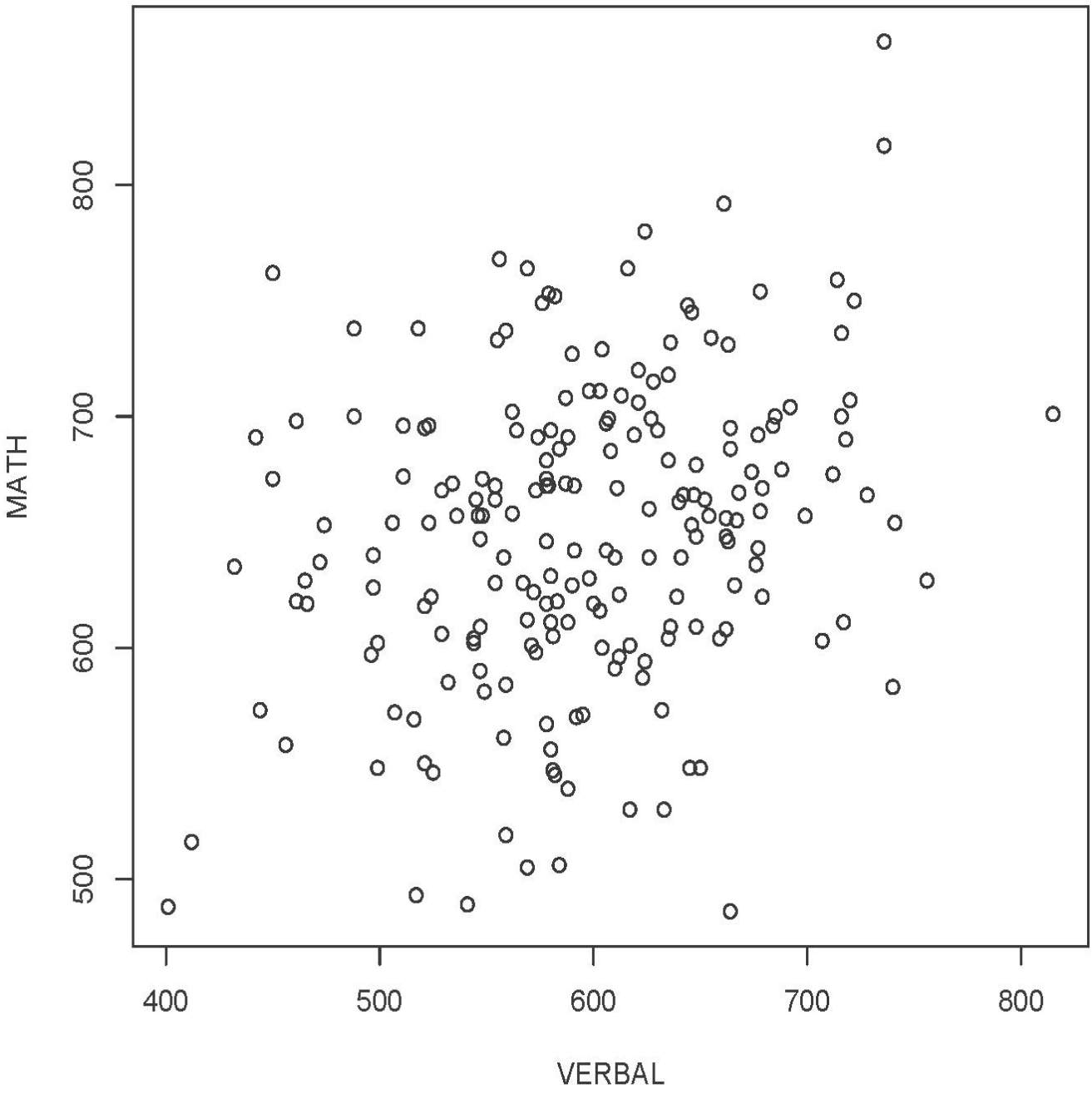
Regression diagnostics with R*

```
> sat = read.table("http://www.utstat.utoronto.ca/~brunner/data/legal/openSAT.data.txt")
> head(sat)
     VERBAL           MATH            GPA
 Min.   :401.0   Min.   :486.0   Min.   :0.690
 1st Qu.:548.8   1st Qu.:607.5   1st Qu.:2.288
 Median :590.5   Median :654.5   Median :2.635
 Mean   :595.7   Mean   :649.5   Mean   :2.630
 3rd Qu.:646.2   3rd Qu.:694.2   3rd Qu.:3.033
 Max.   :815.0   Max.   :862.0   Max.   :3.990
> mod1 = lm(GPA ~ VERBAL+MATH, data=sat); summary(mod1)

Call:
lm(formula = GPA ~ VERBAL + MATH, data = sat)

Residuals:
     Min       1Q   Median       3Q      Max
-2.24875 -0.35113  0.04659  0.38745  1.03527

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.6062975  0.4414062   1.374    0.171
VERBAL      0.0023072  0.0005522   4.178 4.42e-05 ***
MATH        0.0009999  0.0006093   1.641    0.102
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5484 on 197 degrees of freedom
Multiple R-squared: 0.1161,   Adjusted R-squared: 0.1071
F-statistic: 12.93 on 2 and 197 DF,  p-value: 5.284e-06

> attach(sat) # Make variable names accessible
>
> hii = hatvalues(mod1)
>
> summary(hii)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
0.005071 0.007824 0.011309 0.015000 0.019102 0.062370
> 2*(2+1)/100 # Twice mean hat value = 2 * (k+1)/n
[1] 0.06
```

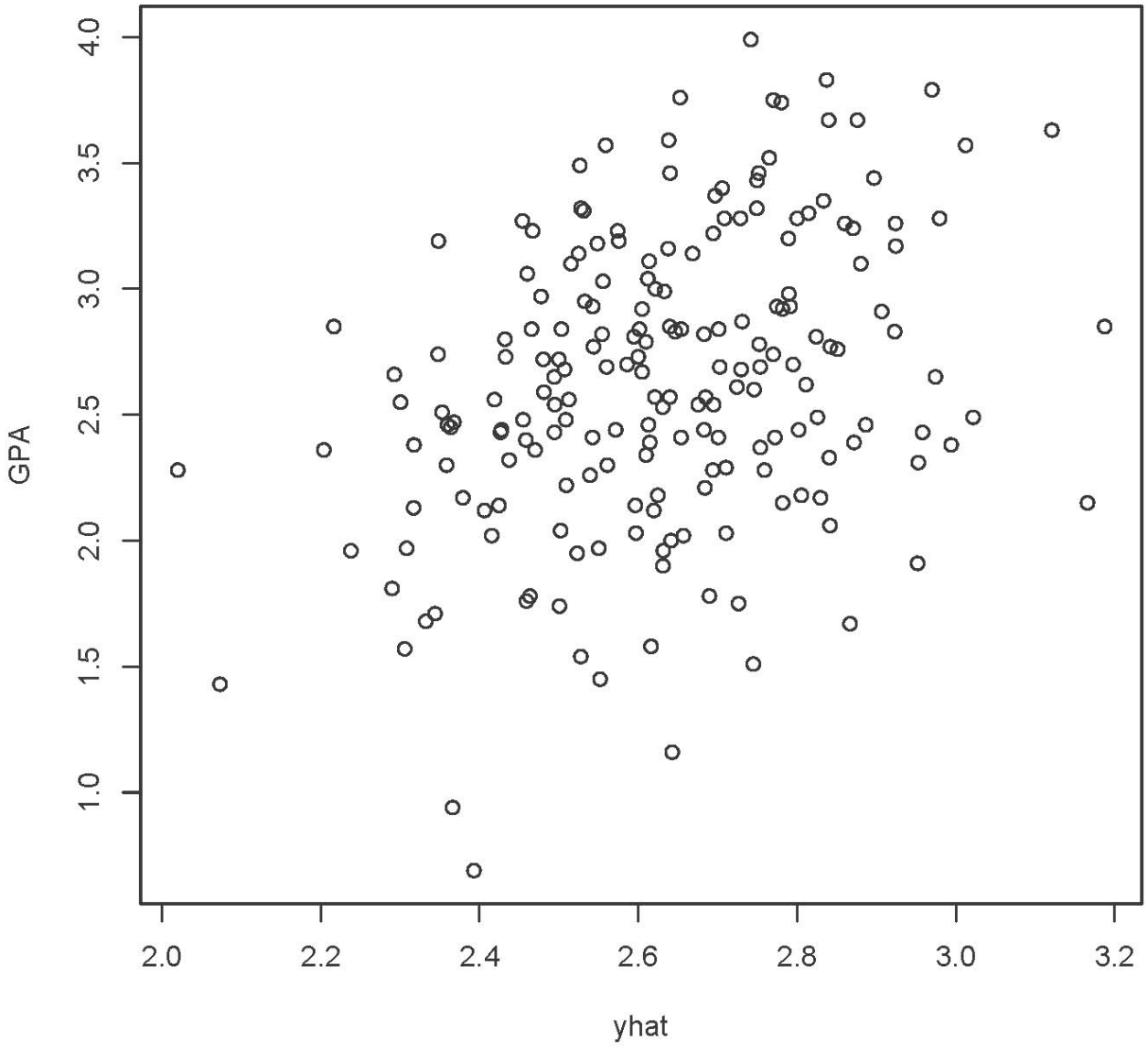
* Copyright information is on the last page.

```
> n = dim(sat)[1]; id = 1:n; id[hii>0.06]
[1] 105
> sat[105,]
    VERBAL MATH  GPA
105    736  862 2.15
> # Excellent SAT, but GPA not great. This is likely not a data error.
> attach(sat)
> plot(VERBAL,MATH)
```



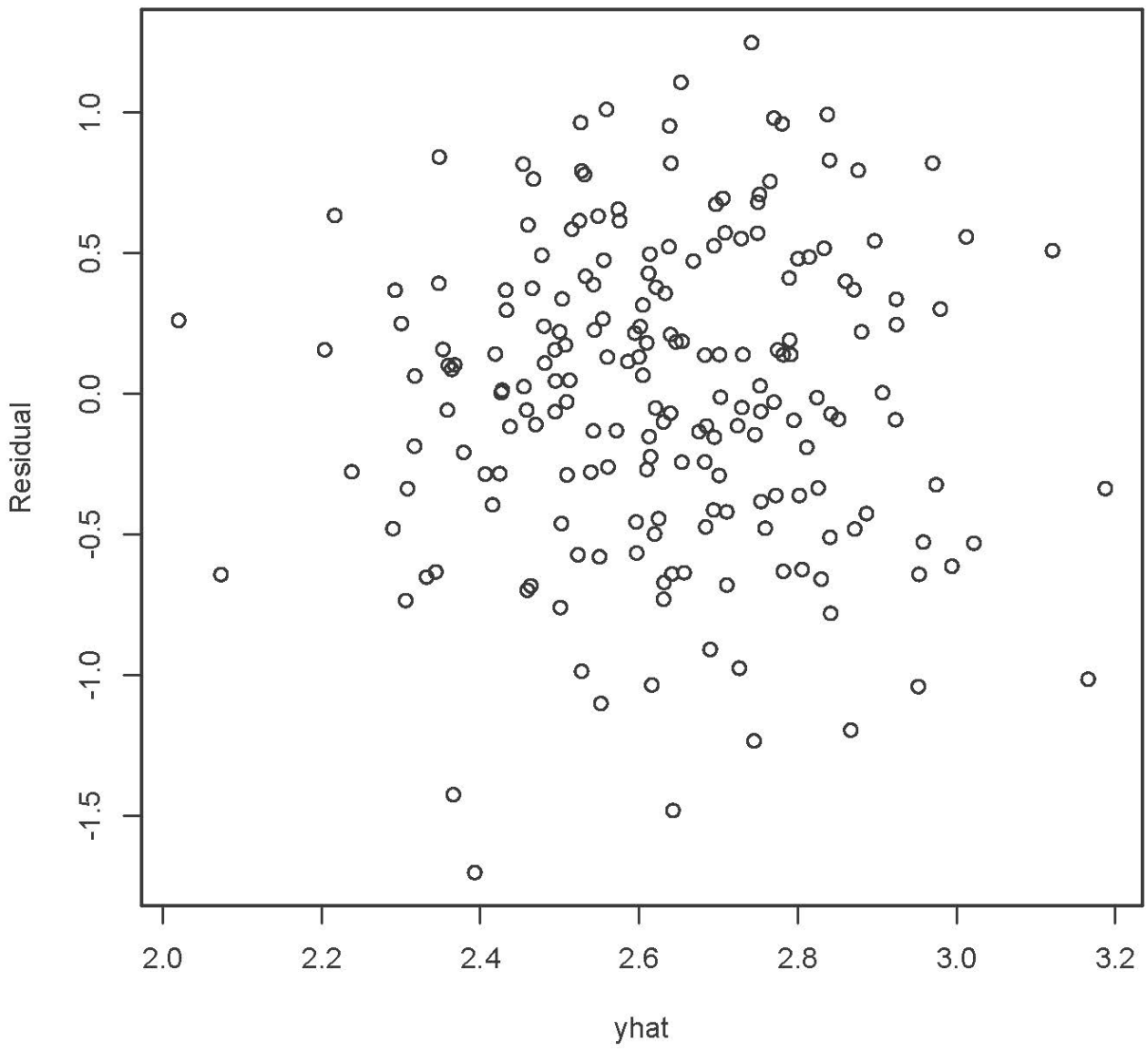

```
> # Keep an eye on Student 105
```

```
> # Plot y-hat versus y
> yhat = fitted.values(mod1)
> plot(yhat,GPA)
>
```



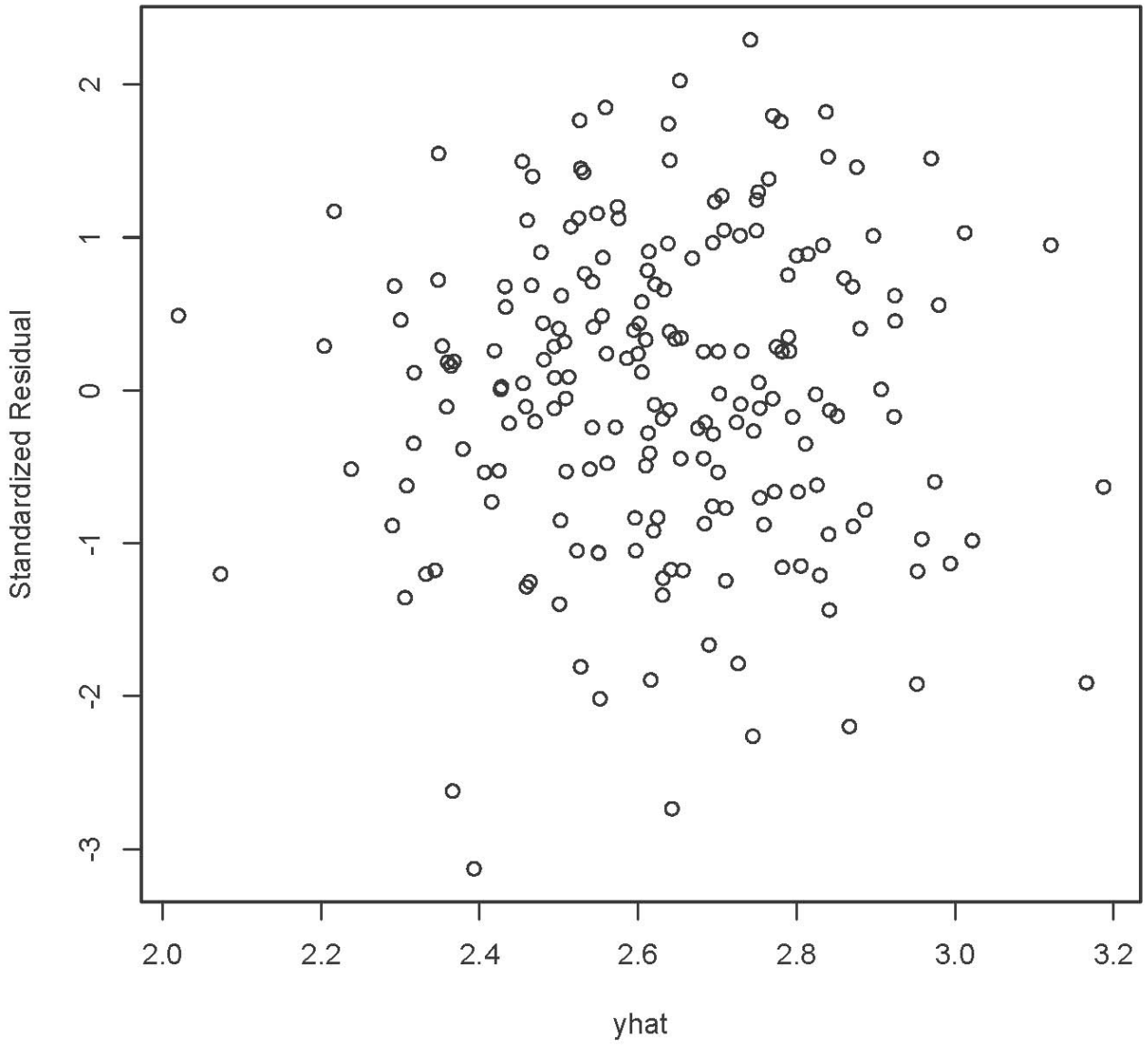

```
> cor(GPA,yhat)^2 # r-squared - R-squared
[1] 0.1160179
```

```
> # Plot y-hat versus residuals
> epsilonhat = residuals(mod1)
> plot(yhat,epsilonhat, ylab='Residual')
```



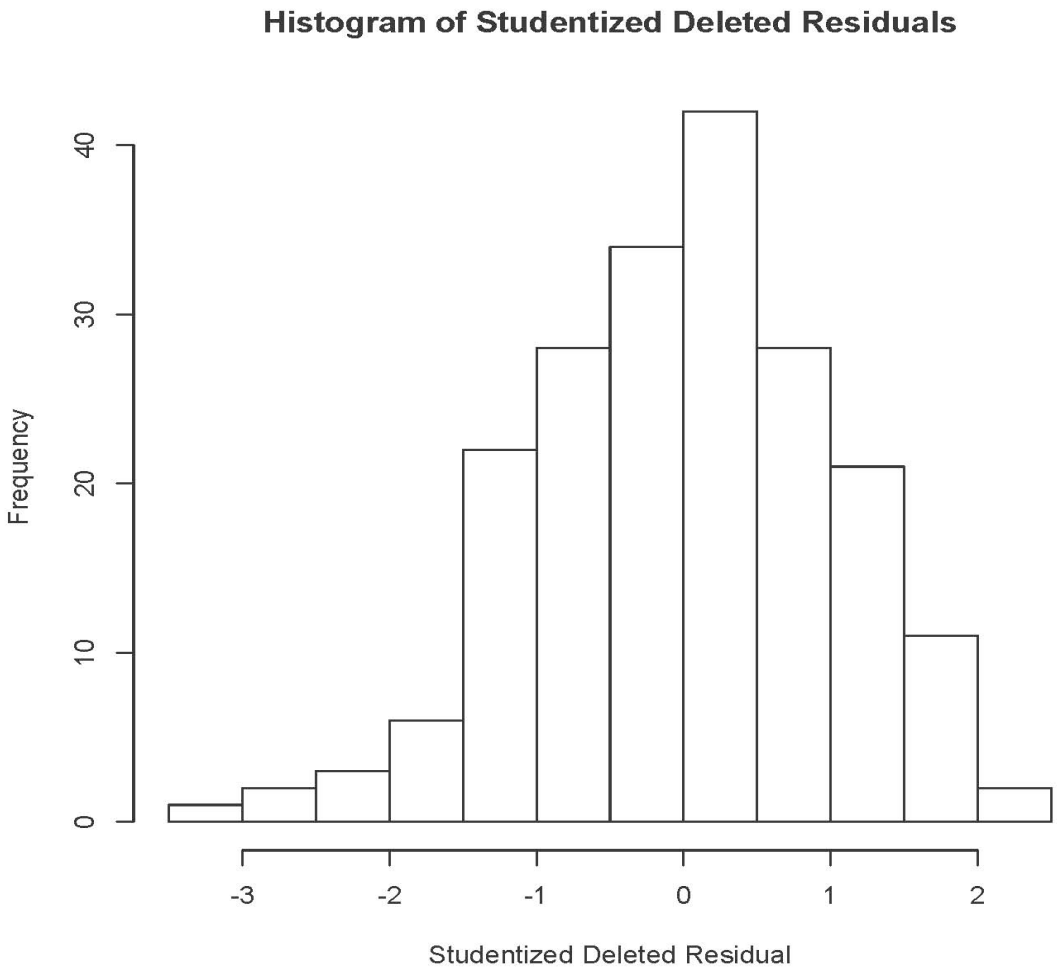

```
> cor(yhat,epsilonhat) # Zero
[1] 2.898153e-16
```

```
> # Compare plot of standardized residuals
> sr = rstandard(mod1)
> plot(yhat,sr,ylab='Standardized Residual')
```



```
> # Three look like possible outliers: Investigate
> suspect = id[sr < -2.5]
> cbind(sat[suspect,],yhat[suspect],sr[suspect])
   VERBAL MATH  GPA yhat[suspect] sr[suspect]
20    497  640 0.69      2.392965   -3.128203
51    645  548 1.16      2.642644   -2.735853
73    516  569 0.94      2.365985   -2.620870


> # Studentized deleted residuals are t-statistics
> sdr = rstudent(mod1) # Studentized deleted residuals
> # Bonferroni critical value for n=200 tests, at joint alpha = 0.05 level
> dfe = df.residual(mod1); dfe
[1] 197
> alpha = 0.05; a = alpha/200; bcrit = qt(1-a/2,dfe-1); bcrit
[1] 3.730706
> sdr[abs(sdr)>bcrit]
named numeric(0)
> summary(sdr)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-3.201000 -0.672800  0.048340 -0.001731  0.710900  2.317000
```

```
> # Do the Studentized deleted residuals look normal?
> hist(sdr,xlab='Studentized Deleted Residual',
+ main='Histogram of Studentized Deleted Residuals')
```

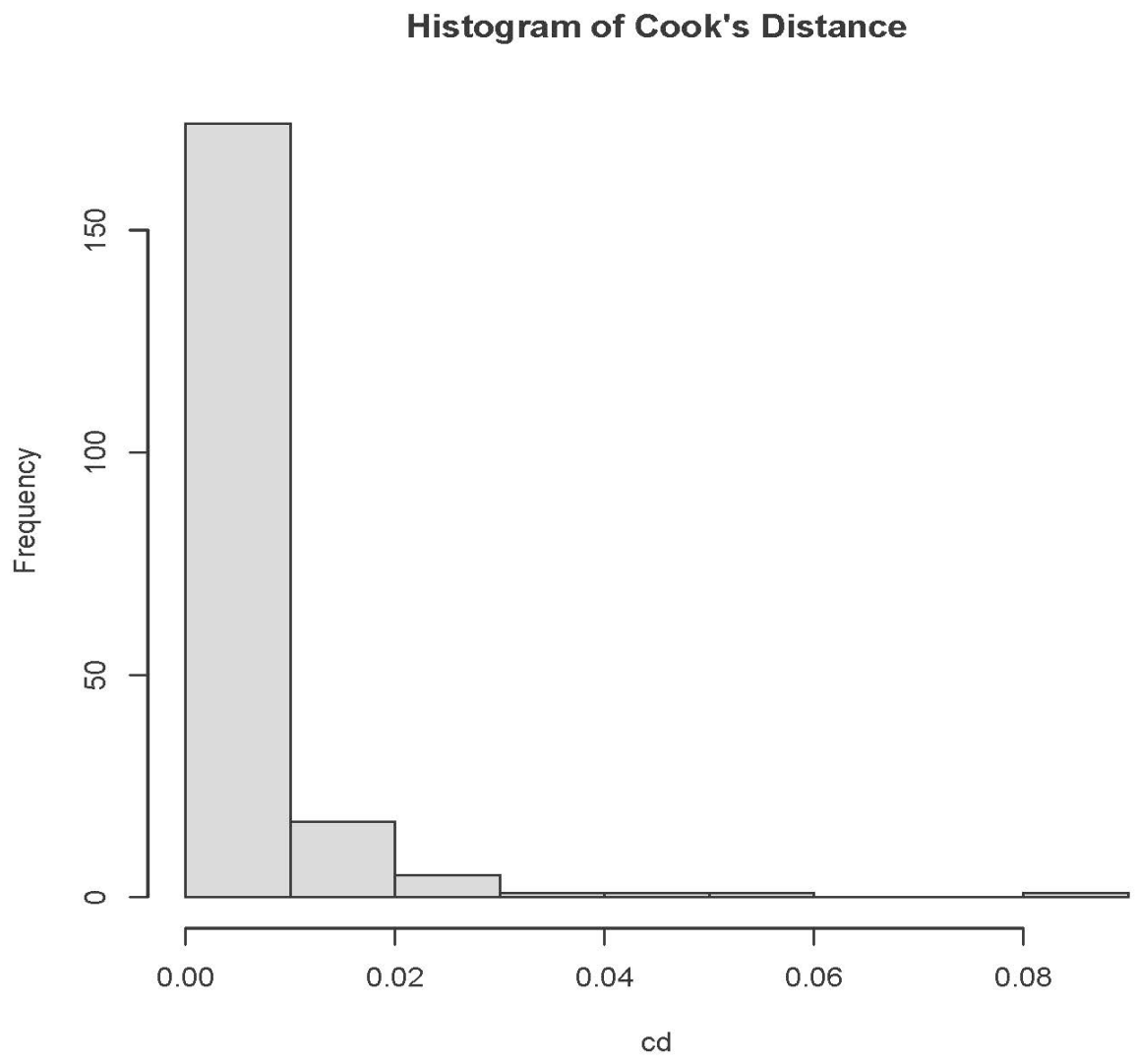



```
> shapiro.test(sdr) # Test for normality

     Shapiro-Wilk normality test

data:  sdr
W = 0.99032, p-value = 0.199
> # Not really needed: max(h_ii) < 0.2
> max(hii)
[1] 0.06236974
```

```
> # Cook's Distance
```

$$D_i = \frac{\sum_{i=1}^{n}(\hat{y}_i - \hat{y}_{(i)})^2}{MSE(k+1)}$$

```
> cd = cooks.distance(mod1); summary(cd)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
1.500e-07 4.459e-04 2.285e-03 4.993e-03 5.835e-03 8.114e-02
> hist(cd, main = "Histogram of Cook's Distance")
```

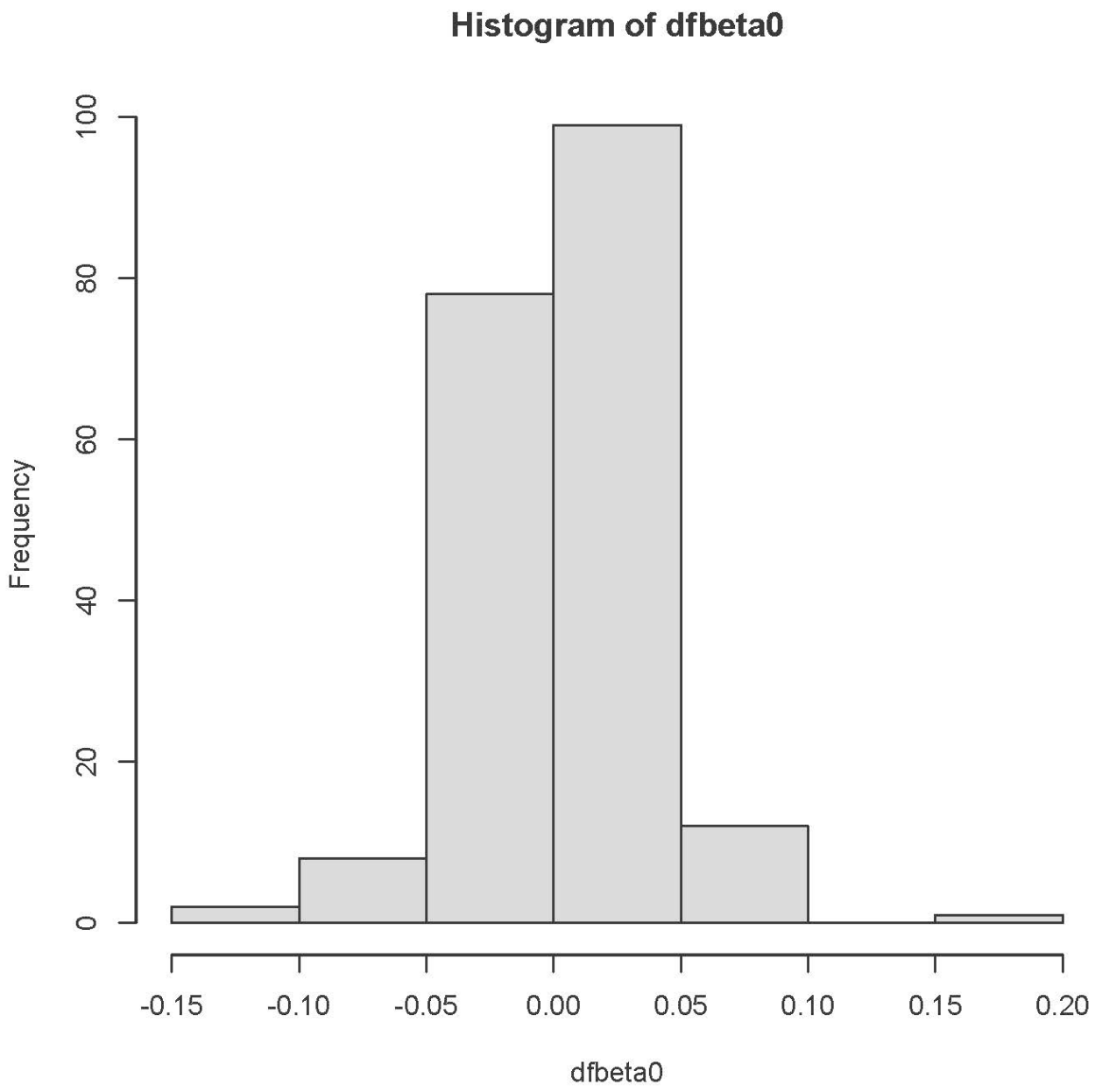



```
> id[cd>0.07]
[1] 105
>
```

$$\text{DFBETA} = \widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_{(i)}$$

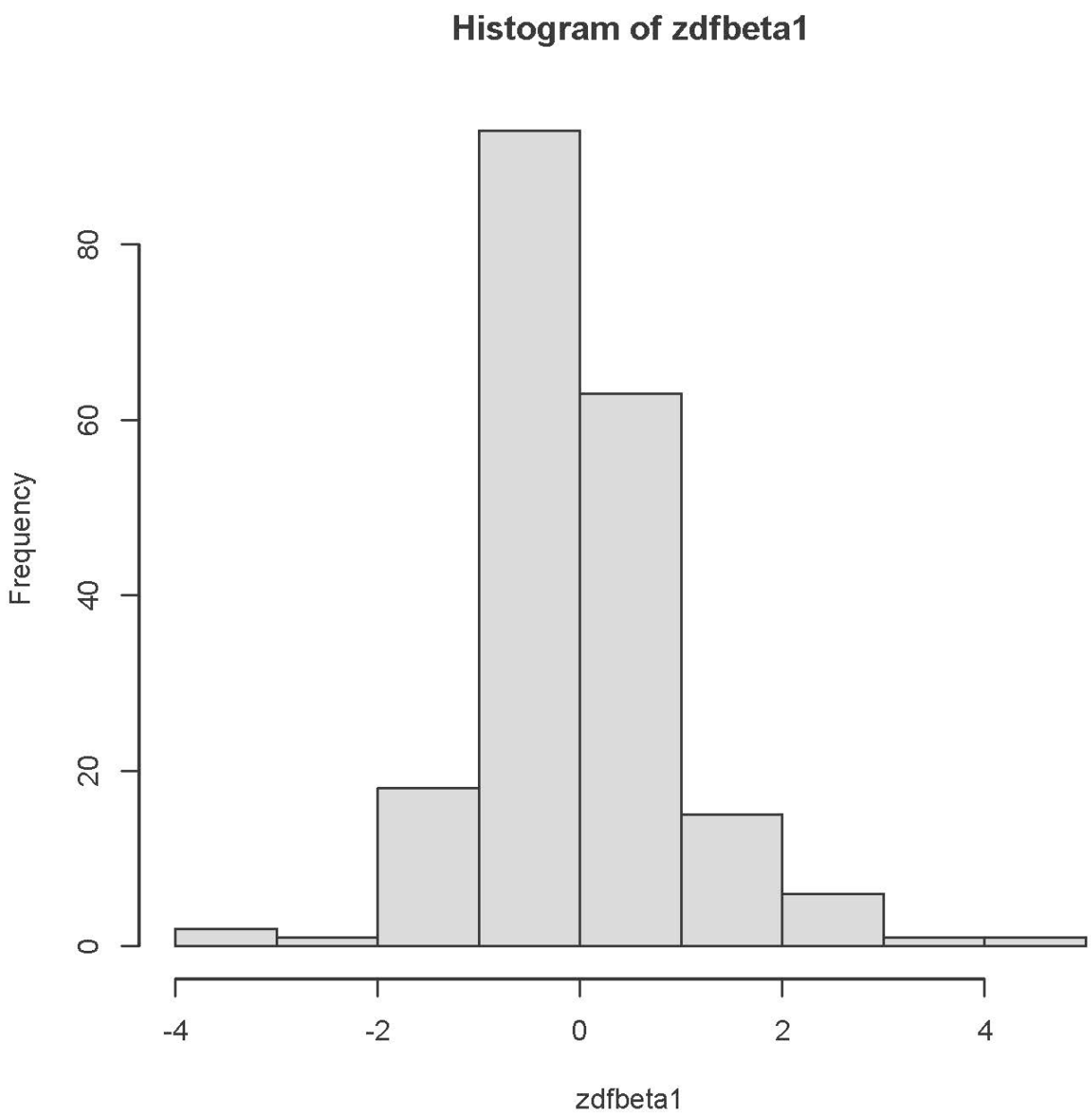
```
> round(dfbeta(mod1),5)
    (Intercept)   VERBAL     MATH
1       0.01108  0.00000 -0.00002
2       0.00834 -0.00002  0.00001
3      -0.01181  0.00003 -0.00001
4       0.00064  0.00000  0.00000
5       0.00402  0.00000 -0.00001
6       0.00071  0.00002 -0.00002
7       0.02276 -0.00005  0.00001
8       0.00222 -0.00001  0.00001
9       0.01277 -0.00004  0.00002
10      0.00659  0.00000 -0.00001
11     -0.01718  0.00003  0.00000
12      0.00937  0.00001 -0.00002
13      0.03158 -0.00001 -0.00003
14      0.00496  0.00010 -0.00009
15      0.07689 -0.00007 -0.00005
16      0.00995 -0.00007  0.00004
17      0.00037  0.00000 -0.00001
18     -0.04318  0.00007  0.00001
19     -0.00747  0.00000  0.00001
20     -0.08770  0.00017 -0.00003

  .         .        .       .
  .         .        .       .
  .         .        .       .

195     0.02110  0.00000 -0.00003
196    -0.02298  0.00005 -0.00001
197    -0.00604  0.00001  0.00000
198    -0.03607 -0.00001  0.00006
199    -0.00363  0.00000  0.00000
200     0.00318 -0.00001  0.00000
```

```
> dfb = dfbeta(mod1)
> dfbeta0 = dfb[,1]; hist(dfbeta0)
```



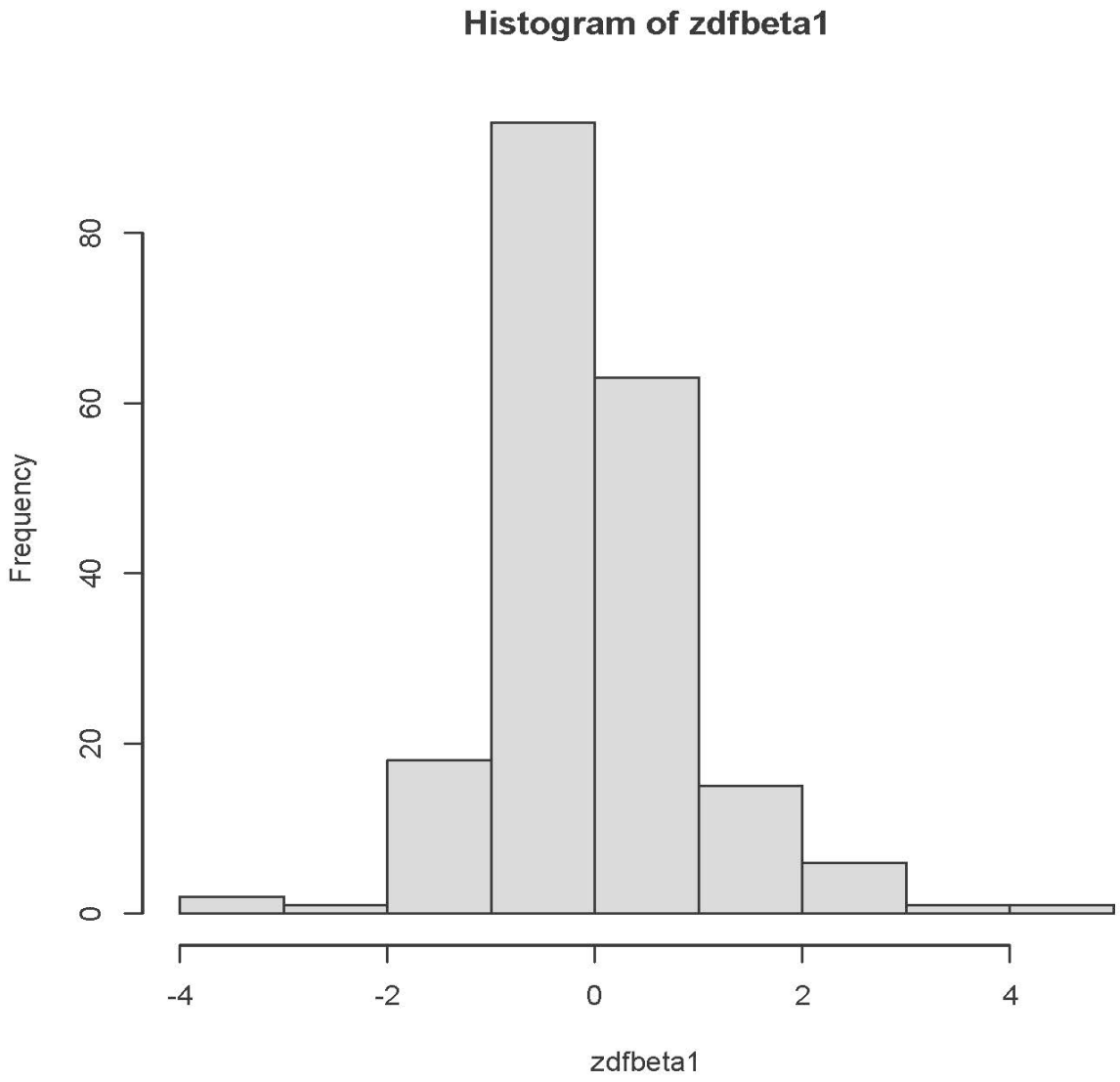

```
> zdfbeta0 = (dfbeta0-mean(dfbeta0))/sd(dfbeta0); summary(zdfbeta0)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-3.77876 -0.39578  0.02852  0.00000  0.34273  5.94714
> id[zdfbeta0>4]
[1] 105
```

```
> dfbeta1 = dfb[,2]
> zdfbeta1 = (dfbeta1-mean(dfbeta1))/sd(dfbeta1); summary(zdfbeta1)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-3.71358 -0.37384 -0.05202  0.00000  0.33814  4.53168
> hist(zdfbeta1)
```



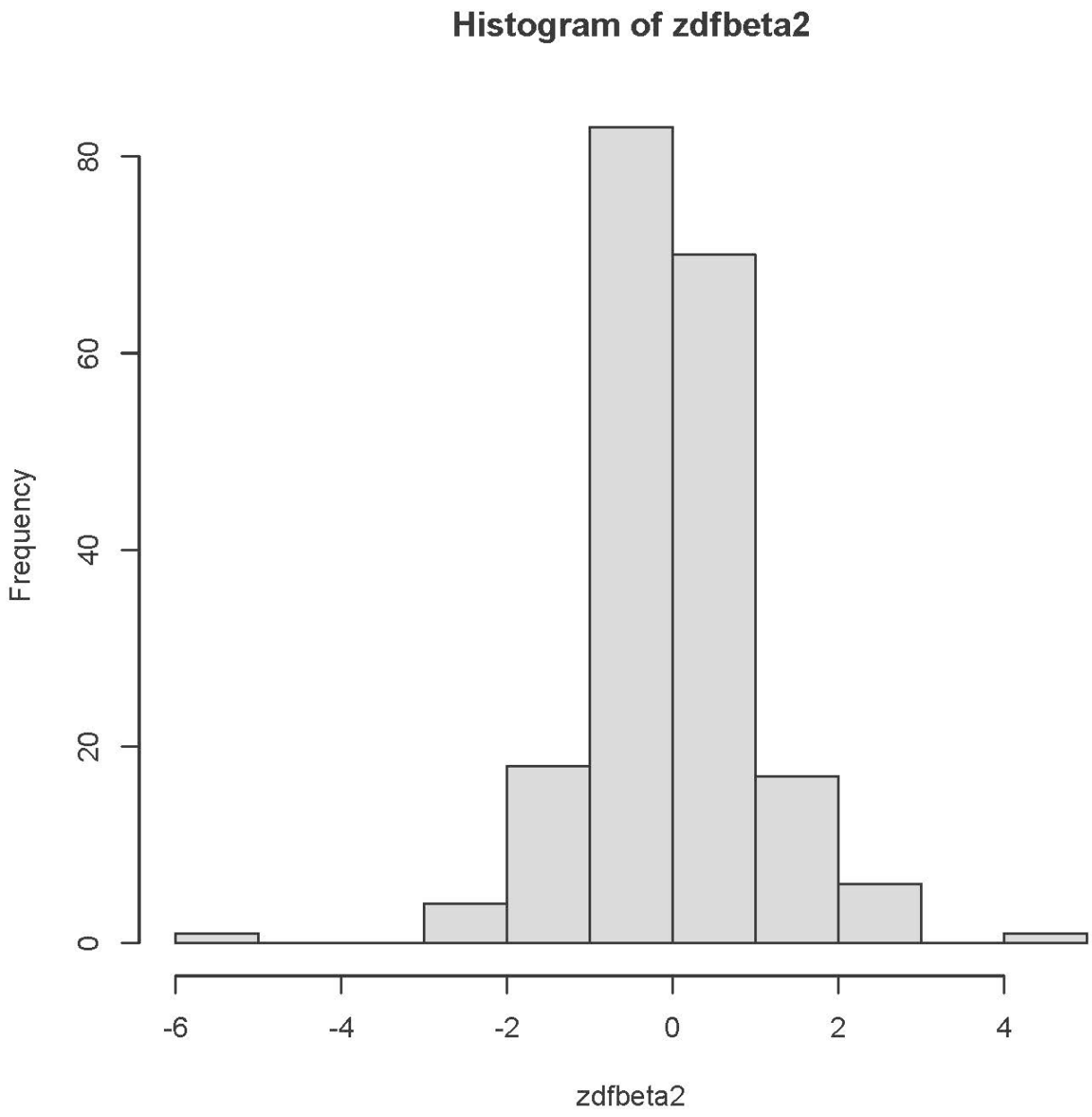

```
> id[zdfbeta1>4]
[1] 20
> # We have seen student 20 before (p. 6).
```

```
> dfbeta1 = dfb[,2]
> zdfbeta1 = (dfbeta1-mean(dfbeta1))/sd(dfbeta1); summary(zdfbeta1)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-3.71358 -0.37384 -0.05202  0.00000  0.33814  4.53168
> hist(zdfbeta1)
```




```
> id[zdfbeta1>4]
[1] 20
> # We have seen student 20 before (p. 6).
```

```
> dfbeta2 = dfb[,3]
> zdfbeta2 = (dfbeta2-mean(dfbeta2))/sd(dfbeta2); summary(zdfbeta2)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-5.47812 -0.38481 -0.01103  0.00000  0.36278  4.92509
> hist(zdfbeta2)
```



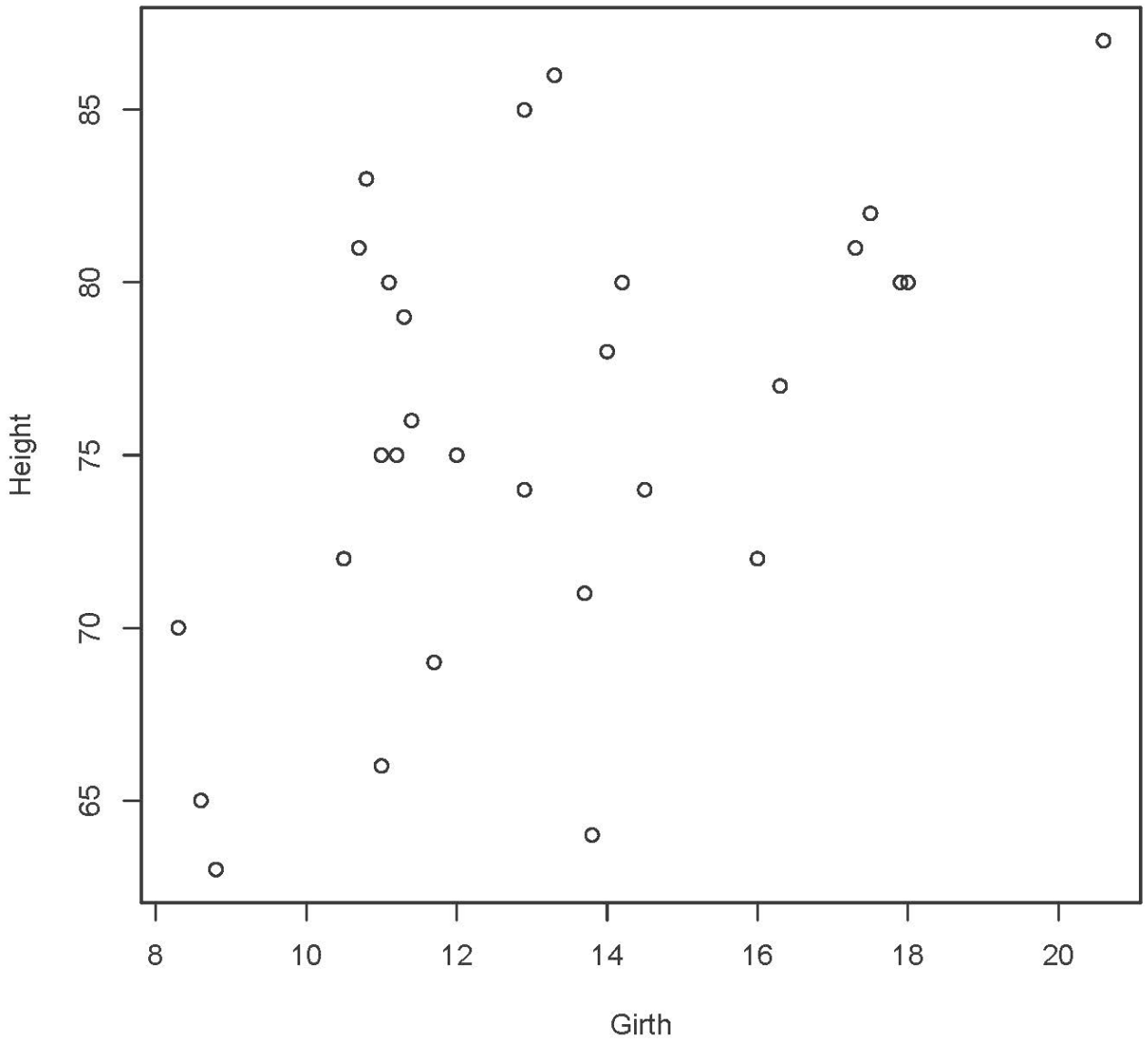

```
> id[abs(zdfbeta2)>4]
[1]  51 105
> # Old friends
```

```
> examine = c(20,51,73,105) # These observations may be influential.
> cbind(sat[examine,],yhat[examine])
    VERBAL MATH  GPA yhat[examine]
20     497  640 0.69      2.392965
51     645  548 1.16      2.642644
73     516  569 0.94      2.365985
105    736  862 2.15      3.165752
> summary(sat) # Again
     VERBAL           MATH            GPA       
 Min.   :401.0   Min.   :486.0   Min.   :0.690  
 1st Qu.:548.8   1st Qu.:607.5   1st Qu.:2.288  
 Median :590.5   Median :654.5   Median :2.635  
 Mean   :595.7   Mean   :649.5   Mean   :2.630  
 3rd Qu.:646.2   3rd Qu.:694.2   3rd Qu.:3.033  
 Max.   :815.0   Max.   :862.0   Max.   :3.990
```

Trees Data

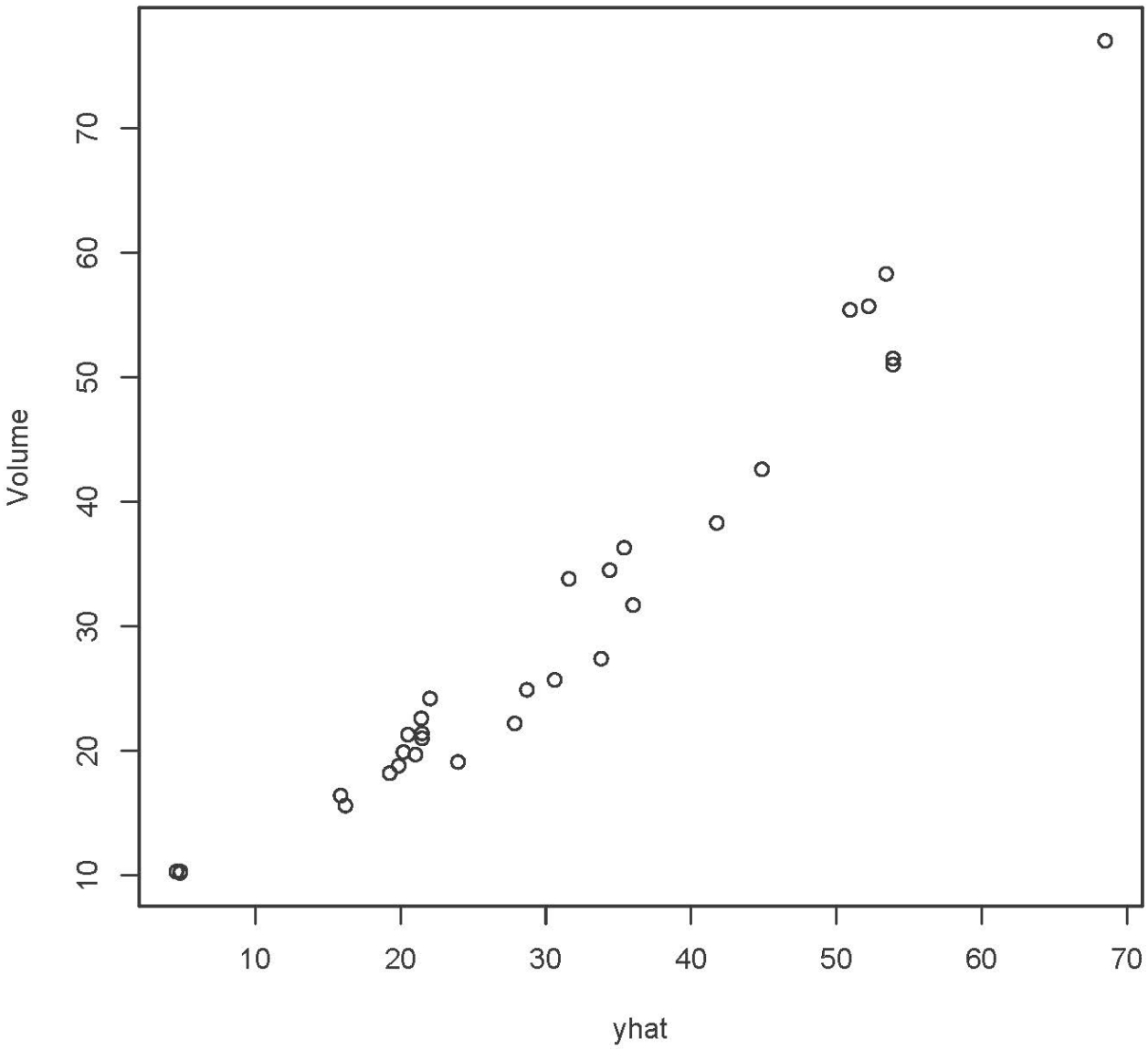
```
> rm(list=ls()) # Remove everything to start
> head(trees)
  Girth Height Volume
1   8.3     70   10.3
2   8.6     65   10.3
3   8.8     63   10.2
4  10.5     72   16.4
5  10.7     81   18.8
6  10.8     83   19.7
> attach(trees)
> mod1 = lm(Volume ~ Girth + Height)
> summary(mod1)

Call:
lm(formula = Volume ~ Girth + Height)

Residuals:
    Min      1Q  Median      3Q     Max
-6.4065 -2.6493 -0.2876  2.2003  8.4847

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -57.9877     8.6382  -6.713 2.75e-07 ***
Girth         4.7082     0.2643  17.816  < 2e-16 ***
Height        0.3393     0.1302   2.607   0.0145 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.882 on 28 degrees of freedom
Multiple R-squared:  0.948,   Adjusted R-squared:  0.9442
F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16

> # Check hat values
> hii = hatvalues(mod1); summary(hii)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.03567 0.05071 0.09201 0.09677 0.11825 0.22706
> n = length(Volume); n
[1] 31
> 2*(2+1)/n # Twice mean hat value
[1] 0.1935484
> sort(hii)
        16         21         15         22         10         12         13
23
0.03566543 0.03580935 0.03764563 0.04541796 0.04797237 0.04809206 0.04809206
0.04994875
         8          4         19         25         14         11         26
9
0.05148096 0.05919131 0.06665975 0.06930648 0.07275901 0.07382512 0.08841762
0.09200658
        27         28         29         30         24          7          1
5
0.09603041 0.10641665 0.10982638 0.10982638 0.11142518 0.11480262 0.11582883
0.12066468
        17         18          2          6          3         20         31
0.13130916 0.14346152 0.14720958 0.15575111 0.17686186 0.21123665 0.22705852

> trees[c(20,31), ] # Rows 20 and 31, all the columns
   Girth Height Volume
20  13.8     64   24.9
31  20.6     87   77.0
```

```
> trees[c(20,31), ] # Rows 20 and 31, all the columns
   Girth Height Volume
20  13.8     64   24.9
31  20.6     87   77.0

> plot(Girth,Height)
```

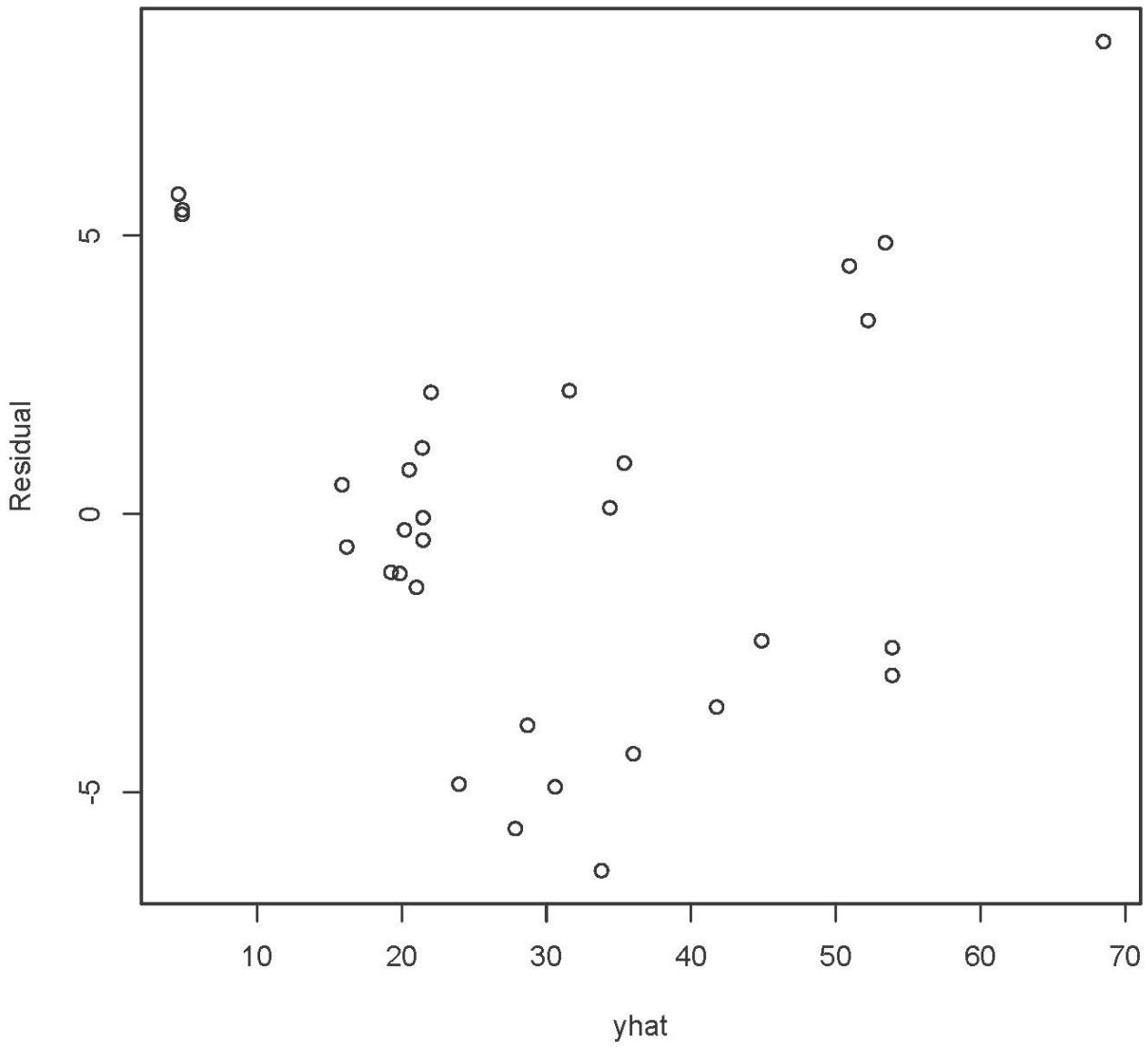



```
> # They are somewhat out of the cluster. No definitive conclusion.
> # Keep an eye on trees 20 and 31.
```

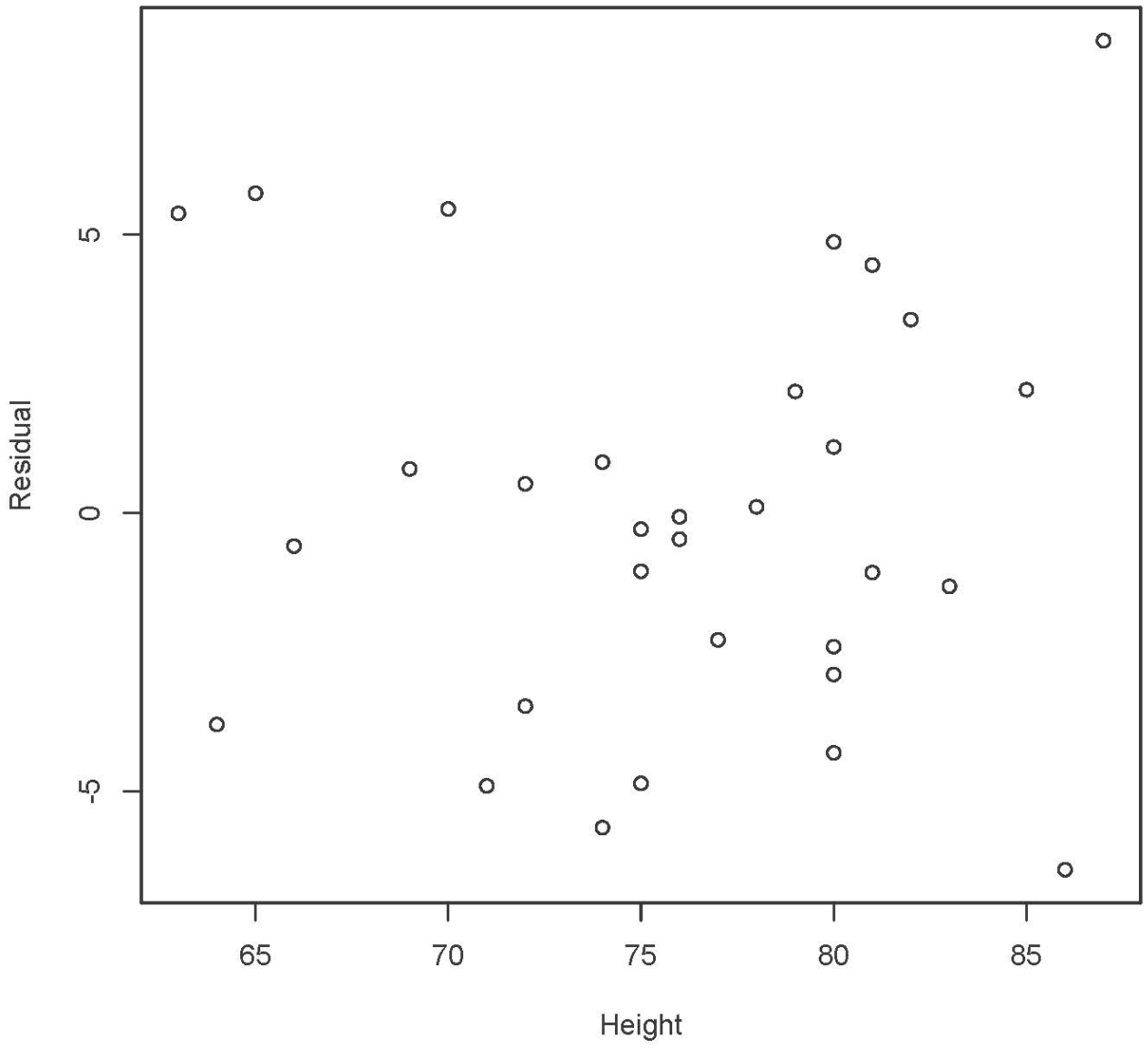
```
> yhat = fitted.values(mod1)
> plot(yhat,Volume)
```



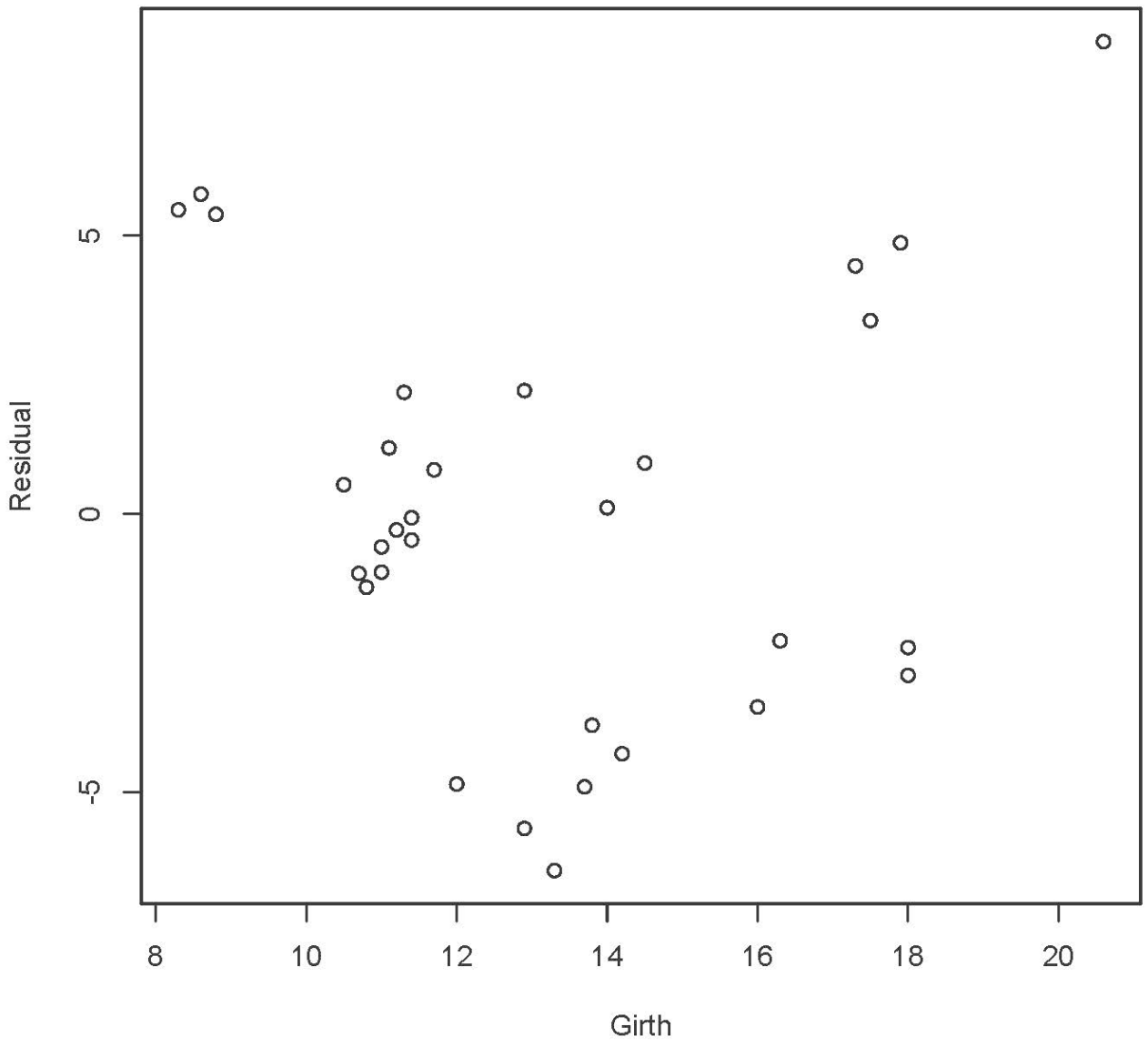
```
> Residual = residuals(mod1)
> plot(yhat,Residual)
```



```
> plot(Height,Residual) # Plotting against variables in the equation
```



```
> plot(Girth,Residual)
```



```
> Girthsq = Girth^2 # Polynomial term -- literally square it
> mod2 = lm(Volume ~ Girth + Girthsq + Height); summary(mod2)

Call:
lm(formula = Volume ~ Girth + Girthsq + Height)

Residuals:
    Min      1Q  Median      3Q     Max
-4.2928 -1.6693 -0.1018  1.7851  4.3489

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -9.92041   10.07911  -0.984 0.333729
Girth       -2.88508    1.30985  -2.203 0.036343 *
Girthsq      0.26862    0.04590   5.852 3.13e-06 ***
Height       0.37639    0.08823   4.266 0.000218 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.625 on 27 degrees of freedom
Multiple R-squared:  0.9771,  Adjusted R-squared:  0.9745
F-statistic: 383.2 on 3 and 27 DF,  p-value: < 2.2e-16

> # Girth squared makes physical sense, because Girth is circumference in
> # inches, C = 2 pi R, and volume of a cylinder is pi R^2 H

> # Proportion of remaining variation
> p = (0.9771-0.948)/(1-0.948); p
[1] 0.5596154
> 5.852^2/(5.852^2 + 27) # p = qF/(qF + n-k-1)
[1] 0.5591542

> Resid2 = residuals(mod2)
> plot(Girth,Resid2)  # Nothing
> plot(Height,Resid2) # Nothing

> sdr = rstudent(mod2)
> summary(sdr)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-1.82124 -0.66287 -0.03926 -0.00292  0.70580  1.82654
>
> # Why is the following unnecessary?
> n = dim(trees)[1]; n
[1] 31
> dfe = df.residual(mod2); dfe # n-k-1
[1] 27
> tcrit = qt(1-0.025/n, dfe-1); tcrit # Bonferroni critical value
[1] 3.519811
>
> shapiro.test(sdr) # Test for normality

      Shapiro-Wilk normality test

data:  sdr
W = 0.96853, p-value = 0.4795
```

```
> # Look at prediction intervals
> cbind(Volume[1:5],predict(mod1,interval='predict')[1:5,])
             fit       lwr      upr
1 10.3  4.837660 -3.561809 13.23713
2 10.3  4.553852 -3.962908 13.07061
3 10.2  4.816981 -3.809144 13.44311
4 16.4 15.874115  7.690594 24.05764
5 18.8 19.869008 11.451358 28.28666
Warning message:
In predict.lm(mod1, interval = "predict") :
  predictions on current data refer to _future_ responses

>
> cbind(Volume[1:5],predict(mod2,interval='predict')[1:5,])
             fit       lwr      upr
1 10.3 10.985950  4.902269 17.06963
2 10.3  9.600406  3.566753 15.63406
3 10.2  9.205421  3.163767 15.24708
4 16.4 16.501775 10.954762 22.04879
5 18.8 20.451204 14.746331 26.15608
Warning message:
In predict.lm(mod2, interval = "predict") :
  predictions on current data refer to _future_ responses

> # Try to improve the model by including interactions
>
> hg = Height*Girth
> hgsq = Height*Girthsq
>
> mod3 = lm(Volume ~ Girth + Height + Girthsq + hg + hgsq)
> summary(mod3)

Call:
lm(formula = Volume ~ Girth + Height + Girthsq + hg + hgsq)

Residuals:
    Min      1Q  Median      3Q     Max
-5.1880 -0.7901 -0.0037  1.9306  3.9483

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 48.914179  90.852925   0.538    0.595
Girth       -8.228180  13.803580  -0.596    0.556
Height      -0.616152   1.250446  -0.493    0.626
Girthsq      0.311160   0.536379   0.580    0.567
hg           0.103075   0.180291   0.572    0.573
hgsq        -0.001764   0.006621  -0.266    0.792

Residual standard error: 2.659 on 25 degrees of freedom
Multiple R-squared:  0.9782,  Adjusted R-squared:  0.9738
F-statistic: 224.3 on 5 and 25 DF,  p-value: < 2.2e-16

> anova(mod2,mod3)
Analysis of Variance Table

Model 1: Volume ~ Girth + Girthsq + Height
Model 2: Volume ~ Girth + Height + Girthsq + hg + hgsq
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     27 186.01
2     25 176.73  2      9.28 0.6564 0.5274
```

```
> # Still not giving up on product terms.
> # Volume of a cylinder is pi r^2 height
>
> # Help says: "This data set provides measurements of the diameter, height
and volume of timber in 31 felled black cherry trees. Note that the
diameter (in inches) is erroneously labelled Girth in the data. It is
measured at 4 ft 6 in above the ground."
>
> Diameter = Girth
> # Diameter in inches, so in feet, Diameter/12 = 2 R
> R = Diameter/24
> cVol =  pi * R^2 * Height
>
> mod4 = lm(Volume ~ cVol); summary(mod4)

Call:
lm(formula = Volume ~ cVol)

Residuals:
    Min      1Q  Median      3Q     Max
-4.6195 -1.1002 -0.1656  1.7451  4.1976

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.29768    0.96356  -0.309     0.76
cVol         0.38950    0.01091  35.711   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.493 on 29 degrees of freedom
Multiple R-squared:  0.9778,    Adjusted R-squared:  0.977
F-statistic:  1275 on 1 and 29 DF,  p-value: < 2.2e-16

> c(summary(mod2)$r.squared, summary(mod3)$r.squared,
+ summary(mod4)$r.squared)
[1] 0.9770528 0.9781976 0.9777654
```

```
> plot(cVol,Volume, xlab = 'Volume of a Cylinder')
> abline(reg=mod4)
```