SHC 798 Assignment 2, 2025

Richard Lubega

2025-10-03

SHC 798 Assignment 2, 2025

Multiple Linear Analysis (MLR)

Question 2: # Energy consumption data from 80 office buildings

```
pacman::p_load(tidymodels)

# Getting started with the dataset in energy.csv :
e.consump <- read.csv(file.choose(), header = TRUE, na.strings = c("NA"))
head(e.consump) # View first few rows of the dataset
```

```
##   energy area occup climate glazing insulation
## 1 1083.5 1887   174       2    47.2      108.5
## 2 1560.9 5445   331       1    41.8      101.4
## 3 1103.5 5576   246       1    24.2      115.3
## 4 1239.7 6304   132       3    47.9      124.9
## 5 1423.2 5749   260       1    32.7       61.7
## 6 1056.0 4778   102       1    49.3       79.0
```

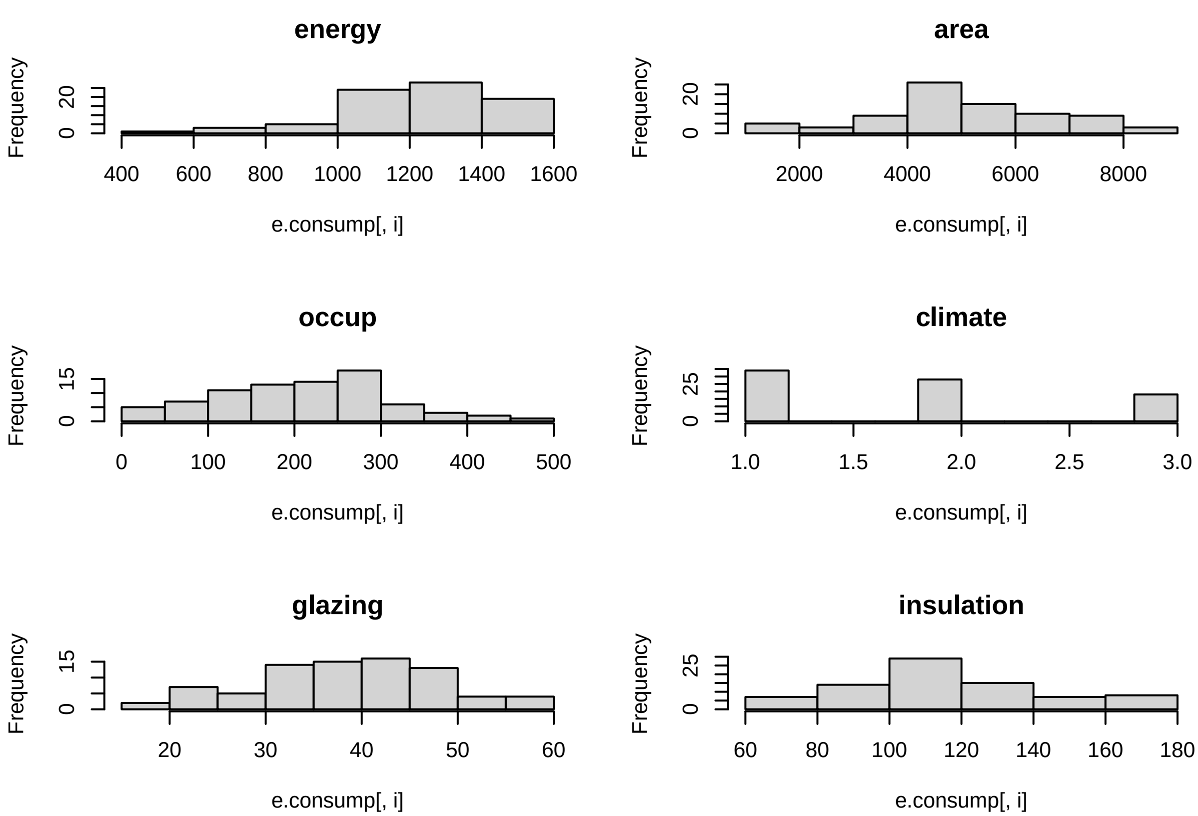
```
summary(e.consump) # Get an overview of the dataset
```

```
##      energy            area          occup          climate        glazing
##  Min.   : 448.7   Min.   :1500   Min.   : 10.0   Min.   :1.0   Min.   :15.00
##  1st Qu.:1095.3   1st Qu.:4101   1st Qu.:133.0   1st Qu.:1.0   1st Qu.:32.00
##  Median :1222.6   Median :4935   Median :205.5   Median :2.0   Median :38.70
##  Mean   :1216.1   Mean   :5015   Mean   :211.9   Mean   :1.8   Mean   :38.68
##  3rd Qu.:1379.9   3rd Qu.:6153   3rd Qu.:281.0   3rd Qu.:2.0   3rd Qu.:45.30
##  Max.   :1571.6   Max.   :8591   Max.   :469.0   Max.   :3.0   Max.   :57.20
##    insulation
##  Min.   : 60.00
##  1st Qu.: 98.08
##  Median :110.45
##  Mean   :115.33
##  3rd Qu.:128.22
##  Max.   :172.10
```

```r
str(e.consump) # inspect the dataset and viewing column data types
```

```
## 'data.frame':    80 obs. of  6 variables:
##  $ energy    : num  1084 1561 1104 1240 1423 ...
##  $ area      : int  1887 5445 5576 6304 5749 4778 4291 8005 6446 4765 ...
##  $ occup     : int  174 331 246 132 260 102 76 423 189 299 ...
##  $ climate   : int  2 1 1 3 1 1 3 3 2 1 ...
##  $ glazing   : num  47.2 41.8 24.2 47.9 32.7 49.3 32 31.4 48.6 38.4 ...
##  $ insulation: num  108.5 101.4 115.3 124.9 61.7 ...
```

```r
# View variables
par(mfrow=c(3,2))
for (i in 1:6) hist(e.consump[,i], main=names(e.consump)[i])
```



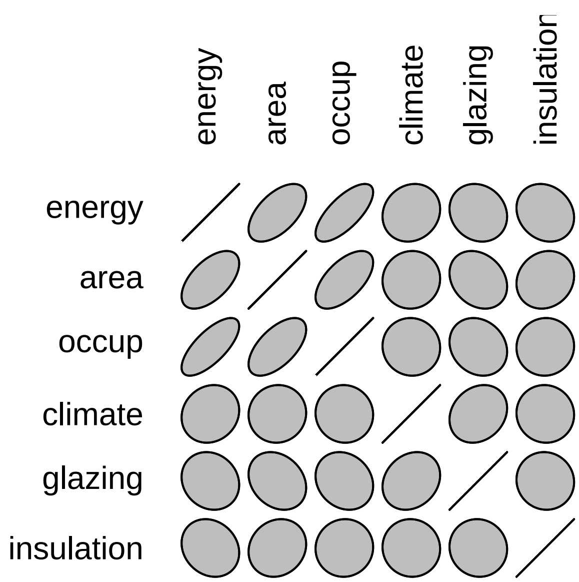

```r
par(mfrow = c(1, 1))

# Part a): Multicollinearity
# (i) Pearson correlation coefficients
cor(e.consump, method = "pearson")
```

```
##               energy       area       occup     climate    glazing
## energy     1.0000000 0.56727188  0.71535501  0.12451307 -0.1348882
## area       0.5672719 1.00000000  0.60076867  0.03597627 -0.2360775
## occup      0.7153550 0.60076867  1.00000000 -0.03118426 -0.1716492
```

```
## climate     0.1245131  0.03597627 -0.03118426  1.00000000  0.2001212
## glazing    -0.1348882 -0.23607748 -0.17164920  0.20012116  1.0000000
## insulation -0.1681677  0.13148613  0.02944892 -0.03287315 -0.0447956
##             insulation
## energy     -0.16816767
## area        0.13148613
## occup       0.02944892
## climate    -0.03287315
## glazing    -0.04479560
## insulation  1.00000000
```

```r
# (ii) An ellipse plot to visualise collinearity
pacman::p_load(ellipse)
plotcorr(cor(e.consump))
```




```r
# (iii) Variance Inflation Factors (VIFs)
pacman::p_load(car)
engy_model <- lm(energy ~ area + occup + climate + glazing + insulation, data = e.consump)
vif(engy_model)
```

```
##       area      occup    climate    glazing insulation
##   1.661848   1.579343   1.055096   1.111013   1.023478
```

Commenting on Multicollinearity In the correlogram (ellipse plot), narrow/elongated ellipses indicate stronger correlation. Energy has elongated ellipses with area (0.5672719) and occupancy (0.71535501), indicating moderate to strong positive correlation. Also, area and occupancy are noticeably correlated with narrow tilted ellipse (0.60076867) which indicates collinearity. Therefore, there is some multicollinearity between area and occupancy, and to a lesser extent between energy and these two variables.

Since all VIF values are very well below 5, there is **no significant multicollinearity** among the predictors for the model, engy_model. This suggests that the predictors can be considered independent of each other for this regression model.

```r
# Part b): Model and Predictor Linearity
# Initial Model Output
summary(engy_model)
```

```
##
## Call:
## lm(formula = energy ~ area + occup + climate + glazing + insulation,
##     data = e.consump)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -522.35  -74.40   11.52   93.61  367.70
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 936.47055  115.55537   8.104 8.23e-12 ***
## area          0.03186    0.01257   2.534  0.01338 *
## occup         1.26073    0.19992   6.306 1.87e-08 ***
## climate      36.38855   21.04601   1.729  0.08798 .
## glazing      -0.32620    1.73189  -0.188  0.85112
## insulation   -1.73568    0.60284  -2.879  0.00521 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 143.1 on 74 degrees of freedom
## Multiple R-squared:  0.6042, Adjusted R-squared:  0.5774
## F-statistic: 22.59 on 5 and 74 DF,  p-value: 1.101e-13
```

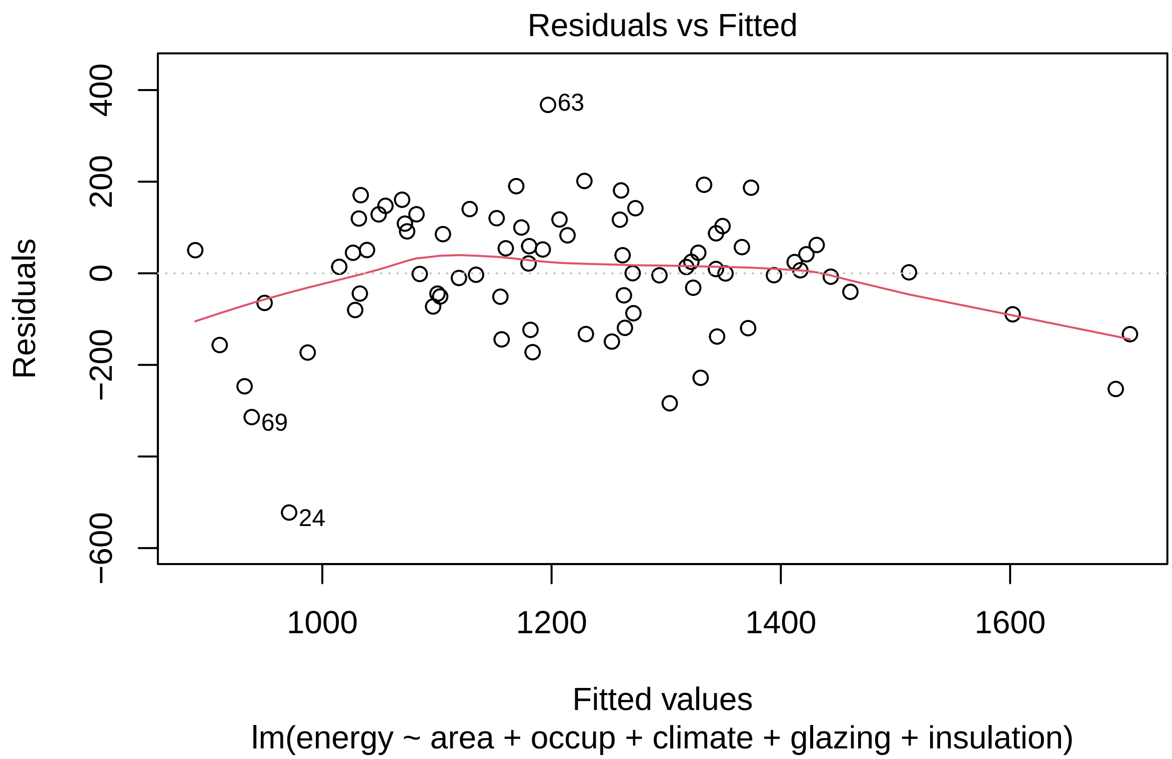
```r
confint(engy_model)
```

```
##                    2.5 %       97.5 %
## (Intercept) 706.22145364 1166.71964585
## area          0.00681188    0.05690576
## occup         0.86238033    1.65908800
## climate      -5.54653691   78.32363945
## glazing      -3.77706386    3.12466606
## insulation   -2.93685538   -0.53449767
```

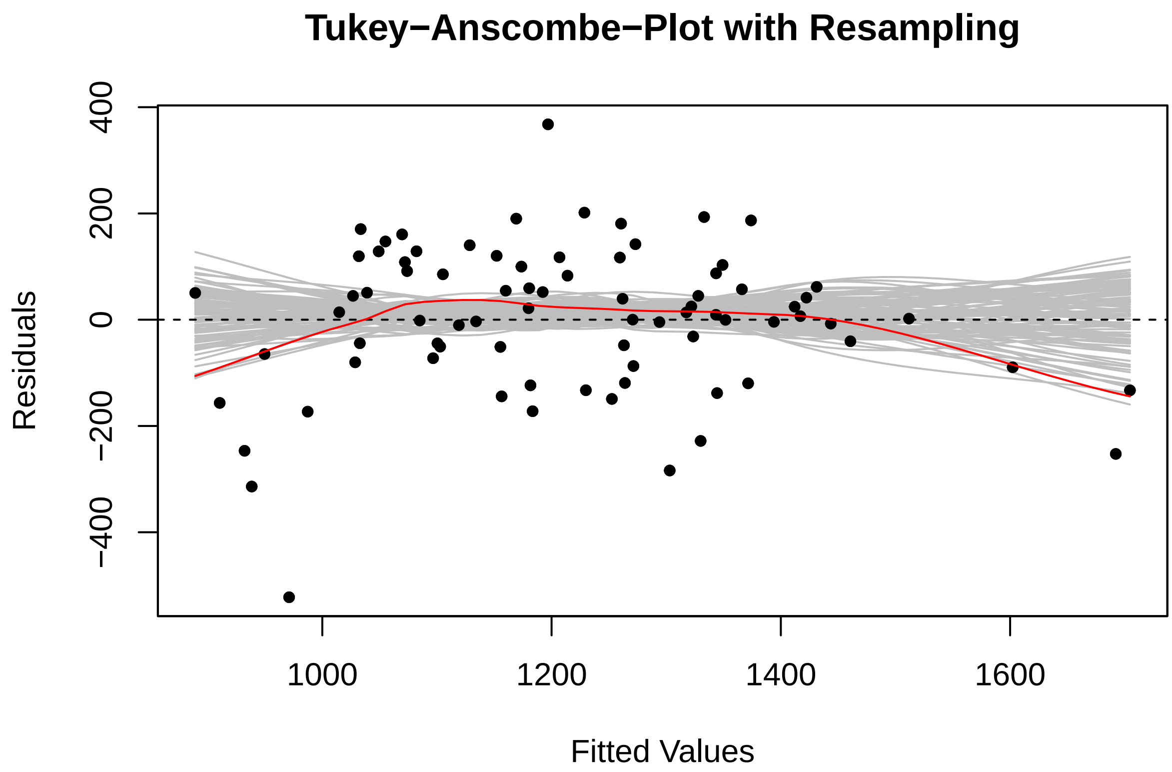
```r
confint(engy_model)["(Intercept)", ]
```

```
##     2.5 %    97.5 %
##  706.2215 1166.7196
```

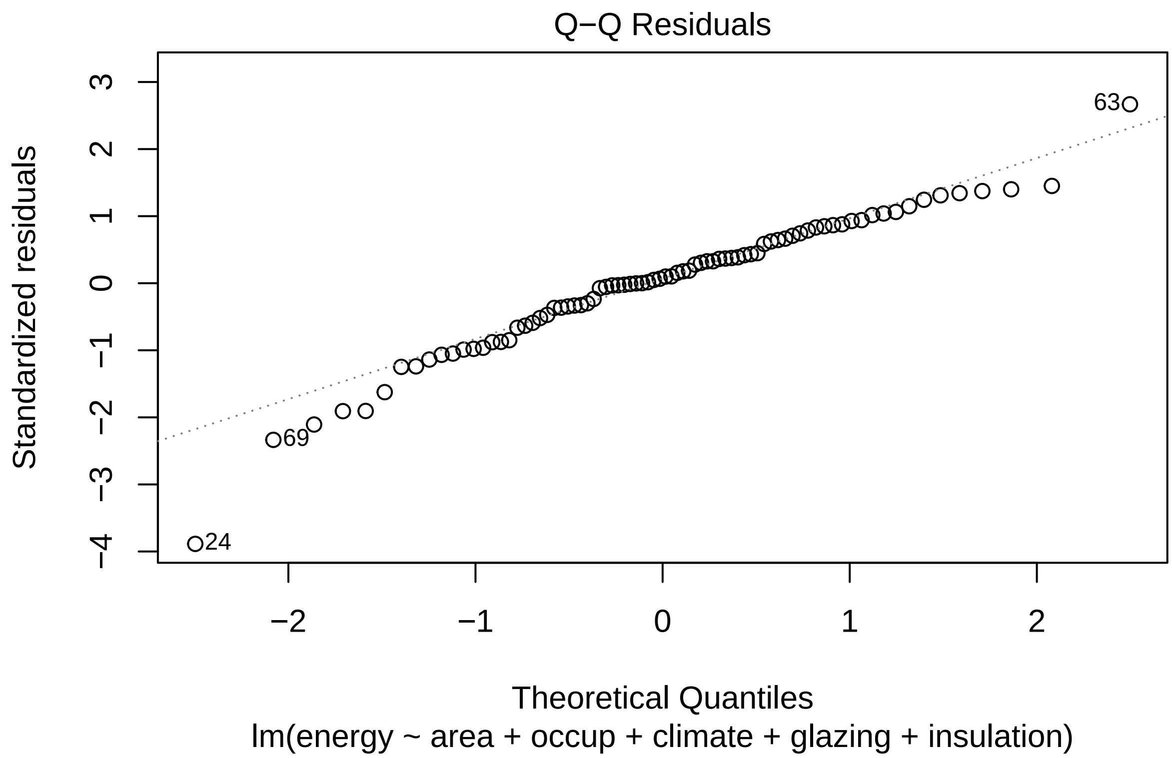
```r
## Residual analysis 1
plot(engy_model, which=1)
```



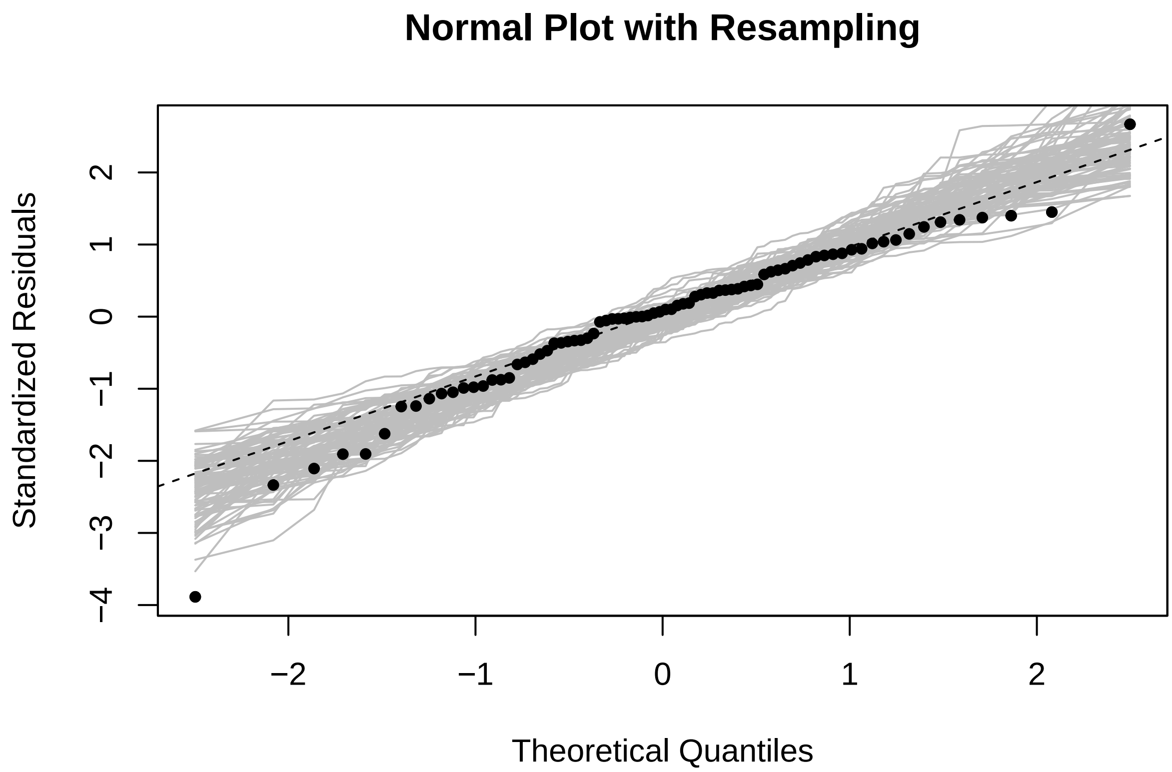

```r
resplot(engy_model, plots = 1)
```



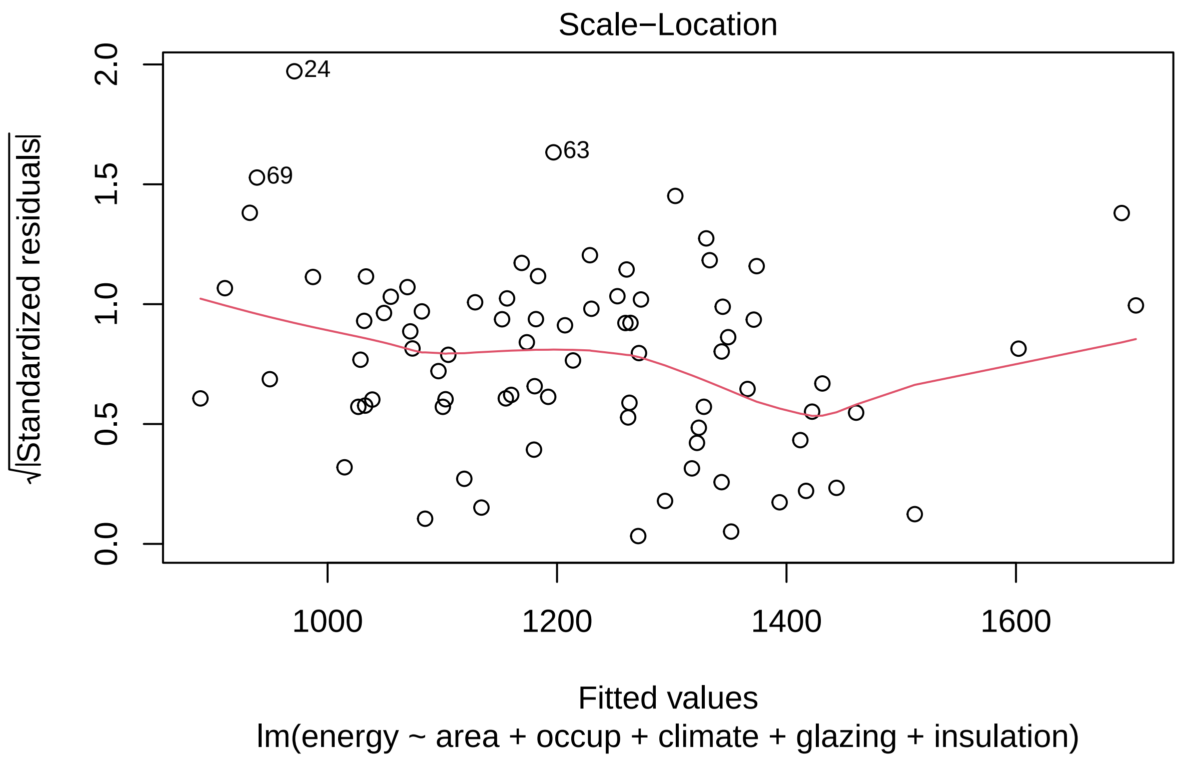

```
plot(engy_model, which = 2)
```



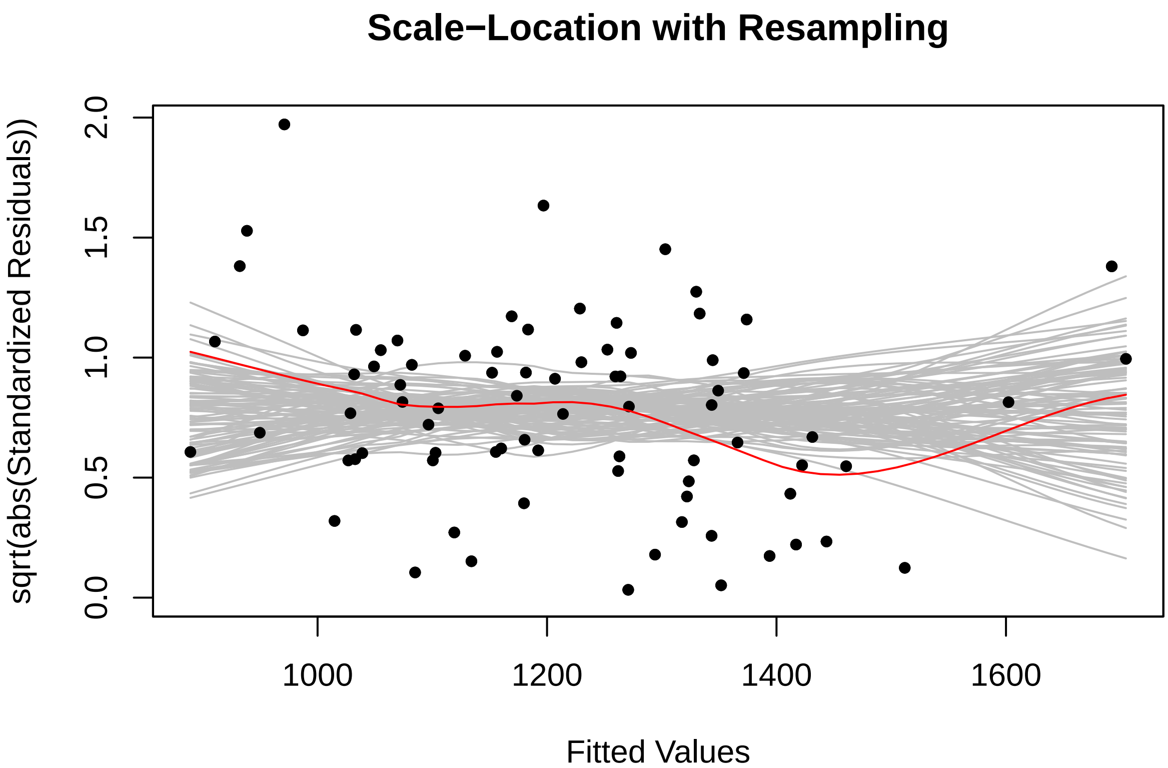

```r
resplot(engy_model, plots = 2)
```



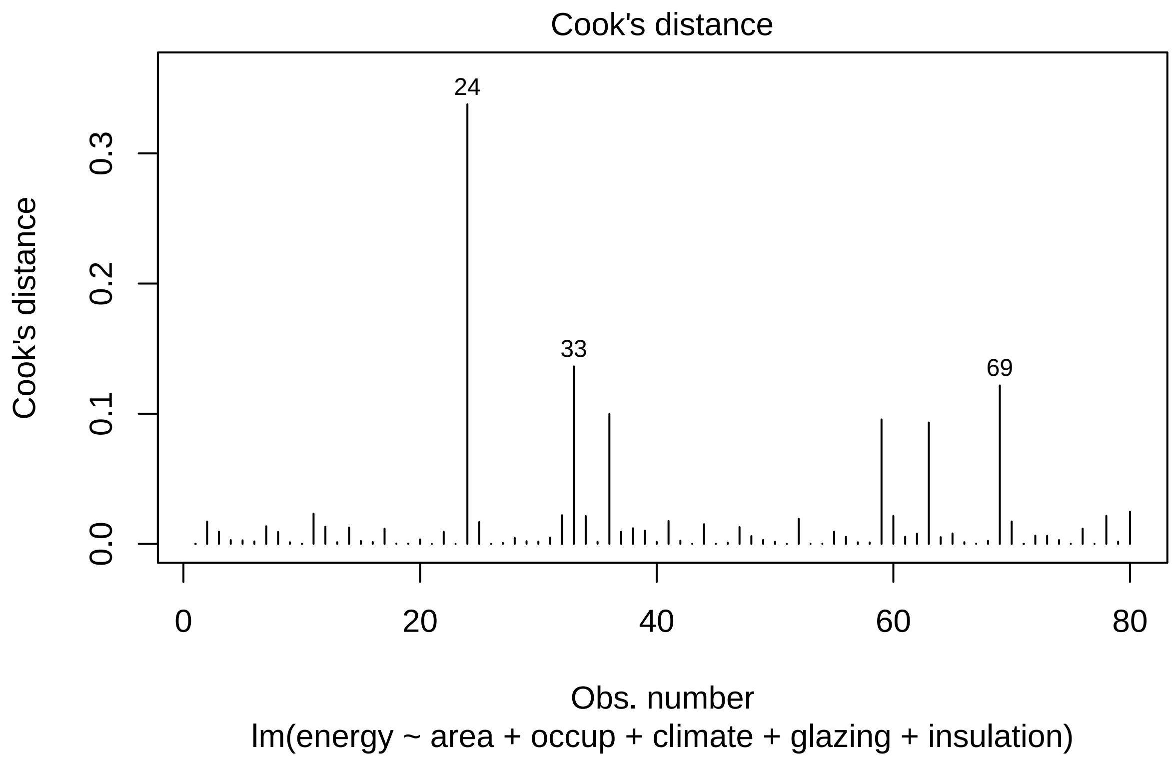

```r
## Scale-location plot
plot(engy_model, which = 3)
```



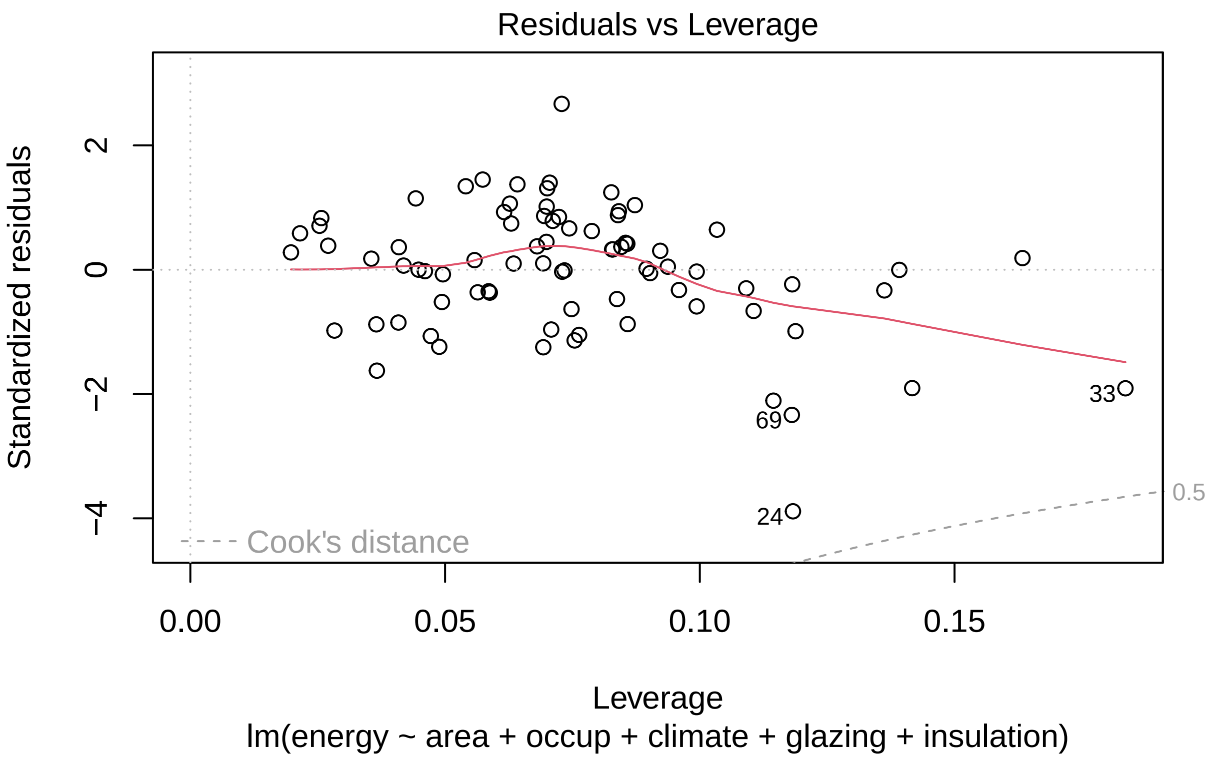

```
resplot(engy_model, plots = 3)
```



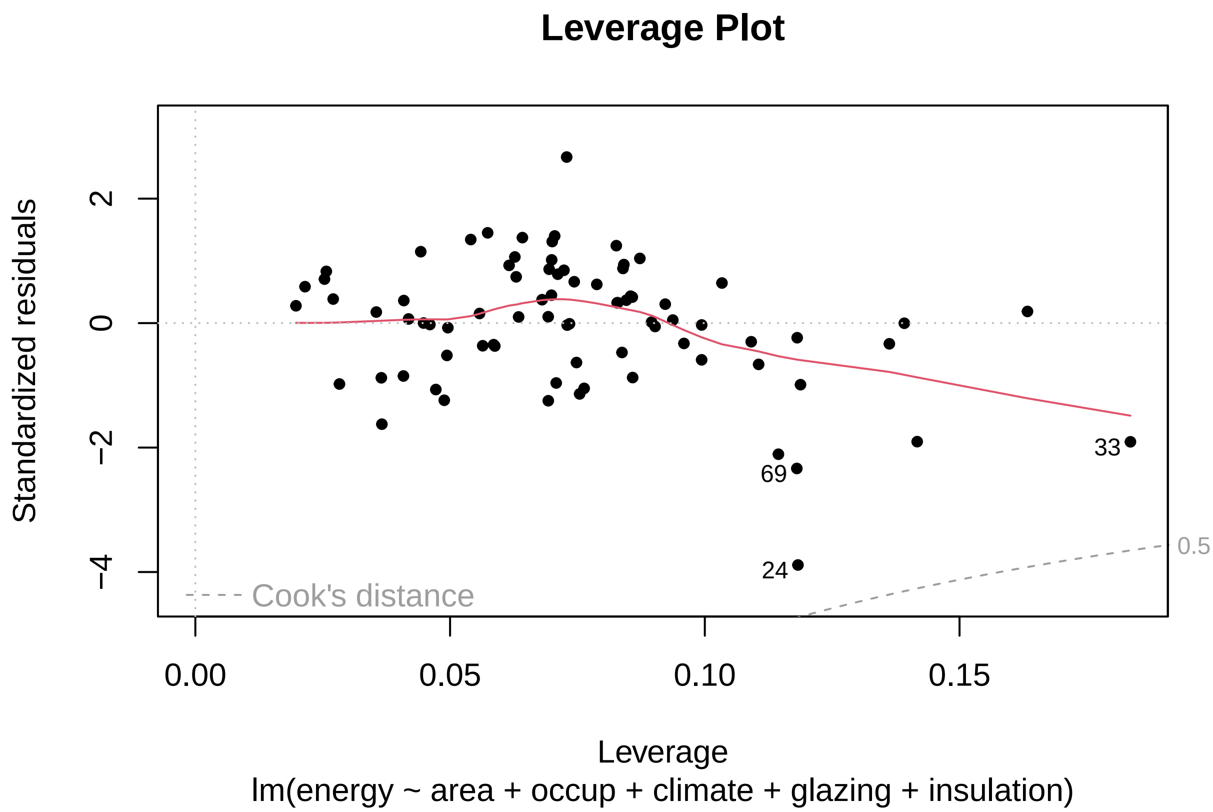

```r
## Cook's Distance plot
plot(engy_model, which = 4)
```



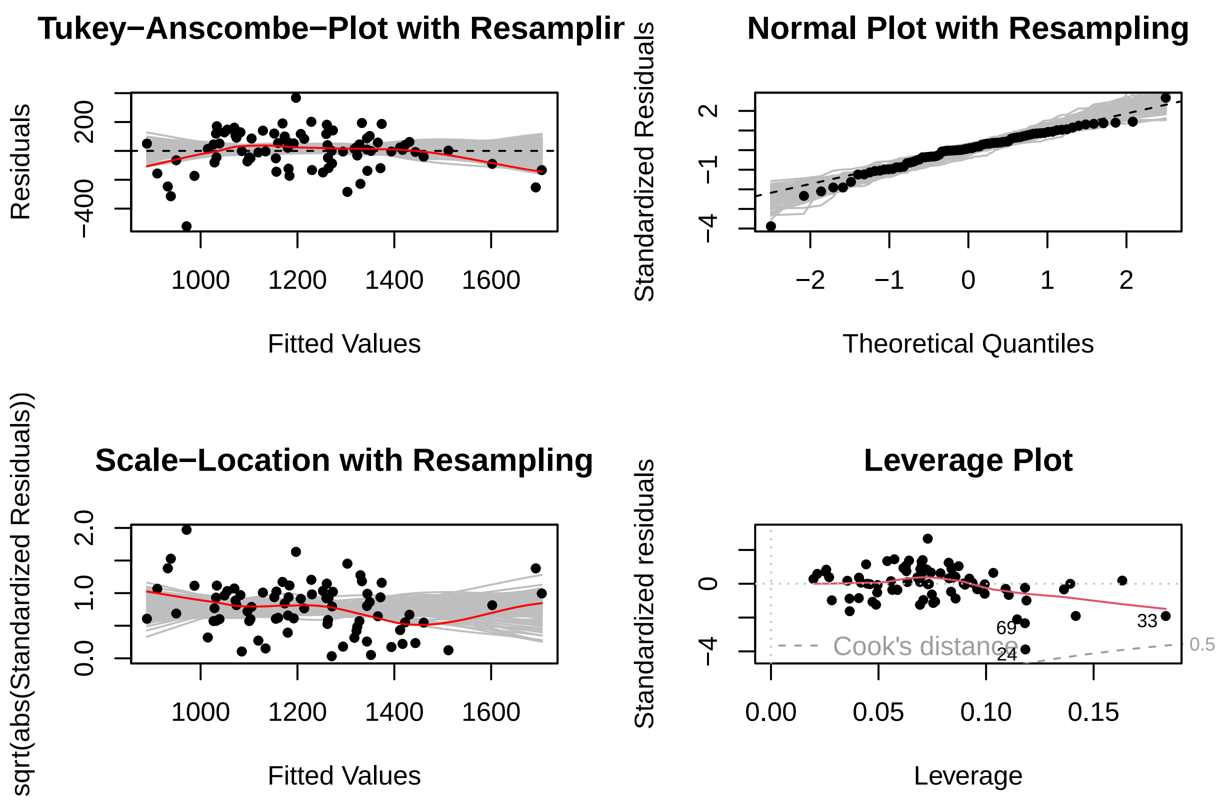

```
plot(engy_model, which = 5)
```



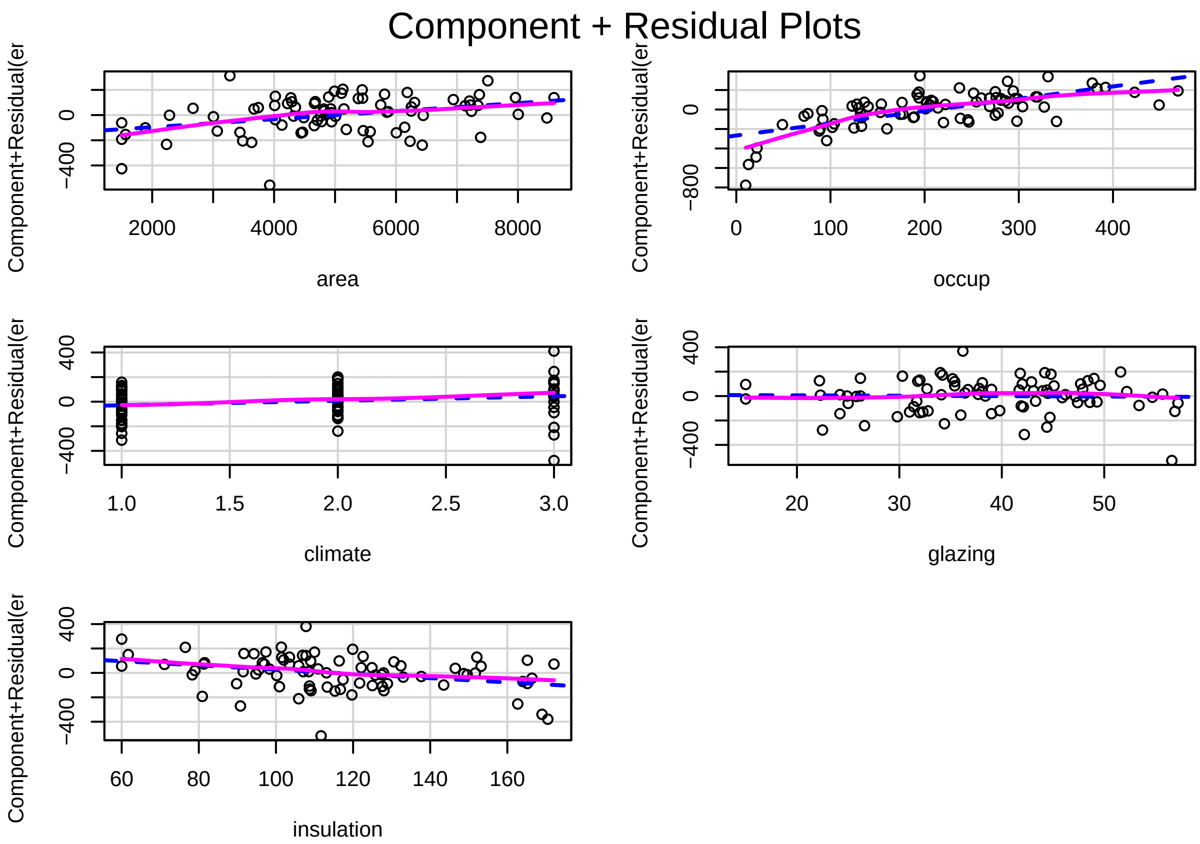

```
resplot(engy_model, plots = 4)
```



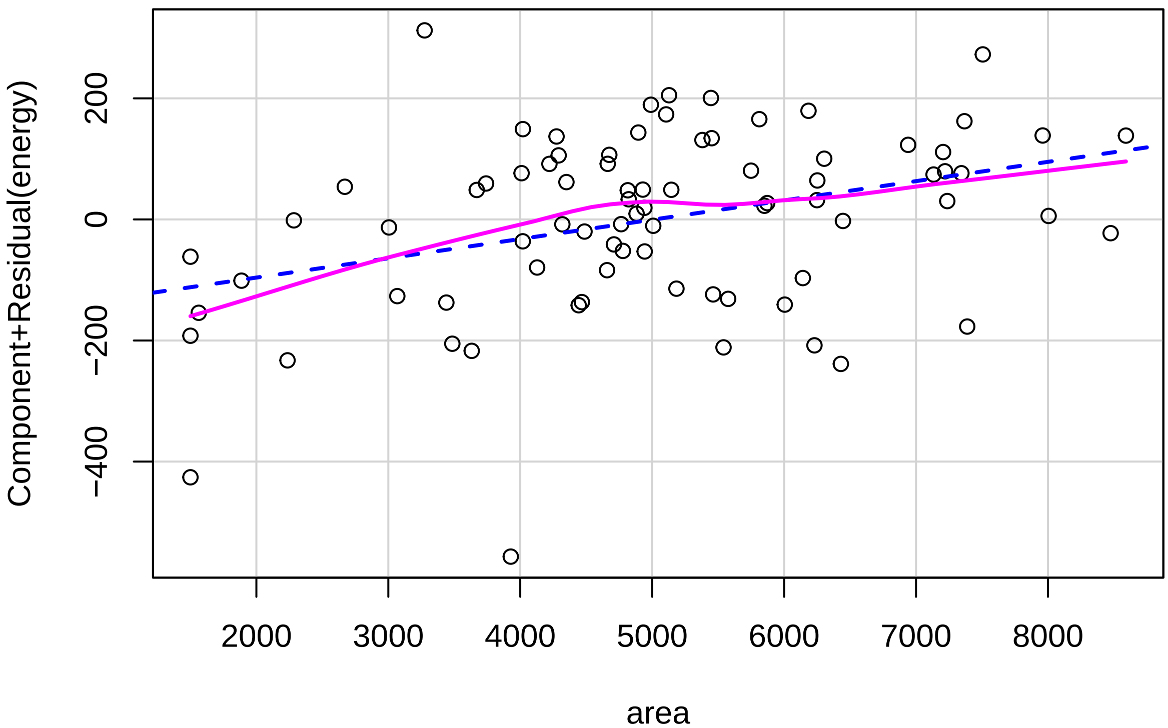

```r
resplot(engy_model)
```

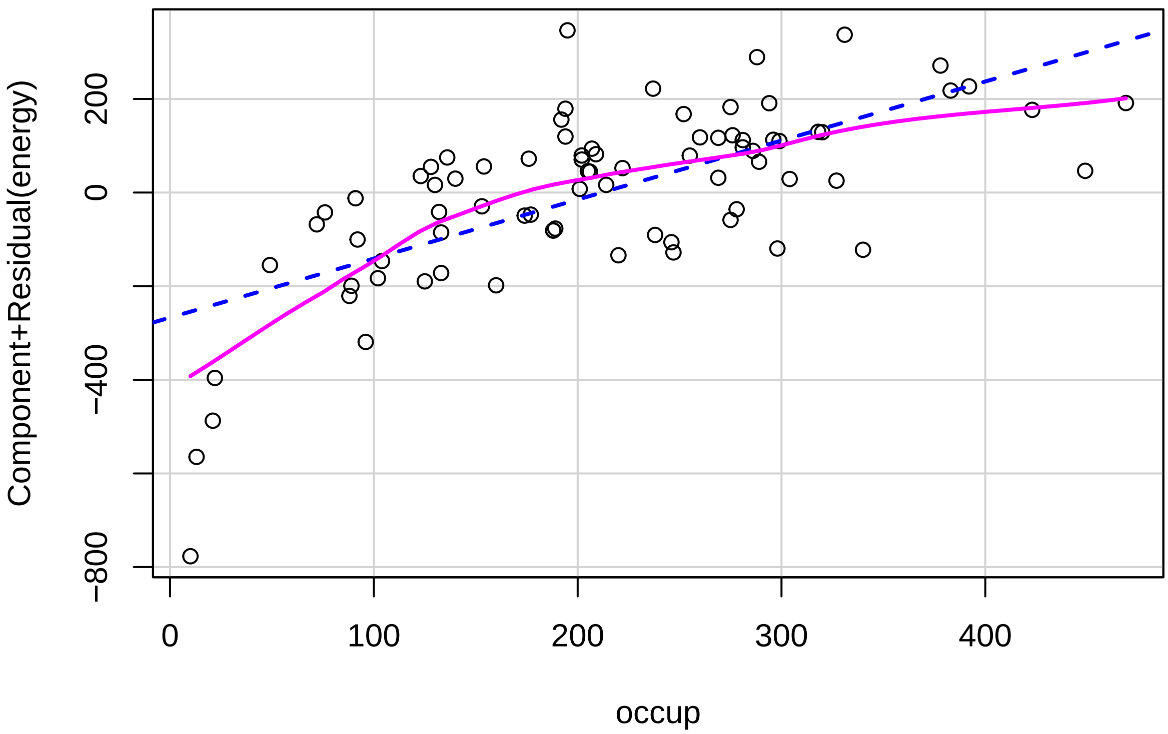



```
# Linearity of each predictor - Use of Partial Residual Plots
pacman::p_load(car)
crPlots(engy_model)
```

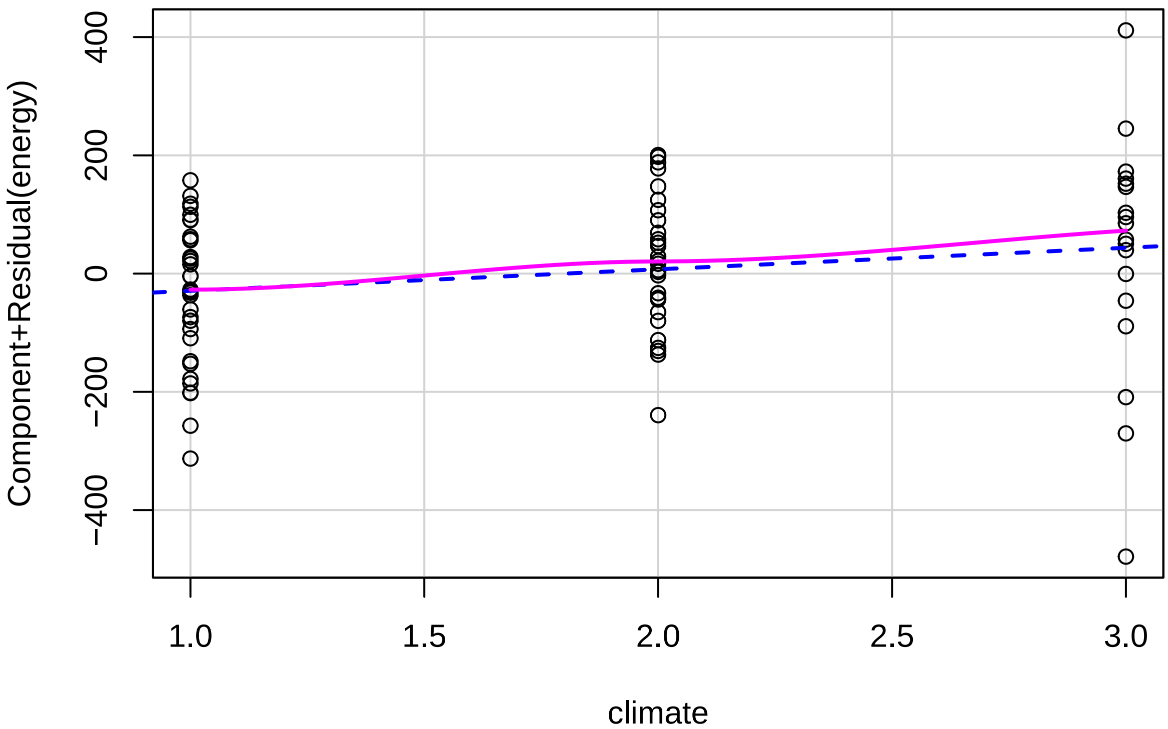



```r
crPlots(engy_model, layout = c(1,1))
```

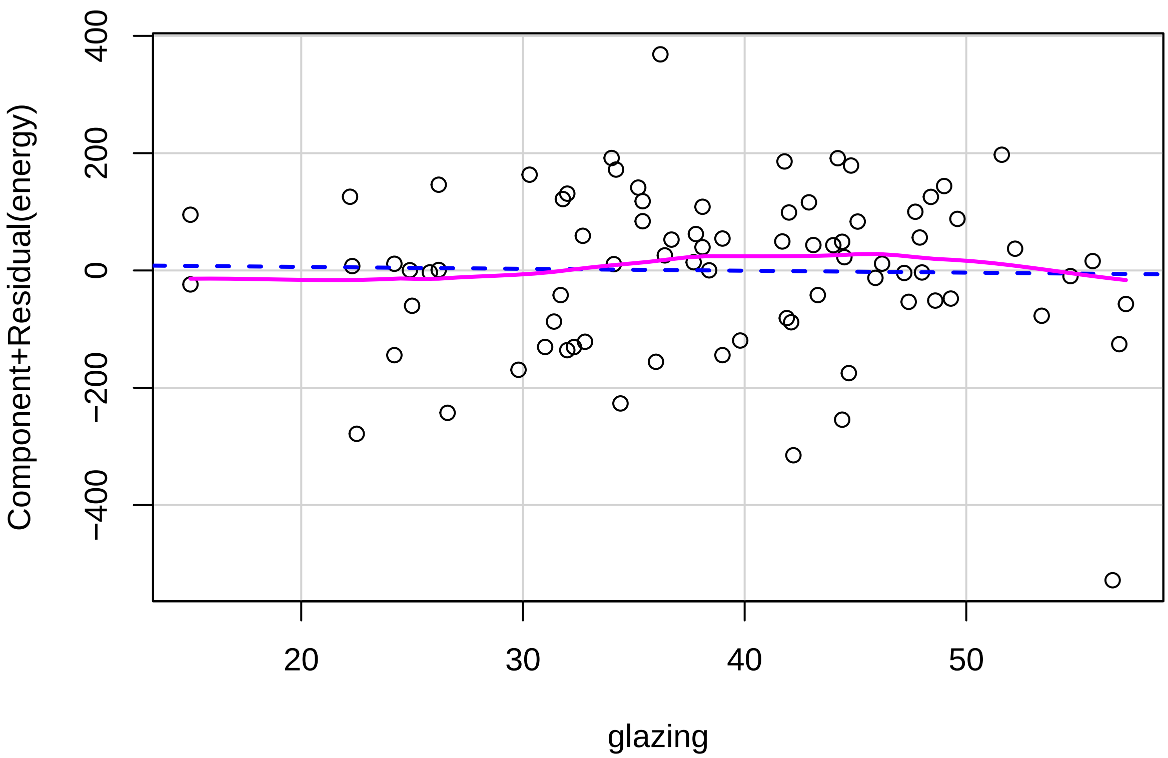

Component+Residual(energy)
200
0
−200
−400
2000
3000
4000
5000
6000
7000
8000
area



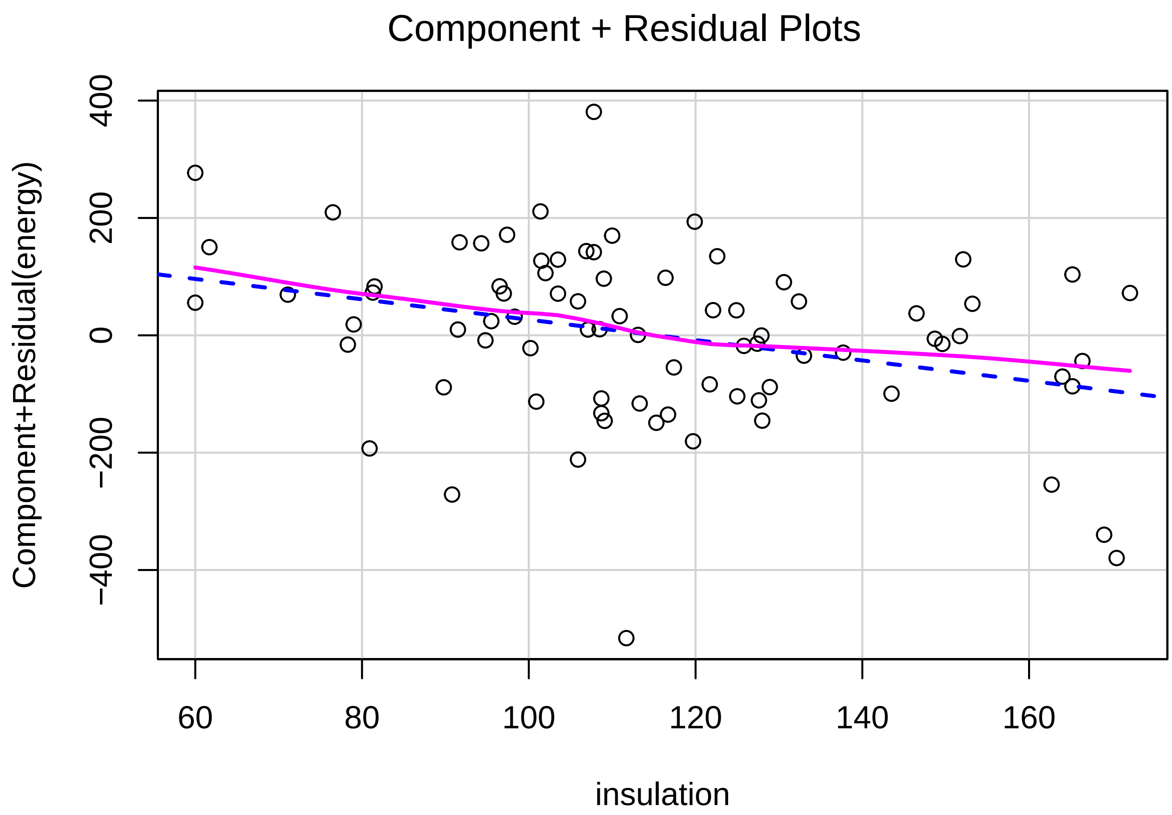
Component+Residual(energy)
200
0
−400
−800
0
100
200
300
400
occup



Component+Residual(energy)
400
200
0
−200
−400
1.0
1.5
2.0
2.5
3.0
climate



Component+Residual(energy)
400
200
0
−200
−400
20
30
40
50
glazing

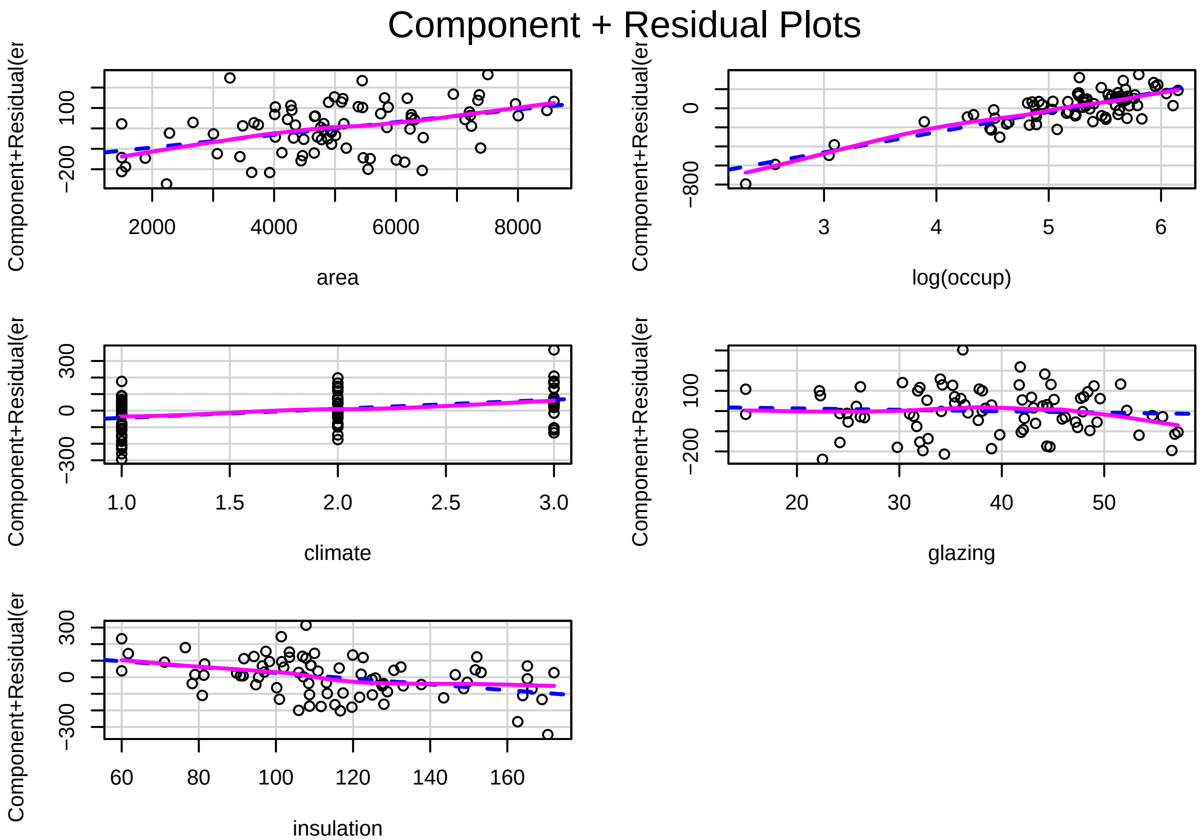



```r
# Transformed Model 1
engy_model2 <- lm(energy ~ area + log(occup) + climate + glazing + insulation, data = e.consump)
summary(engy_model2)
```

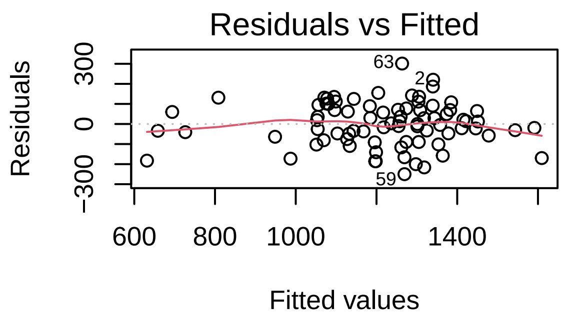
```
##
## Call:
## lm(formula = energy ~ area + log(occup) + climate + glazing +
##     insulation, data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -250.129  -66.554    1.599   72.610  301.157
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  89.134103 127.287741   0.700 0.485963
## area          0.031008   0.009216   3.365 0.001217 **
## log(occup)  212.531379  20.365584  10.436 3.42e-16 ***
## climate      55.078048  16.765693   3.285 0.001559 **
## glazing      -0.694371   1.365478  -0.509 0.612602
## insulation   -1.744152   0.474978  -3.672 0.000452 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112.9 on 74 degrees of freedom
```

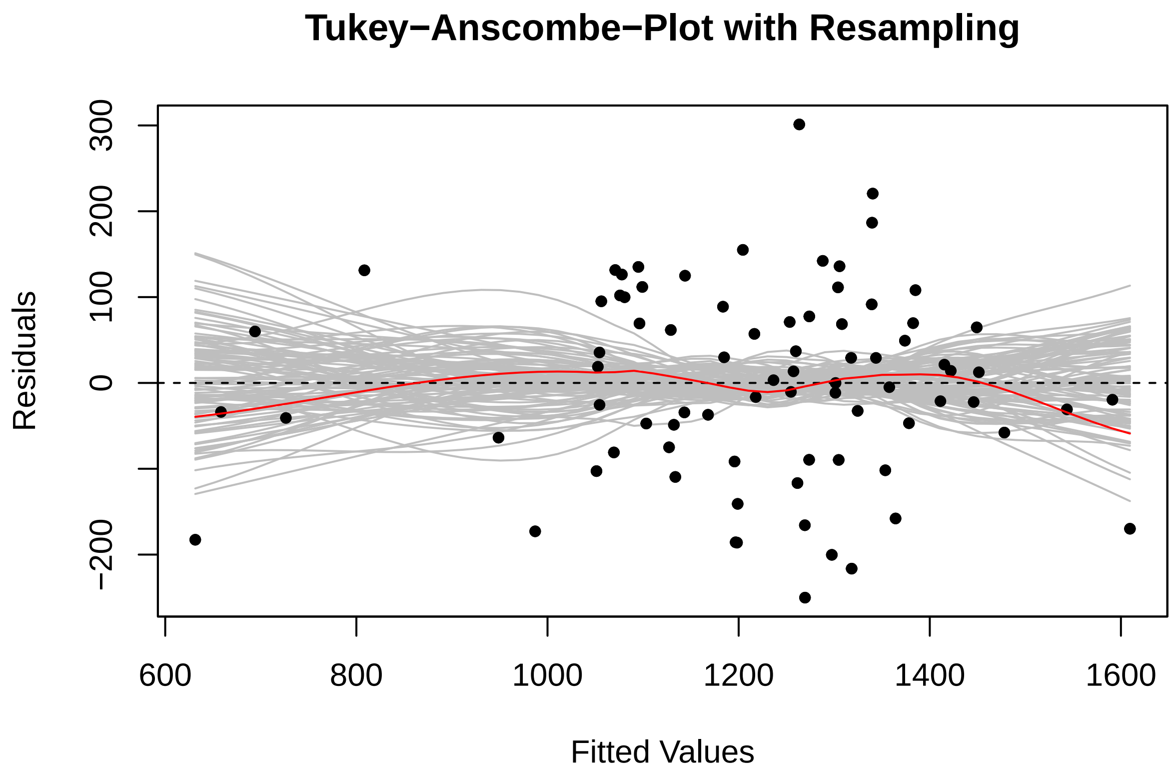
```
## Multiple R-squared:  0.7538, Adjusted R-squared:  0.7371
## F-statistic: 45.31 on 5 and 74 DF,  p-value: < 2.2e-16
```

```r
crPlots(engy_model2)
```

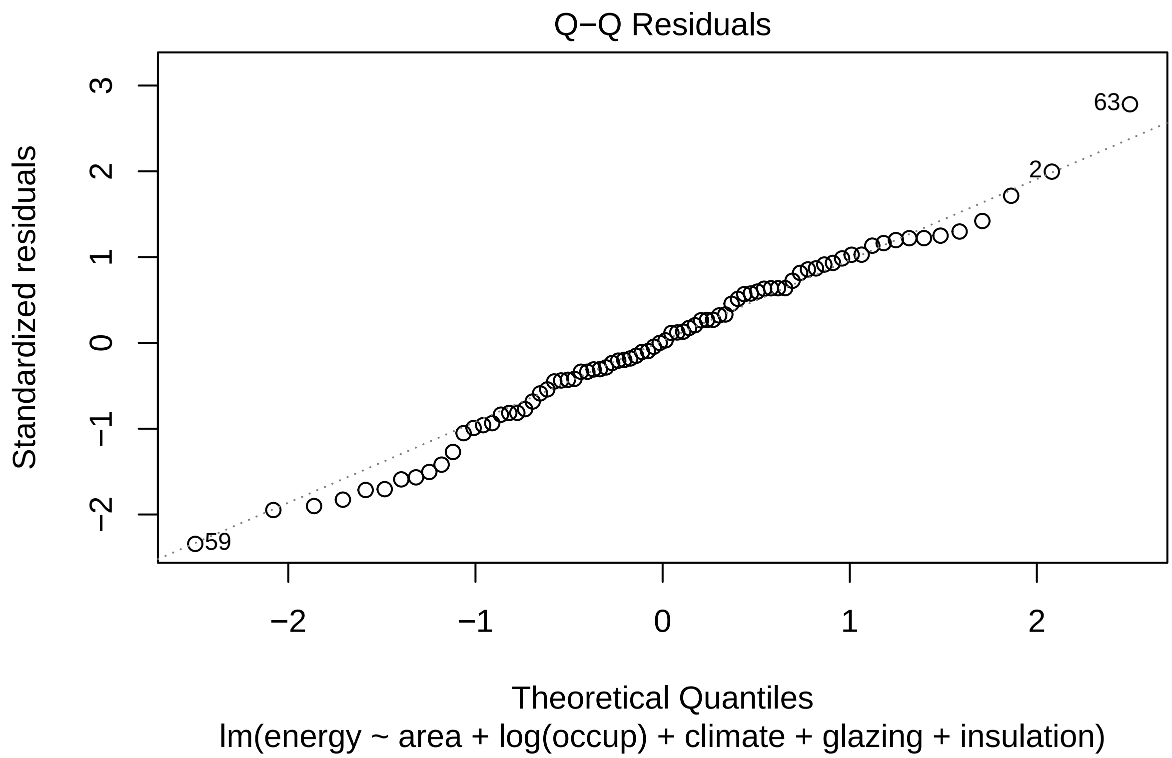



```r
## Residual analysis 2
plot(engy_model2, which=1)
resplot(engy_model2, plots = 1)
```

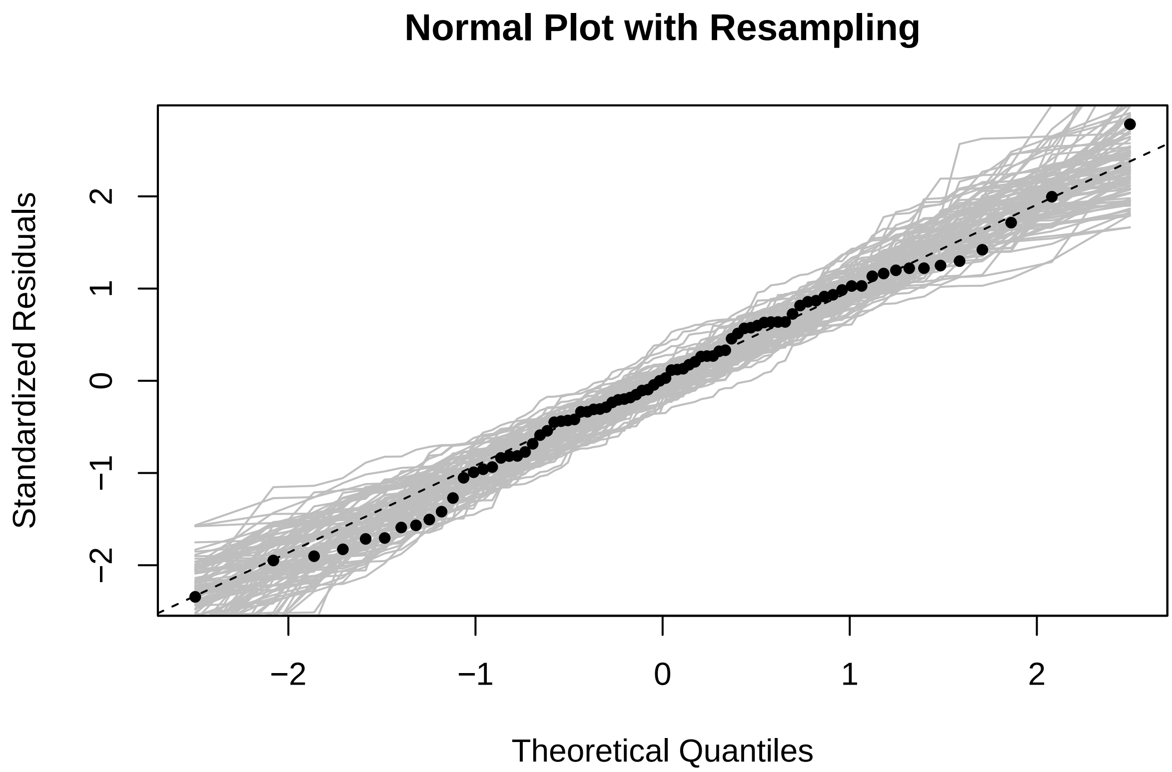

Residuals vs Fitted
63
2
59
300
0
−300
600
800
1000
1400
Residuals
Fitted values



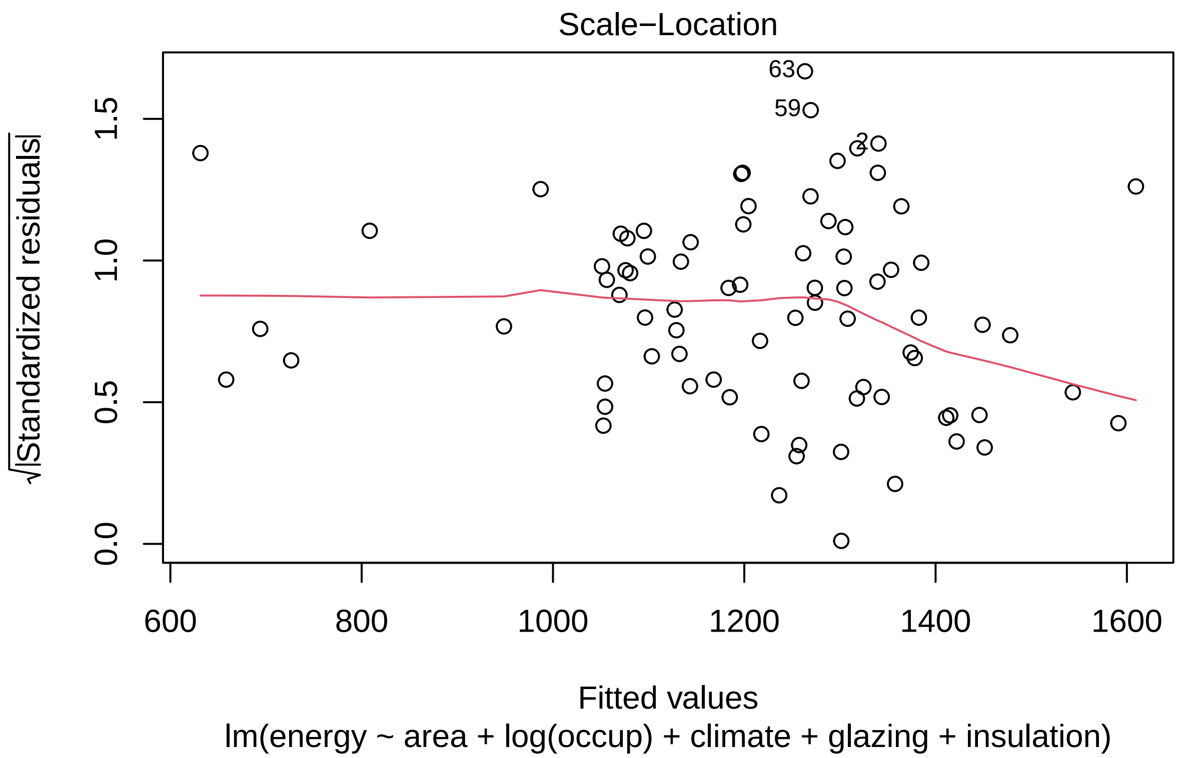

```
plot(engy_model2, which = 2)
```



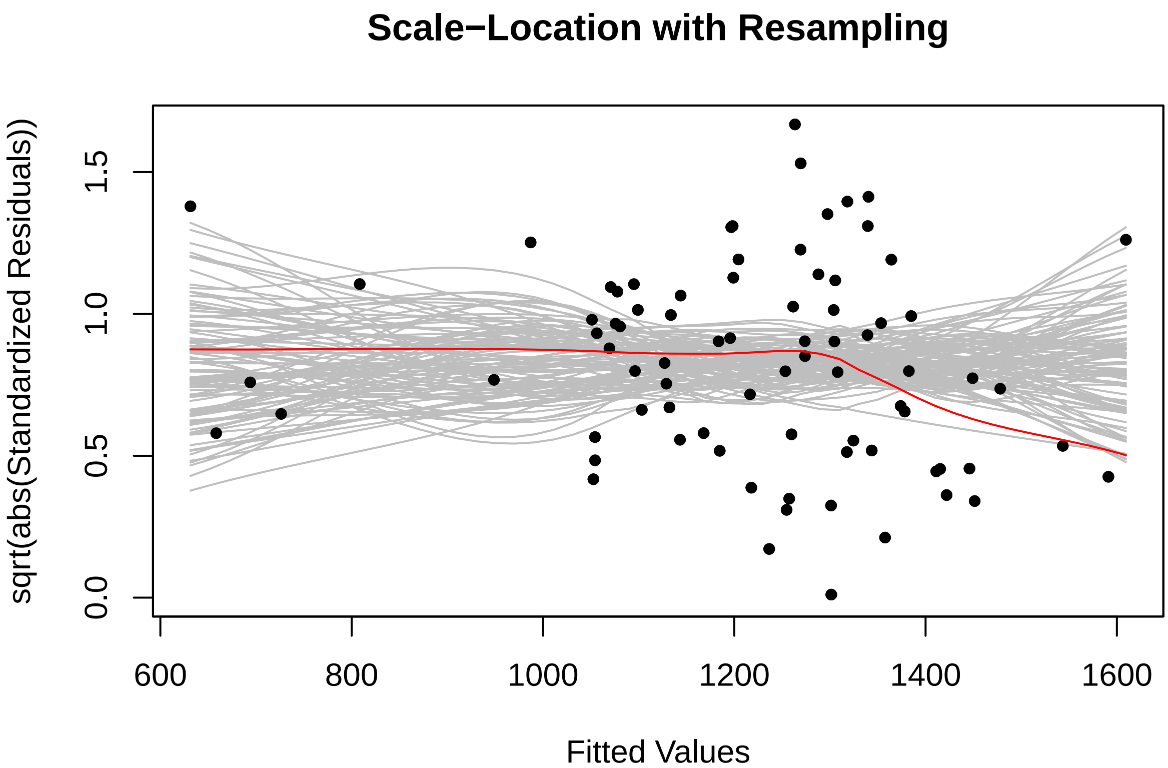

```
resplot(engy_model2, plots = 2)
```



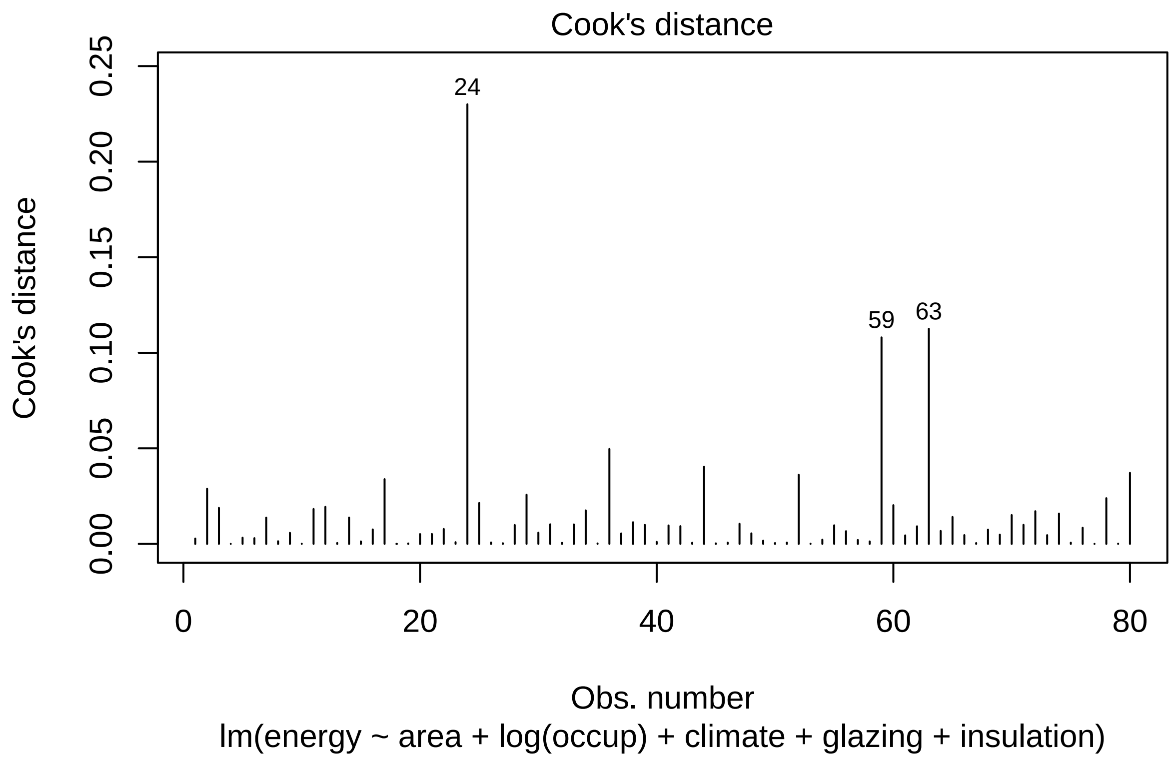

```r
## Scale-location plot
plot(engy_model2, which = 3)
```



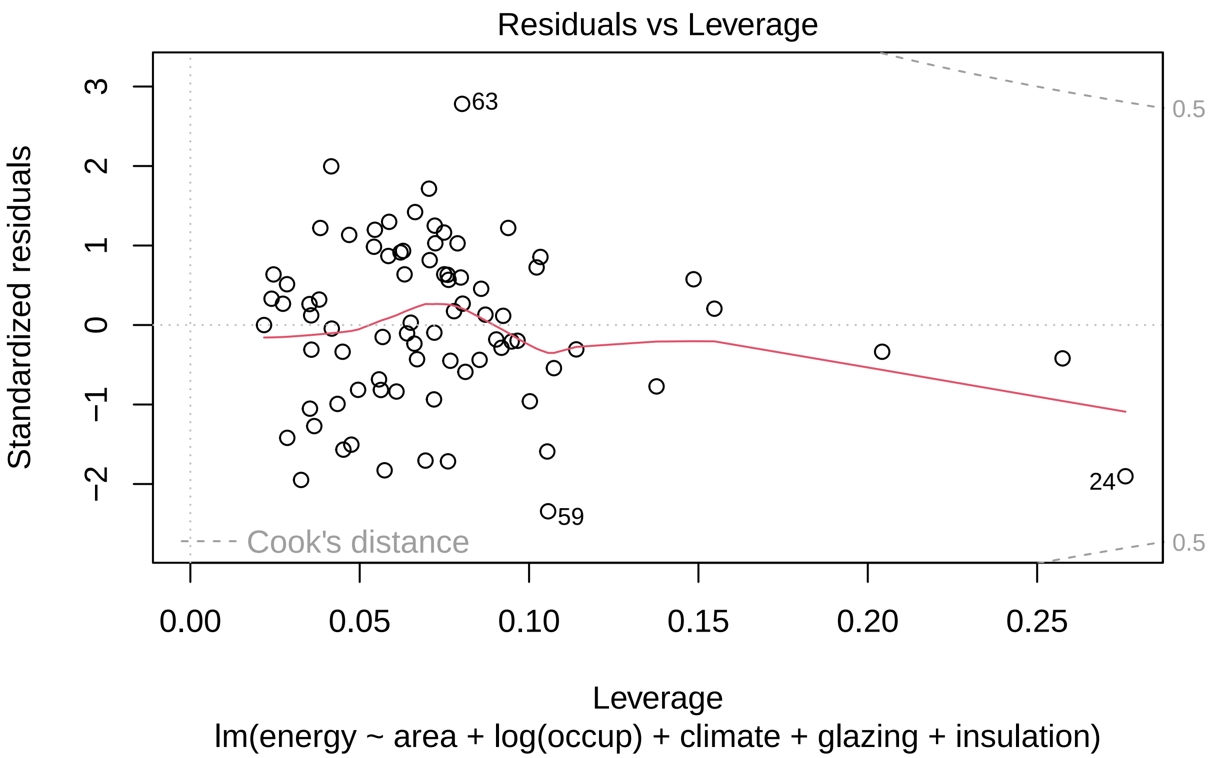

```r
resplot(engy_model2, plots = 3)
```



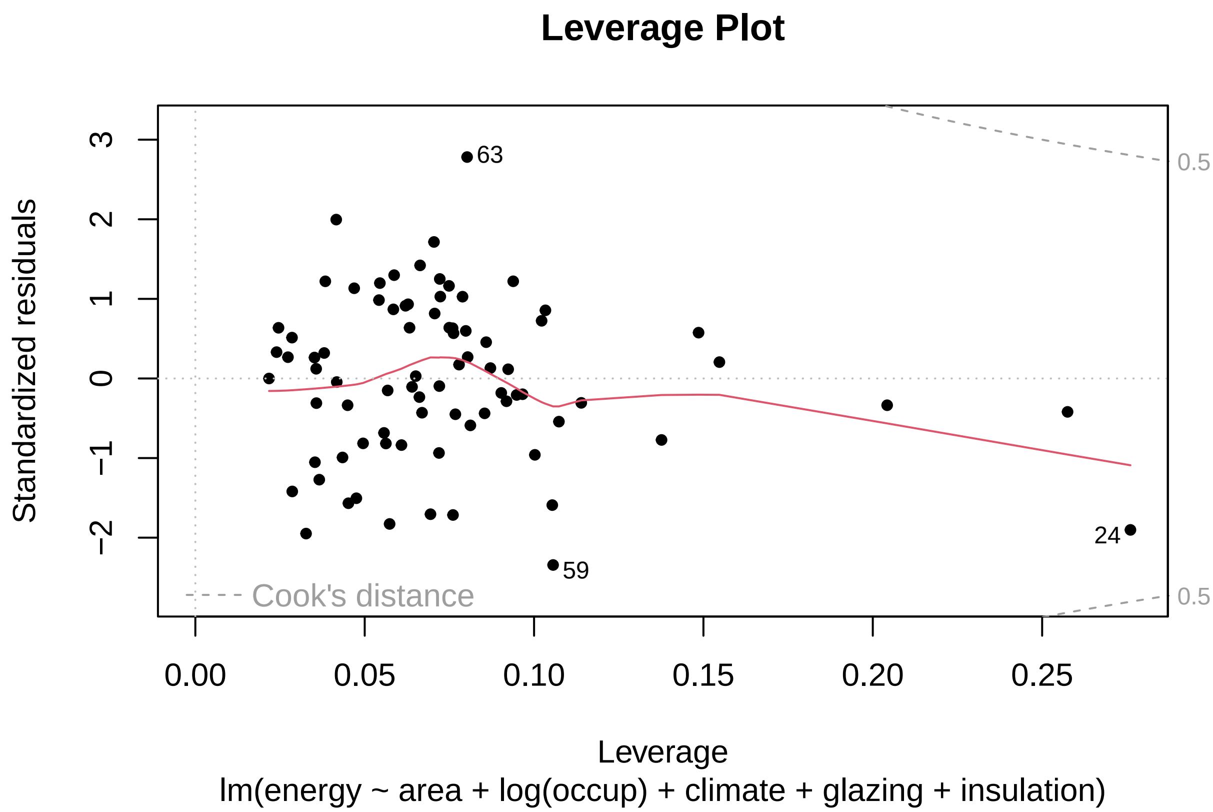

```r
## Cook's Distance plot
plot(engy_model2, which = 4)
```



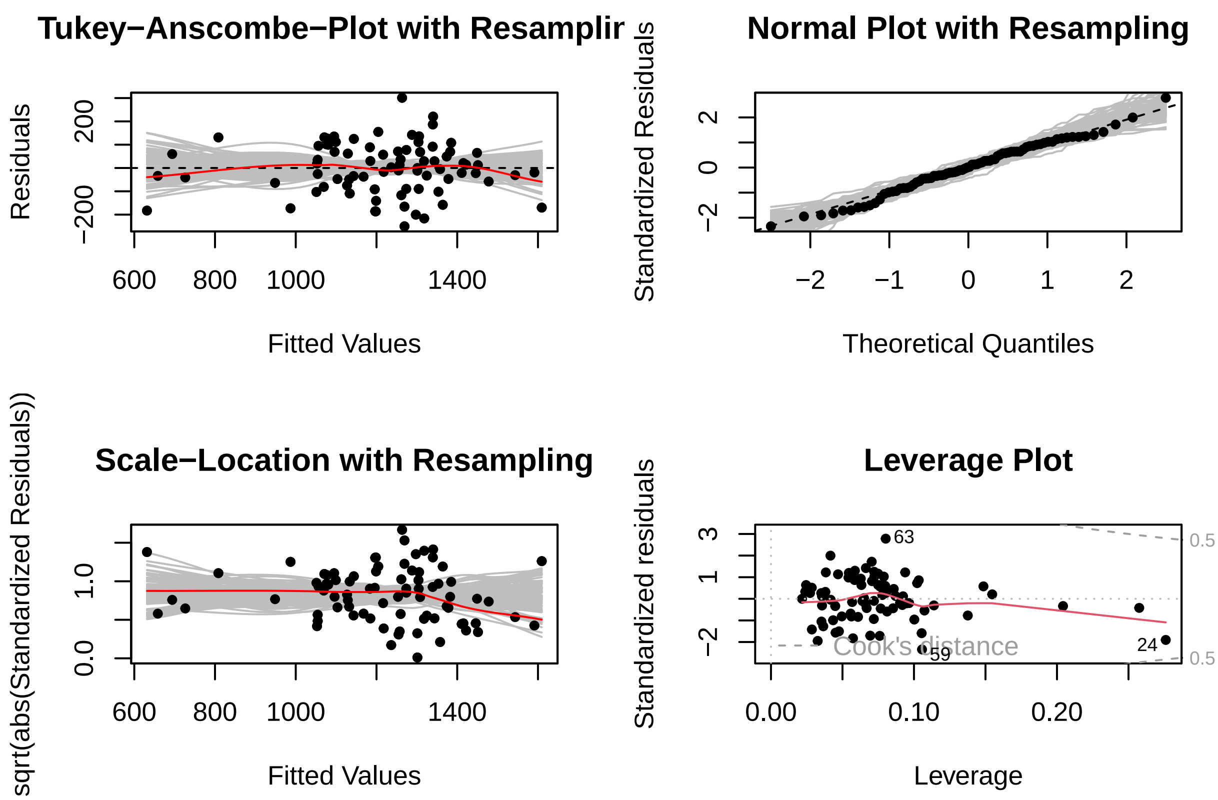

```
plot(engy_model2, which = 5)
```




```r
resplot(engy_model2, plots = 4)
```




```r
resplot(engy_model2)
```

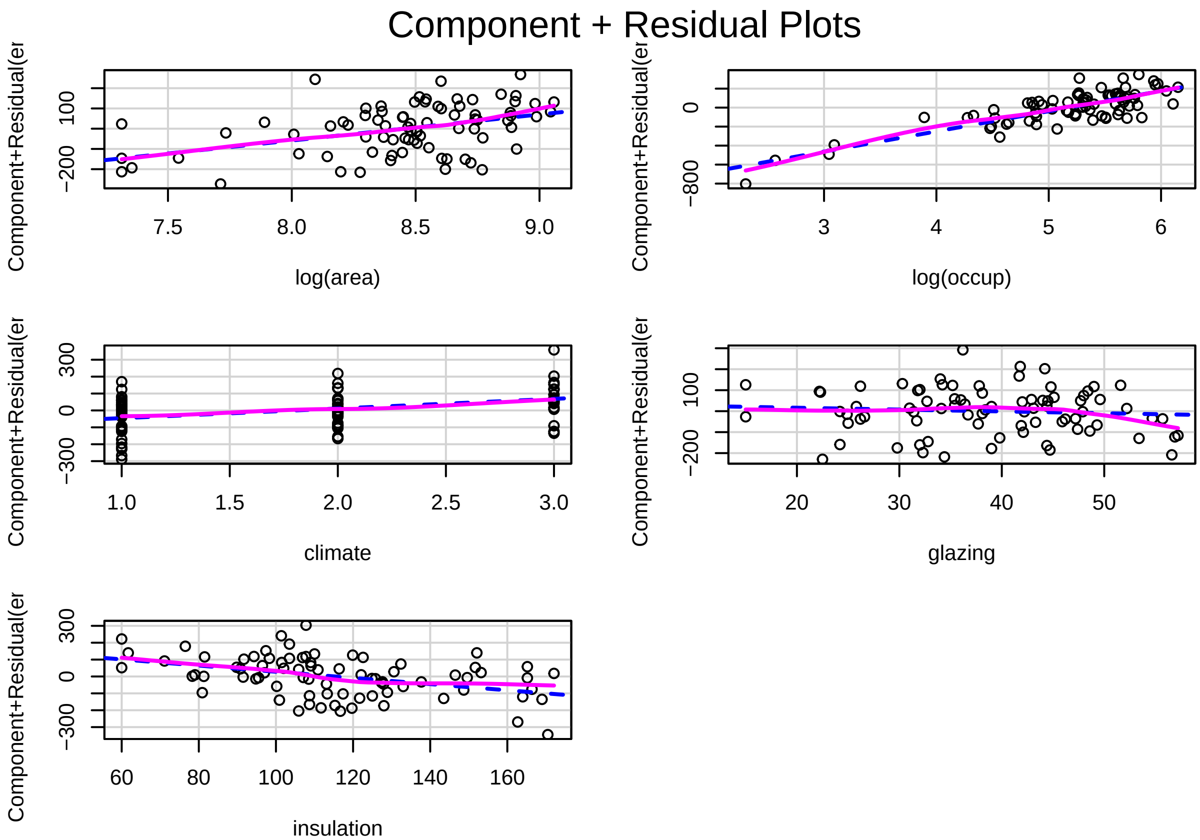



```r
# Transformed Model 2
engy_model3 <- lm(energy ~ log(area) + log(occup) + climate + glazing + insulation, data = e.consump)
summary(engy_model3)
```

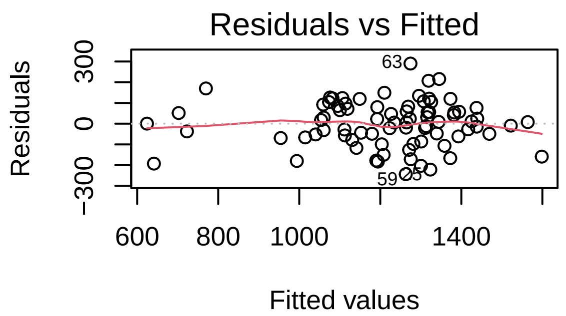
```
##
## Call:
## lm(formula = energy ~ log(area) + log(occup) + climate + glazing +
##     insulation, data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -243.595  -62.706    6.811   77.199  290.031
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -829.3248   295.0838  -2.810 0.006327 **
## log(area)    128.5629    38.1860   3.367 0.001209 **
## log(occup)   212.6420    20.3453  10.452 3.19e-16 ***
## climate       56.9442    16.6877   3.412 0.001047 **
## glazing       -0.8729     1.3548  -0.644 0.521390
## insulation    -1.8293     0.4790  -3.819 0.000276 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112.9 on 74 degrees of freedom
```

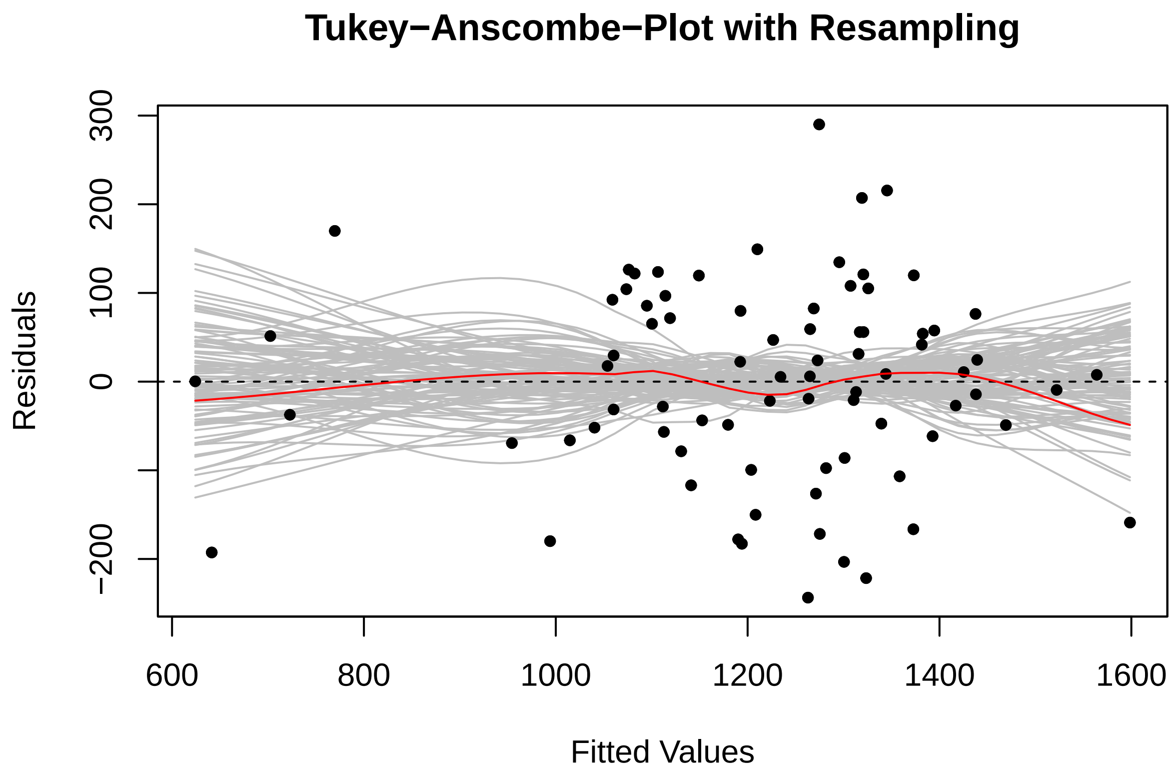
```
## Multiple R-squared:  0.7538, Adjusted R-squared:  0.7372
## F-statistic: 45.32 on 5 and 74 DF,  p-value: < 2.2e-16
```

```r
crPlots(engy_model3)
```

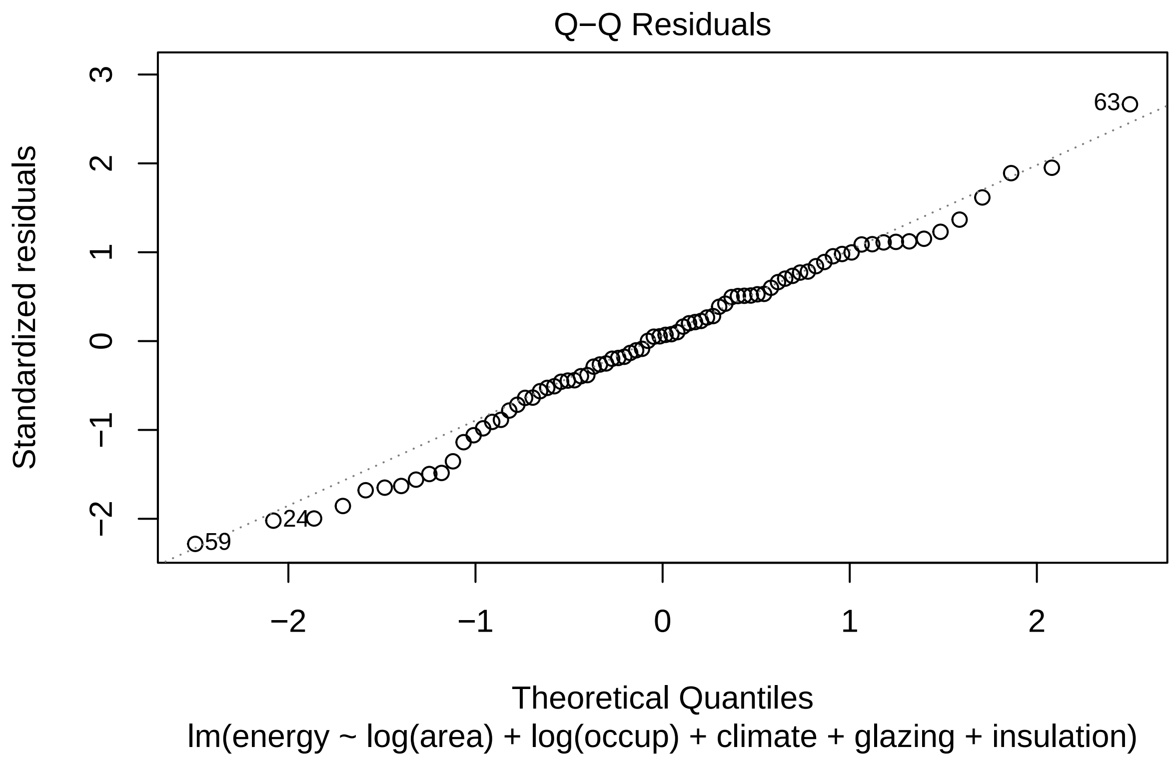



```r
## Residual analysis 3
plot(engy_model3, which=1)
resplot(engy_model3, plots = 1)
```

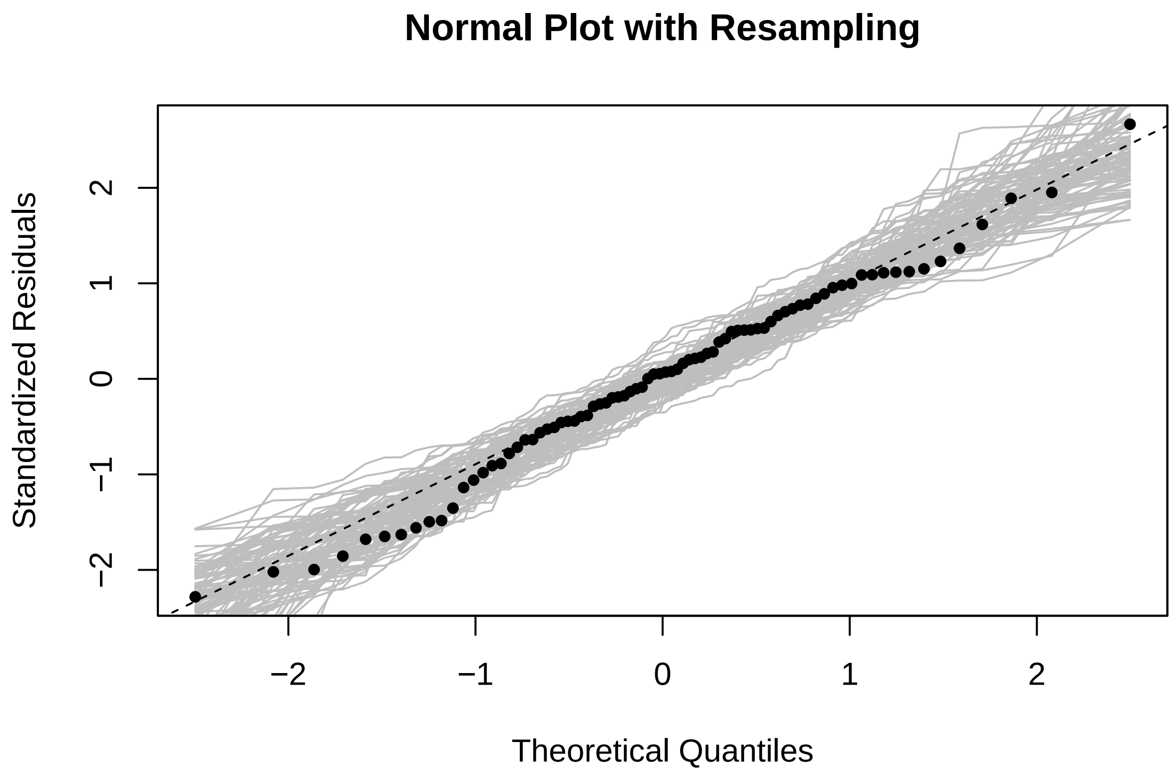

Residuals vs Fitted
Residuals
300
0
−300
600
800
1000
1400
Fitted values
63
59
25



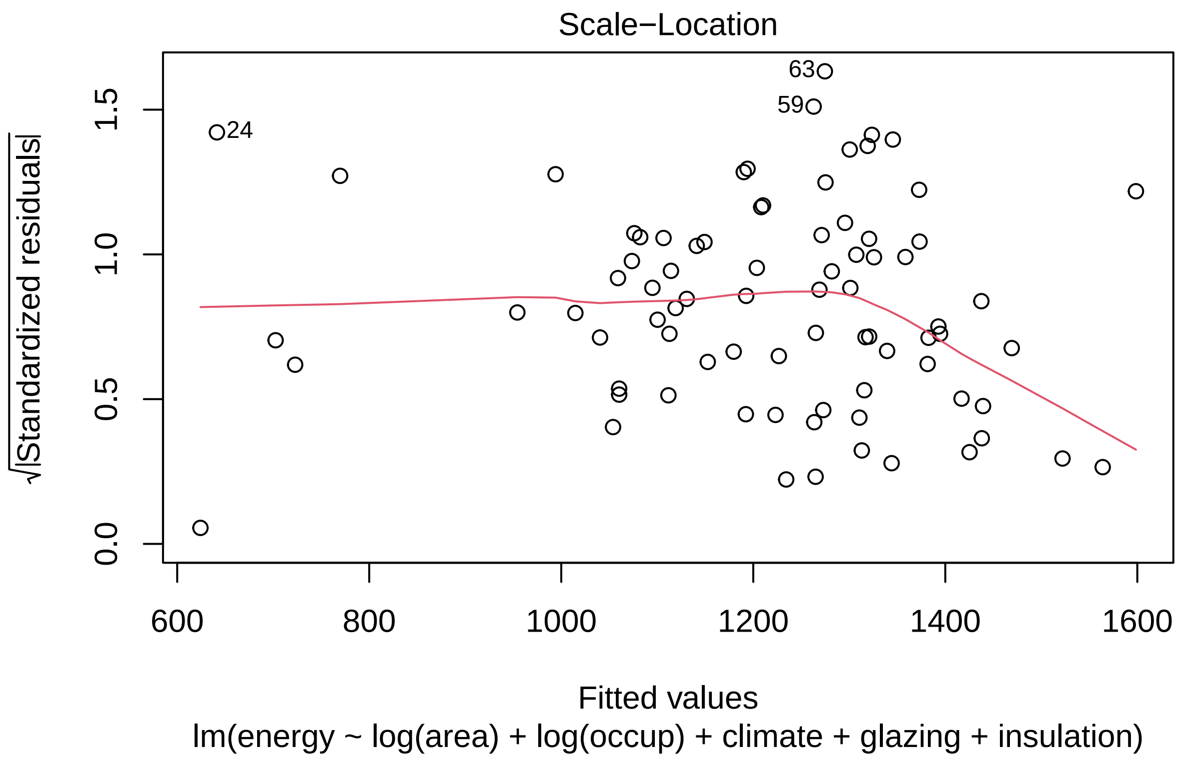

```r
plot(engy_model3, which = 2)
```



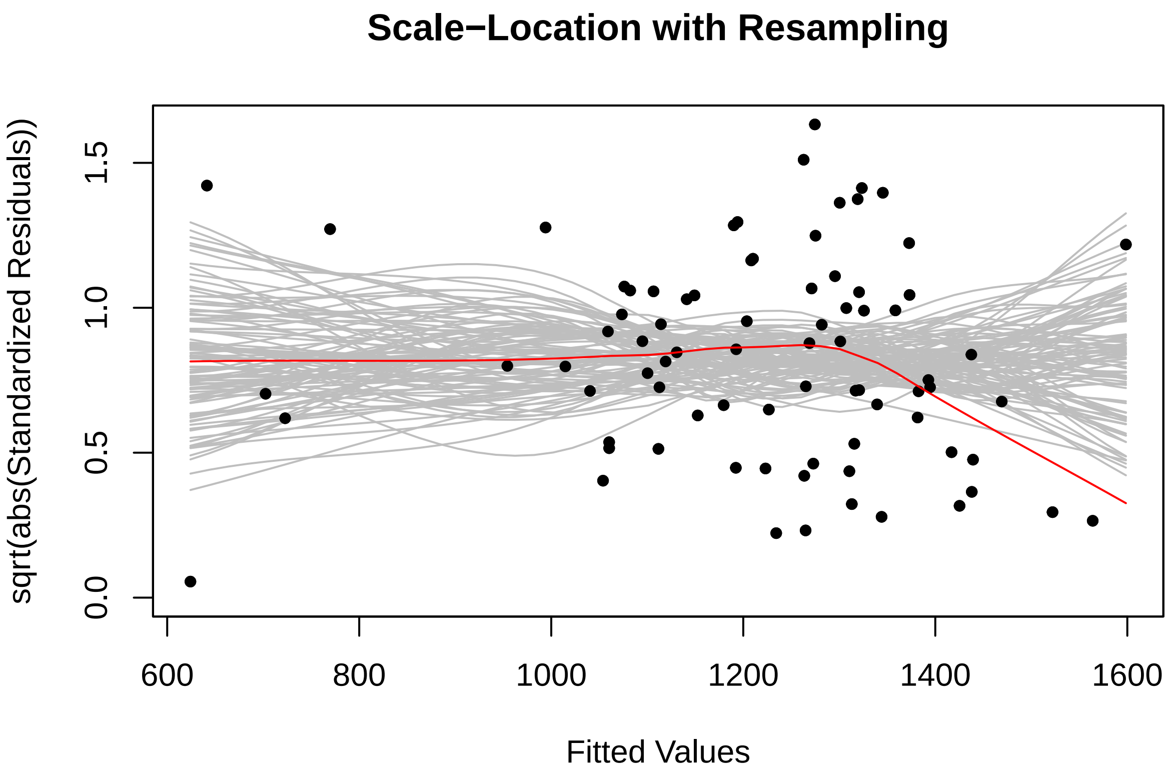

```
resplot(engy_model3, plots = 2)
```



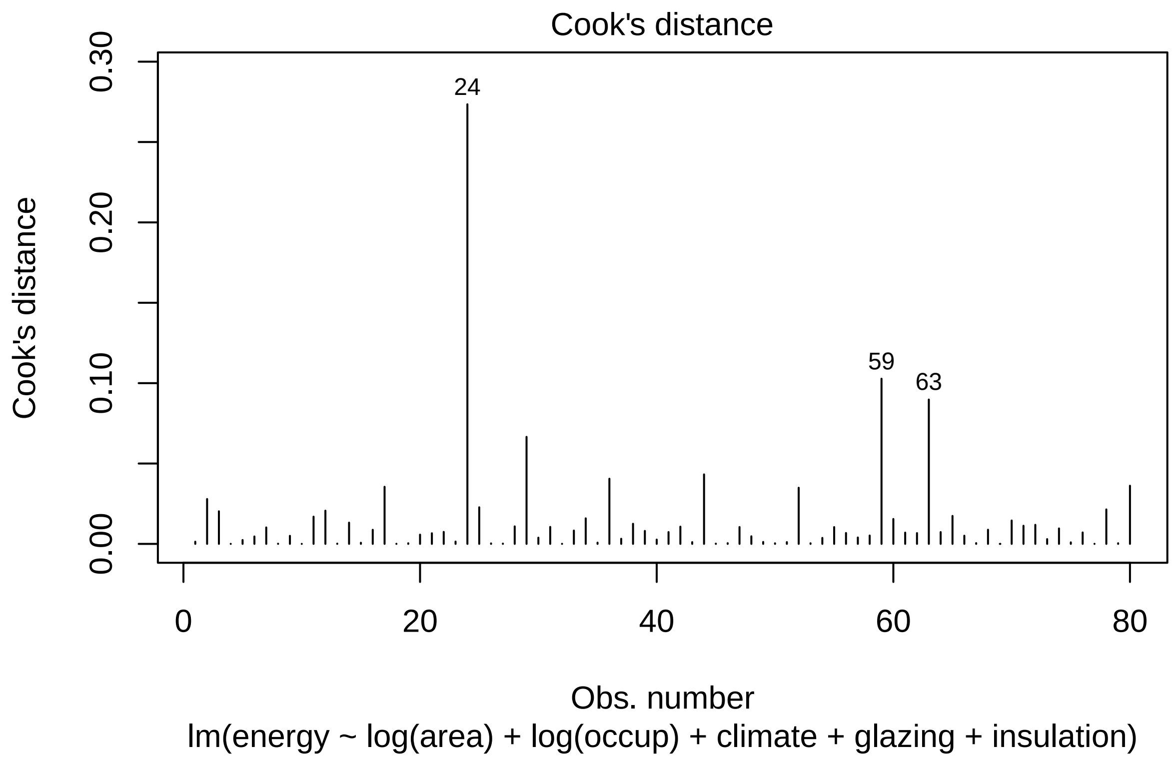

```
## Scale-location plot
plot(engy_model3, which = 3)
```



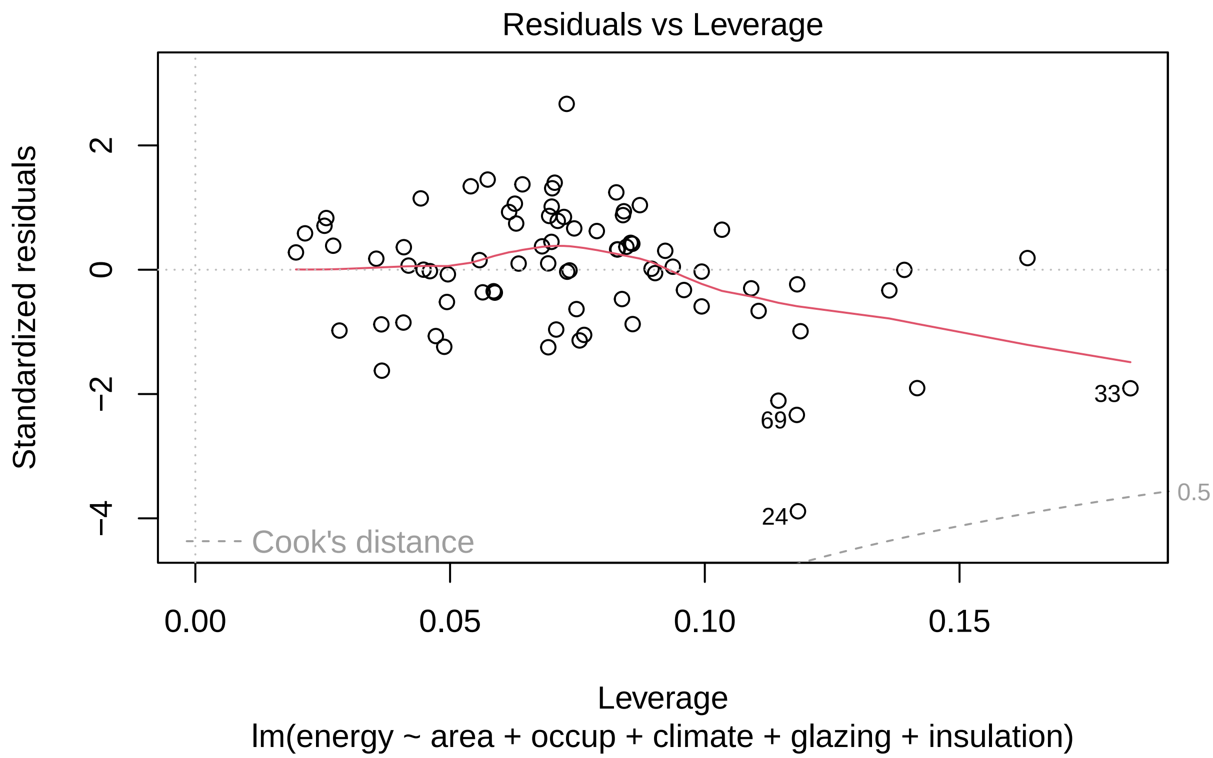

```
resplot(engy_model3, plots = 3)
```



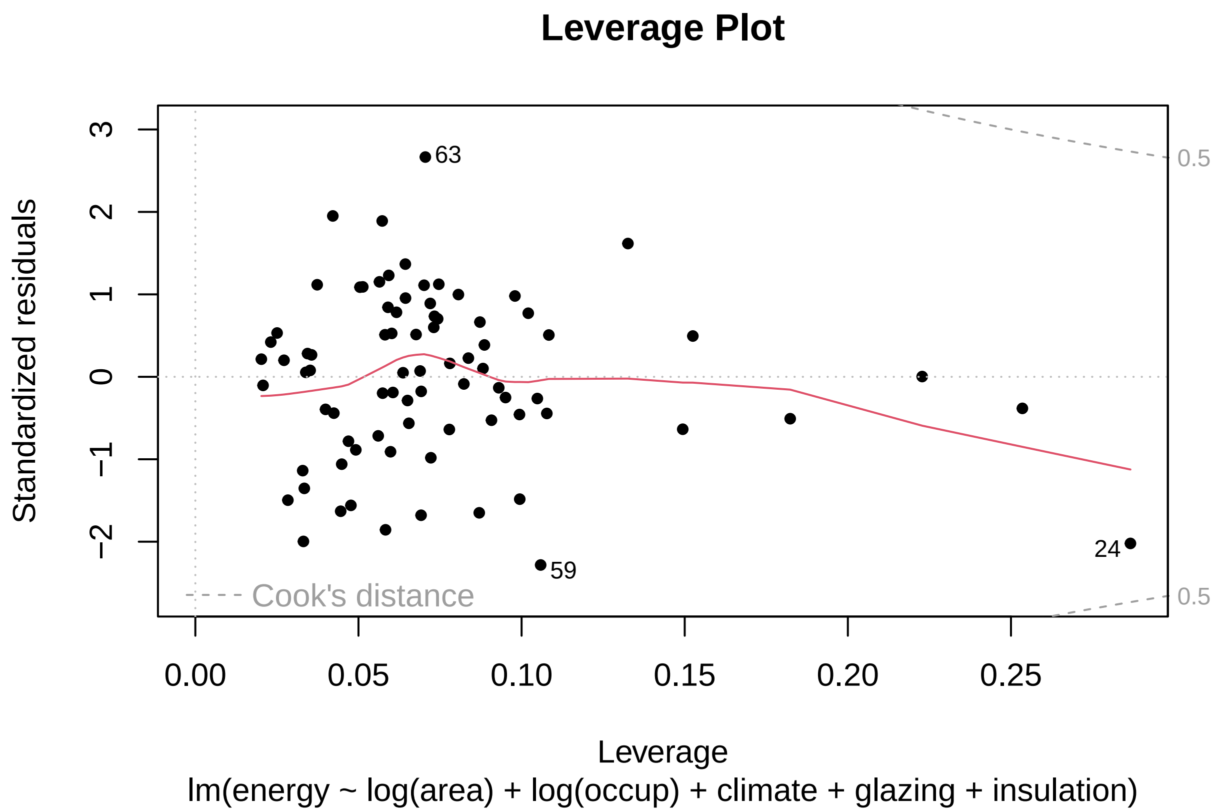

```r
## Cook's Distance plot
plot(engy_model3, which = 4)
```



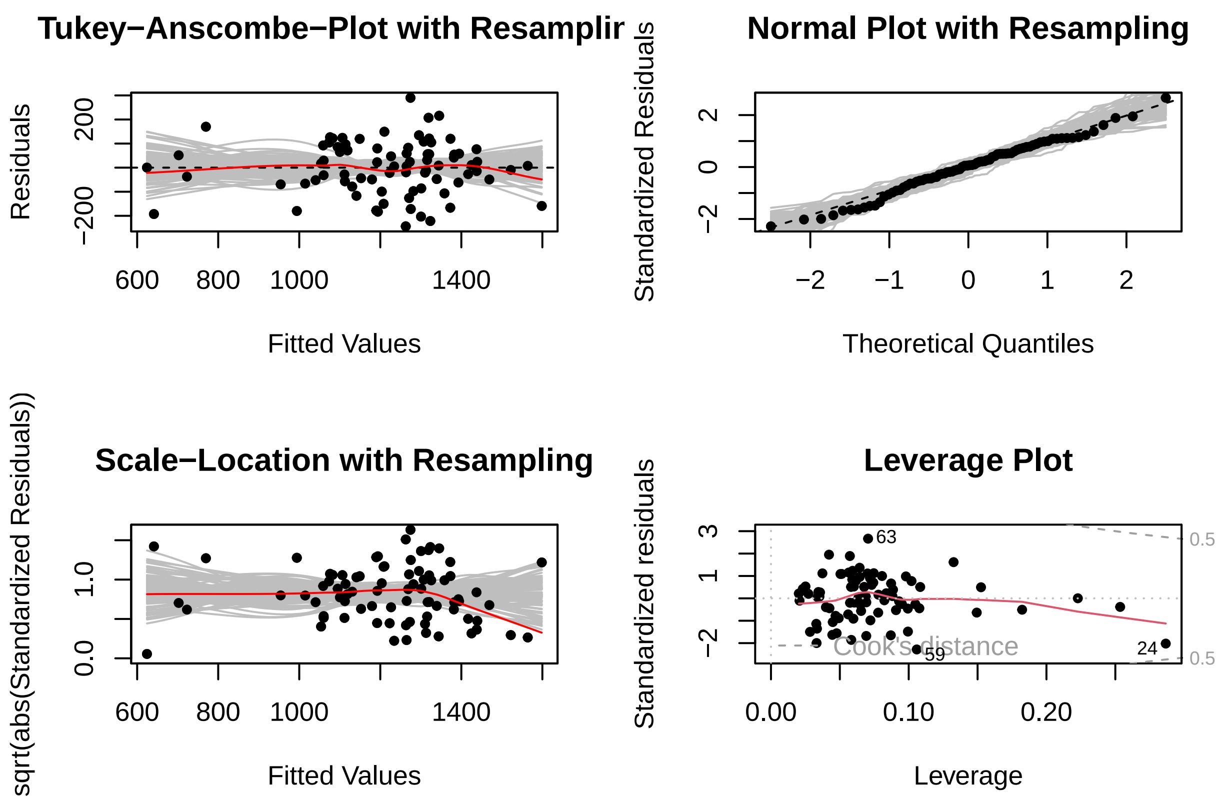

```
plot(engy_model, which = 5)
```




```r
resplot(engy_model3, plots = 4)
```




```r
resplot(engy_model3)
```



From the partial plots of the initial/original (engy_model) predictors, the variables area and occupancy clearly deviate from the dotted blue line which indicates non-linearity. In the first transformed model (engy_model2), the linearity of both variables are seen to improve. Also, the model diagnostics are better for this transformed model. From the Adjusted R2, this model also fits the data better (0.7371 > 0.5774) than the original/initial model.

In the second transformed model (engy_model3), the variable linearity, residual plots and model fit are better than both the original and the first transformed model (engy_model2).

Therefore, this model is taken as the most appropriate.

```
# Part c) Variable Selection starting with the transformed model

# Backward Elimination with AIC
engy.back <- stats::step(engy_model3, direction="backward")
```

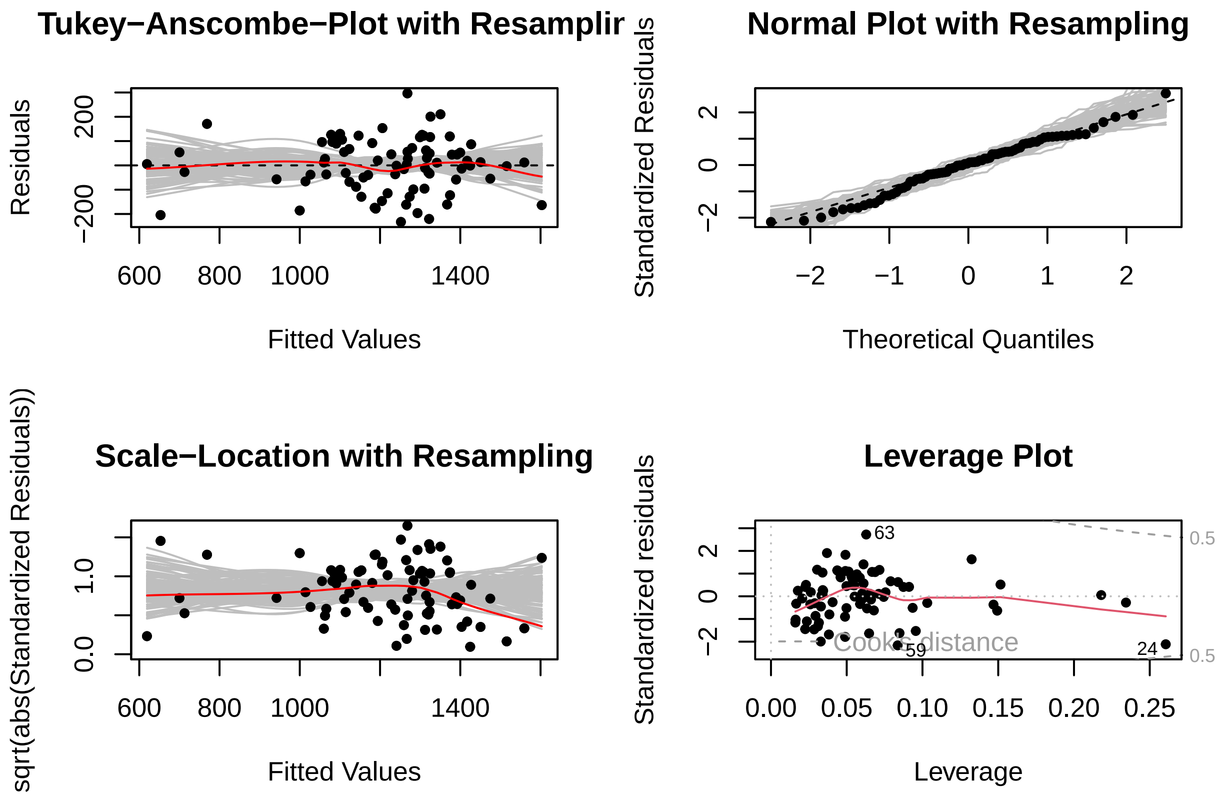
```
## Start:  AIC=761.97
## energy ~ log(area) + log(occup) + climate + glazing + insulation
##
##               Df Sum of Sq     RSS    AIC
## - glazing      1      5288  948090 760.41
## <none>                      942802 761.97
## - log(area)    1    144415 1087217 771.37
## - climate      1    148352 1091154 771.66
## - insulation   1    185826 1128628 774.36
## - log(occup)   1   1391746 2334548 832.50
##
```

```
## Step:  AIC=760.41
## energy ~ log(area) + log(occup) + climate + insulation
##
##              Df Sum of Sq     RSS    AIC
## <none>                     948090 760.41
## - climate     1    143097 1091187 769.66
## - log(area)   1    159606 1107696 770.86
## - insulation  1    185718 1133808 772.73
## - log(occup)  1   1394266 2342356 830.77
```

```r
summary(engy.back)
```

```
##
## Call:
## lm(formula = energy ~ log(area) + log(occup) + climate + insulation,
##     data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -232.755  -60.103    8.202   75.275  296.391
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -897.3616   274.4641  -3.270 0.001628 **
## log(area)    132.9696    37.4216   3.553 0.000662 ***
## log(occup)   212.8157    20.2640  10.502  < 2e-16 ***
## climate       54.7563    16.2747   3.365 0.001211 **
## insulation    -1.8288     0.4771  -3.833 0.000261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112.4 on 75 degrees of freedom
## Multiple R-squared:  0.7524, Adjusted R-squared:  0.7392
## F-statistic: 56.99 on 4 and 75 DF,  p-value: < 2.2e-16
```

```r
resplot(engy.back)
```




```r
# AIC Stepwise Model Search: Both Directions Approach
# starting with the null model
engy_null <- lm(energy ~ 1, data = e.consump) # Intercept-only model
sc <- list(lower=engy_null, upper=engy_model3)
engy.b1 <- stats::step(engy_null, scope = sc, direction = "both")
```

```
## Start:  AIC=864.1
## energy ~ 1
##
##                 Df Sum of Sq     RSS    AIC
## + log(occup)   1   2409474 1420285 786.75
## + log(area)    1   1164697 2665062 837.10
## + insulation   1    108307 3721452 863.81
## <none>                     3829759 864.10
## + glazing      1     69682 3760078 864.63
## + climate      1     59375 3770385 864.85
##
## Step:  AIC=786.75
## energy ~ log(occup)
##
##                 Df Sum of Sq     RSS    AIC
## + climate      1    181658 1238627 777.80
## + insulation   1    140139 1280146 780.44
## + log(area)    1    127714 1292571 781.21
## <none>                     1420285 786.75
```

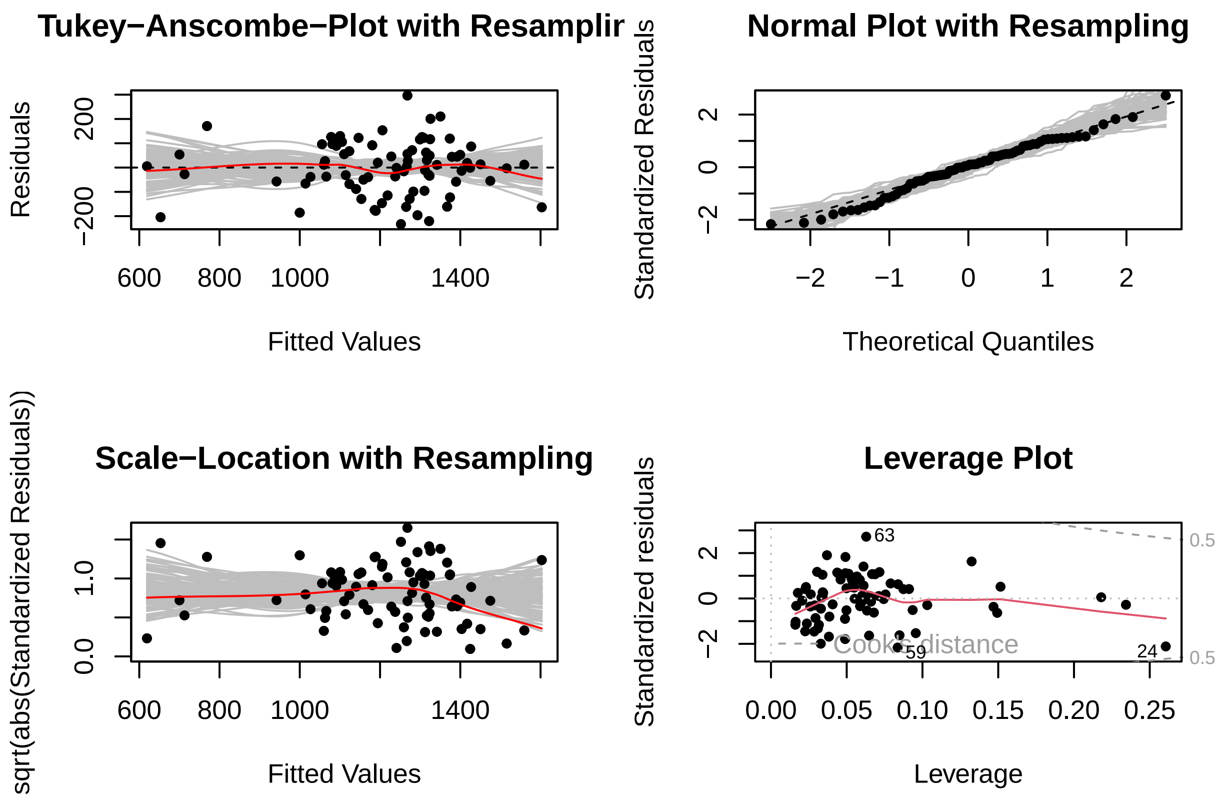
```
## + glazing     1      2292 1417993 788.62
## - log(occup)  1   2409474 3829759 864.10
##
## Step:  AIC=777.8
## energy ~ log(occup) + climate
##
##              Df Sum of Sq     RSS    AIC
## + insulation  1    130931 1107696 770.86
## + log(area)   1    104819 1133808 772.73
## <none>                    1238627 777.80
## + glazing     1     16872 1221755 778.70
## - climate     1    181658 1420285 786.75
## - log(occup)  1   2531758 3770385 864.85
##
## Step:  AIC=770.86
## energy ~ log(occup) + climate + insulation
##
##              Df Sum of Sq     RSS    AIC
## + log(area)   1    159606  948090 760.41
## <none>                    1107696 770.86
## + glazing     1     20479 1087217 771.37
## - insulation  1    130931 1238627 777.80
## - climate     1    172450 1280146 780.44
## - log(occup)  1   2559478 3667174 864.63
##
## Step:  AIC=760.41
## energy ~ log(occup) + climate + insulation + log(area)
##
##              Df Sum of Sq     RSS    AIC
## <none>                     948090 760.41
## + glazing     1      5288  942802 761.97
## - climate     1    143097 1091187 769.66
## - log(area)   1    159606 1107696 770.86
## - insulation  1    185718 1133808 772.73
## - log(occup)  1   1394266 2342356 830.77
```

```r
summary(engy.b1)
```

```
##
## Call:
## lm(formula = energy ~ log(occup) + climate + insulation + log(area),
##     data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -232.755  -60.103    8.202   75.275  296.391
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -897.3616   274.4641  -3.270 0.001628 **
## log(occup)   212.8157    20.2640  10.502  < 2e-16 ***
## climate       54.7563    16.2747   3.365 0.001211 **
## insulation    -1.8288     0.4771  -3.833 0.000261 ***
## log(area)    132.9696    37.4216   3.553 0.000662 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112.4 on 75 degrees of freedom
## Multiple R-squared:  0.7524, Adjusted R-squared:  0.7392
## F-statistic: 56.99 on 4 and 75 DF,  p-value: < 2.2e-16
```

```r
resplot(engy.b1)
```




```r
# starting with the full model
engy.b2 <- stats::step(engy_model3, scope = sc, direction = "both")
```

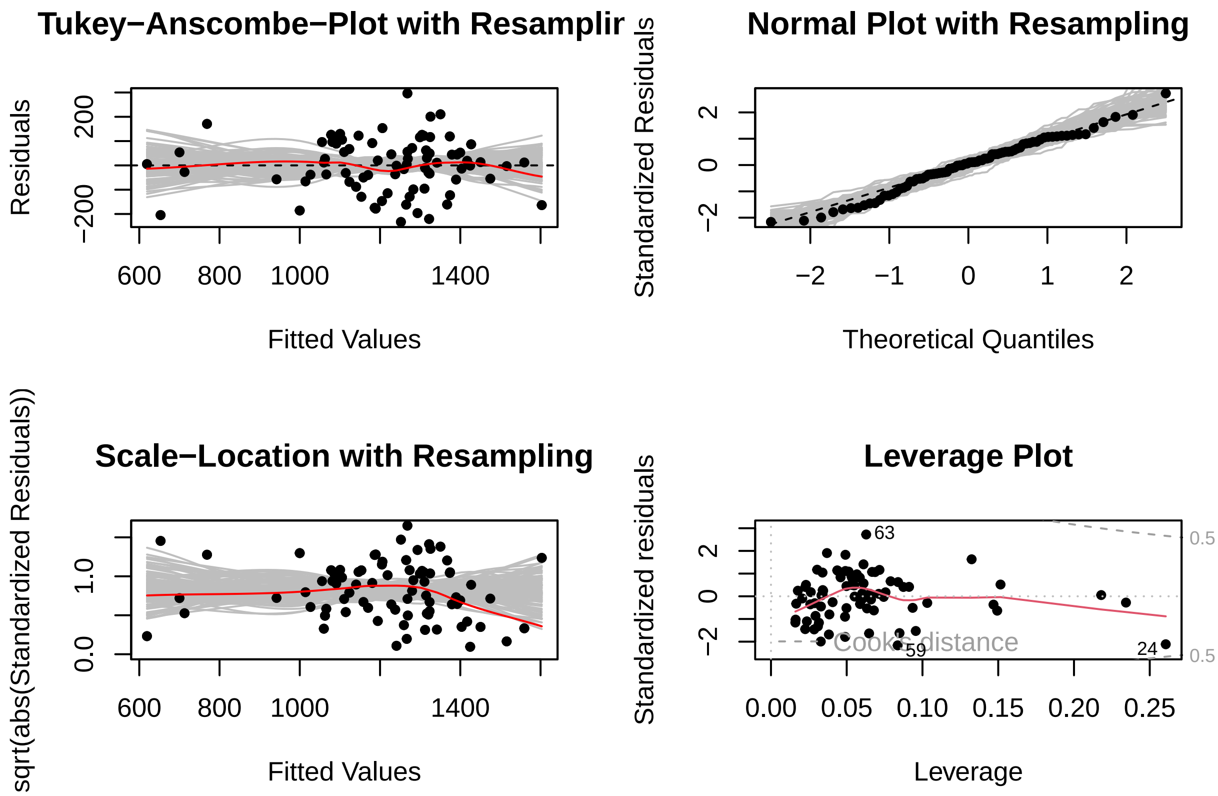
```
## Start:  AIC=761.97
## energy ~ log(area) + log(occup) + climate + glazing + insulation
##
##              Df Sum of Sq     RSS    AIC
## - glazing     1      5288  948090 760.41
## <none>                     942802 761.97
## - log(area)   1    144415 1087217 771.37
## - climate     1    148352 1091154 771.66
## - insulation  1    185826 1128628 774.36
## - log(occup)  1   1391746 2334548 832.50
##
## Step:  AIC=760.41
## energy ~ log(area) + log(occup) + climate + insulation
```

```
##
##              Df Sum of Sq     RSS    AIC
## <none>                     948090 760.41
## + glazing     1      5288  942802 761.97
## - climate     1    143097 1091187 769.66
## - log(area)   1    159606 1107696 770.86
## - insulation  1    185718 1133808 772.73
## - log(occup)  1   1394266 2342356 830.77
```

```r
summary(engy.b2)
```

```
##
## Call:
## lm(formula = energy ~ log(area) + log(occup) + climate + insulation,
##     data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -232.755  -60.103    8.202   75.275  296.391
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -897.3616   274.4641  -3.270 0.001628 **
## log(area)    132.9696    37.4216   3.553 0.000662 ***
## log(occup)   212.8157    20.2640  10.502  < 2e-16 ***
## climate       54.7563    16.2747   3.365 0.001211 **
## insulation    -1.8288     0.4771  -3.833 0.000261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112.4 on 75 degrees of freedom
## Multiple R-squared:  0.7524, Adjusted R-squared:  0.7392
## F-statistic: 56.99 on 4 and 75 DF,  p-value: < 2.2e-16
```

```r
resplot(engy.b2)
```




```
# starting with a model somewhere in the middle
engy.mid <- lm(energy ~  climate + glazing, data = e.consump)
engy.b3<- stats::step(engy.mid, scope = sc, direction = "both")
```

```
## Start:  AIC=864.66
## energy ~ climate + glazing
##
##                Df Sum of Sq     RSS    AIC
## + log(occup)    1   2446745 1221755 778.70
## + log(area)     1   1063969 2604532 839.26
## + insulation    1    111540 3556961 864.19
## - climate       1     91577 3760078 864.63
## <none>                      3668500 864.66
## - glazing       1    101884 3770385 864.85
##
## Step:  AIC=778.7
## energy ~ climate + glazing + log(occup)
##
##                Df Sum of Sq     RSS    AIC
## + insulation    1    134538 1087217 771.37
## + log(area)     1     93127 1128628 774.36
## - glazing       1     16872 1238627 777.80
## <none>                      1221755 778.70
## - climate       1    196238 1417993 788.62
## - log(occup)    1   2446745 3668500 864.66
```

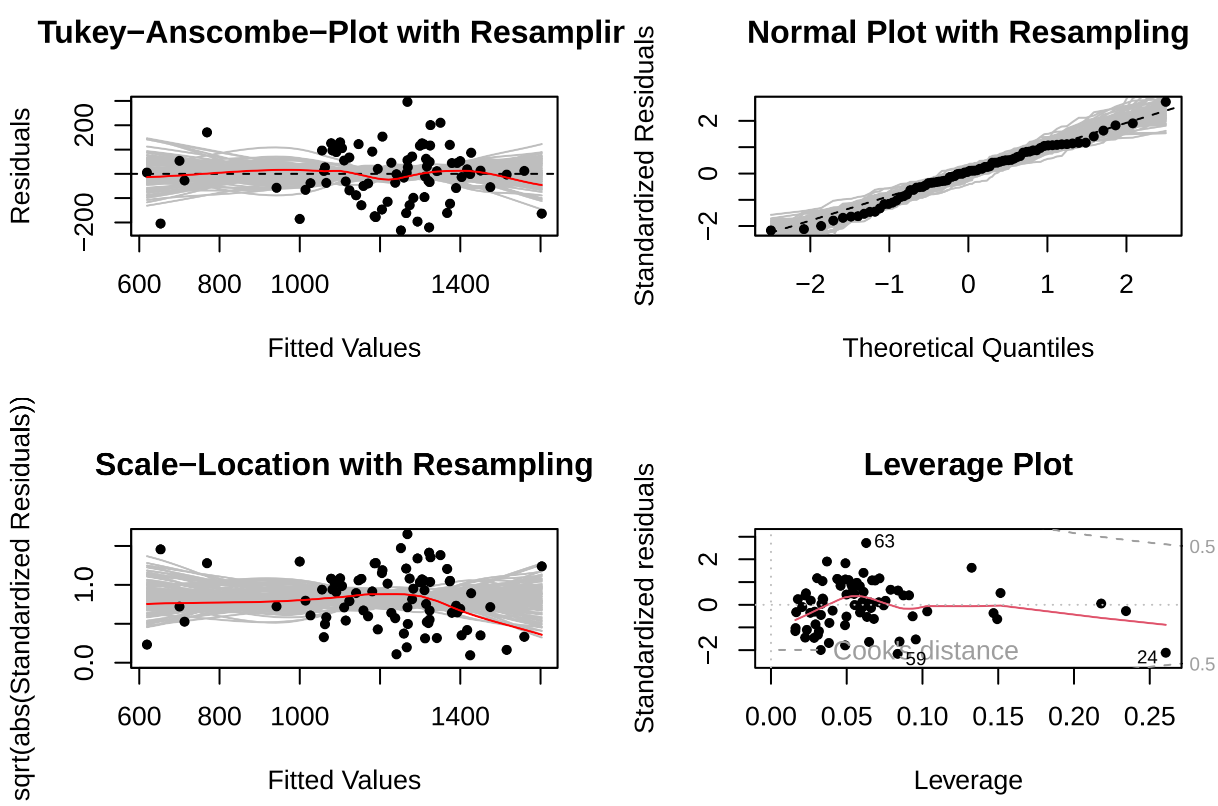
```
##
## Step:  AIC=771.37
## energy ~ climate + glazing + log(occup) + insulation
##
##               Df Sum of Sq     RSS    AIC
## + log(area)   1    144415  942802 761.97
## - glazing     1     20479 1107696 770.86
## <none>                    1087217 771.37
## - insulation  1    134538 1221755 778.70
## - climate     1    188917 1276134 782.19
## - log(occup)  1   2469744 3556961 864.19
##
## Step:  AIC=761.97
## energy ~ climate + glazing + log(occup) + insulation + log(area)
##
##               Df Sum of Sq     RSS    AIC
## - glazing     1      5288  948090 760.41
## <none>                     942802 761.97
## - log(area)   1    144415 1087217 771.37
## - climate     1    148352 1091154 771.66
## - insulation  1    185826 1128628 774.36
## - log(occup)  1   1391746 2334548 832.50
##
## Step:  AIC=760.41
## energy ~ climate + log(occup) + insulation + log(area)
##
##               Df Sum of Sq     RSS    AIC
## <none>                     948090 760.41
## + glazing     1      5288  942802 761.97
## - climate     1    143097 1091187 769.66
## - log(area)   1    159606 1107696 770.86
## - insulation  1    185718 1133808 772.73
## - log(occup)  1   1394266 2342356 830.77
```

```
summary(engy.b3)
```

```
##
## Call:
## lm(formula = energy ~ climate + log(occup) + insulation + log(area),
##     data = e.consump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -232.755  -60.103    8.202   75.275  296.391
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -897.3616   274.4641  -3.270 0.001628 **
## climate       54.7563    16.2747   3.365 0.001211 **
## log(occup)   212.8157    20.2640  10.502  < 2e-16 ***
## insulation    -1.8288     0.4771  -3.833 0.000261 ***
## log(area)    132.9696    37.4216   3.553 0.000662 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 112.4 on 75 degrees of freedom
## Multiple R-squared:  0.7524, Adjusted R-squared:  0.7392
## F-statistic: 56.99 on 4 and 75 DF,  p-value: < 2.2e-16
```

```r
resplot(engy.b3)
```



In all the reduced models from applying variable selection (i.e., engy.back, engy.b1, engy.b2 and engy.b3), the variable glazing was dropped. There are no major improvements in residual plots for all the models. Also, no noticeable changes on predictor significance or model fit.

```r
# Part d) 5-fold cross validation
set.seed(123) # Set seed for reproducibility
n <- nrow(e.consump) # Number of observations and folds
k <- 5 # Number of folds
sb <- round(seq(0, n, length = (k + 1)))  # Fold boundaries

# Initialize vectors to store MSPE for each model
mspe_full <- numeric(k)
mspe_reduced <- numeric(k)

# 5-fold cross-validation for full model (engy_model3)
for (i in 1:k) {
  test <- (sb[k + 1 - i] + 1):sb[k + 2 - i]
  train <- (1:n)[-test]
  fit_full <- lm(energy ~ log(area) + log(occup) + climate + glazing + insulation, data = e.consump[tra
```

```r
  pred_full <- predict(fit_full, newdata = e.consump[test, ])
  mspe_full[i] <- mean((e.consump$energy[test] - pred_full)^2, na.rm = FALSE)
}

# 5-fold cross-validation for reduced model (dropping glazing)
for (i in 1:k) {
  test <- (sb[k + 1 - i] + 1):sb[k + 2 - i]  # Same fold split for comparability
  train <- (1:n)[-test]
  fit_reduced <- lm(energy ~ log(area) + log(occup) + climate + insulation, data = e.consump[train, ])
  pred_reduced <- predict(fit_reduced, newdata = e.consump[test, ])
  mspe_reduced[i] <- mean((e.consump$energy[test] - pred_reduced)^2, na.rm = FALSE)
}

# Calculate overall MSPE for each model
mspe_full_mean <- mean(mspe_full, na.rm = TRUE)
mspe_reduced_mean <- mean(mspe_reduced, na.rm = TRUE)

# Report results
cat("MSPE per fold for Full Model:", mspe_full, "\n")
```

```
## MSPE per fold for Full Model: 12804.69 23320.92 11287.79 18497.25 11118.67
```

```r
cat("MSPE per fold for Reduced Model:", mspe_reduced, "\n")
```

```
## MSPE per fold for Reduced Model: 12862.88 21751.54 11583.43 18541.62 11307.16
```

```r
cat("MSPE for Full Model:", mspe_full_mean, "\n")
```

```
## MSPE for Full Model: 15405.86
```

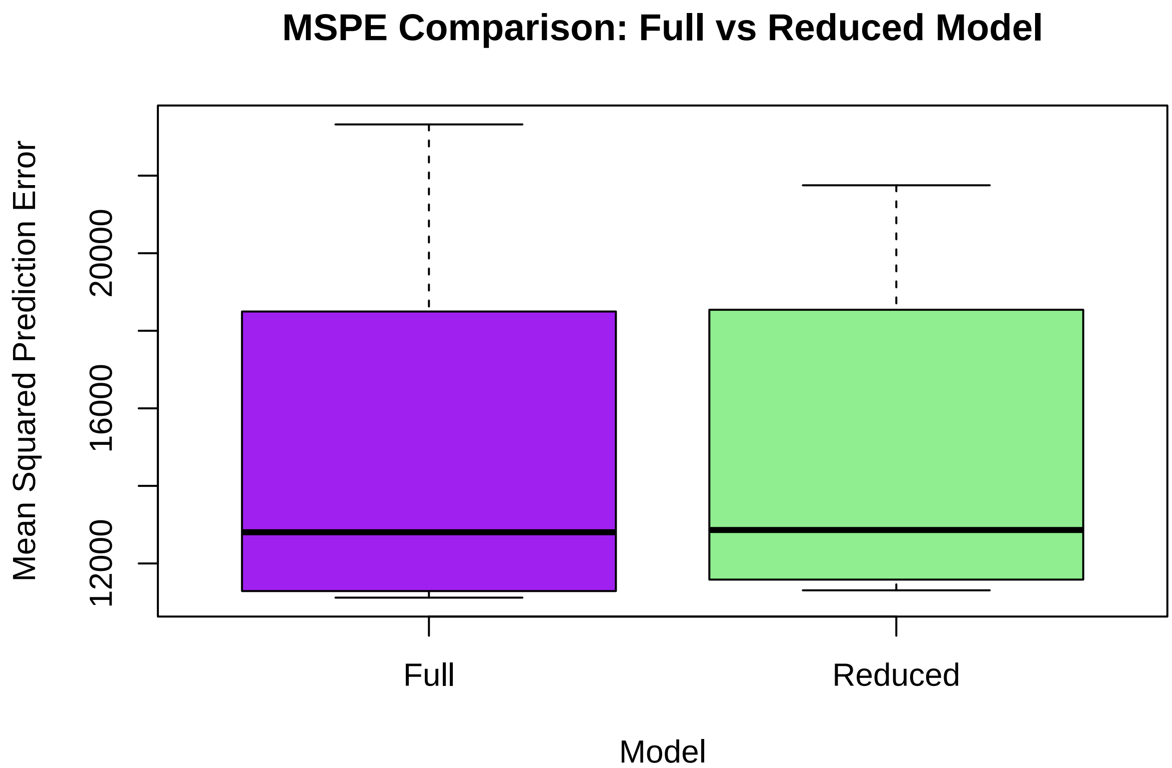
```r
cat("MSPE for Reduced Model:", mspe_reduced_mean, "\n")
```

```
## MSPE for Reduced Model: 15209.33
```

```r
# Checking relative increase in MSPE
relative_increase <- ((mspe_reduced_mean - mspe_full_mean) / mspe_full_mean) * 100
cat("Relative increase in MSPE (%):", relative_increase, "\n")
```

```
## Relative increase in MSPE (%): -1.275735
```

```r
# Box plots Using MSPEs
# Combining MSPEs into a data frame for plotting
mspe_data <- data.frame(
  MSPE = c(mspe_full, mspe_reduced),
  Model = factor(rep(c("Full", "Reduced"), each = k))
)
# Generating box plots
boxplot(MSPE ~ Model, data = mspe_data,
        main = "MSPE Comparison: Full vs Reduced Model",
        ylab = "Mean Squared Prediction Error",
        col = c("purple", "lightgreen"),
        border = "black")
```



From the cross-validation exercise, The MSPE for the reduced model is less (-1.275735%) than the full model. Therefore, the variable glazing can be said to reduce the predictive power in model. Therefore, the reduced model is preferable for prediction purposes.