

# Assignment 2

Chelsea Nowlin

6/19/2020

```
# install.packages("ISLR")
```

```
library(ISLR)
```

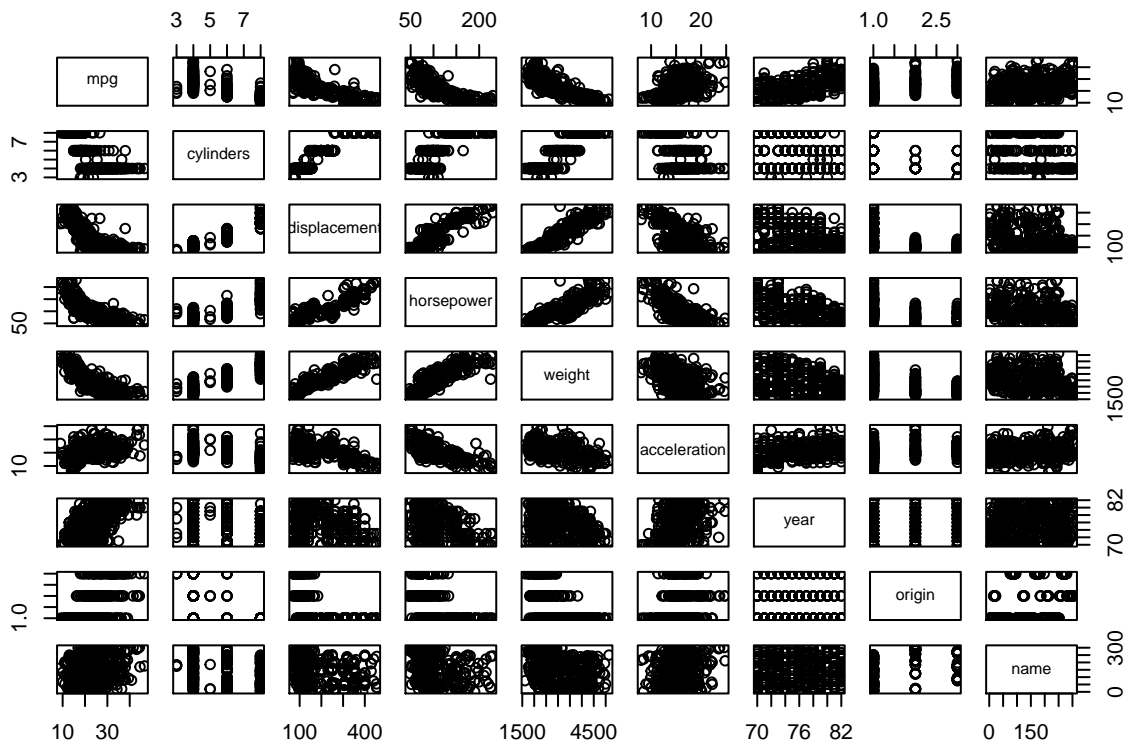
```
## Warning: package 'ISLR' was built under R version 3.6.3
```

```
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 3.6.3
```

```
# Multilinear Regression
```

```
data(Auto)  
pairs(Auto)
```



```
names(Auto)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"
## [5] "weight"       "acceleration" "year"         "origin"
## [9] "name"
```

```
cor(Auto[1:8])
```

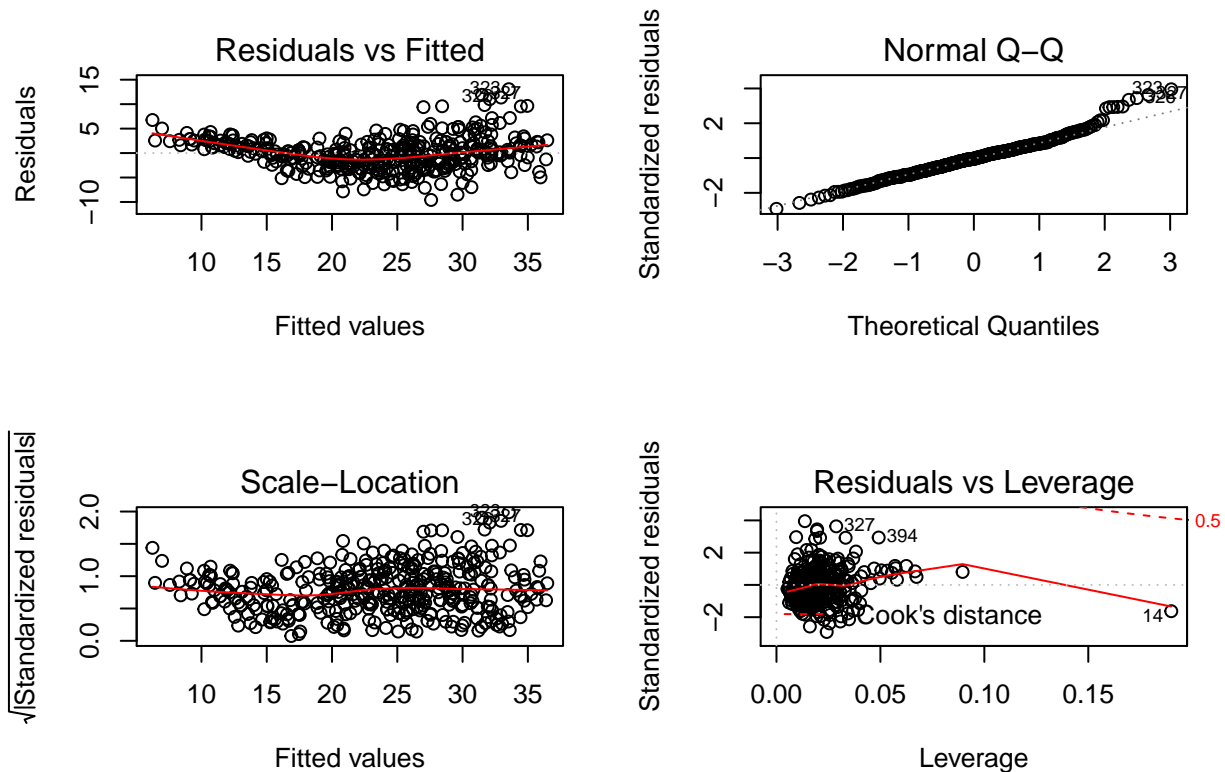
```
##           mpg cylinders displacement horsepower    weight
## mpg          1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year         0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin       0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##           acceleration    year    origin
## mpg          0.4233285  0.5805410  0.5652088
## cylinders    -0.5046834 -0.3456474 -0.5689316
## displacement -0.5438005 -0.3698552 -0.6145351
## horsepower   -0.6891955 -0.4163615 -0.4551715
## weight       -0.4168392 -0.3091199 -0.5850054
## acceleration  1.0000000  0.2903161  0.2127458
## year         0.2903161  1.0000000  0.1815277
## origin       0.2127458  0.1815277  1.0000000
```

```
fit2 <- lm(mpg ~ . - name, data = Auto)
summary(fit2)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.218435   4.644294  -3.707  0.00024 ***
## cylinders    -0.493376   0.323282  -1.526  0.12780
## displacement  0.019896   0.007515   2.647  0.00844 **
## horsepower   -0.016951   0.013787  -1.230  0.21963
## weight       -0.006474   0.000652  -9.929 < 2e-16 ***
## acceleration  0.080576   0.098845   0.815  0.41548
## year         0.750773   0.050973  14.729 < 2e-16 ***
## origin       1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
```

```
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 2))
plot(fit2)
```

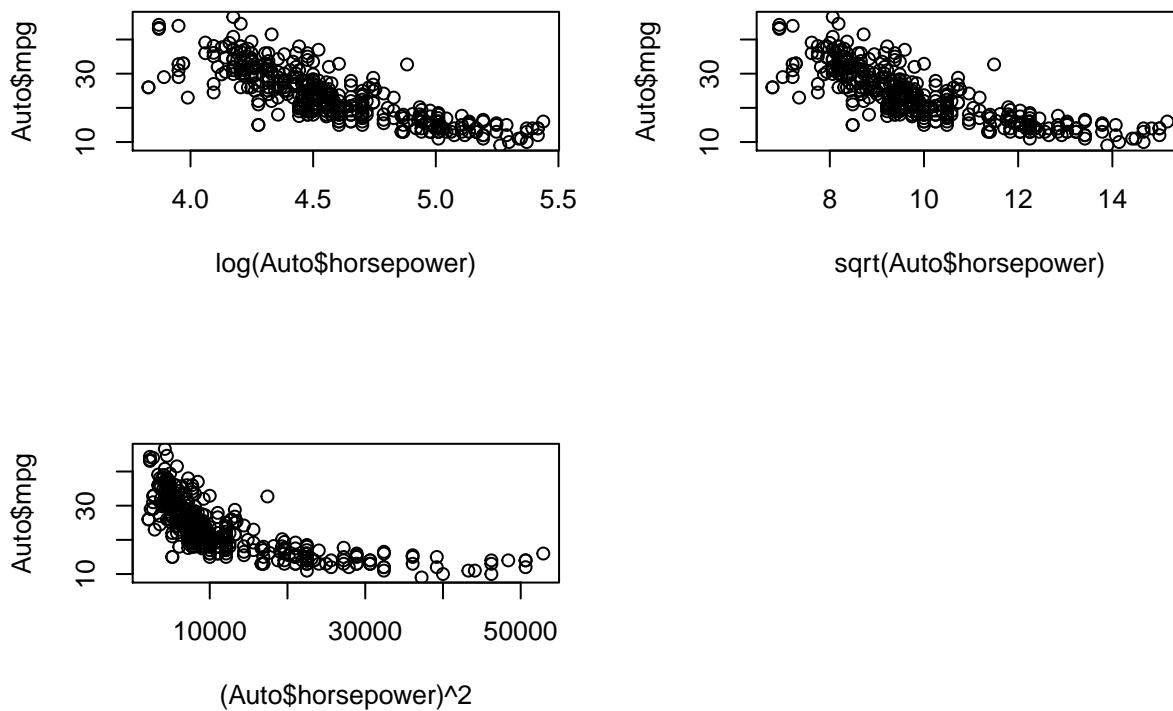


```
fit3 <- lm(mpg ~ cylinders * displacement+displacement * weight, data = Auto[, 1:8])
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders * displacement + displacement *
##     weight, data = Auto[, 1:8])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.2934  -2.5184  -0.3476   1.8399  17.7723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.262e+01  2.237e+00  23.519  < 2e-16 ***
## cylinders      7.606e-01  7.669e-01   0.992   0.322
## displacement -7.351e-02  1.669e-02  -4.403 1.38e-05 ***
## weight        -9.888e-03  1.329e-03  -7.438 6.69e-13 ***
## cylinders:displacement -2.986e-03  3.426e-03  -0.872   0.384
```

```
## displacement:weight      2.128e-05  5.002e-06  4.254 2.64e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.103 on 386 degrees of freedom
## Multiple R-squared:  0.7272, Adjusted R-squared:  0.7237
## F-statistic: 205.8 on 5 and 386 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 2))
plot(log(Auto$horsepower), Auto$mpg)
plot(sqrt(Auto$horsepower), Auto$mpg)
plot((Auto$horsepower)^2, Auto$mpg)
```



```
# Qualitative Predictors
```

```
fix(Carseats)
names(Carseats)
```

```
## [1] "Sales"      "CompPrice"  "Income"     "Advertising" "Population"
## [6] "Price"      "ShelveLoc"  "Age"        "Education"   "Urban"
## [11] "US"
```

```
lm.fit=lm(Sales~.+Price:Urban+US,data=Carseats)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Sales ~ . + Price:Urban + US, data = Carseats)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.8723 -0.7011  0.0298  0.6723  3.3937
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.5386942   0.7371346    7.514 4.02e-13 ***
## CompPrice      0.0928744   0.0041576   22.338 < 2e-16 ***
## Income         0.0157950   0.0018475    8.549 2.91e-16 ***
## Advertising    0.1233524   0.0111724   11.041 < 2e-16 ***
## Population     0.0002072   0.0003709    0.559  0.577
## Price         -0.0943707   0.0043411  -21.739 < 2e-16 ***
## ShelfLocGood   4.8492903   0.1533223   31.628 < 2e-16 ***
## ShelfLocMedium 1.9557229   0.1263016   15.485 < 2e-16 ***
## Age           -0.0461214   0.0031964  -14.429 < 2e-16 ***
## Education      -0.0206165   0.0198153   -1.040  0.299
## UrbanYes       0.2845509   0.5712614    0.498  0.619
## USYes         -0.1855349   0.1501027   -1.236  0.217
## Price:UrbanYes -0.0014091   0.0048808   -0.289  0.773
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.02 on 387 degrees of freedom
## Multiple R-squared:  0.8734, Adjusted R-squared:  0.8695
## F-statistic: 222.6 on 12 and 387 DF, p-value: < 2.2e-16
```

```
attach(Carseats)
contrasts(ShelveLoc)
```

```
##           Good Medium
## Bad           0      0
## Good          1      0
## Medium        0      1
```

```
lm.fit=lm(Sales~.+Price:ShelveLoc,data=Carseats)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Sales ~ . + Price:ShelveLoc, data = Carseats)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9005 -0.6943  0.0239  0.6649  3.4204
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.0823218   0.7901950    7.697 1.18e-13 ***
## CompPrice      0.0928186   0.0041874   22.166 < 2e-16 ***
```

```
## Income            0.0157690  0.0018557   8.497 4.27e-16 ***
## Advertising       0.1232481  0.0111422  11.061 < 2e-16 ***
## Population        0.0001862  0.0003728   0.499  0.6178
## Price            -0.0986980  0.0046879 -21.054 < 2e-16 ***
## ShelveLocGood     4.1785991  0.7494181   5.576 4.64e-08 ***
## ShelveLocMedium   1.5344694  0.6329676   2.424  0.0158 *
## Age              -0.0464310  0.0032158 -14.439 < 2e-16 ***
## Education        -0.0221904  0.0197846  -1.122  0.2627
## UrbanYes         0.1247657  0.1131754   1.102  0.2710
## USYes            -0.1827611  0.1500758  -1.218  0.2240
## Price:ShelveLocGood 0.0057959  0.0063142   0.918  0.3592
## Price:ShelveLocMedium 0.0036944  0.0054168   0.682  0.4956
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.02 on 386 degrees of freedom
## Multiple R-squared:  0.8737, Adjusted R-squared:  0.8694
## F-statistic: 205.4 on 13 and 386 DF,  p-value: < 2.2e-16
```

```
attach(Carseats)
```

```
## The following objects are masked from Carseats (pos = 3):
##
##   Advertising, Age, CompPrice, Education, Income, Population,
##   Price, Sales, ShelveLoc, Urban, US
```

```
contrasts(ShelveLoc)
```

```
##           Good Medium
## Bad           0       0
## Good          1       0
## Medium        0       1
```

```
#install.packages("Rmisc")
```

```
library(Rmisc)
```

```
## Warning: package 'Rmisc' was built under R version 3.6.3
```

```
## Loading required package: lattice
```

```
## Loading required package: plyr
```

```
myData <- Carseats
CI(Carseats$ShelveLoc, ci=0.95)
```

```
## Warning in mean.default(x): argument is not numeric or logical: returning
## NA
```

```
## Warning in qt(ci + (1 - ci)/2, df = n - 1): NaNs produced
```

```
## upper mean lower
##      NA      NA      NA
```

```
confint(lm.fit)
```

```
##              2.5 %      97.5 %
## (Intercept)    4.5286967907  7.6359468440
## CompPrice      0.0845856984  0.1010515389
## Income         0.0121203674  0.0194176506
## Advertising    0.1013410799  0.1451551553
## Population     -0.0005468093  0.0009191648
## Price          -0.1079149004 -0.0894810304
## ShelvelocGood  2.7051465912  5.6520515717
## ShelvelocMedium 0.2899735095  2.7789652884
## Age           -0.0527536069 -0.0401083721
## Education      -0.0610894563  0.0167085653
## UrbanYes       -0.0977516742  0.3472831395
## USYes          -0.4778295223  0.1123072528
## Price:ShelvelocGood -0.0066186981  0.0182104076
## Price:ShelvelocMedium -0.0069557012  0.0143445885
```

```
# Non Linear Regression
```

```
str(Boston)
```

```
## 'data.frame':  506 obs. of  14 variables:
## $ crim : num  0.00632 0.02731 0.02729 0.03237 0.06905 ...
## $ zn : num  18 0 0 0 0 12.5 12.5 12.5 12.5 ...
## $ indus : num  2.31 7.07 7.07 2.18 2.18 2.18 7.87 7.87 7.87 ...
## $ chas : int  0 0 0 0 0 0 0 0 0 ...
## $ nox : num  0.538 0.469 0.469 0.458 0.458 0.458 0.524 0.524 0.524 ...
## $ rm : num  6.58 6.42 7.18 7 7.15 ...
## $ age : num  65.2 78.9 61.1 45.8 54.2 58.7 66.6 96.1 100 85.9 ...
## $ dis : num  4.09 4.97 4.97 6.06 6.06 ...
## $ rad : int  1 2 2 3 3 3 5 5 5 ...
## $ tax : num  296 242 242 222 222 222 311 311 311 311 ...
## $ ptratio: num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 15.2 15.2 15.2 ...
## $ black : num  397 397 393 395 397 ...
## $ lstat : num  4.98 9.14 4.03 2.94 5.33 ...
## $ medv : num  24 21.6 34.7 33.4 36.2 28.7 22.9 27.1 16.5 18.9 ...
```

```
lm.fit=lm(medv~lstat, data=Boston)
attach(Boston)
lm.fit=lm(medv~lstat)
```

```
lm.fit2=lm(medv~lstat+I(lstat^2))
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = medv ~ lstat + I(lstat^2))
##
## Residuals:
```

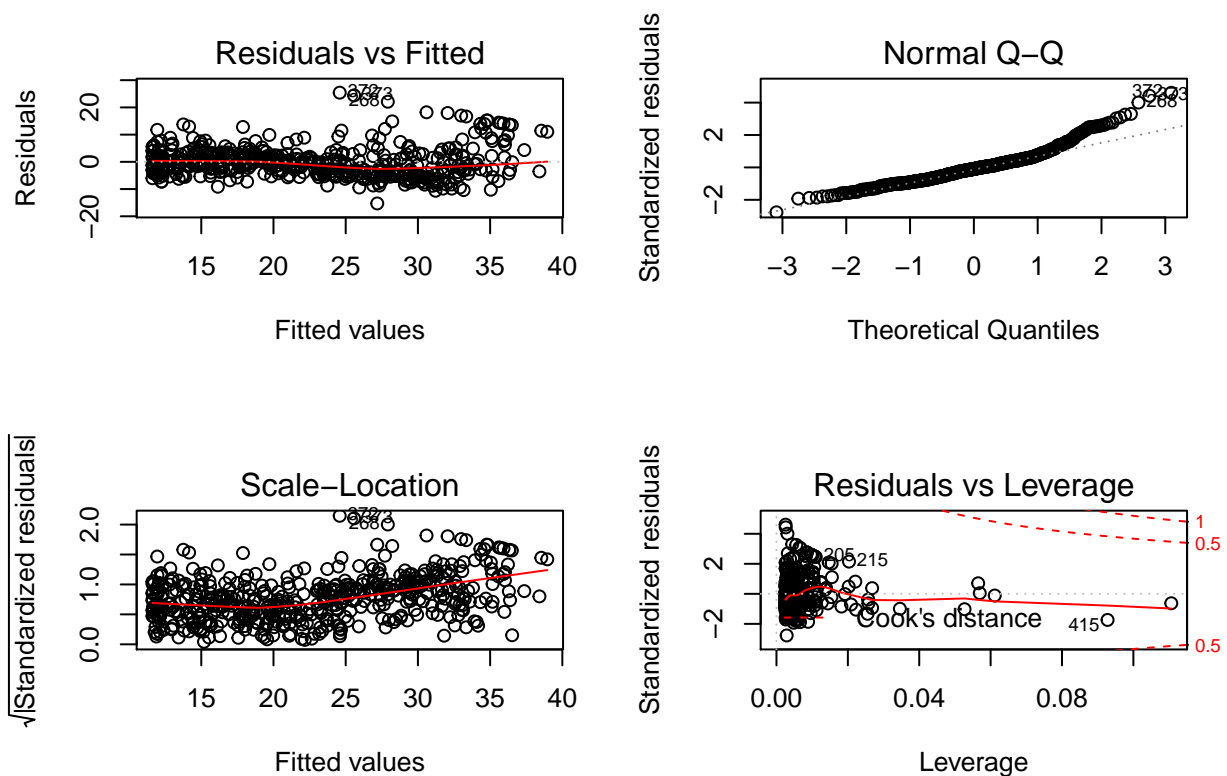
```
##      Min      1Q   Median      3Q      Max
## -15.2834 -3.8313 -0.5295   2.3095  25.4148
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 42.862007   0.872084   49.15  <2e-16 ***
## lstat      -2.332821   0.123803  -18.84  <2e-16 ***
## I(lstat^2)  0.043547   0.003745   11.63  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.524 on 503 degrees of freedom
## Multiple R-squared:  0.6407, Adjusted R-squared:  0.6393
## F-statistic: 448.5 on 2 and 503 DF,  p-value: < 2.2e-16
```

```
lm.fit=lm(medv~lstat)
anova(lm.fit,lm.fit2)
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ lstat
## Model 2: medv ~ lstat + I(lstat^2)
##   Res.Df  RSS Df Sum of Sq    F    Pr(>F)
## 1     504 19472
## 2     503 15347   1    4125.1 135.2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(2,2))
plot(lm.fit2)
```





```
lm.fit5=lm(medv~poly(lstat,5))
summary(lm.fit5)
```

```
##
## Call:
## lm(formula = medv ~ poly(lstat, 5))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5433  -3.1039  -0.7052   2.0844  27.1153
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.5328    0.2318  97.197 < 2e-16 ***
## poly(lstat, 5)1 -152.4595    5.2148 -29.236 < 2e-16 ***
## poly(lstat, 5)2   64.2272    5.2148  12.316 < 2e-16 ***
## poly(lstat, 5)3  -27.0511    5.2148  -5.187 3.10e-07 ***
## poly(lstat, 5)4   25.4517    5.2148   4.881 1.42e-06 ***
## poly(lstat, 5)5  -19.2524    5.2148  -3.692 0.000247 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.215 on 500 degrees of freedom
## Multiple R-squared:  0.6817, Adjusted R-squared:  0.6785
## F-statistic: 214.2 on 5 and 500 DF, p-value: < 2.2e-16
```

```
summary(lm(medv~log(rm),data=Boston))
```

```
##
## Call:
## lm(formula = medv ~ log(rm), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.487  -2.875  -0.104   2.837  39.816
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -76.488      5.028  -15.21  <2e-16 ***
## log(rm)       54.055      2.739   19.73  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.915 on 504 degrees of freedom
## Multiple R-squared:  0.4358, Adjusted R-squared:  0.4347
## F-statistic: 389.3 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
coefficients(lm.fit2)
```

```
## (Intercept)      lstat  I(lstat^2)
## 42.86200733 -2.33282110  0.04354689
```

```
coefficients(lm.fit5)
```

```
##      (Intercept) poly(lstat, 5)1 poly(lstat, 5)2 poly(lstat, 5)3
##      22.53281      -152.45955      64.22724      -27.05110
## poly(lstat, 5)4 poly(lstat, 5)5
##      25.45173      -19.25242
```