Homework 4

Zejie Gao

Due date:March 17th 2023 at 23:59 PT

1. This question uses the Auto dataset available in the ISLR package. The dataset under the name *Auto* is automatically available once the ISLR package is loaded.

```
library(ISLR)
data(Auto)
library("tidyverse")

## ── Attaching packages ─────────────────────────────────────── tidyve
rse 1.3.2 ──
## ✓ ggplot2 3.3.6     ✓ purrr   0.3.4
## ✓ tibble  3.1.8     ✓ dplyr   1.1.0
## ✓ tidyr   1.2.0     ✓ stringr 1.4.0
## ✓ readr   2.1.2     ✓ forcats 0.5.1
## ── Conflicts ────────────────────────────────────────── tidyverse_co
nflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()

library("dplyr")
library("lmtest")

## 载入需要的程辑包：zoo
##
## 载入程辑包：'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

library("MASS")

##
## 载入程辑包：'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

The dataset *Auto* contains the following information for 392 vehicles:

- mpg: miles per gallon

- cylinders: number of cylinders (between 4 and 8)
- displacement: engine displacement (cu.inches)
- horsepower: engine horsepower
- weight: vehicle weight (lbs)
- acceleration: time to accelerate from 0 to 60 mph (seconds)
- year: model year
- origin: origin of the vehicle (numerically coded as 1: American, 2: European, 3: Japanese)
- name: vehicle name

Our goal is to analyze several linear models where *mpg* is the response variable.

(a) **(2 pts)** In this data set, which predictors are qualitative, and which predictors are quantitative?
In this data set, mpg, displacement, horsepower, weight and acceleration are quantitative, and the rest of the predictors such as cylinders, year and origin are qualitative.

```
summary(Autod)

##       mpg        cylinders  displacement     horsepower        weigh
t
##  Min.   : 9.00   3:  4     Min.   : 68.0   Min.   : 46.0   Min.   :1
613
##  1st Qu.:17.00   4:199     1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2
225
##  Median :22.75   5:  3     Median :151.0   Median : 93.5   Median :2
804
##  Mean   :23.45   6: 83     Mean   :194.4   Mean   :104.5   Mean   :2
978
##  3rd Qu.:29.00   8:103     3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3
615
##  Max.   :46.60             Max.   :455.0   Max.   :230.0   Max.   :5
140
##

##   acceleration        year     origin                  name
##  Min.   : 8.00   73     : 40   1:245   amc matador       :  5
##  1st Qu.:13.78   78     : 36   2: 68   ford pinto        :  5
##  Median :15.50   76     : 34   3: 79   toyota corolla    :  5
##  Mean   :15.54   75     : 30           amc gremlin       :  4
##  3rd Qu.:17.02   82     : 30           amc hornet        :  4
##  Max.   :24.80   70     : 29           chevrolet chevette:  4
##                  (Other):193           (Other)           :365
```

(b) **(2 pts)** Fit a MLR model to the data, in order to predict mpg using all of the other predictors except for name.

For each predictor in the fitted MLR model, comment on whether you can reject the null hypothesis that there is no linear association between that predictor and mpg, conditional on the other predictors in the model. Looking at the analysis of summary table, we see that all the predictors except for acceleration and displacement have a very low p-value (less than 0.05), indicating strong evidence that there is a linear association between each of these predictors and mpg, conditional on the other predictors in the model. As acceleration, the p-value (0.3315) is greater than 0.05, suggesting that fail to reject the null hypothesis that there is no linear association between between acceleration and mpg, after controlling for the other predictors in the model. As displacement, the p-value (0.081785) is silgtly grater than 0.05; thus, they don't have linear association when using 5% significant level. Although there are variables within the predictor "year" (specifically, year71 and year72) that are not statistically significant, it is still reasonable to consider "year" as a predictor of the outcome variable due to the presence of other variables within the predictor that do show statistical significance (namely, year77 and year78).

```
lmod<- lm(mpg~ cylinders + displacement + horsepower + weight + acceler
ation + year + origin, Autod)
summary(lmod)

## 
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight + 
##     acceleration + year + origin, data = Autod)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -7.9267 -1.6678 -0.0506  1.4493 11.6002 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  30.9168415  2.3608985  13.095  < 2e-16 ***
## cylinders4    6.9399216  1.5365961   4.516 8.48e-06 ***
## cylinders5    6.6377310  2.3372687   2.840 0.004762 ** 
## cylinders6    4.2973139  1.7057848   2.519 0.012182 *  
## cylinders8    6.3668129  1.9687277   3.234 0.001331 ** 
## displacement  0.0118246  0.0067755   1.745 0.081785 .  
## horsepower   -0.0392323  0.0130356  -3.010 0.002795 ** 
## weight       -0.0051802  0.0006241  -8.300 1.99e-15 ***
## acceleration  0.0036080  0.0868925   0.042 0.966902    
## year71        0.9104285  0.8155744   1.116 0.265019    
## year72       -0.4903062  0.8038193  -0.610 0.542257    
## year73       -0.5528934  0.7214463  -0.766 0.443947    
## year74        1.2419976  0.8547434   1.453 0.147056    
## year75        0.8704016  0.8374036   1.039 0.299297    
## year76        1.4966598  0.8019080   1.866 0.062782 .  
## year77        2.9986967  0.8198949   3.657 0.000292 ***
## year78         2.9737783  0.7792185   3.816 0.000159 ***
## year79         4.8961763  0.8248124   5.936 6.74e-09 ***
## year80         9.0589316  0.8751948  10.351  < 2e-16 ***
## year81         6.4581580  0.8637018   7.477 5.58e-13 ***
## year82         7.8375850  0.8493560   9.228  < 2e-16 ***
## origin2        1.6932853  0.5162117   3.280 0.001136 **
## origin3        2.2929268  0.4967645   4.616 5.41e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.848 on 369 degrees of freedom
## Multiple R-squared:  0.8744, Adjusted R-squared:  0.8669
## F-statistic: 116.8 on 22 and 369 DF,  p-value: < 2.2e-16
```

(c) **(2 pts)** What mpg do you predict for a Japanese car with three cylinders, displacement 100, horsepower of 85, weight of 3000, acceleration of 20, built in the year 1980?

```
new_data <- data.frame(cylinders = factor(3, levels = levels(Autod$cyli
nders)),
                       displacement = 100,
                       horsepower = 85,
                       weight = 3000,
                       acceleration = 20,
                       year = factor(80, levels = levels(Autod$year)),
                       origin = factor(3, levels = levels(Autod$origin)
))
predicted_mpg <- predict(lmod, newdata = new_data, interval = "predicti
on")
predicted_mpg

##        fit      lwr      upr
## 1 24.64804 18.24614 31.04993
```

(d) **(2 pts)** On average, holding all other predictor variables fixed, what is the difference between the mpg of a Japanese car and the mpg of an European car?

Therefore, on average, holding all other predictor variables fixed, the mpg of a Japanese car is 0.5996415 higher than the mpg of an European car.

```
summary(lmod)

##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin, data = Autod)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9267 -1.6678 -0.0506  1.4493 11.6002
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  30.9168415  2.3608985  13.095  < 2e-16 ***
## cylinders4    6.9399216  1.5365961   4.516 8.48e-06 ***
## cylinders5    6.6377310  2.3372687   2.840 0.004762 ** 
## cylinders6    4.2973139  1.7057848   2.519 0.012182 *  
## cylinders8    6.3668129  1.9687277   3.234 0.001331 ** 
## displacement  0.0118246  0.0067755   1.745 0.081785 .  
## horsepower   -0.0392323  0.0130356  -3.010 0.002795 ** 
## weight       -0.0051802  0.0006241  -8.300 1.99e-15 ***
## acceleration  0.0036080  0.0868925   0.042 0.966902    
## year71        0.9104285  0.8155744   1.116 0.265019    
## year72       -0.4903062  0.8038193  -0.610 0.542257    
## year73       -0.5528934  0.7214463  -0.766 0.443947    
## year74        1.2419976  0.8547434   1.453 0.147056    
## year75        0.8704016  0.8374036   1.039 0.299297    
## year76        1.4966598  0.8019080   1.866 0.062782 .  
## year77        2.9986967  0.8198949   3.657 0.000292 ***
## year78        2.9737783  0.7792185   3.816 0.000159 ***
## year79        4.8961763  0.8248124   5.936 6.74e-09 ***
## year80        9.0589316  0.8751948  10.351  < 2e-16 ***
## year81        6.4581580  0.8637018   7.477 5.58e-13 ***
## year82        7.8375850  0.8493560   9.228  < 2e-16 ***
## origin2       1.6932853  0.5162117   3.280 0.001136 ** 
## origin3       2.2929268  0.4967645   4.616 5.41e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.848 on 369 degrees of freedom
## Multiple R-squared:  0.8744, Adjusted R-squared:  0.8669 
## F-statistic: 116.8 on 22 and 369 DF,  p-value: < 2.2e-16
```

```
dif_mpg_J_E <- 2.2929268-1.6932853; dif_mpg_J_E
```

```
## [1] 0.5996415
```

(e) **(2 pts)** Fit a model to predict *mpg* using origin and horsepower, as well as an interaction between origin and horsepower. Present the summary output of the fitted model, and write out the fitted linear model.

$$\widehat{\text{mpg}} = 34.476496 - 0.121320 * \text{horsepower} + 10.99723 * I(\text{origin=2}) + 14.339718 * I(\text{origin=3}) - 0.100515 * \text{horsepower} * I(\text{origin=2}) - 0.108723 * \text{horsepower} * I(\text{origin=3}).$$

```
mod2 <-lm(mpg~ horsepower + origin + horsepower:origin, Autod)
summary(mod2)
```

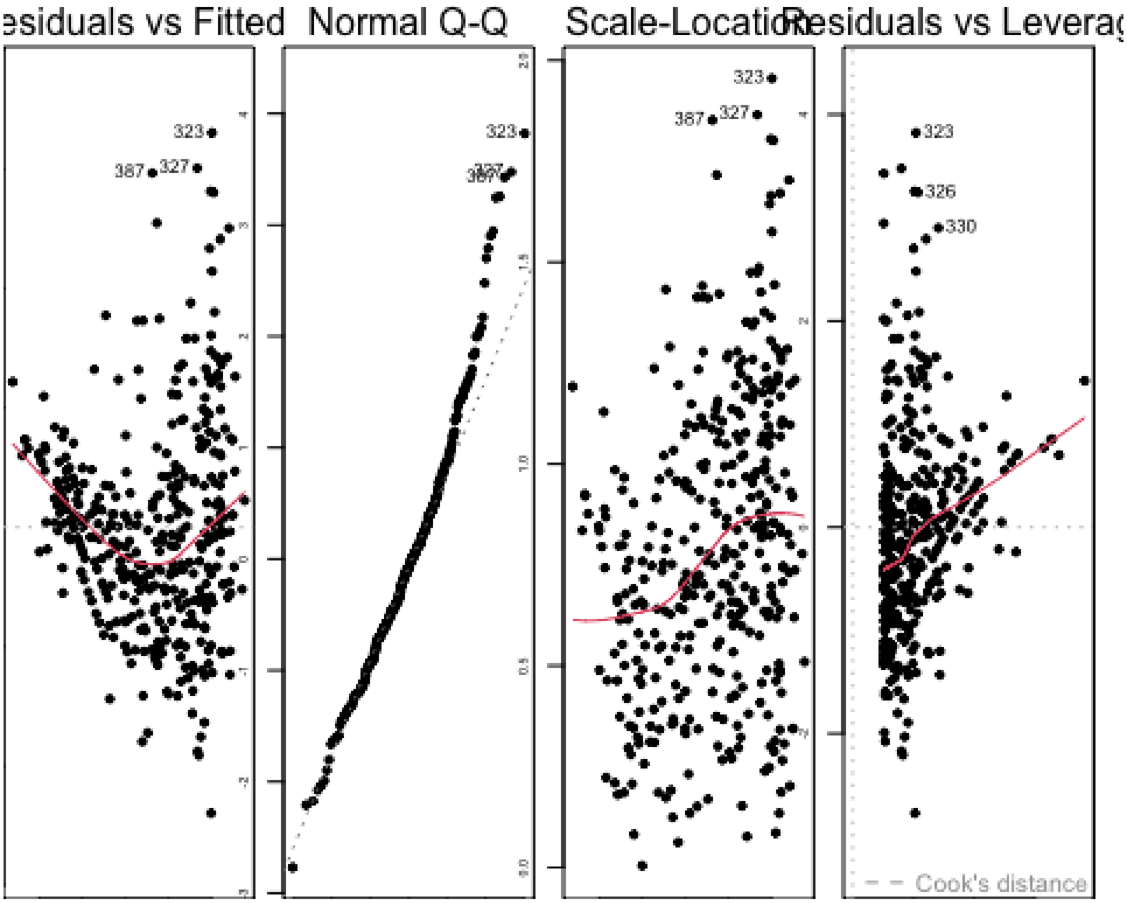
```
## 
## Call:
## lm(formula = mpg ~ horsepower + origin + horsepower:origin, data = A
utod)
## 
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.7415  -2.9547  -0.6389   2.3978  14.2495
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          34.476496   0.890665  38.709  < 2e-16 ***
## horsepower           -0.121320   0.007095 -17.099  < 2e-16 ***
## origin2              10.997230   2.396209   4.589 6.02e-06 ***
## origin3              14.339718   2.464293   5.819 1.24e-08 ***
## horsepower:origin2 -0.100515   0.027723  -3.626 0.000327 ***
## horsepower:origin3 -0.108723   0.028980  -3.752 0.000203 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.422 on 386 degrees of freedom
## Multiple R-squared:  0.6831, Adjusted R-squared:  0.679
## F-statistic: 166.4 on 5 and 386 DF,  p-value: < 2.2e-16
```

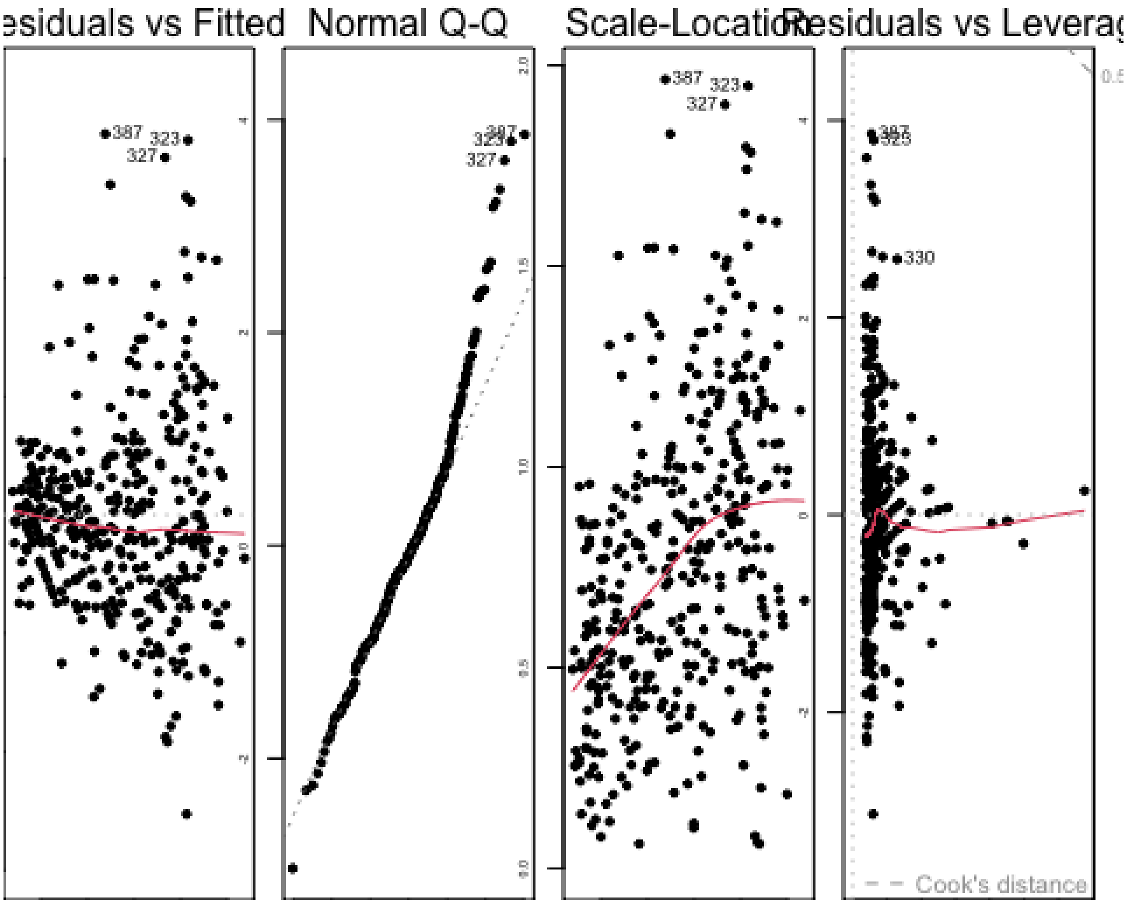
(f) **(2 pts)** If we are fitting a polynomial regression with mpg as the response variable and weight as the predictor, what should be a proper degree of that polynomial?

The p-values in each model’s output indicate whether each predictor variable’s coefficient is significantly different from zero. A p-value less than 0.05 suggests strong evidence against the null hypothesis that the coefficient is equal to zero, and we can conclude that the predictor variable is significantly associated with the response variable. From there model below, only model 3 have p-value that is bigger than 0.05, suggesting that weight^3 is a significant predictor of mpg in m3. Additional, the residual vs fitted value plot in m2 is more flatter than that in m1. Thus, second should be a proper degree of that polynomial, quadratic models.

```
summary(m1 <- lm(mpg~weight,Autod))$coefficient

##                 Estimate    Std. Error   t value      Pr(>|t|)
## (Intercept) 46.216524549 0.7986724633  57.86668 1.623069e-193
## weight      -0.007647343 0.0002579633 -29.64508 6.015296e-102

summary(m2 <- lm(mpg~weight + I(weight^2),Autod))$coefficient

##                  Estimate   Std. Error   t value     Pr(>|t|)
## (Intercept)  6.225547e+01 2.993076e+00 20.799832 3.848779e-65
## weight      -1.849561e-02 1.972056e-03 -9.378849 5.609944e-19
## I(weight^2)  1.696565e-06 3.059491e-07  5.545252 5.429177e-08

summary(m3 <- lm(mpg~weight + I(weight^2) + I(weight^3),Autod))$coeffic
ient

##                  Estimate   Std. Error     t value     Pr(>|t|)
## (Intercept)  6.169524e+01 1.104305e+01  5.58679434 4.360869e-08
## weight      -1.792978e-02 1.091485e-02 -1.64269604 1.012560e-01
## I(weight^2)  1.515412e-06 3.450428e-06  0.43919548 6.607644e-01
## I(weight^3)  1.846219e-11 3.502615e-10  0.05270974 9.579903e-01

par(mfrow = c(1, 4), mar = c(0,0,1.5,1))
plot(m1, cex.main = 1, cex.lab = 0.5, cex.axis = 0.5, pch = 20)
```



```
plot(m2, cex.main = 1, cex.lab = 0.5, cex.axis = 0.5, pch = 20)
```



(g) **(4 pts)** Perform a backward selection, starting with the full model which includes all predictors (except for name). What is the best model based on the AIC criterion? What are the predictor variables in that best model?

The AIC value will decrease as the model fits the data better.In the first step, the acceleration variable is removed from the model, resulting in a lower AIC value of 840.72. This means that the model without acceleration is a better fit for the data than the original model.In the second step, the remaining predictor variables are cylinders, displacement, horsepower, weight, year, and origin. The output shows that no other variables should be removed from the model since the AIC value remains the same as before. Therefore, this is the best model based on AIC criterion. The AIC values indicate that cylinders, displacement, horsepower, weight, year, and origin are the predictor variables in that best model. formula = mpg ~ cylinders + displacement + horsepower + weight + year + origin

```
step(lmod, direction = "backward")

## Start:  AIC=842.72
## mpg ~ cylinders + displacement + horsepower + weight + acceleration
+
##     year + origin
##
##                Df Sum of Sq    RSS     AIC
## - acceleration  1      0.01 2992.1  840.72
## <none>                      2992.1  842.72
## - displacement  1     24.70 3016.8  843.94
## - horsepower    1     73.45 3065.5  850.23
## - origin        2    183.21 3175.3  862.02
## - cylinders     4    472.77 3464.8  892.23
## - weight        1    558.60 3550.7  907.82
## - year         12   2831.60 5823.7 1079.78
##
## Step:  AIC=840.72
## mpg ~ cylinders + displacement + horsepower + weight + year +
##     origin
##
##                Df Sum of Sq    RSS     AIC
## <none>                      2992.1  840.72
## - displacement  1     24.88 3017.0  841.97
## - horsepower    1    115.58 3107.7  853.58
## - origin        2    183.45 3175.5  860.05
## - cylinders     4    476.39 3468.5  890.64
## - weight        1    730.02 3722.1  924.31
## - year         12   2841.52 5833.6 1078.45

##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     year + origin, data = Autod)
##
## Coefficients:
##  (Intercept)    cylinders4    cylinders5    cylinders6    cylinders8

##    30.970678      6.948983      6.646736      4.305068      6.372326

## displacement    horsepower        weight        year71        year72

##     0.011793     -0.039554     -0.005168      0.905750     -0.492137

##       year73        year74        year75        year76        year77

##    -0.555066      1.237612      0.865415      1.492399      2.994879

##       year78        year79        year80        year81        year82

##     2.970303      4.892261      9.055269      6.452705      7.833655

##      origin2       origin3
##     1.693186      2.293670
```

2. Use the *fat* data set available from the *faraway* package. Use the percentage of body fat: *siri* as the response, and the other variables, except *bronzek* and

density as potential predictors. Remove every tenth observation from the data for use as a test sample. Use the remaining data as a training sample, building the following models:

```
library(faraway)
data(fat)
head(fat)

##   brozek siri density age weight height adipos  free neck chest abdo
m   hip
## 1   12.6 12.3  1.0708  23 154.25  67.75   23.7 134.9 36.2  93.1  85.
2  94.5
## 2    6.9  6.1  1.0853  22 173.25  72.25   23.4 161.3 38.5  93.6  83.
0  98.7
## 3   24.6 25.3  1.0414  22 154.00  66.25   24.7 116.0 34.0  95.8  87.
9  99.2
## 4   10.9 10.4  1.0751  26 184.75  72.25   24.9 164.7 37.4 101.8  86.
4 101.2
## 5   27.8 28.7  1.0340  24 184.25  71.25   25.6 133.1 34.4  97.3 100.
0 101.9
## 6   20.6 20.9  1.0502  24 210.25  74.75   26.5 167.0 39.0 104.5  94.
4 107.8
##   thigh knee ankle biceps forearm wrist
## 1  59.0 37.3  21.9   32.0    27.4  17.1
## 2  58.7 37.3  23.4   30.5    28.9  18.2
## 3  59.6 38.9  24.0   28.8    25.2  16.6
## 4  60.1 37.3  22.8   32.4    29.4  18.2
## 5  63.2 42.2  24.0   32.2    27.7  17.7
## 6  66.0 42.0  25.6   35.7    30.6  18.8

fat <- subset(fat, select = c(2, 4:18))
test_indices <- seq(10, nrow(fat), by=10)
test_data <- fat[test_indices,]
training_data <- fat[-test_indices, ]
```

(a) **(5 pts)** Linear regression with all the predictors.

```
MLR_f <- lm(siri~.,training_data);summary(MLR_f)

##
## Call:
## lm(formula = siri ~ ., data = training_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8314 -0.6722  0.1828  0.9150  6.6619
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.591885   6.448868  -1.953 0.052193 .
## age           0.007978   0.012320   0.648 0.517983
## weight        0.362999   0.023314  15.570  < 2e-16 ***
## height        0.049026   0.040315   1.216 0.225315    
## adipos       -0.514032   0.114074  -4.506 1.09e-05 ***
## free         -0.564773   0.014889 -37.933  < 2e-16 ***
## neck          0.016525   0.089863   0.184 0.854272    
## chest         0.120219   0.039590   3.037 0.002694 ** 
## abdom         0.140108   0.042186   3.321 0.001056 ** 
## hip           0.006197   0.056101   0.110 0.912148    
## thigh         0.195057   0.054460   3.582 0.000424 ***
## knee          0.106637   0.093534   1.140 0.255542    
## ankle         0.125118   0.081303   1.539 0.125325    
## biceps        0.096199   0.064656   1.488 0.138278    
## forearm       0.230775   0.073332   3.147 0.001888 ** 
## wrist         0.139279   0.206804   0.673 0.501378    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.55 on 211 degrees of freedom
## Multiple R-squared:  0.9692, Adjusted R-squared:  0.967 
## F-statistic: 442.5 on 15 and 211 DF,  p-value: < 2.2e-16
```

(b) **(5 pts)** Ridge regression.

```
library(glmnet)
```

```
## 载入需要的程辑包：Matrix
```

```
## 
## 载入程辑包：'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
## 
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-6
```

```
x <- scale(data.matrix(fat)[,-1])
x <- x[-test_indices,]
y <- training_data$siri
ridge_model <- cv.glmnet(x, y, alpha = 0);ridge_model
```

```
## 
## Call:  cv.glmnet(x = x, y = y, alpha = 0) 
## 
## Measure: Mean-Squared Error 
## 
##     Lambda Index Measure    SE Nonzero
## min 0.6942   100   8.309 2.025      15
## 1se 1.7600    90  10.262 1.178      15
```

```
best_lambda <- ridge_model$lambda.min
best_lambda
```

```
## [1] 0.6941839
```

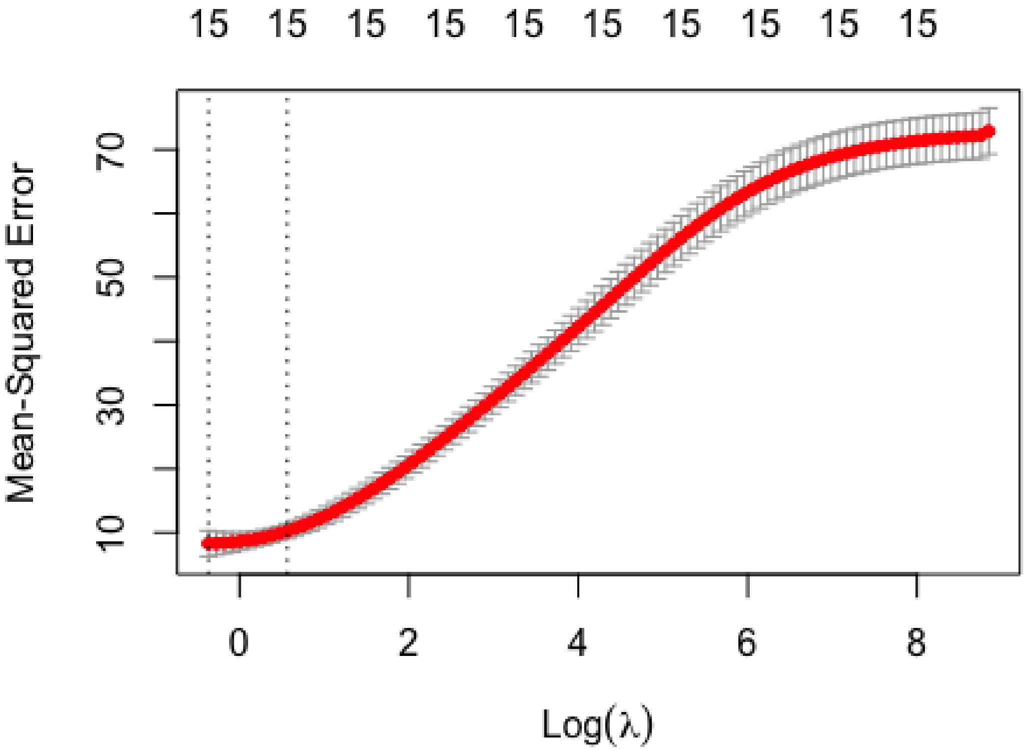
```
best_model <- glmnet(x, y, alpha = 0,lambda = best_lambda);best_model
```

```
##
## Call:  glmnet(x = x, y = y, alpha = 0, lambda = best_lambda)
##
##   Df  %Dev Lambda
## 1 15 92.82 0.6942
```

```
coef(best_model, s = "lambda.min")
```

```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                      s1
## (Intercept) 19.18924478
## age          0.38880181
## weight       2.30750495
## height       0.52787998
## adipos       0.46286715
## free        -6.43075409
## neck         0.09936367
## chest        1.27848771
## abdom        3.19204398
## hip          1.03998826
## thigh        1.01855289
## knee         0.72980876
## ankle        0.23945778
## biceps       0.47686916
## forearm      0.52888997
## wrist       -0.33165866
```

```
plot(ridge_model)
```


15
15
15
15
15
15
15
15
15
15
Mean-Squared Error
10
30
50
70
0
2
4
6
8
Log(λ)