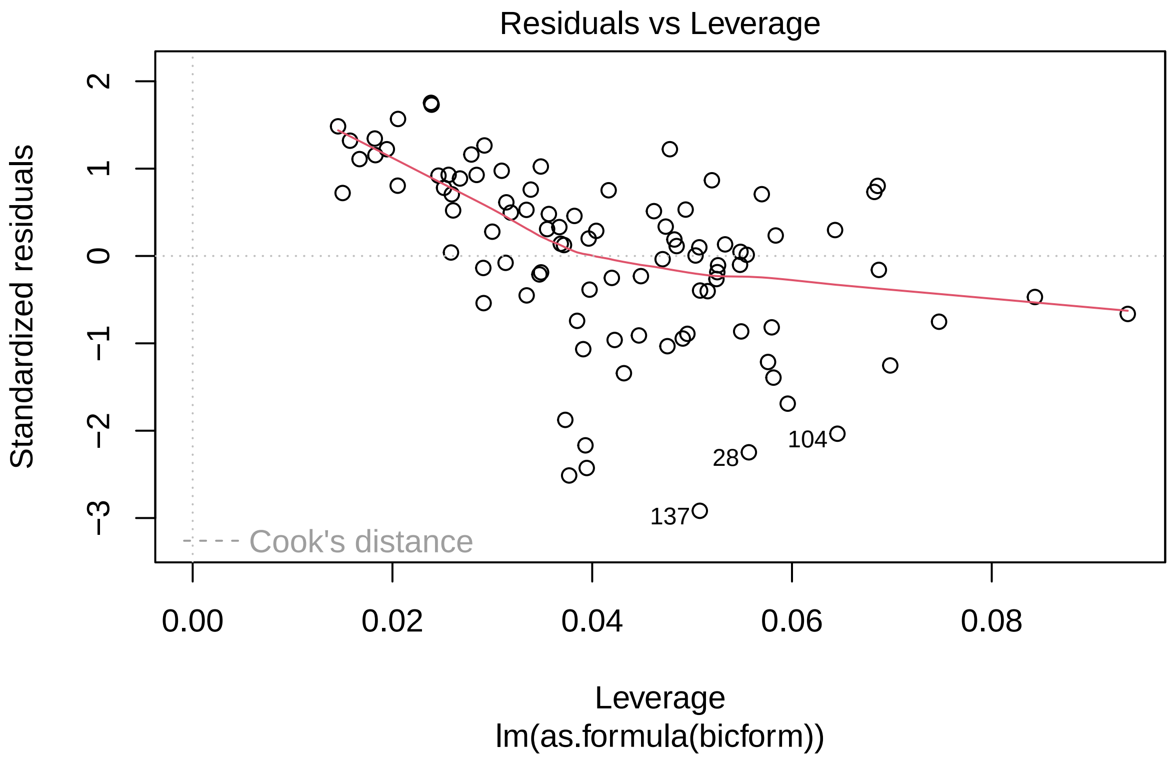
```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -6.7751 -1.1695  0.2922  1.7302  4.1282 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.471e+02  6.816e-01 362.551   <2e-16 ***
## Area        3.672e-04  3.312e-05  11.088   <2e-16 ***
## Elev        1.479e-02  1.897e-03   7.798    1e-11 ***
## Soil        1.128e-01  1.118e-02  10.089   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.384 on 91 degrees of freedom
## Multiple R-squared:  0.8017, Adjusted R-squared:  0.7951 
## F-statistic: 122.6 on 3 and 91 DF,  p-value: < 2.2e-16
```

```
# Let us look at the assessment plots of our final model

plot(modbic)
```




Q−Q Residuals
Standardized residuals
2
1
0
−1
−2
−3
137
100
136
−2
−1
0
1
2
Theoretical Quantiles
lm(as.formula(bicform))


Scale−Location
√|Standardized residuals|
1.5
1.0
0.5
0.0
137
136
100
250
255
260
265
270
Fitted values
lm(as.formula(bicform))



```
# Finding the BIC score of the final model modbic

BIC(modbic)
```

```
## [1] 453.3425
```