HW3

David Schultheiss

9/21/2020

##Problem 1 a)

```
set.seed(1)
x = runif(100,0,1)
y = 16 + 12*x + rnorm(100, mean=0, sd=0.5)
SimulatedData = data.frame(cbind(x,y))
```

B0= 16 , B1= 12 , Variance = 1 , Variance of residuals = .5. R Squared is .9737 so y explains 97.37% of the variance

b)

```
reg1 = lm(data= SimulatedData, y~x)
summary(reg1)
```

```
##
## Call:
## lm(formula = y ~ x, data = SimulatedData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.92489 -0.28111 -0.04353  0.26214  1.25830
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.9103     0.1029  154.61   <2e-16 ***
## x            12.1562     0.1767   68.78   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4705 on 98 degrees of freedom
## Multiple R-squared:  0.9797, Adjusted R-squared:  0.9795
## F-statistic:  4731 on 1 and 98 DF,  p-value: < 2.2e-16
```

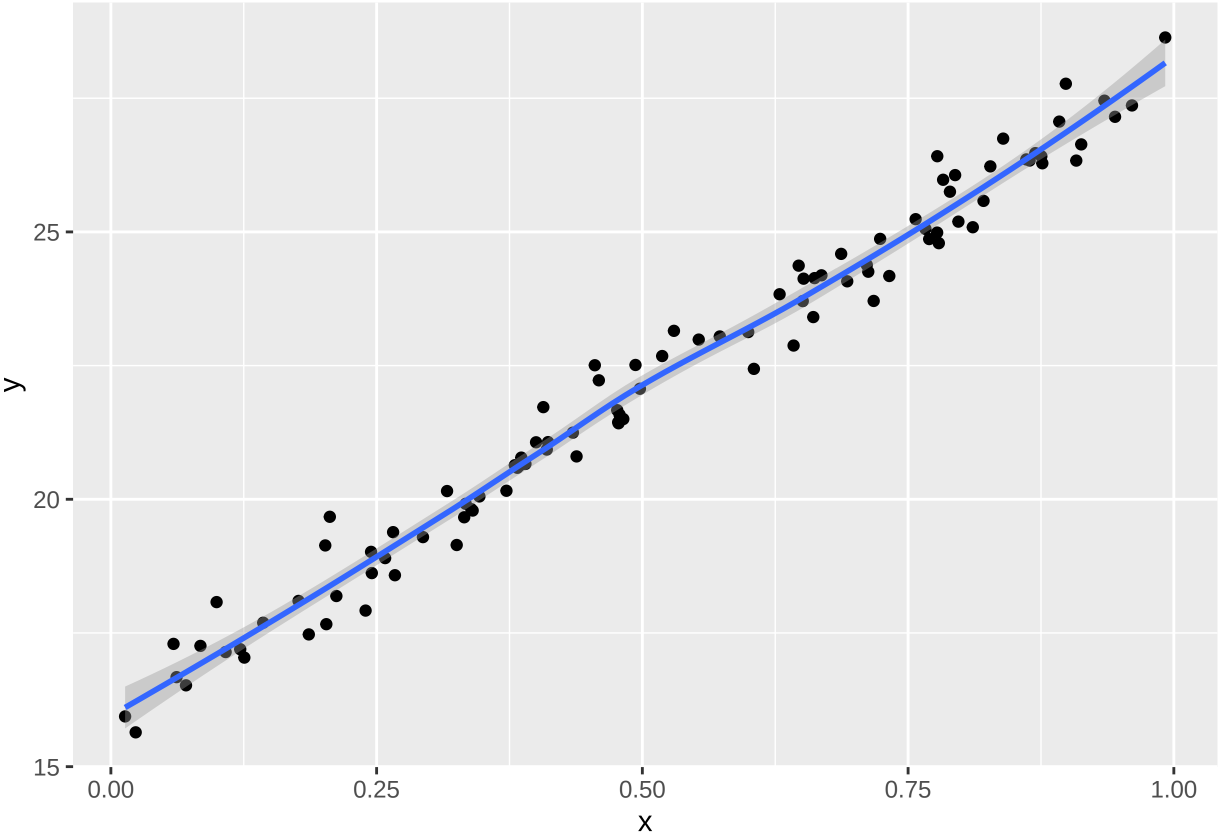
c)

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.2
```

```
ggplot(data= SimulatedData, mapping = aes(x= x, y= y)) +
  geom_point() +
  geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



d)

```
y2 = 16 + 12*x + rnorm(100, mean=0, sd=2)
SimulatedData2 = data.frame(cbind(x,y))

reg2 = lm(data= SimulatedData2, y2~x)
summary(reg2)
```

```
##
## Call:
## lm(formula = y2 ~ x, data = SimulatedData2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7520 -1.3692 -0.1497  1.2277  4.7963
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.2355     0.4605   35.25   <2e-16 ***
## x            11.6331     0.7909   14.71   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.106 on 98 degrees of freedom
## Multiple R-squared:  0.6882, Adjusted R-squared:  0.685
## F-statistic: 216.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

With more variance in the data, R squared falls. Estimates of B0/B1 are also less accurate.

e)

```
reg3 = lm(data= SimulatedData2, y2~x + I(x^2))
summary(reg3)
```

```
##
## Call:
## lm(formula = y2 ~ x + I(x^2), data = SimulatedData2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6411 -1.1869 -0.1057  1.2774  4.6339
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.9545     0.7528  22.522   <2e-16 ***
## x             7.7092     3.3488   2.302   0.0235 *
## I(x^2)        3.8726     3.2119   1.206   0.2309
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.101 on 97 degrees of freedom
## Multiple R-squared:  0.6928, Adjusted R-squared:  0.6865
## F-statistic: 109.4 on 2 and 97 DF,  p-value: < 2.2e-16
```

```
reg4 = lm(data= SimulatedData2, y2~x + I(x^3))
summary(reg4)
```

```
##
## Call:
## lm(formula = y2 ~ x + I(x^3), data = SimulatedData2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6086 -1.2037 -0.0801  1.2921  4.6077
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   16.750      0.657  25.493  < 2e-16 ***
## x              9.520      2.083   4.570 1.43e-05 ***
## I(x^3)         2.344      2.138   1.096    0.276
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.104 on 97 degrees of freedom
## Multiple R-squared:  0.692,  Adjusted R-squared:  0.6857
## F-statistic:   109 on 2 and 97 DF,  p-value: < 2.2e-16
```

```
reg5 = lm(data= SimulatedData2, y2~x + I(x^8))
summary(reg5)
```

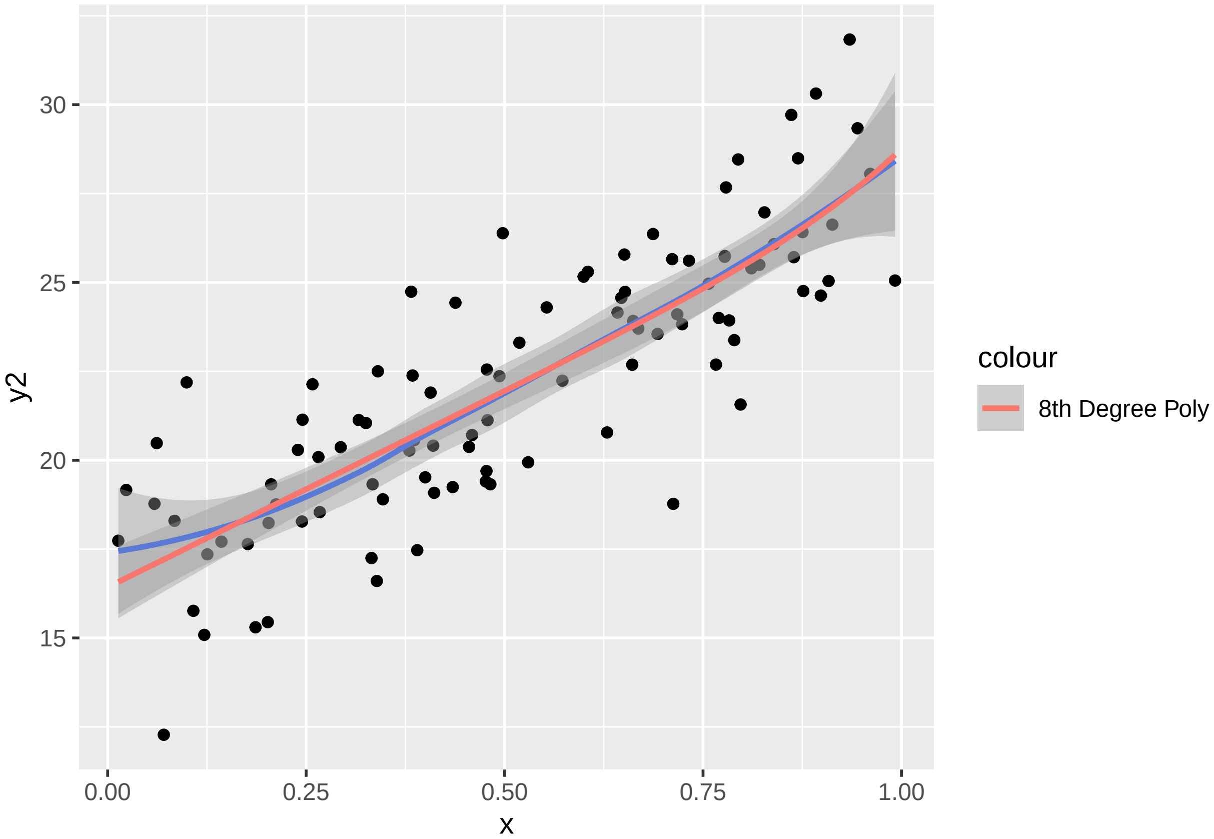
```
##
## Call:
## lm(formula = y2 ~ x + I(x^8), data = SimulatedData2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5989 -1.3153 -0.1187  1.2158  4.6647
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.4274     0.5268  31.184  < 2e-16 ***
## x            11.0273     1.1276   9.779 4.02e-16 ***
## I(x^8)        1.3062     1.7291   0.755    0.452
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.11 on 97 degrees of freedom
## Multiple R-squared:  0.6901, Adjusted R-squared:  0.6837
## F-statistic:   108 on 2 and 97 DF,  p-value: < 2.2e-16
```

The R-Squared goes up a very small amount for each iteration of the model above. This does not mean the model is improving, however. Each of the terms is insignificant (we know our real model is linear). Adding more terms will not reduce R-Squared but will lead to overfitting.

f)

```
ggplot(data= SimulatedData2, mapping = aes(x= x, y= y2)) +
  geom_point() +
  geom_smooth() +
  stat_smooth(aes(color= '8th Degree Poly'),
              method= 'lm', formula = y2~x + I(x^8))
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



##Problem 2 a)

```r
library(MASS)
data('UScereal')
summary(UScereal)
```

```
##  mfr       calories         protein             fat            sodium     
##  G:22   Min.   : 50.0   Min.   : 0.7519   Min.   :0.000   Min.   :  0.0  
##  K:21   1st Qu.:110.0   1st Qu.: 2.0000   1st Qu.:0.000   1st Qu.:180.0  
##  N: 3   Median :134.3   Median : 3.0000   Median :1.000   Median :232.0  
##  P: 9   Mean   :149.4   Mean   : 3.6837   Mean   :1.423   Mean   :237.8  
##  Q: 5   3rd Qu.:179.1   3rd Qu.: 4.4776   3rd Qu.:2.000   3rd Qu.:290.0  
##  R: 5   Max.   :440.0   Max.   :12.1212   Max.   :9.091   Max.   :787.9  
##      fibre            carbo           sugars          shelf      
##  Min.   : 0.000   Min.   :10.53   Min.   : 0.00   Min.   :1.000  
##  1st Qu.: 0.000   1st Qu.:15.00   1st Qu.: 4.00   1st Qu.:1.000  
##  Median : 2.000   Median :18.67   Median :12.00   Median :2.000  
##  Mean   : 3.871   Mean   :19.97   Mean   :10.05   Mean   :2.169  
##  3rd Qu.: 4.478   3rd Qu.:22.39   3rd Qu.:14.00   3rd Qu.:3.000  
##  Max.   :30.303   Max.   :68.00   Max.   :20.90   Max.   :3.000  
##    potassium          vitamins 
##  Min.   : 15.00   100%    : 5  
##  1st Qu.: 45.00   enriched:57  
##  Median : 96.59   none    : 3  
##  Mean   :159.12                
##  3rd Qu.:220.00                
##  Max.   :969.70                
```

We can see fat and sugar both have min values of 0, so there are both fat free and sugar free cereals included.G is most common manufacturer. Mean fiber content is 2g.

b/c)

```
library(GGally)
```

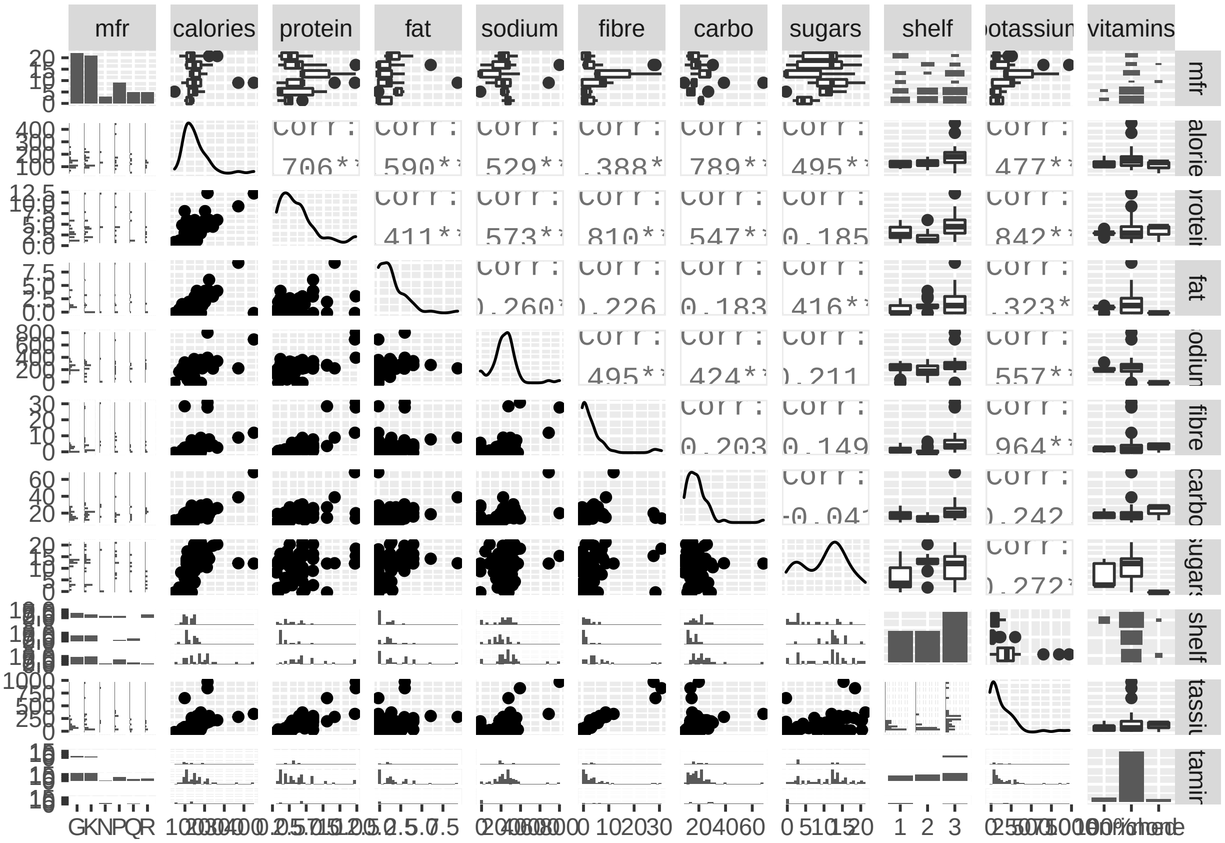
```
## Warning: package ’GGally’ was built under R version 3.6.2
```

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

```
UScereal$shelf = factor(UScereal$shelf)
ggpairs(UScereal)
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



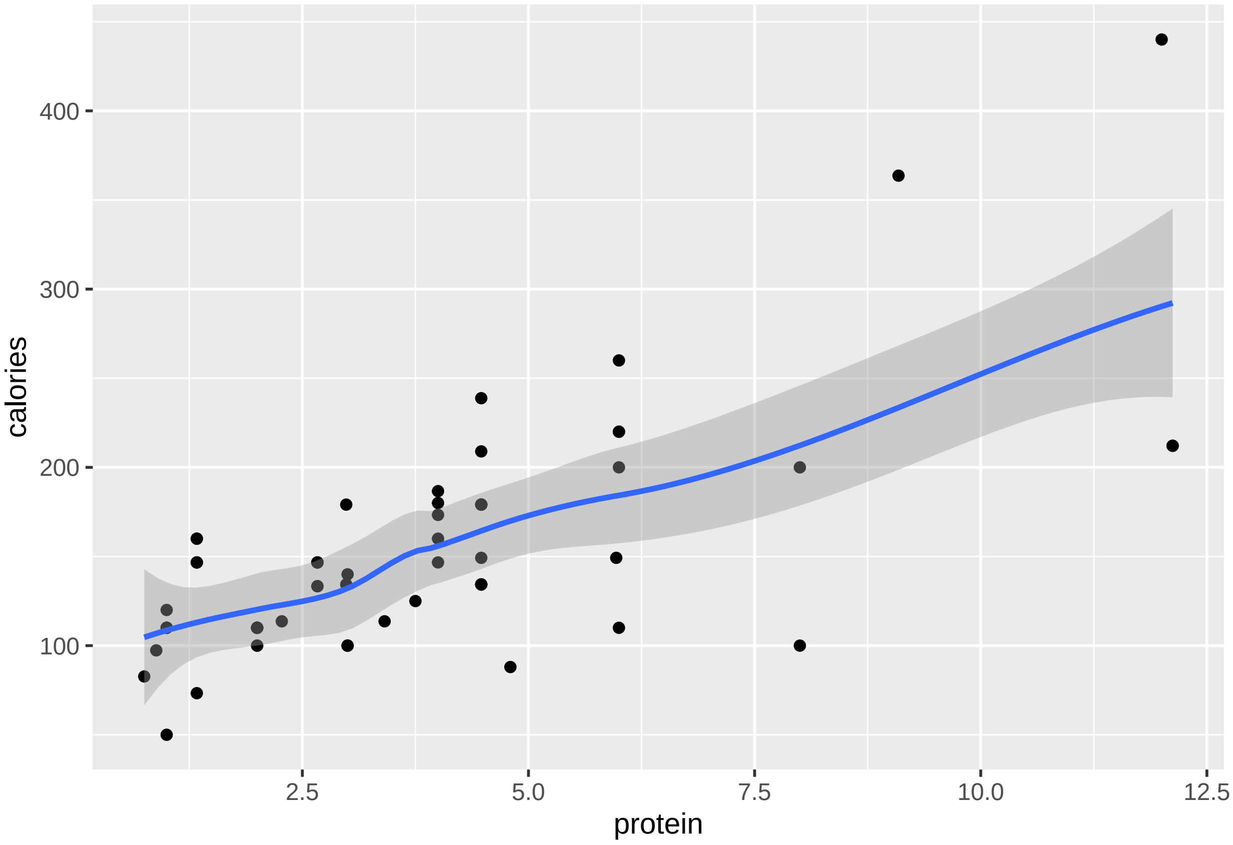
Calories has a clear relationship with fat, protein, and sugar. I am surprised protein has a higher correlation to calories than fat. I'm pretty sure protein is 4 calories/gram, and fat is double that.

Potassium and fiber have an extremely high correlation (.96) Potassium and protein also have a high correlation (.84)

d)

```
ggplot(data= UScereal, mapping = aes(x= protein, y= calories)) +
  geom_point() +
  geom_smooth()
```

```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



e)

```
reg6= lm(data= UScereal, calories~protein)
summary(reg6)
```

```
##
## Call:
## lm(formula = calories ~ protein, data = UScereal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -121.379  -21.379    0.883   16.458  151.925
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   87.986      9.528   9.234 2.55e-13 ***
## protein       16.674      2.107   7.913 5.07e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44.55 on 63 degrees of freedom
## Multiple R-squared:  0.4985, Adjusted R-squared:  0.4905
## F-statistic: 62.61 on 1 and 63 DF,  p-value: 5.071e-11
```

```
confint(reg6)
```

```
##                2.5 %    97.5 %
## (Intercept) 68.94460 107.02658
## protein     12.46312  20.88519
```

Y = 87.99 + 16.67*protein

49.8% of the variance explained by the model. A 1 unit increase in protein causes calories to rise by 16.67.

f)

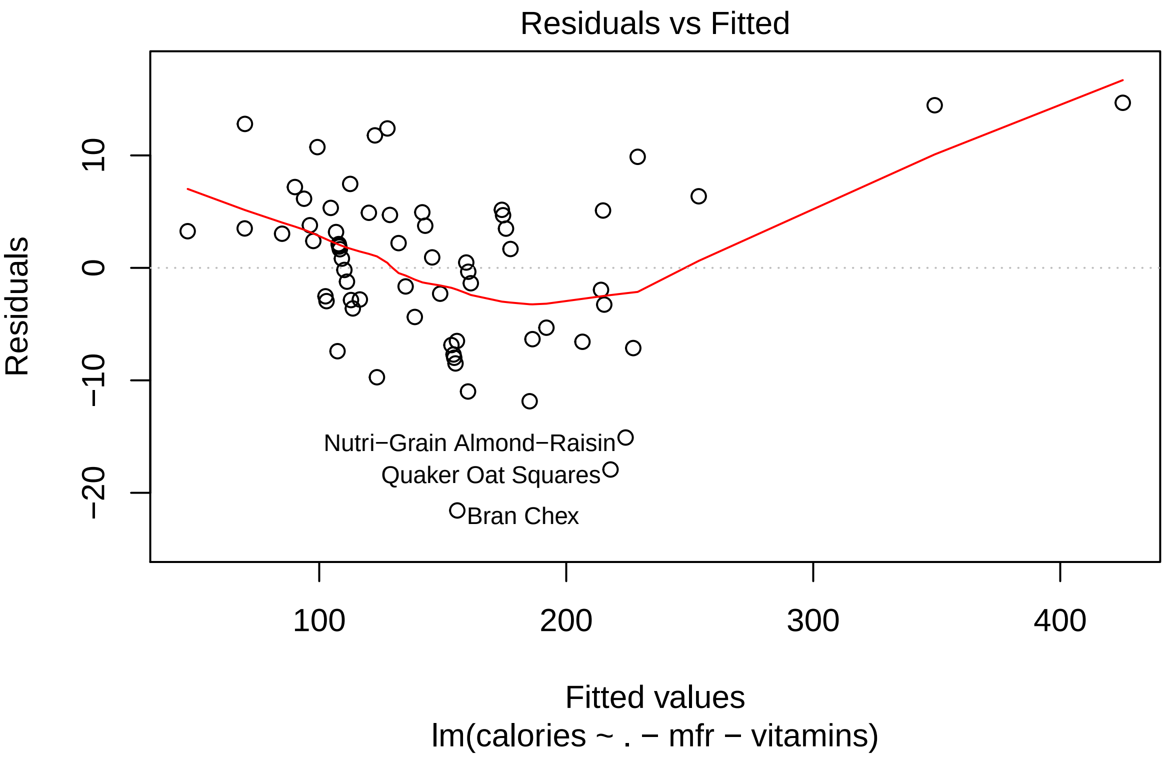
```
reg7= lm(data= UScereal, calories~. -mfr -vitamins)
summary(reg7)
```

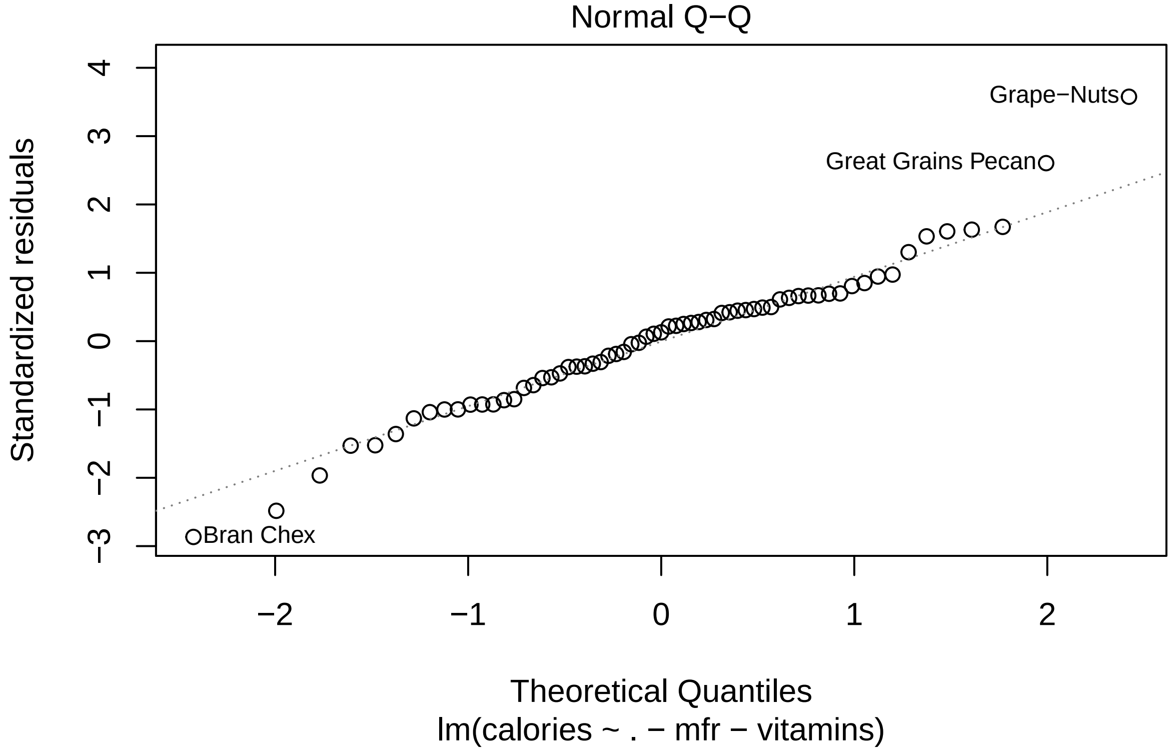
```
##
## Call:
## lm(formula = calories ~ . - mfr - vitamins, data = UScereal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.5687  -4.3661   0.9305   4.7052  14.6765
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -21.07195    3.46940  -6.074 1.22e-07 ***
## protein       4.90350    0.98867   4.960 7.15e-06 ***
## fat           9.26014    0.75644  12.242  < 2e-16 ***
## sodium        0.01193    0.01010   1.182 0.242456
## fibre         2.73587    0.74230   3.686 0.000523 ***
## carbo         4.88255    0.17850  27.353  < 2e-16 ***
## sugars        4.50385    0.23030  19.557  < 2e-16 ***
## shelf2        1.76313    3.05034   0.578 0.565615
## shelf3        1.12629    2.81599   0.400 0.690734
## potassium    -0.11271    0.02834  -3.978 0.000205 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 8.008 on 55 degrees of freedom
## Multiple R-squared:  0.9859, Adjusted R-squared:  0.9835
## F-statistic: 425.8 on 9 and 55 DF,  p-value: < 2.2e-16
```

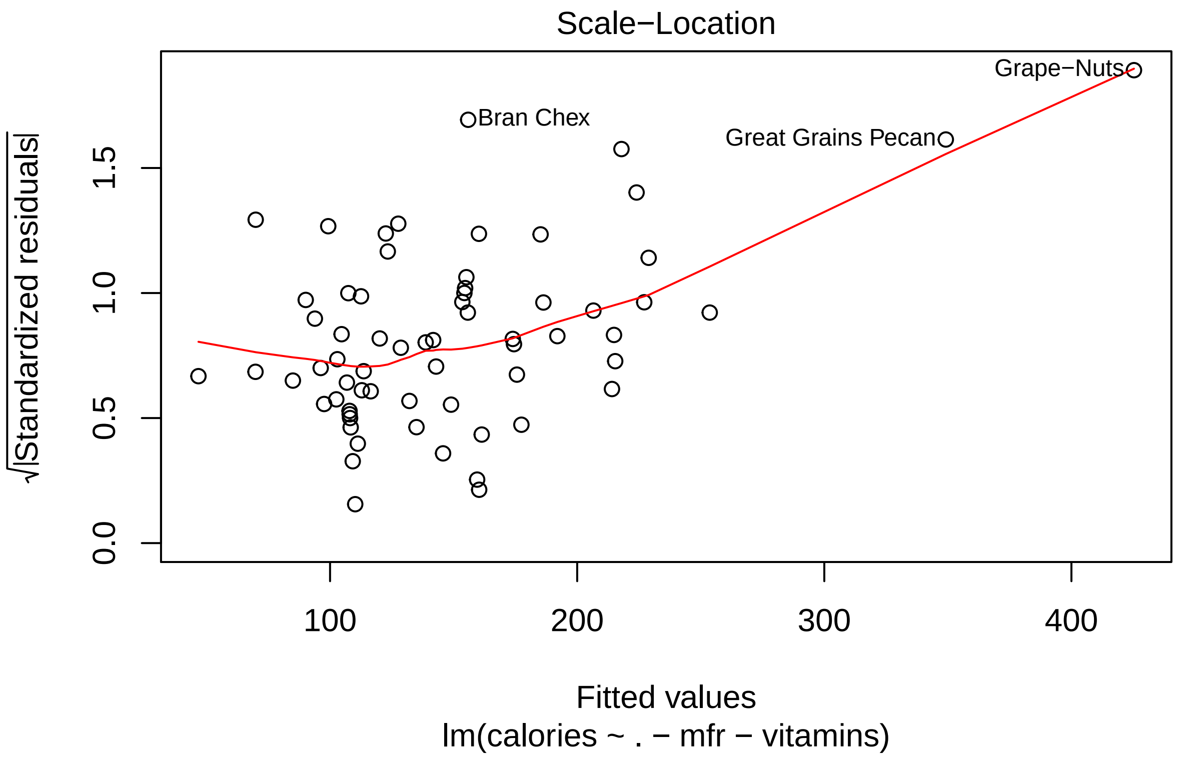
Protein, fat, fiber, carbo, sugars, and potassium are all significant. A negative coefficient for potassium means cereals with more potassium had slighlty lower calories. The shelf variables were insignificant.

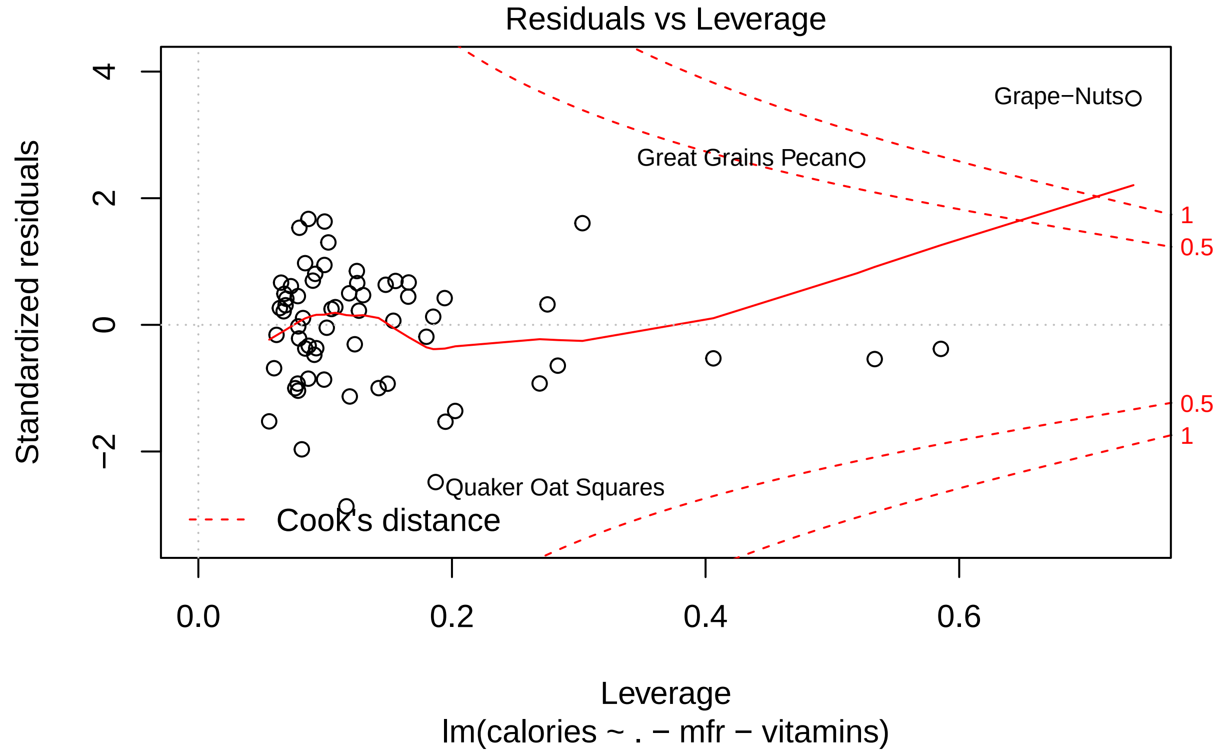
g)

```
plot(reg7)
```


Residuals vs Fitted
Residuals
10
0
−10
−20
100
200
300
400
Nutri−Grain Almond−Raisin
Quaker Oat Squares
Bran Chex
Fitted values
lm(calories ~ . − mfr − vitamins)


Normal Q−Q
Standardized residuals
4
3
2
1
0
−1
−2
−3
−2
−1
0
1
2
Grape−Nuts
Great Grains Pecan
Bran Chex
Theoretical Quantiles
lm(calories ~ . − mfr − vitamins)





Our residual plot looks good, there are a couple of high leverage point off to the right. Data looks linear and no heteroscedasticity. here may be some collinearity. As we saw in our pairs plot, some variables were highly correlated.

Two problematic points coming from ‘Grape-Nuts’ and ‘Great Grains Pecan’.