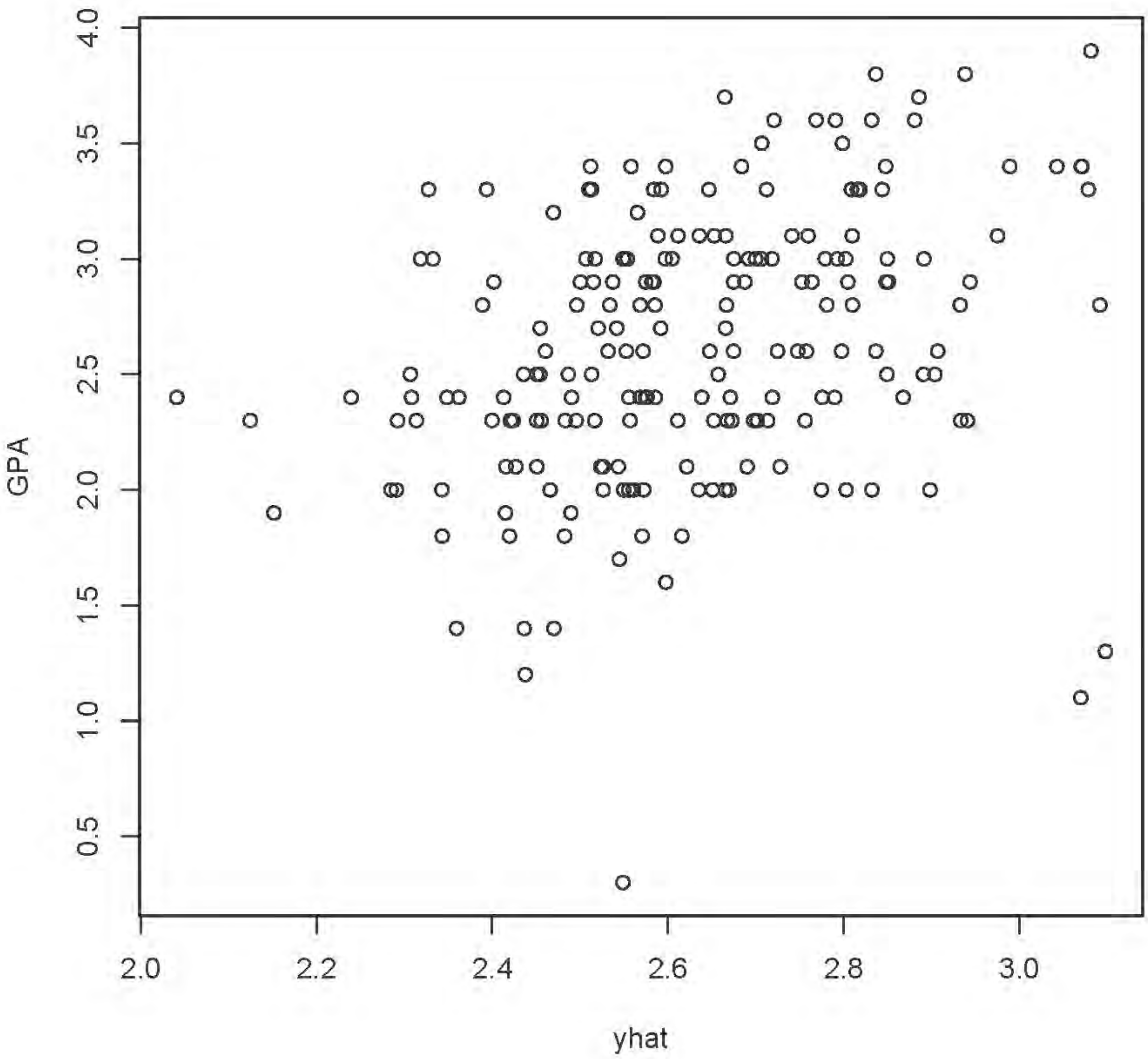
Regression diagnostics with R[*]

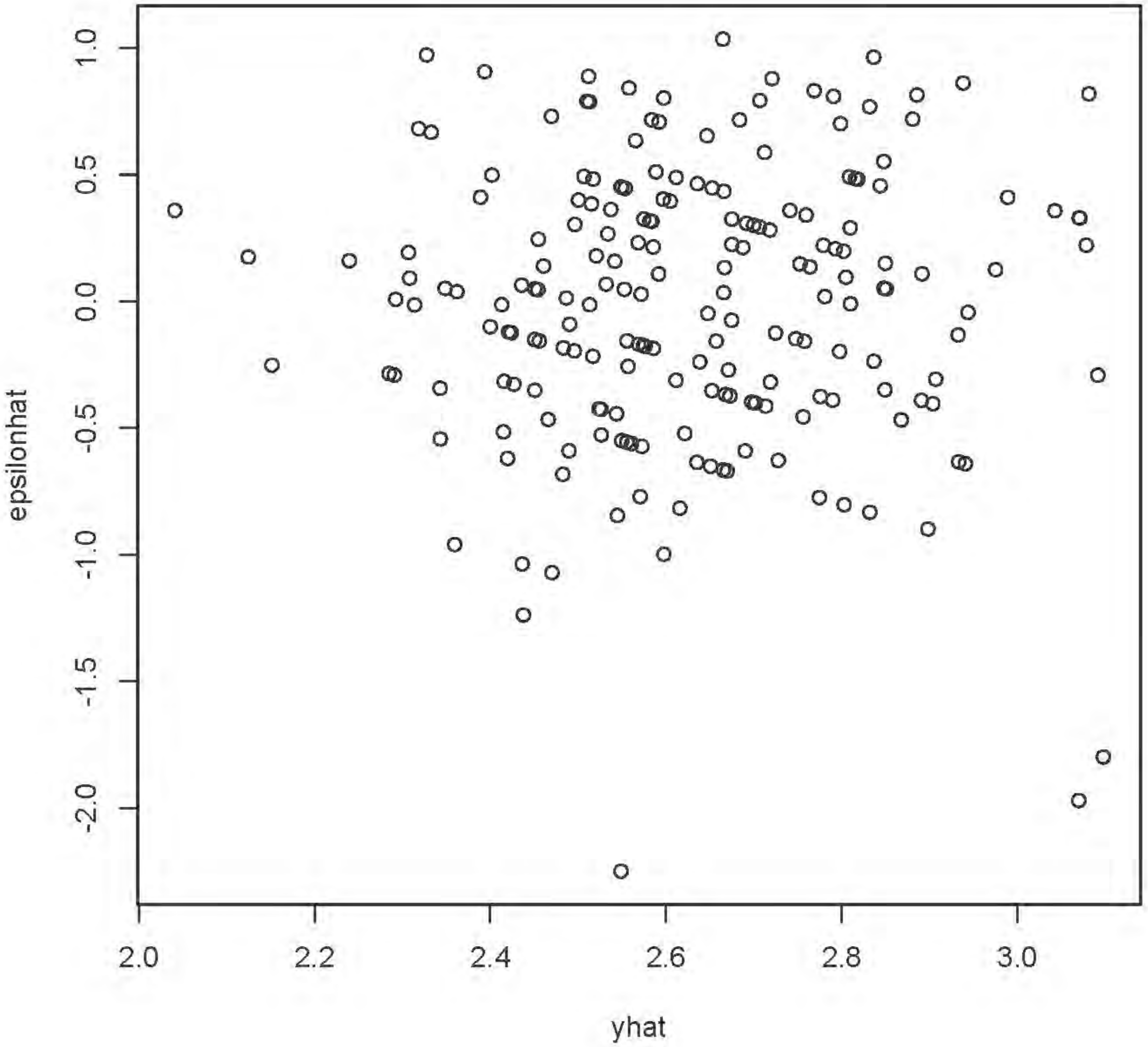
```
> sat = 
read.table("http://www.utstat.utoronto.ca/~brunner/302f13/code_n_data/lectu
re/sat.data")
> head(sat)
  VERBAL MATH GPA
1    623  509 2.6
2    454  471 2.3
3    643  700 2.4
4    585  719 3.0
5    719  710 3.1
6    693  643 2.9
> mod1 = lm(GPA ~ VERBAL+MATH, data=sat); summary(mod1)

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
> 
```

[*] Copyright information is on the last page.

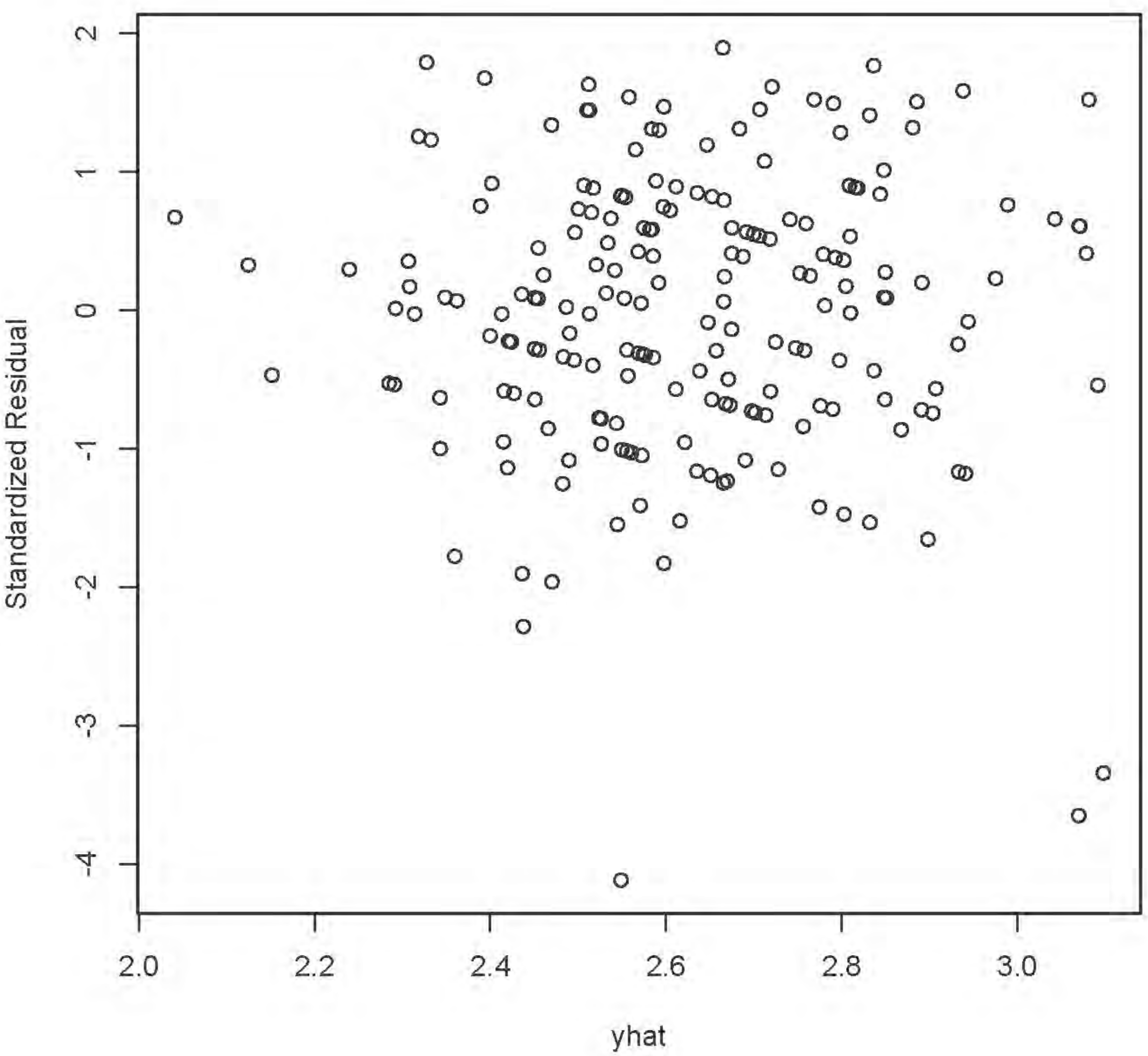
```
> # Plot y-hat versus y
> yhat = mod1$fitted.values
> plot(yhat,GPA)
>
```



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



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

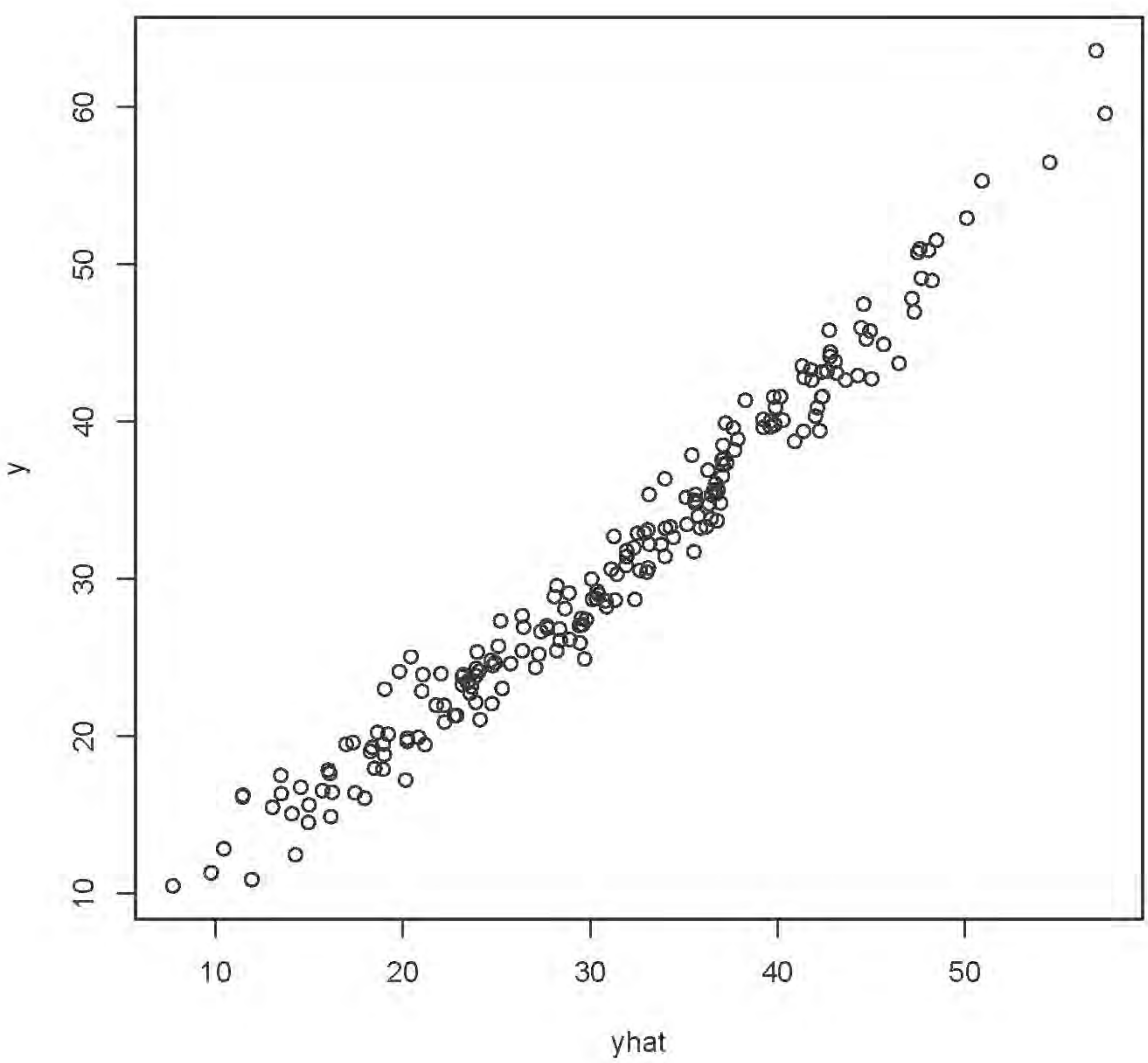


```
> # Three look like possible outliers: Investigate

> id = 1:200
> suspect = id[sr < -3]
> cbind(sat[suspect,],yhat[suspect],epsilonhat[suspect])
    VERBAL MATH GPA yhat[suspect] epsilonhat[suspect]
121    780  692 1.3      3.097791           -1.797791
131    578  609 0.3      2.548754           -2.248754
136    760  710 1.1      3.069645           -1.969645

> # 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 = mod1$df.residual; dfe
[1] 197
> alpha = 0.05; a = alpha/200; bcrit = qt(1-a/2,dfe-1); bcrit
[1] 3.730706
> sdr[abs(sdr)>bcrit]
      131       136 
-4.293141 -3.768640 
>
```

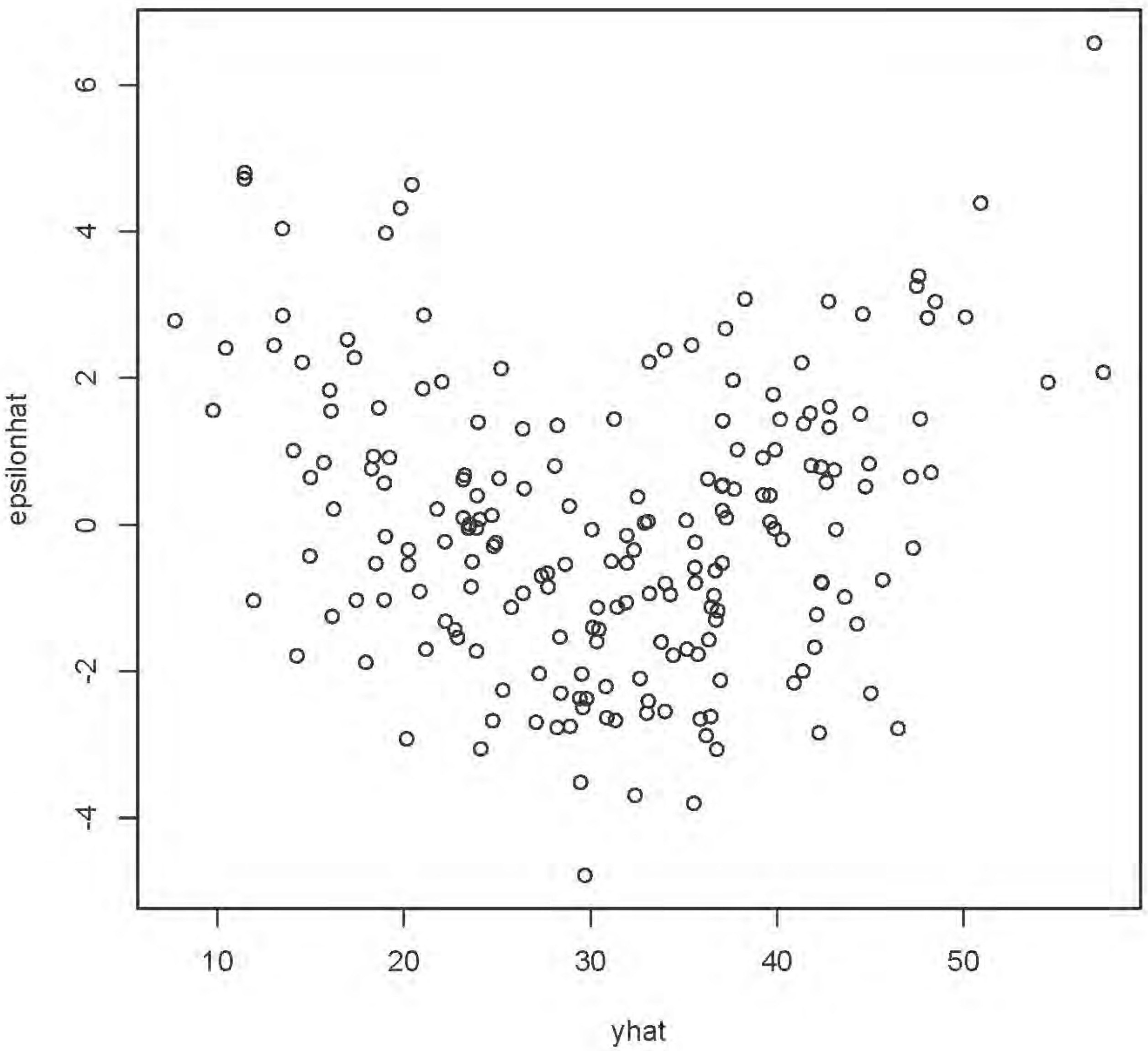
I feel that all three suspicious points are worthy of investigation.

```
> # Detecting curvilinear trends

> curvy =
read.table("http://www.utstat.toronto.edu/~brunner/302f13/code_n_data/lectu
re/curvy.data")
> mod1 = lm(y~x1+x2,data=curvy); summary(mod1)

Call:
lm(formula = y ~ x1 + x2, data = curvy)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.7813 -1.3667 -0.0649  1.3356  6.5690 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -22.4665     0.8172 -27.493  < 2e-16 ***
x1            0.9598     0.1474   6.511 6.05e-10 ***
x2            9.9476     0.1509  65.904  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 1.945 on 197 degrees of freedom
Multiple R-squared: 0.9663,   Adjusted R-squared: 0.9659 
F-statistic:  2823 on 2 and 197 DF,  p-value: < 2.2e-16 

> attach(curvy)
> yhat = mod1$fitted.values; epsilonhat = mod1$residuals
> 
```

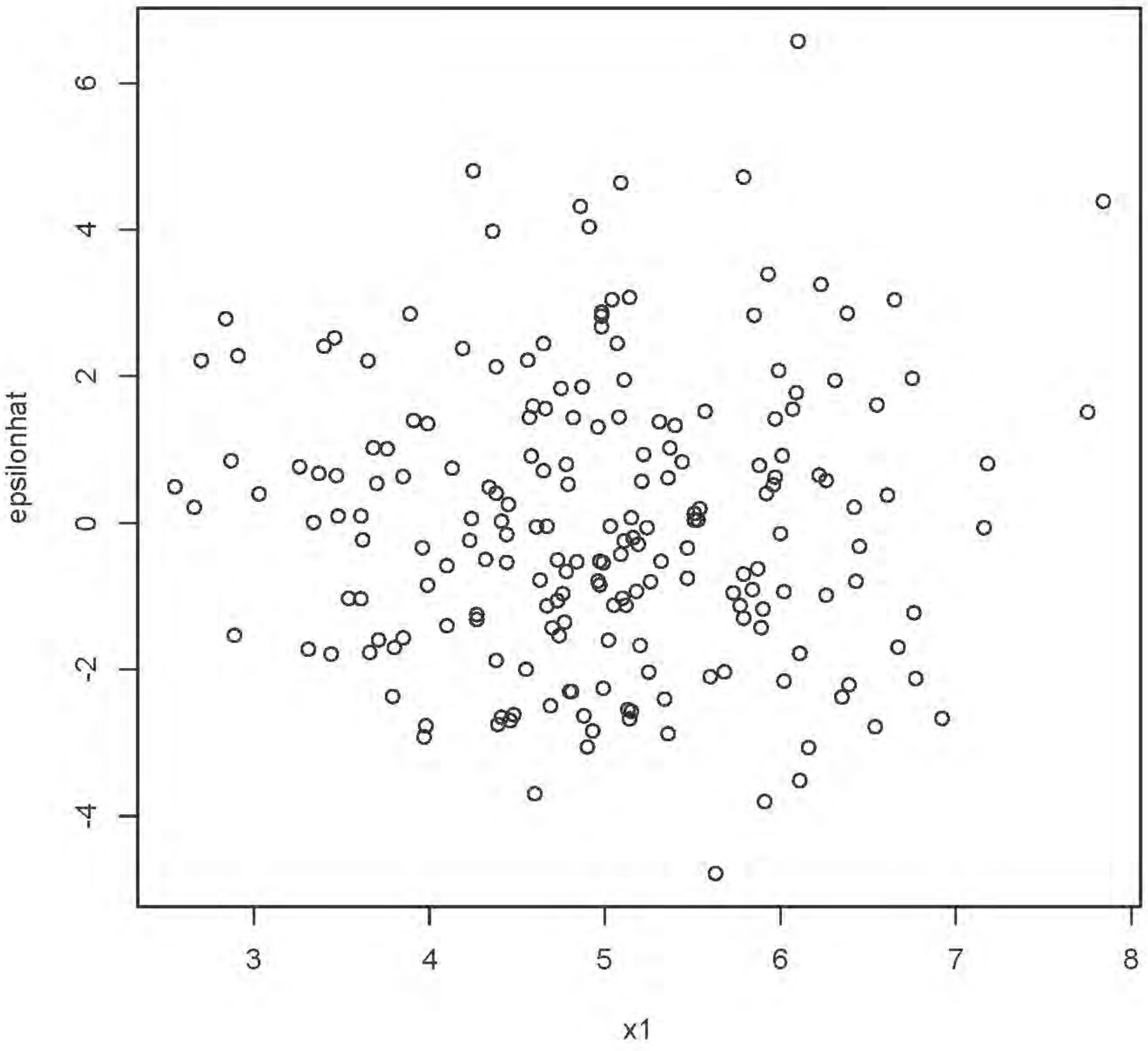
```
> # Plot y-hat versus y
> plot(yhat,y)
```



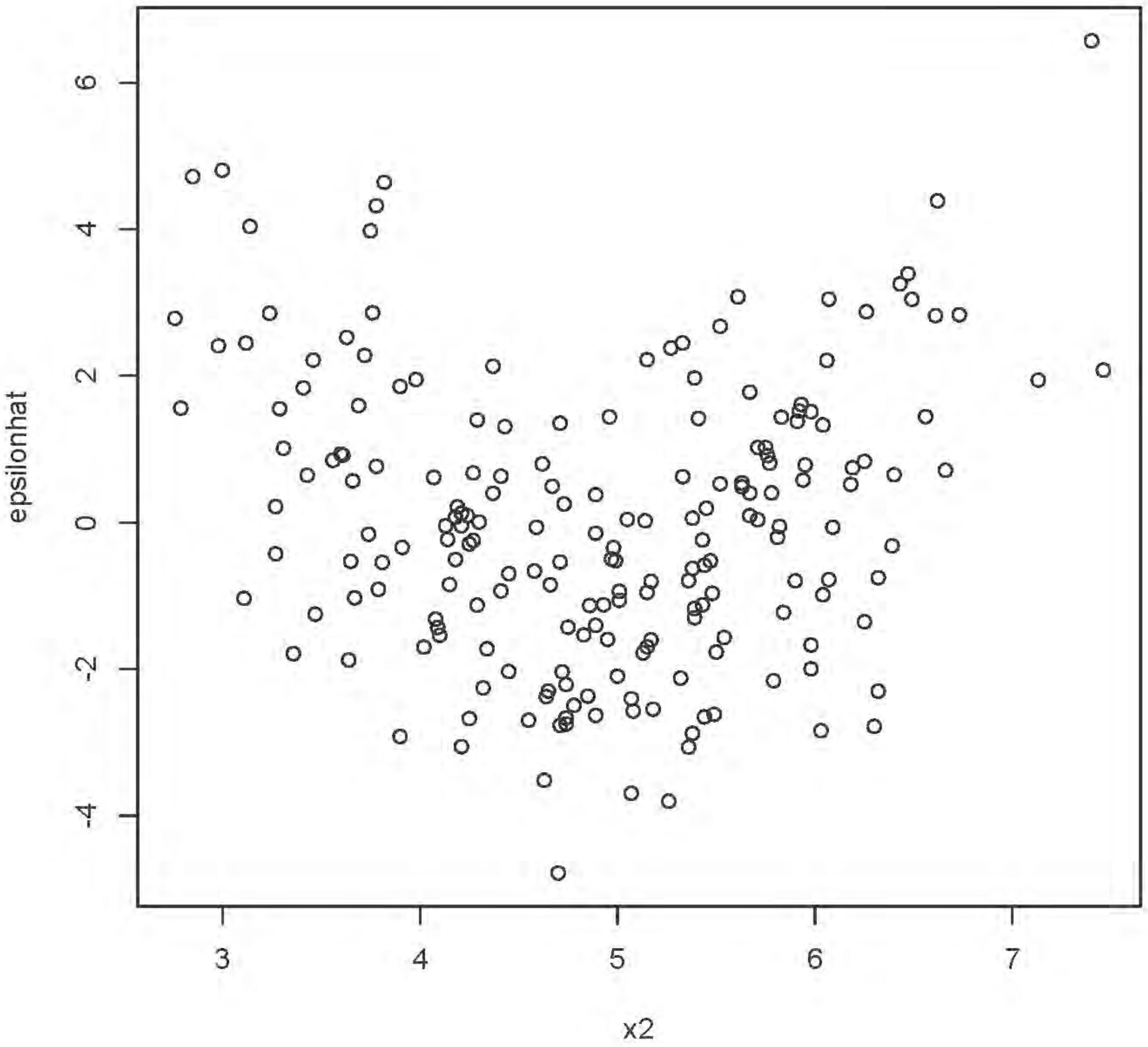
```
> # Plot y-hat versus residuals
> plot(yhat,epsilonhat)
>
```



```
> plot(yhat,rstandard(mod1)) # Same picture
> plot(yhat,rstudent(mod1))  # Same picture
> # Plot residuals against variables in the model
> plot(x1,epsilonhat)
```

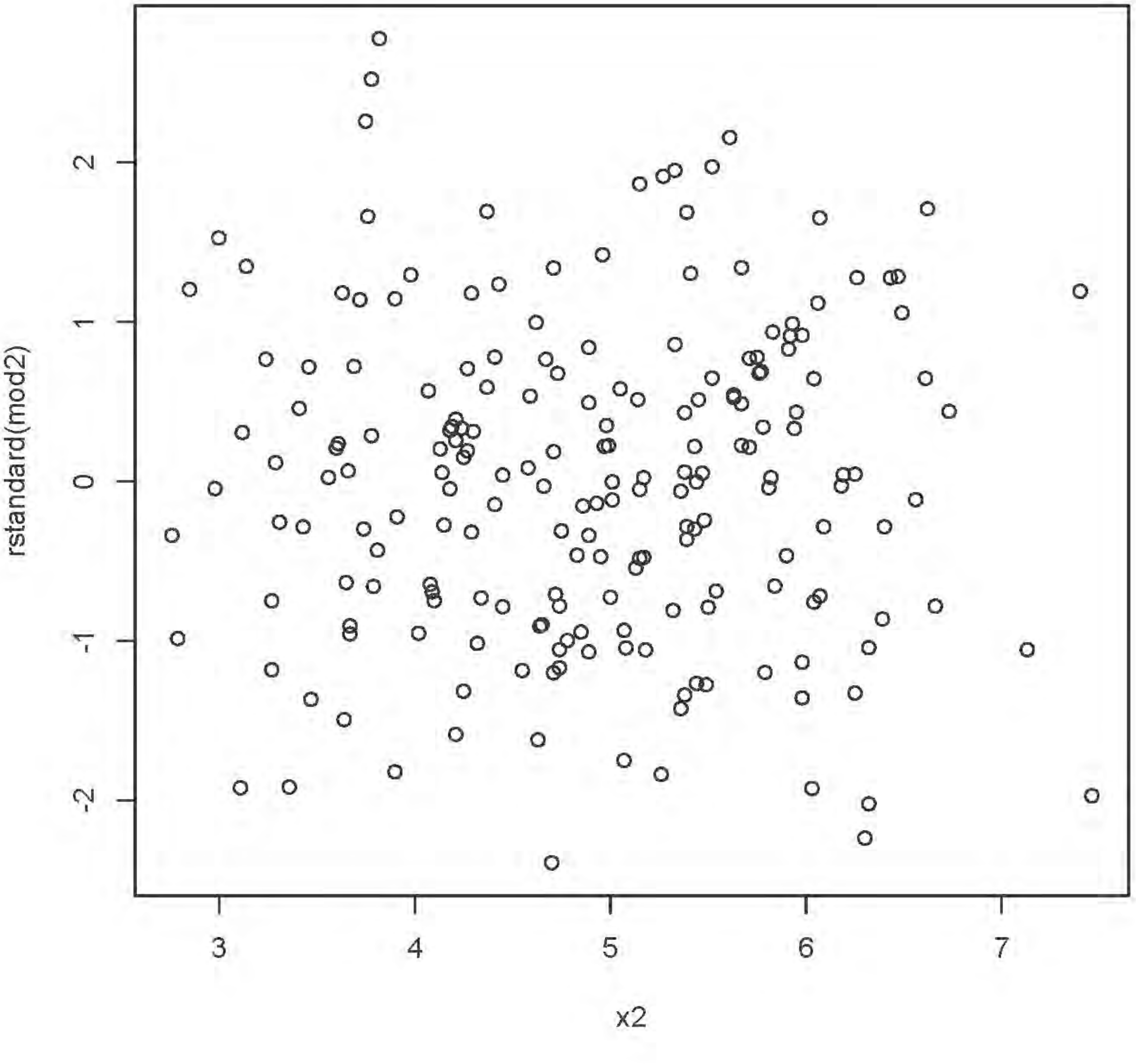


```
plot(x2,epsilonhat)
```



```
> # Try adding x2^2 to the model
> x2sq = x2^2
> mod2 = lm(y~x1+x2+x2sq,data=curvy); summary(mod2)

Call:
lm(formula = y ~ x1 + x2 + x2sq, data = curvy)

Residuals:
   Min     1Q Median     3Q    Max 
-3.884 -1.219  0.041  1.101  4.509 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.46607    2.41019  -0.608    0.544    
x1           0.91057    0.12407   7.339 5.58e-12 ***
x2           1.08960    0.98266   1.109    0.269    
x2sq         0.90596    0.09966   9.090  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.635 on 196 degrees of freedom
Multiple R-squared: 0.9763,	Adjusted R-squared: 0.9759 
F-statistic:  2690 on 3 and 196 DF,  p-value: < 2.2e-16

> 
```

```
> plot(x2,rstandard(mod2)) # In practice you would look at all the residual plots again.
```



Non-constant variance (“Heteroscedasticity”)

Used Car Sales, n = number of dealerships

Y = Price paid by customer

x1 = Number of sales people

x2 = Average sales force years of experience

x3 = Average sales force years of education

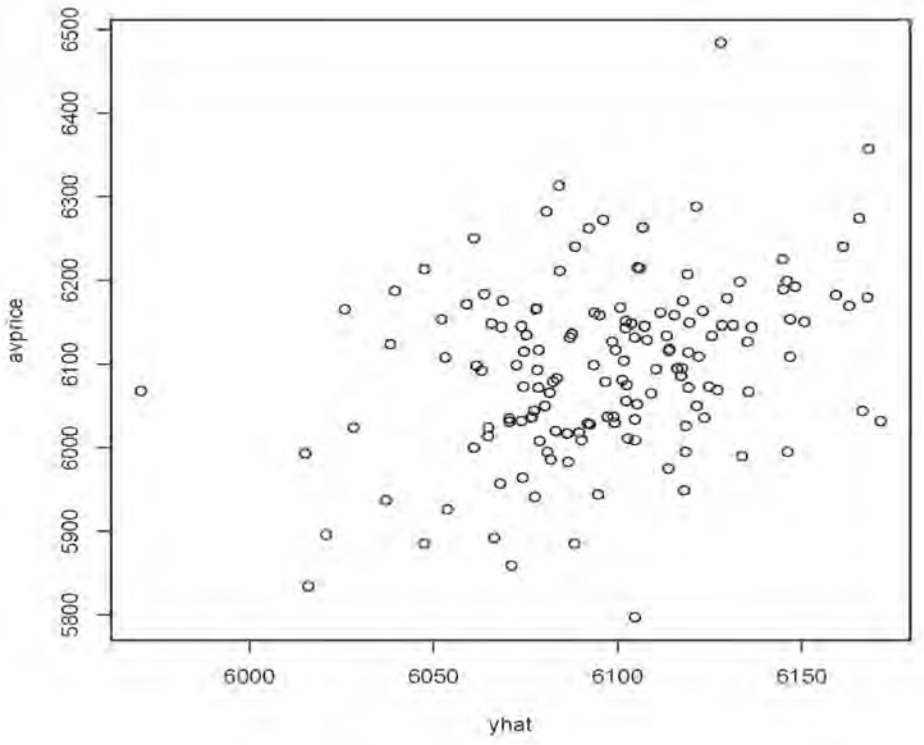
x4 = Percent women on sales force

x5 = Average income of census tract where dealership is located, in thousands

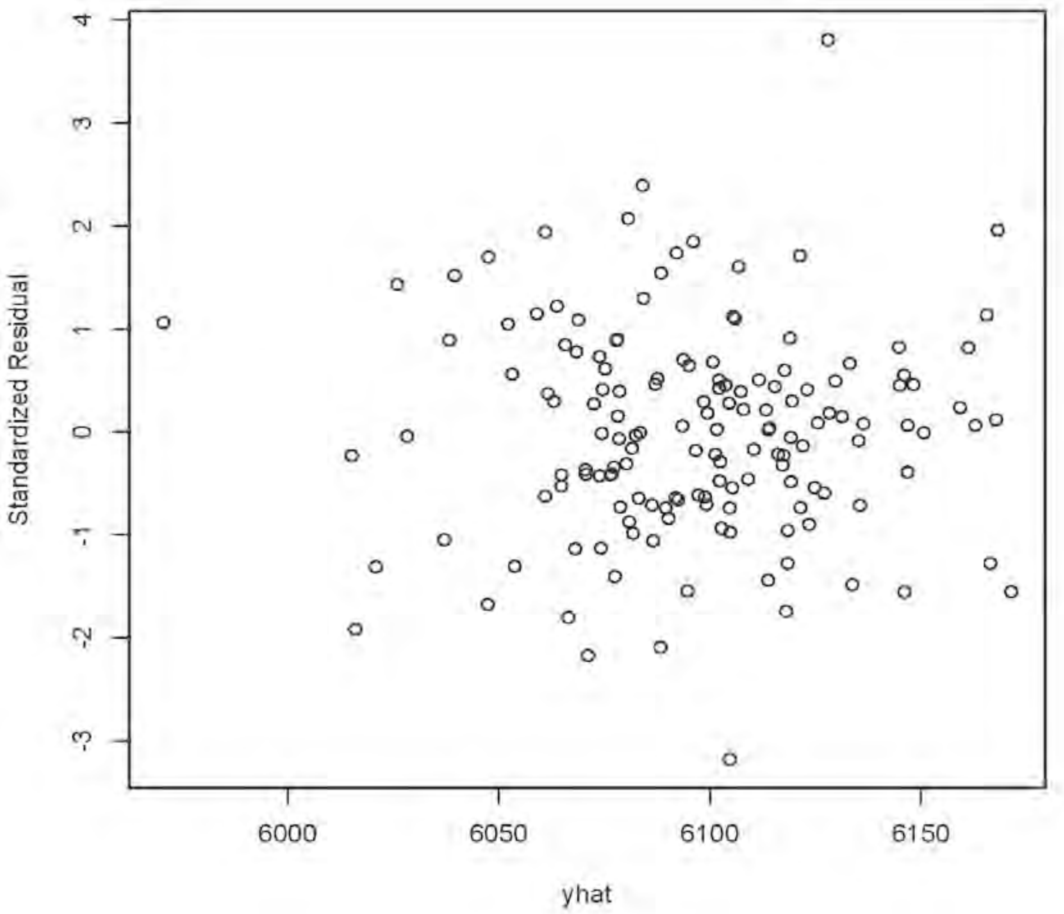
x6 = Number of cars sold, not in model

```
> carsales =
read.table("http://www.utstat.toronto.edu/~brunner/302f13/code_n_data/lectu
re/carsales.data")
> head(carsales)
  salesforce yrsexp yrseduc women income nsales avprice
1         20   4.25    12.7    35     24     78    6099
2         16   6.42    13.8     6     87     77    6161
3          6   9.25    13.8    17     56     22    6094
4         17   6.08    11.6    24     26     71    6030
5         12   5.58    12.7     8     26     49    6108
6          8   7.33    15.5    50     42     33    6136

> auto = lm(avprice ~ salesforce + yrsexp + yrseduc + women + income,
data=carsales)
> summary(auto)

Call:
lm(formula = avprice ~ salesforce + yrsexp + yrseduc + women +
    income, data = carsales)

Residuals:
    Min      1Q  Median      3Q     Max
-307.71  -62.60    0.80   54.38  356.01
```

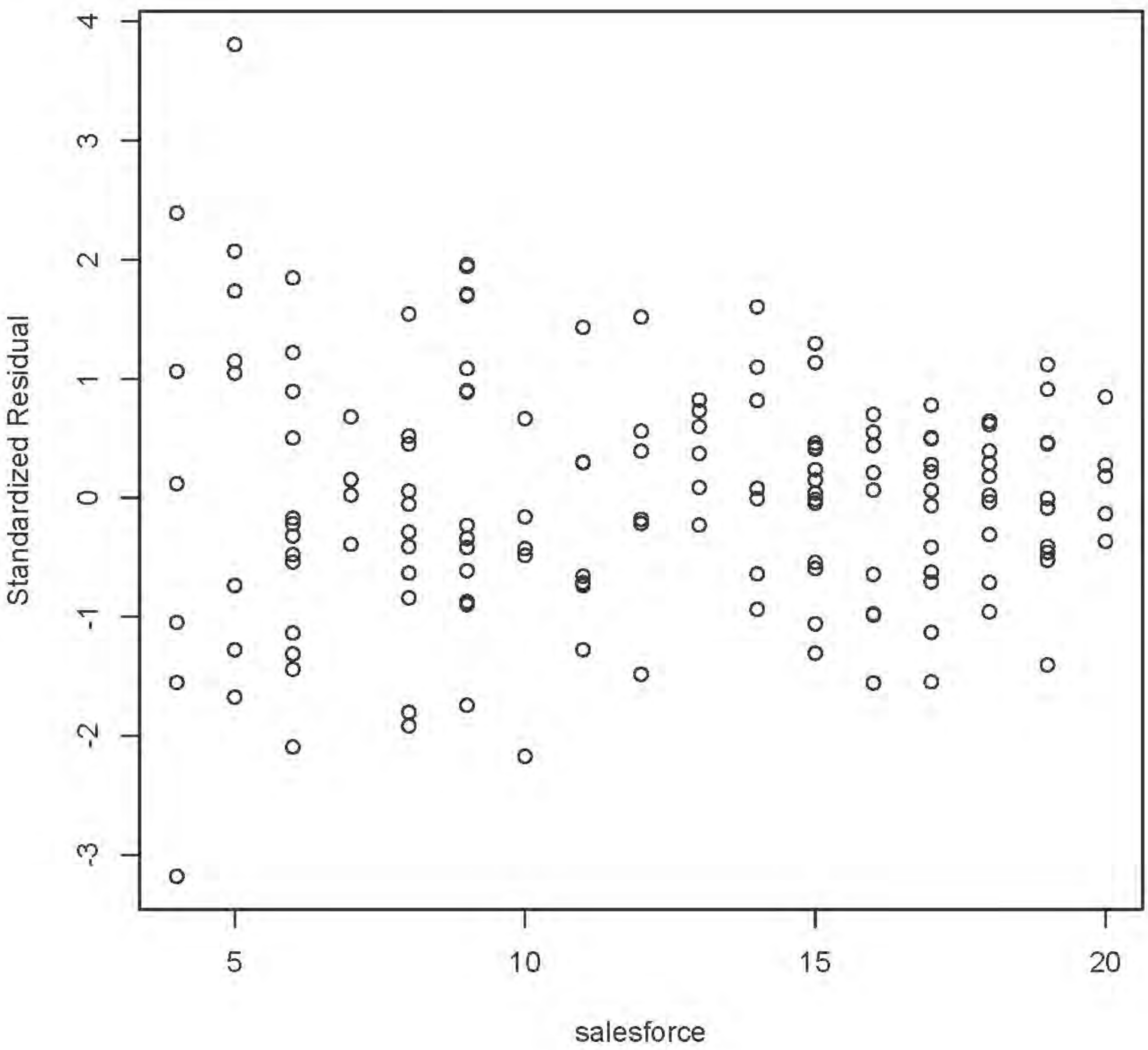
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6099.1463    96.8659  62.965  < 2e-16 ***
salesforce     1.5577     1.6865   0.924  0.35724
yrsexp        14.4620     5.3166   2.720  0.00733 **
yrseduc      -13.3943     7.0424  -1.902  0.05918 .
women          1.0157     0.6621   1.534  0.12721
income         0.6383     0.3099   2.059  0.04126 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 98.94 on 144 degrees of freedom
Multiple R-squared: 0.1103,   Adjusted R-squared: 0.07939
F-statistic:  3.57 on 5 and 144 DF,  p-value: 0.0045

> yhat = auto$fitted.values ; sr = rstandard(auto)
> attach(carsales) # Make variable names accessible without dollar signs
The following object(s) are masked from 'package:datasets':

    women
> plot(yhat,avprice) # Plot y-hat versus y
```



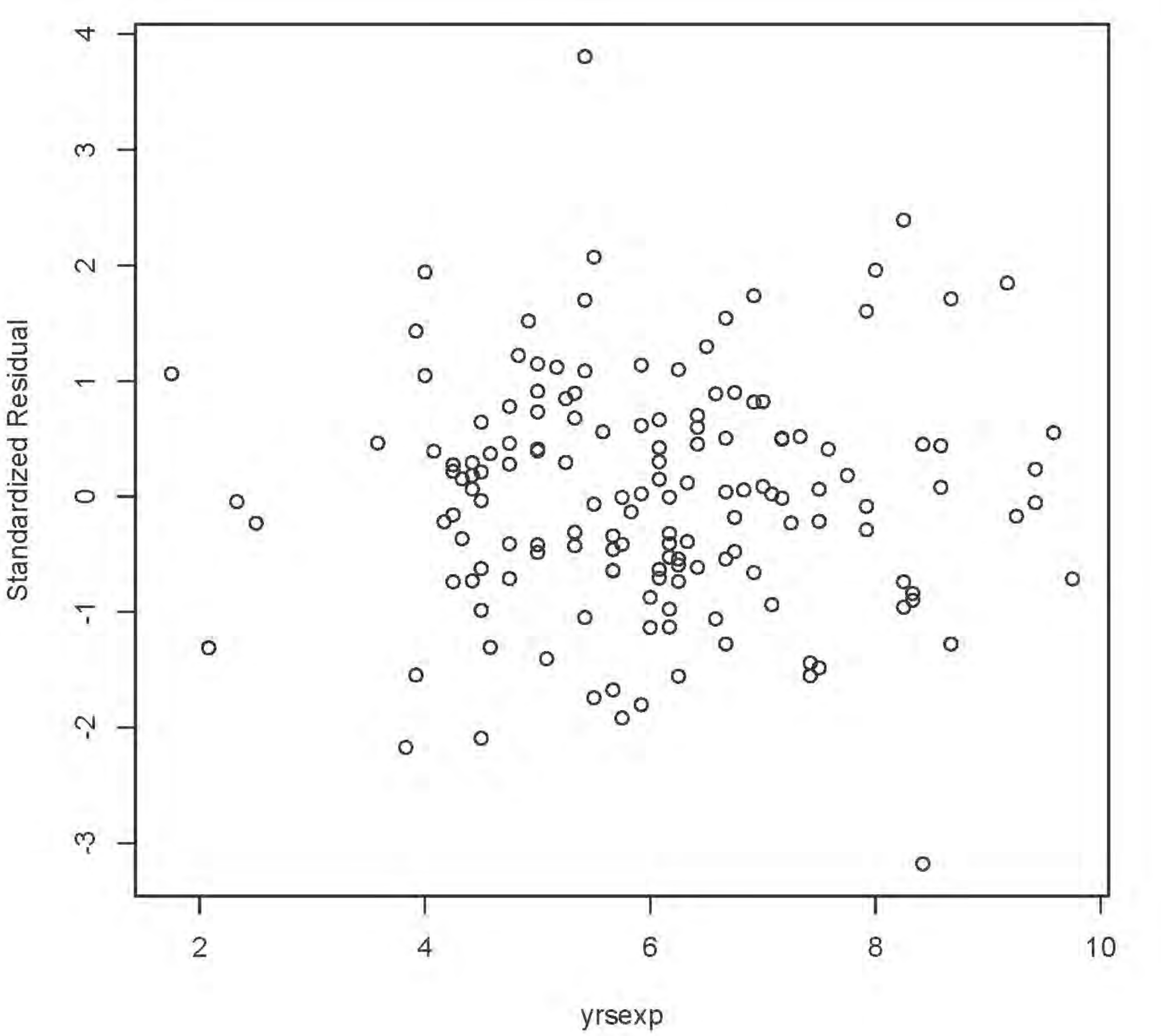
```
> # Plot y-hat versus standardized residual
> plot(yhat,sr,ylab='Standardized Residual')
> # One possible high outlier, one low
> n = length(nsales); n
[1] 150
> # Studentized deleted residuals are t-statistics
> sdr = rstudent(auto) # Studentized deleted residuals
> # Bonferroni critical value for n=200 tests, at joint alpha = 0.05 level
> dfe = auto$df.residual; dfe
[1] 144
> alpha = 0.05; a = alpha/n; bcrit = qt(1-a/2,dfe-1); bcrit
[1] 3.676863
> sdr[abs(sdr)>bcrit]
      37
4.001458
>
> # Looks like one high outlier. Keep investigating.
```



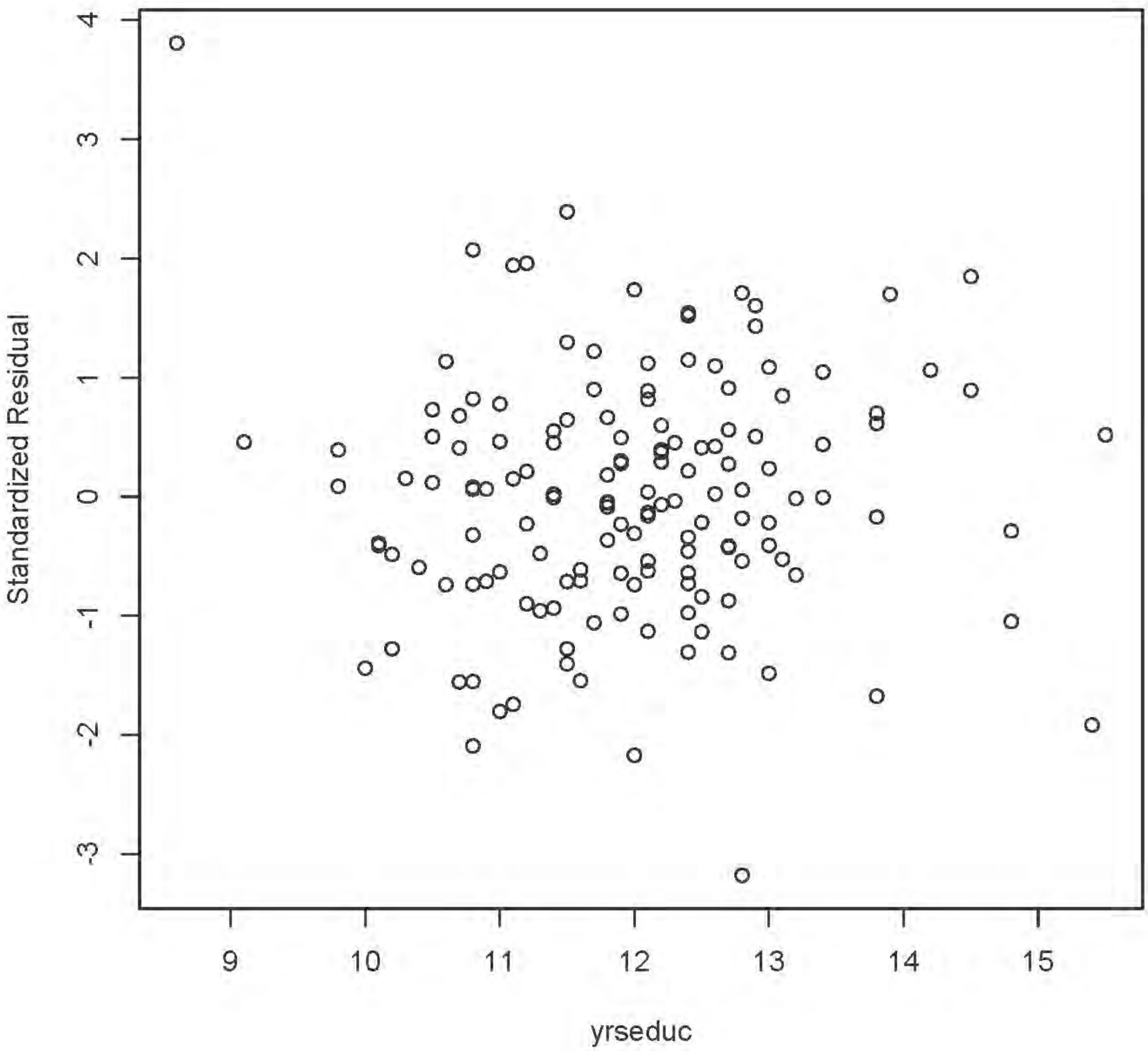
```
> # Plot standardized residuals against variables in the model
> plot(salesforce,sr,ylab='Standardized Residual')
```



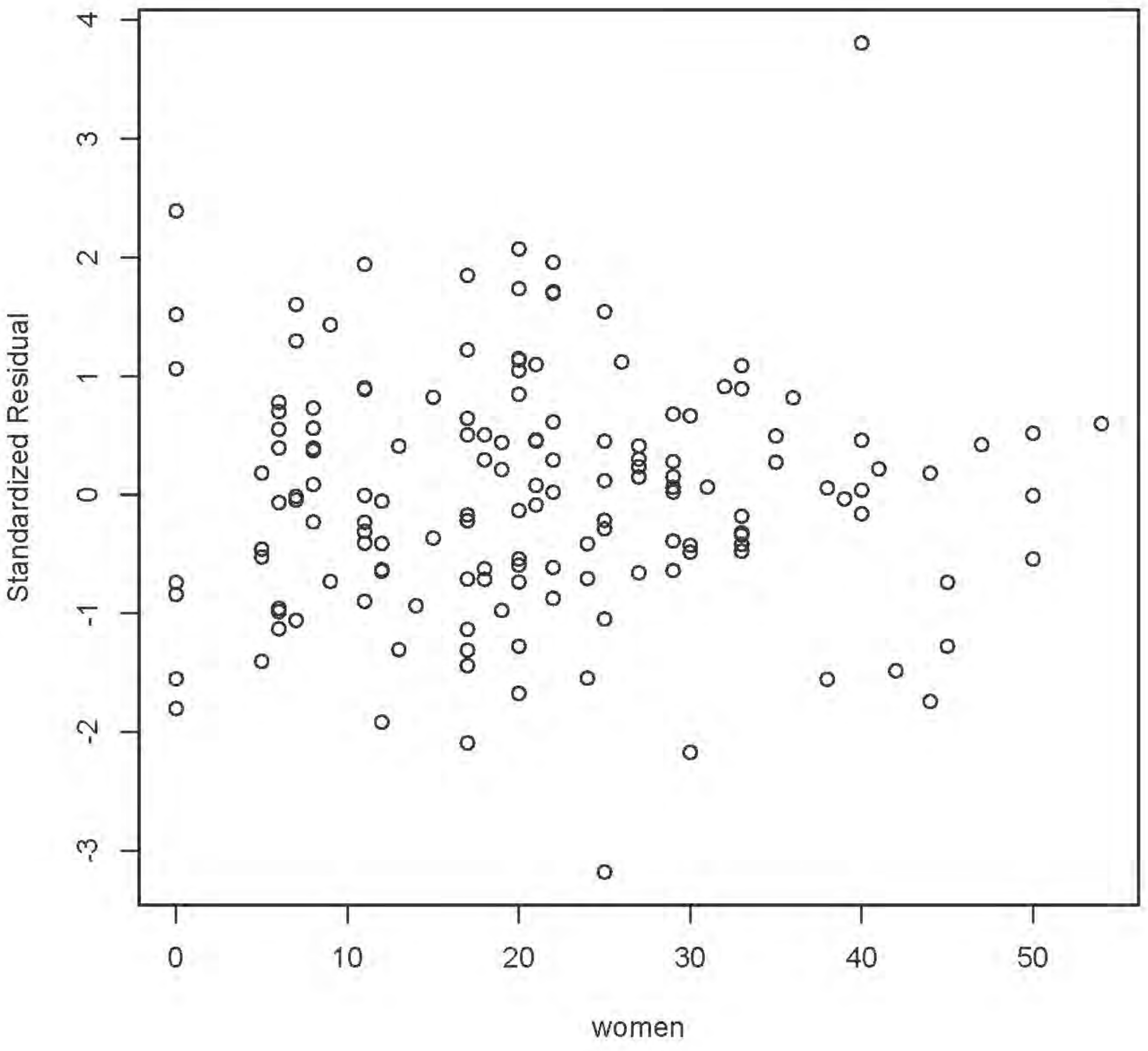
```
> # Possible non-constant variance
> plot(yrsexp,sr,ylab='Standardized Residual')
```



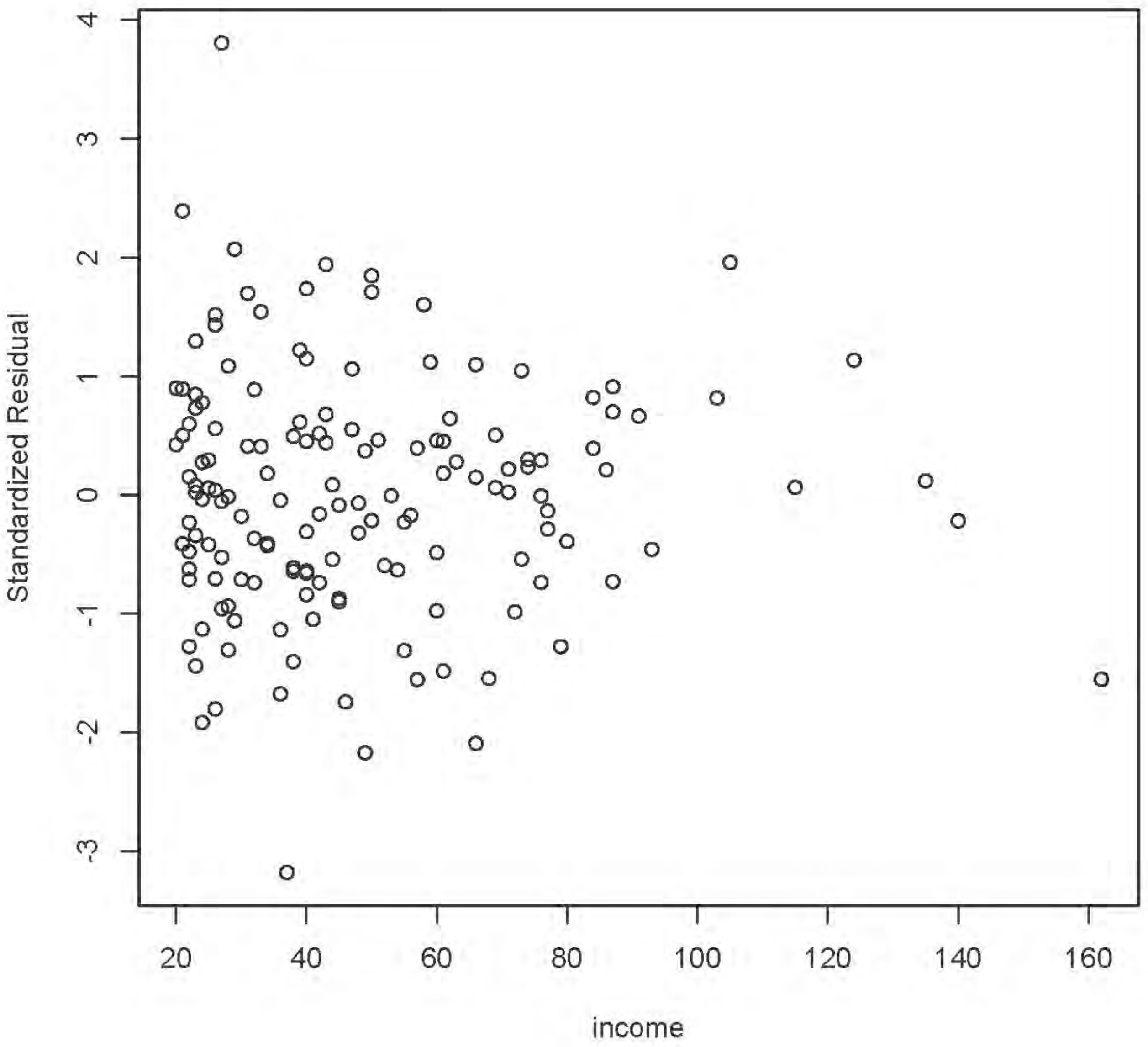
```
> plot(yrseduc,sr,ylab='Standardized Residual')
```



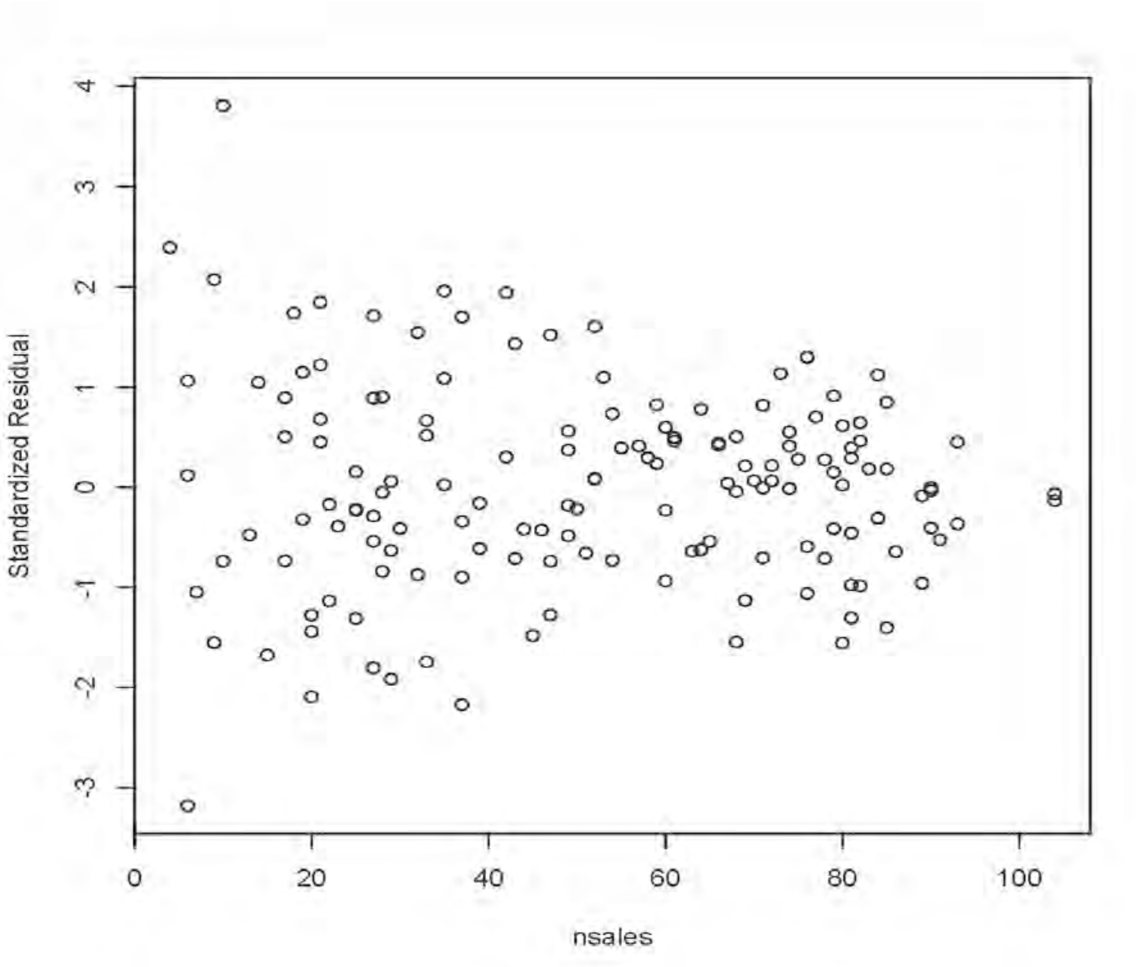
```
> plot(women,sr,ylab='Standardized Residual')
```



```
> plot(income,sr,ylab='Standardized Residual')
```



```
> # Plot standardized residuals against variable(s) NOT in the model
> plot(nsales,sr,ylab='Standardized Residual')
```



```
> # Likely non-constant variance, and it makes sense.
> # Var(x-bar) = sigma-squared/n
```

We need ways to deal with non-constant variance.

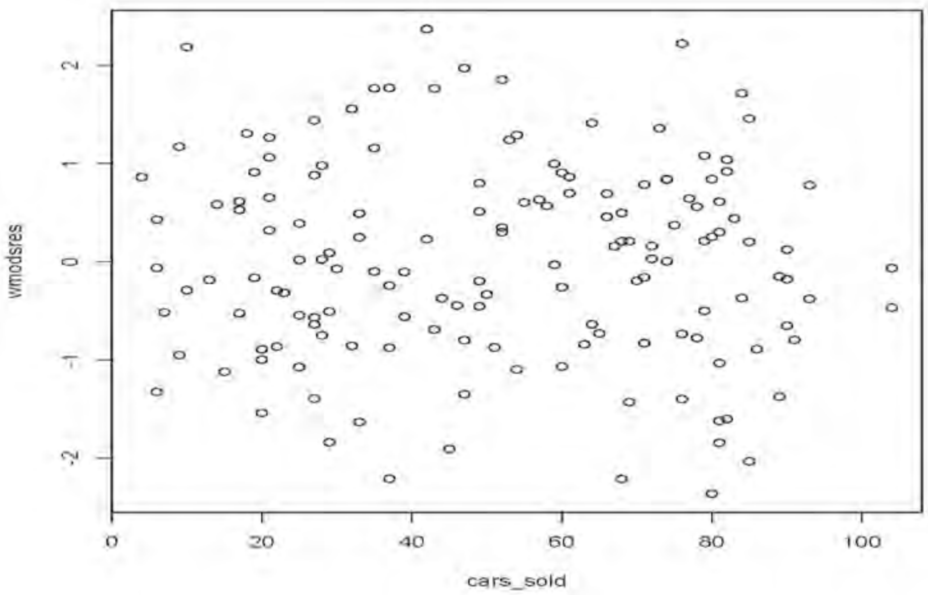
```
> help(lm)
```

`weights` an optional vector of weights to be used in the fitting process. Should be `NULL` or a numeric vector. If non-NULL, weighted least squares is used with weights `weights` (that is, minimizing `sum(w*e^2)`); otherwise ordinary least squares is used. See also 'Details',

Non-`NULL` `weights` can be used to indicate that different observations have different variances (with the values in `weights` being inversely proportional to the variances); or equivalently, when the elements of `weights` are positive integers *w_i*, that each response *y_i* is the mean of *w_i* unit-weight observations (including the case that there are *w_i* observations equal to *y_i* and the data have been summarized).

```
> wmod = lm(avprice ~ salesforce + yrsexp + yrseduc + women + income,
weights = nsales, data=carsales)
>
> summary(wmod)

Call:
lm(formula = avprice ~ salesforce + yrsexp + yrseduc + women +
    income, data = carsales, weights = nsales)

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-1320.26  -408.78     5.58   390.12  1334.63
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5975.0290    90.4117  66.087  < 2e-16 ***
salesforce     1.7696     1.6318   1.084 0.280004    
yrsexp        16.6338     4.6872   3.549 0.000523 ***
yrseduc       -6.5467     6.6998  -0.977 0.330137    
women          1.1456     0.5423   2.113 0.036357 *  
income         1.0846     0.2741   3.957 0.000119 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 575 on 144 degrees of freedom
Multiple R-squared: 0.1869,   Adjusted R-squared: 0.1587 
F-statistic: 6.621 on 5 and 144 DF,  p-value: 1.407e-05

> wmodsres = rstandard(wmod)
> cars_sold = carsales$nsales ; plot(cars_sold,wmodsres)
```


wmodsres
cars_sold