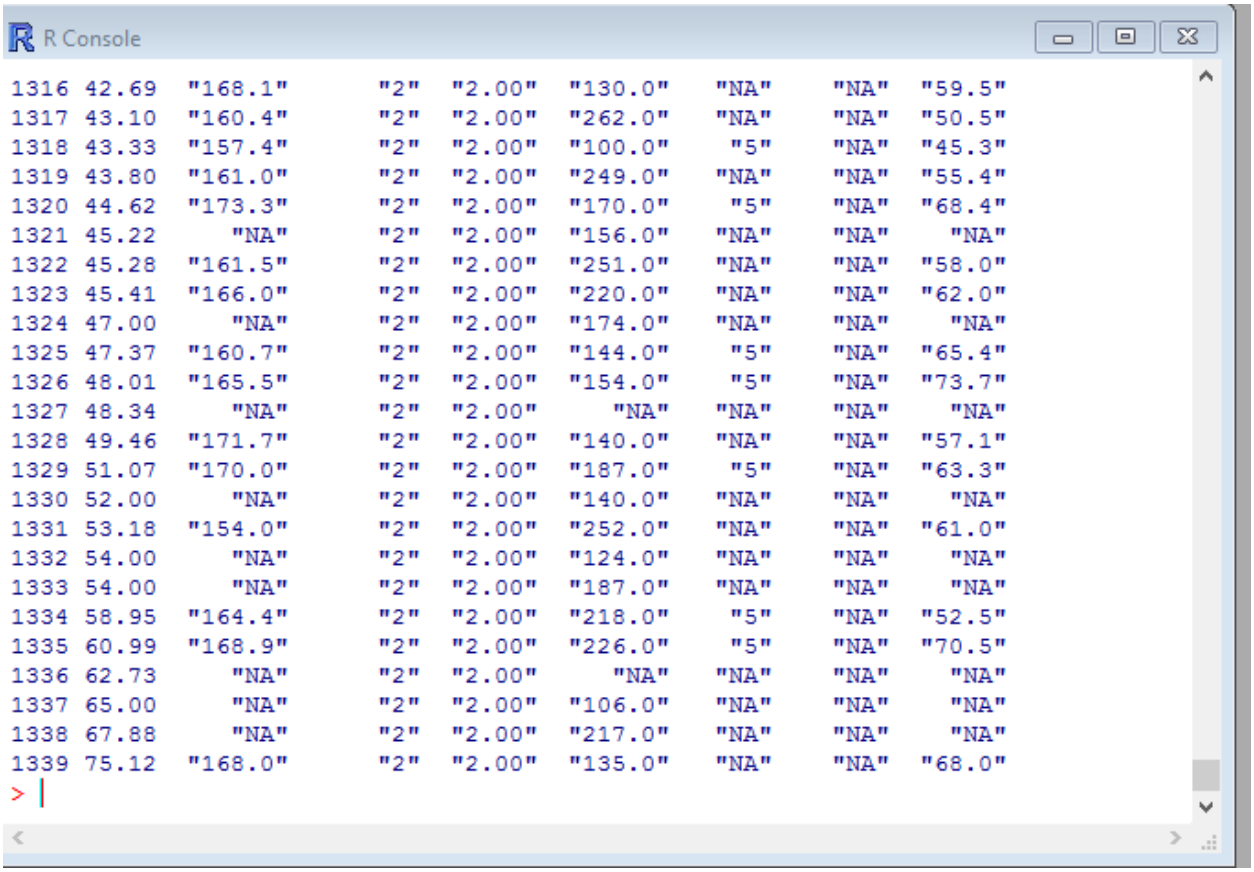
Daniel Hanks Jr
IST687 - Exercise 9: Making Predictions

juul <-read.csv("juul2.csv", header=TRUE)

#read dataset into R.



#create variables based on csv file

age <- juul$age

height <- juul$height

menarche <- juul$menarche

sex <- juul$sex

igf1 <- juul$igf1

tanner <- juul$tanner

testvol <- juul$testvol

weight <- juul$weightstr(juul)

#inspect data confirming 1139 observations of 8 variables

```
> str(juul)
'data.frame':   1339 obs. of  8 variables:
 $ age     : num  NA NA NA NA NA 0.17 0.17 0.17 0.17 0.17 ...
 $ height  : Factor w/ 600 levels " \"110.8\"," \"111.5\"",..: 600 600 600 600$
 $ menarche: Factor w/ 3 levels " \"1\"," \"2\"",..: 3 3 3 3 3 3 3 3 3 ...
 $ sex     : Factor w/ 3 levels " \"1.00\"," \"2.00\"",..: 3 3 3 3 3 1 1 1 1 1$
 $ igf1    : Factor w/ 501 levels " \"100.0\"," \"101.0\"",..: 490 487 53 55 2$
 $ tanner  : Factor w/ 6 levels " \"1\"," \"2\"",..: 6 6 6 6 6 1 1 1 1 1 ...
 $ testvol : Factor w/ 26 levels " \"1\"," \"10\"",..: 26 26 26 26 26 26 26 26$
 $ weight  : Factor w/ 518 levels " \"14.1\"," \"17.9\"",..: 518 518 518 518 5$
>
```

#I was unable to get complte.cases() to work so I ended up filtering the data in Excel to remove the #cases that were missing data. I then redid the above steps with the new data file juul_filtered.csv.

#create data frame with 8 variables
juulDF <- data.frame(age, height, menarche, sex, igf1, tanner, testvol, weight)

R Console

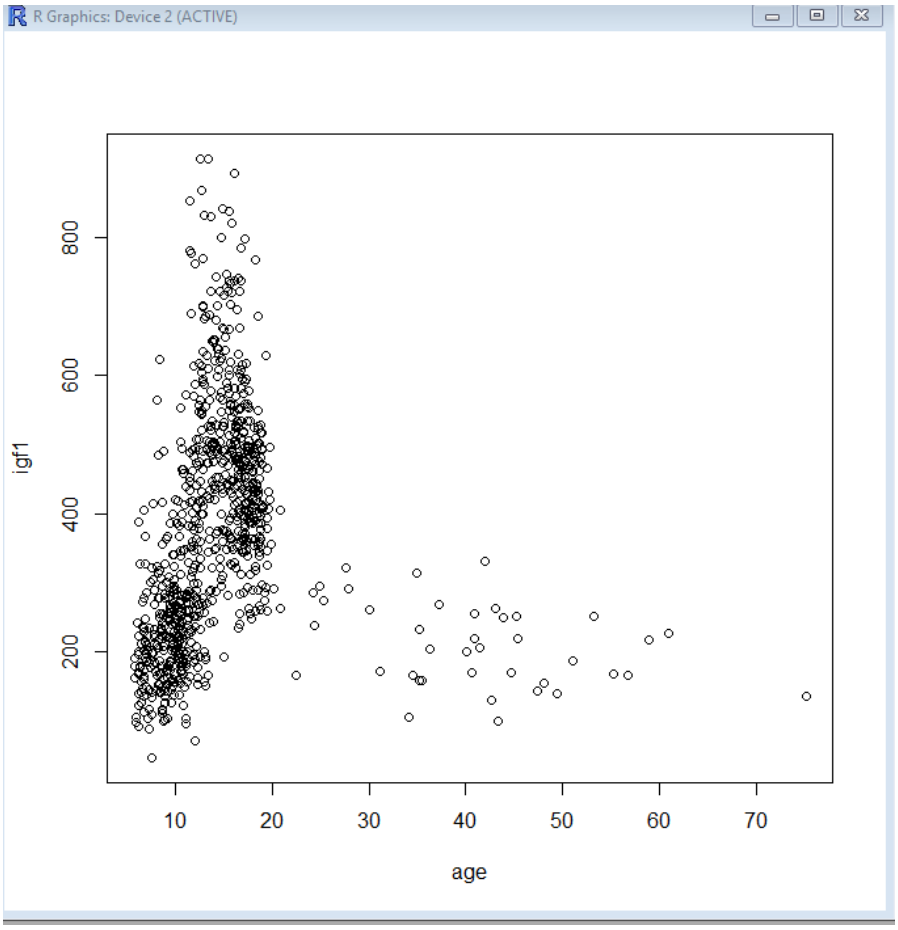
```
> juulDF
      age height menarche sex igf1 tanner testvol weight
1    6.00  111.6       NA   1   98     98       1   19.1
2    6.08  116.7       NA   1  242    242       1   21.7
3    6.26  120.3       NA   1  196    196       1   24.7
4    6.40  115.5       NA   1  179    179       1   19.6
5    6.42  115.6       NA   1  126    126       1   20.6
6    6.43  116.1       NA   1  142    142       1   20.2
7    6.61  130.3       NA   1  236    236       1   28.0
8    6.63  122.2       NA   1  148    148       2   21.6
9    6.70  126.2       NA   1  174    174       1   26.1
10   6.72  125.6       NA   1  136    136       1   22.6
11   6.72  121.0       NA   1  164    164       1   24.4
12   6.76  123.2       NA   1  160    160       1   22.8
13   6.84  122.5       NA   1  215    215       1   24.4
14   6.89  126.1       NA   1  214    214      NA   19.9
15   6.90  133.7       NA   1  328    328       1   28.0
16   6.91  119.2       NA   1  367    367       1   21.5
17   7.04  130.0       NA   1  149    149       1   27.4
18   7.07  124.2       NA   1  187    187       1   26.9
19   7.22  126.4       NA   1  103    103       1   26.4
20   7.24  123.7       NA   1  145    145       1   24.7
21   7.25  131.2       NA   1  117    117       1   28.4
22   7.26  123.1       NA   1   88     88       1   25.1
23   7.29  131.3       NA   1  186    186       1   26.2
```

#Test data frame for the expected 858 observations as show here:

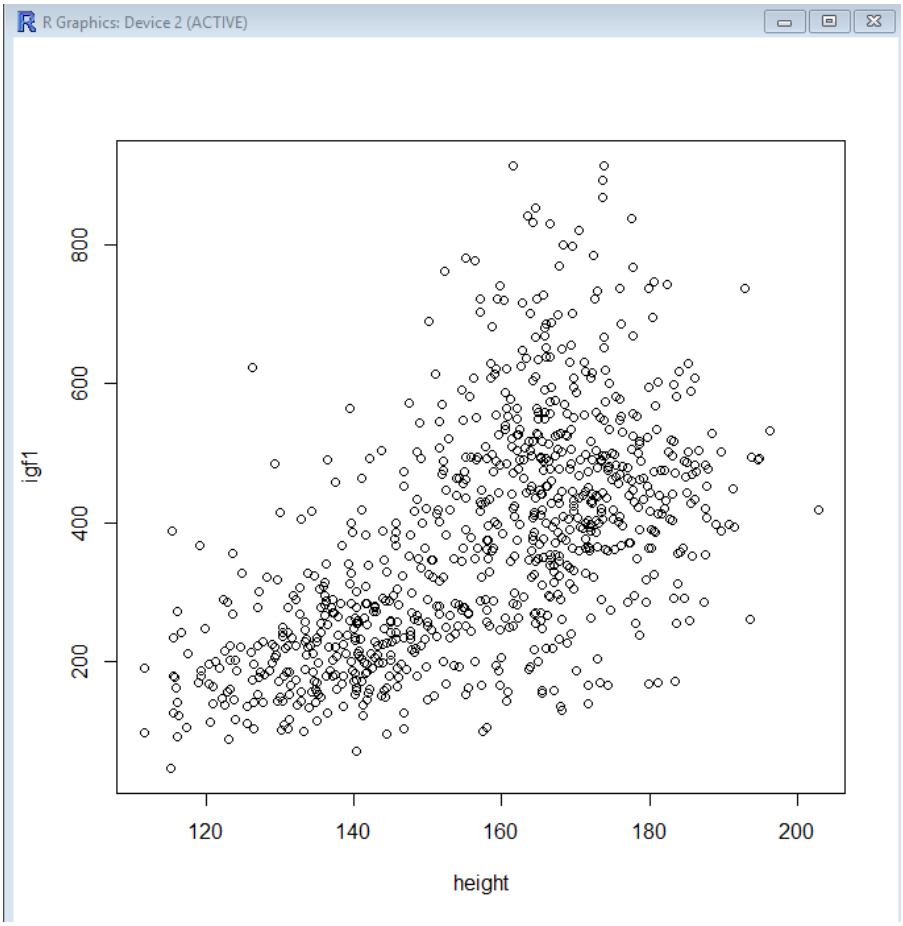
```
> str(juulDF)
'data.frame':   858 obs. of  8 variables:
 $ age      : num  6 6.08 6.26 6.4 6.42 6.43 6.61 6.63 6.7 6.72 ...
 $ height   : num  112 117 120 116 116 ...
 $ menarche: Factor w/ 3 levels " NA","1","2": 1 1 1 1 1 1 1 1 1 ...
 $ sex      : int  1 1 1 1 1 1 1 1 1 ...
 $ igf1     : int  98 242 196 179 126 142 236 148 174 136 ...
 $ tanner   : int  98 242 196 179 126 142 236 148 174 136 ...
 $ testvol : Factor w/ 25 levels " NA","1","10",..: 2 2 2 2 2 2 2 12 2 2 ...
 $ weight   : num  19.1 21.7 24.7 19.6 20.6 20.2 28 21.6 26.1 22.6 ...
```

#The dependent variable in this exercise is IGF1 so it should be on the Y-axis as shown:

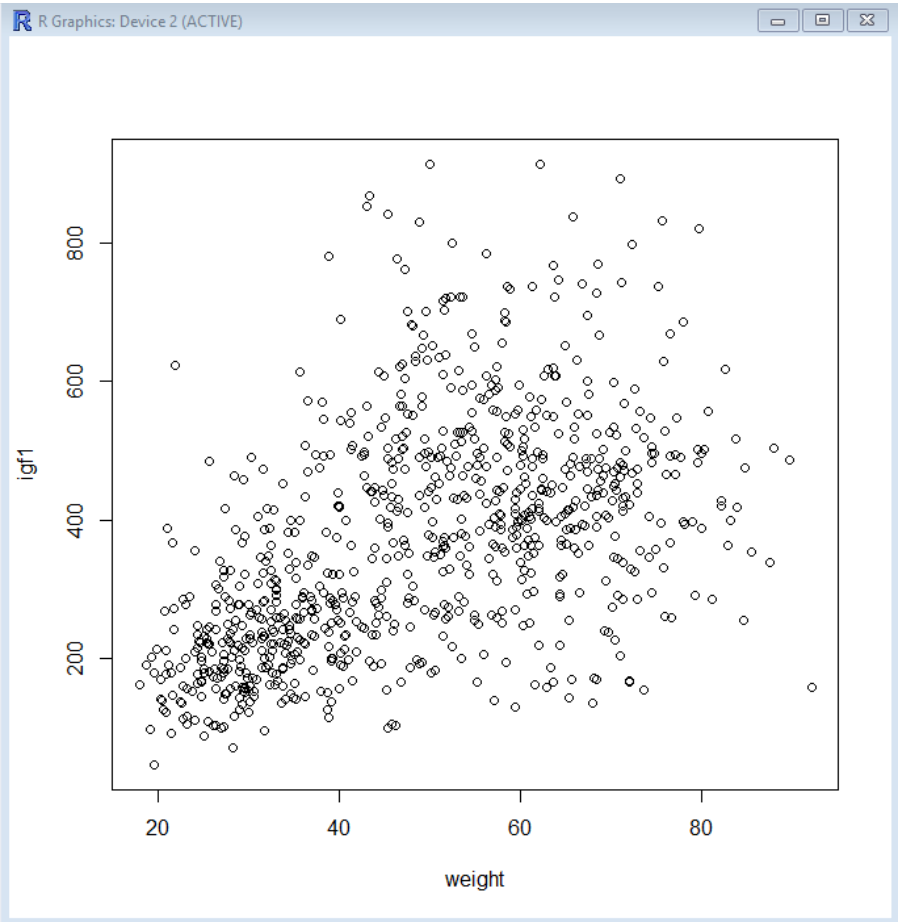
Plot (igf1~age)



plot(igf1 ~ height)



plot(igf1 ~ weight)

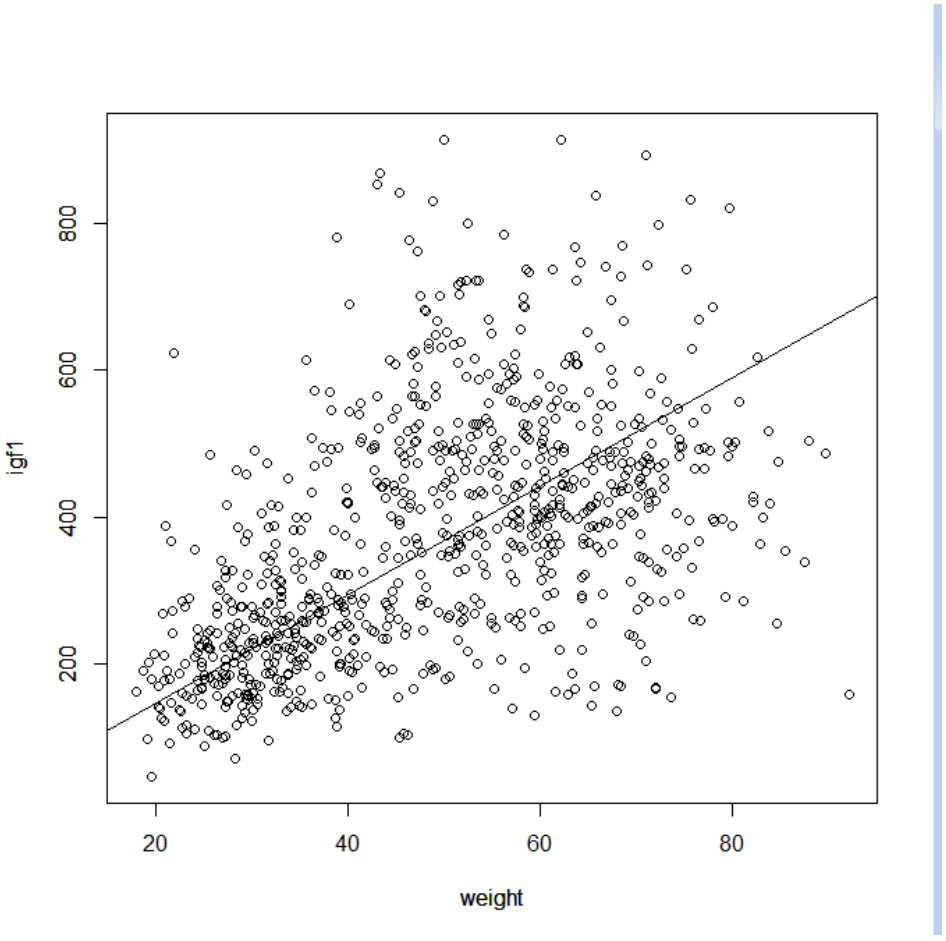


#create linear model and show summary of it

```
> model1 <-lm(igf1 ~ weight-1)
> summary(model1)

Call:
lm(formula = igf1 ~ weight - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-520.90  -69.80   -6.03   91.11  547.96

Coefficients:
       Estimate Std. Error t value Pr(>|t|)
weight  7.37418    0.09772   75.46   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 146.9 on 857 degrees of freedom
Multiple R-squared:  0.8692,    Adjusted R-squared:  0.869
F-statistic:  5694 on 1 and 857 DF,  p-value: < 2.2e-16

> |
```

#add a line of best fit based on model

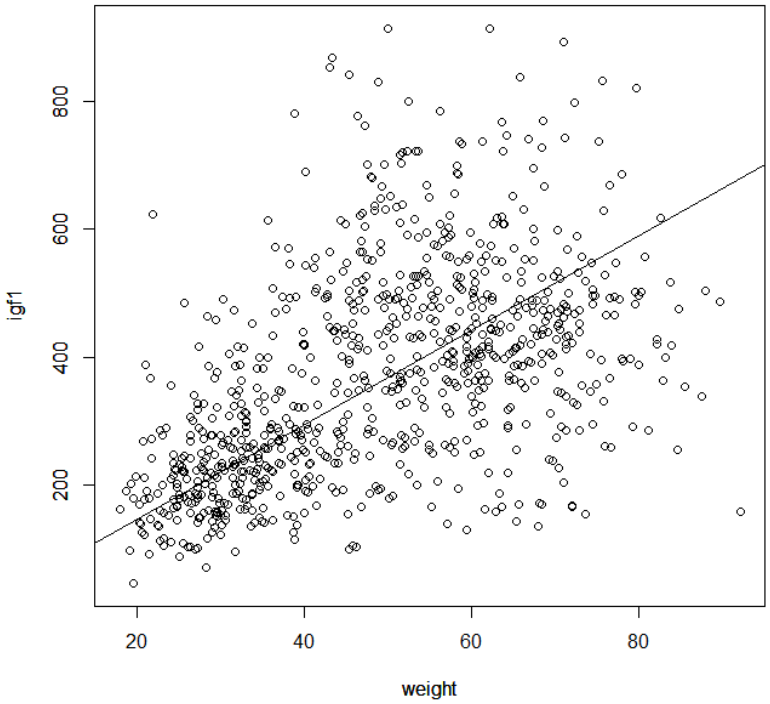
abline(model1)



#create second linear model and show summary

```
> model2 <-lm(igf1 ~ weight + height)
> summary(model2)

Call:
lm(formula = igf1 ~ weight + height)

Residuals:
    Min      1Q  Median      3Q     Max
-341.26  -87.38  -18.53   74.62  515.58

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -546.7308    71.1492  -7.684 4.22e-14 ***
weight        -1.2150     0.7260  -1.674   0.0946 .
height         6.2308     0.6525   9.550  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 135 on 855 degrees of freedom
Multiple R-squared:  0.3369,    Adjusted R-squared:  0.3353
F-statistic: 217.2 on 2 and 855 DF,  p-value: < 2.2e-16
```

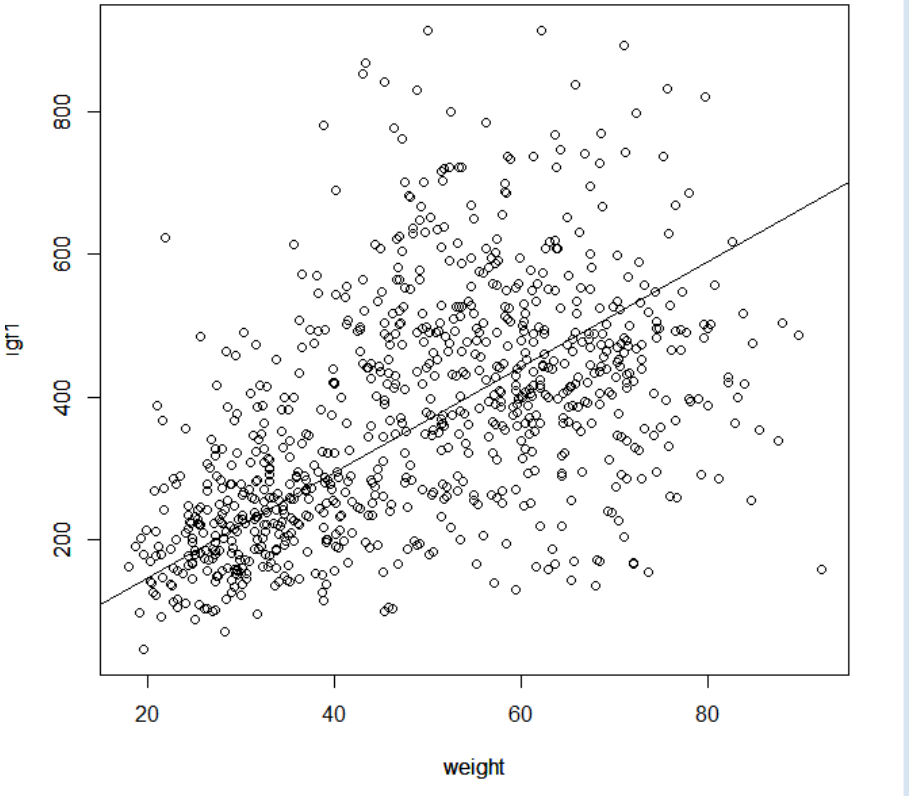
#add a line of best fit based on model
abline(model2)



#create third linear model and show summary

```
> model3 <-lm(igf1 ~ weight + height)
> summary(model3)

Call:
lm(formula = igf1 ~ weight + height)

Residuals:
    Min      1Q  Median      3Q     Max
-341.26  -87.38  -18.53   74.62  515.58

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -546.7308    71.1492  -7.684 4.22e-14 ***
weight        -1.2150     0.7260  -1.674   0.0946 .
height         6.2308     0.6525   9.550  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 135 on 855 degrees of freedom
Multiple R-squared:  0.3369,    Adjusted R-squared:  0.3353
F-statistic: 217.2 on 2 and 855 DF,  p-value: < 2.2e-16
```

Abline(model3)



Based on these models, and an r-square value of 1 meaning perfectly predicted I think the first model works the best with the R-square of .8692.