­

Ch3. Statistical Learning

-Lab & Exercises 8, 9, 13, 14

­­

**학번 : 182STG12**

**이름 : 오혜윤**

c

Problem Description

In Chapter 3 lab, we learn about linear regression and making a function. Using Boston dataset, we do linear regression. The function about linear regression is called lm(). The basic syntax is lm(y∼x,data), where y is the response, x is the predictor, and data is the data set in which these two variables are kept. After developing a linear regression model, we use predict() for predicting y. The predict() function can be used to produce confidence intervals and prediction intervals.

Of course, we can add interaction terms and non-linear predictors. The syntax : tells R to include an interaction term between two variables. The syntax \* simultaneously includes two variables, and the interaction term × as predictors. Also, we can create a non-linear predictor using I(X^2), log(X) and poly(X,2).

In Chapter 3 exercises, using Auto and simulation dataset, we want to investigate the predictors. ‘Auto’ dataset is about cars and there are 11 variables in this dataset. By creating some plots highlighting the relationships among the predictors and using some summary functions, we can understand what relationship exists between the variables and what characteristics are in the variables.

Results

**CH3. Lab Review**

In the lab, it was mostly about simple linear regression and making a function. I already knew them, so it was not that surprising. In the second part, there were some commands about making a function. There were not many details about how to make a function, but it was useful. Especially, I have not ever made the function like LoadLibraries() but I think I should try it next time.

**CH3 Exercises**

**8. This question involves the use of simple linear regression on the Auto data set.**

**(a) Use the lm() function to perform a simple linear regression with mpg as the response and horsepower as the predictor. Use the summary() function to print the results. Comment on the output.**

**For example:**

**i. Is there a relationship between the predictor and the response?**

We can answer this question by testing the hypothesis H0:βi=0. The p-value corresponding to the F-statistic is < 2.2e-16, this indicates a clear evidence of a relationship between “mpg” and “horsepower”.

**ii. How strong is the relationship between the predictor and the response?**

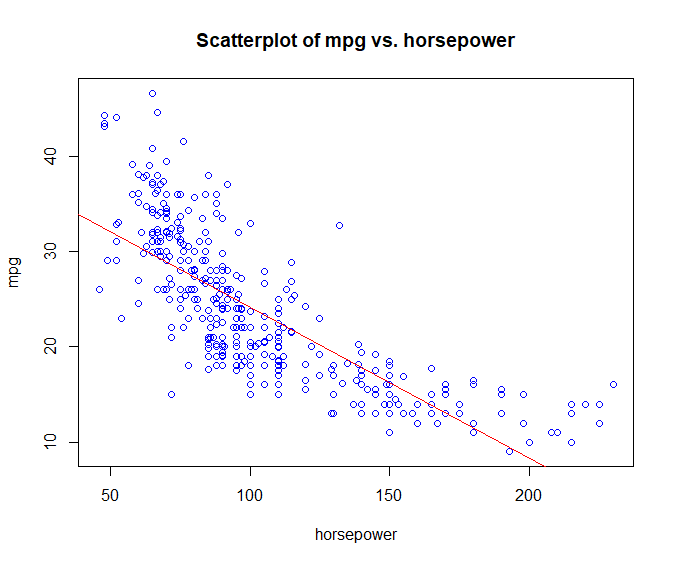
To calculate the residual error relative to the response we use the mean of the response and the RSE. The mean of mpg is 23.4459. The RSE of the lm.fit was 4.906. We may also note that as the R2 is equal to 0.6059, almost 60.59% of the variability in “mpg” can be explained using “horsepower”.

**iii. Is the relationship between the predictor and the response positive or negative?**

As the coefficient of “horsepower” is negative, the relationship is also negative. The more horsepower an automobile has the linear regression indicates the less mpg fuel efficiency the automobile will have.

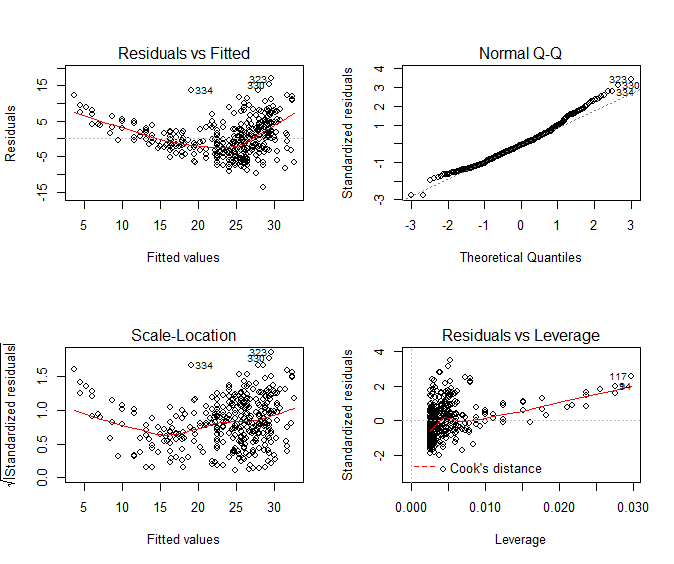
**iv. What is the predicted mpg associated with a horsepower of 98? What are the associated 95% confidence and prediction intervals?**

|  |  |  |
| --- | --- | --- |
| Fit | Lwr | upr |
| 24.46708 | 23.97308 | 24.96108 |
| Fit | **Lwr** | **upr** |
| 24.46708 | 14.8094 | 34.12476 |

 **(b) Plot the response and the predictor. Use the abline() function to display the least squares regression line.**

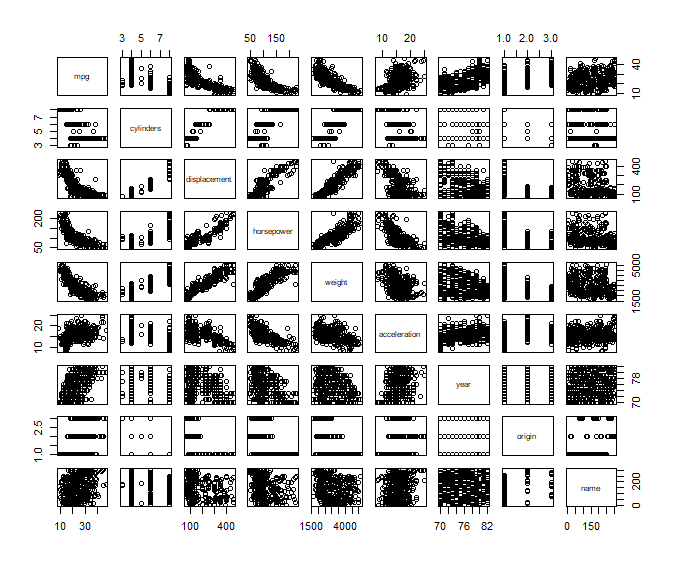
**(c) Use the plot() function to produce diagnostic plots of the least squares regression fit. Comment on any problems you see with the fit.**

The plot of residuals versus fitted values indicates the presence of nonlinearity in the data. The plot of standardized residuals versus leverage indicates the presence of a few outliers (higher than 2 or lower than -2) and a few high leverage points.

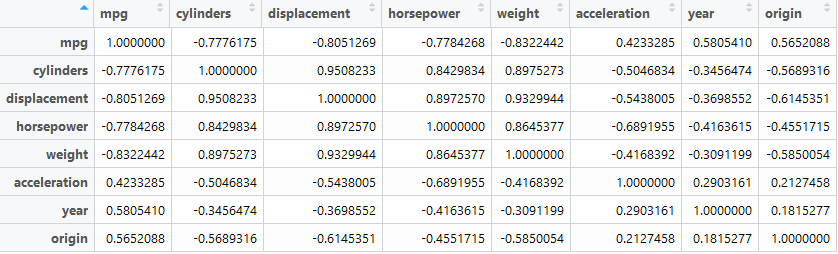


**9. This question involves the use of multiple linear regression on the Auto data set.**

**(a) Produce a scatterplot matrix which includes all of the variables in the data set.**



**(b) Compute the matrix of correlations between the variables using the function cor(). You will need to exclude the name variable, cor() which is qualitative.**



**(c) Use the lm() function to perform a multiple linear regression with mpg as the response and all other variables except name as the predictors. Use the summary() function to print the results. Comment on the output.**

**For instance:**

**i. Is there a relationship between the predictors and the response?**

We can answer this question by again testing the hypothesis H0:βi=0. The p-value corresponding to the F-statistic is < 2.2e-16, this indicates a clear evidence of a relationship between “mpg” and the other predictors.

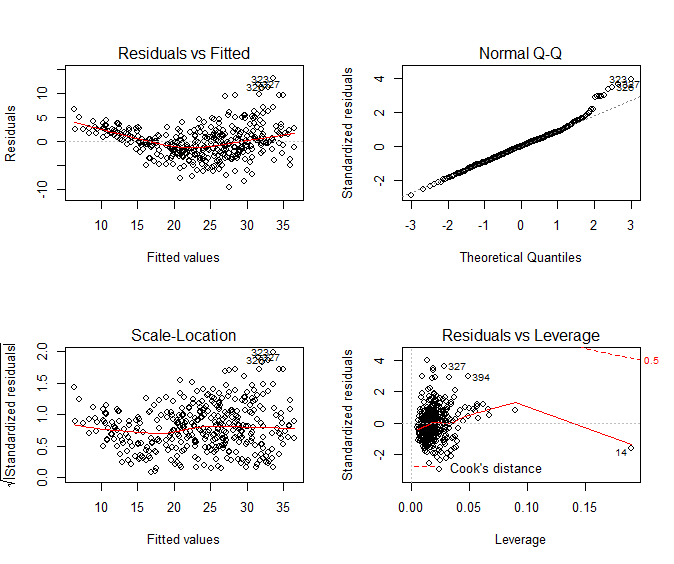
**ii. Which predictors appear to have a statistically significant relationship to the response?**

We can answer this question by checking the p-values associated with each predictor’s t-statistic. We may conclude that all predictors are statistically significant except “cylinders”, “horsepower” and “acceleration”.

**iii. What does the coefficient for the year variable suggest?**

The coefficient of the “year” variable suggests that the average effect of an increase of 1 year is an increase of 0.7508 in “mpg” (all other predictors remaining constant). In other words, cars become more fuel efficient every year by almost 1 mpg / year.

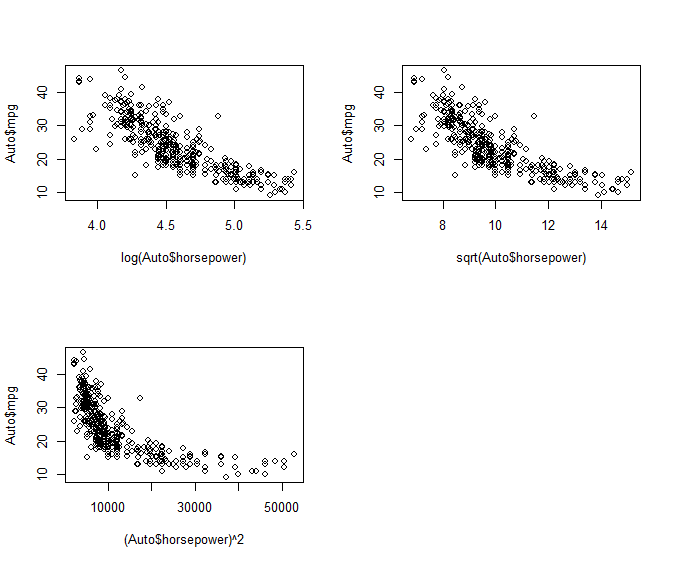
**(d) Use the plot() function to produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit. Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage?**



As before, the plot of residuals versus fitted values indicates the presence of mild nonlinearity in the data. The plot of standardized residuals versus leverage indicates the presence of a few outliers (higher than 2 or lower than -2) and one high leverage point (point 14).

**(e) Use the \* and : symbols to fit linear regression models with interaction effects. Do any interactions appear to be statistically significant?**

From the p-values, we can see that the interaction between displacement and weight is statistically significant, while the interaction between cylinders and displacement is not.

**(f) Try a few different transformations of the variables, such as log(X), √X, X2. Comment on your findings.**

We limit ourselves to examining “horsepower” as an only predictor. It seems that the log transformation plot is the most linear plot.

**13. In this exercise you will create some simulated data and will fit simple linear regression models to it. Make sure to use set.seed(1) prior to starting part (a) to ensure consistent results.**

**(a) Using the rnorm() function, create a vector, x, containing 100 observations drawn from a N(0, 1) distribution. This represents a feature, X.**

set.seed(1)

x <- rnorm(100)

By writing this code above, we create a vector x .

**(b) Using the rnorm() function, create a vector, eps, containing 100 observations drawn from a N(0, 0.25) distribution i.e. a normal distribution with mean zero and variance 0.25.**

eps <- rnorm(100, sd = sqrt(0.25))

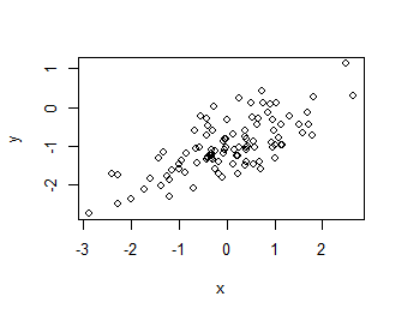
By writing this code above, we create a vector eps.

**(c) Using x and eps, generate a vector y according to the model Y = −1 + 0.5X + ϵ. (3.39)**

**What is the length of the vector y? What are the values of β0 and β1 in this linear model?**

The length of the vector y is 100. Also, the values of β0 and β1 are −1 and 0.5 respectively**.**

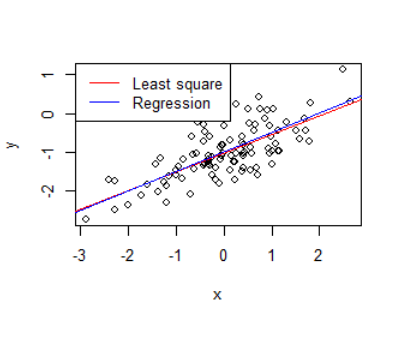
**(d) Create a scatterplot displaying the relationship between x and y. Comment on what you observe.**



**(e) Fit a least squares linear model to predict y using x. Comment on the model obtained. How do ˆ β0 and ˆ β1 compare to β0 and β1?**

The values of β^0 and β^1 are pretty close to β0 and β1. (The values of β^0 and β^1 are -1.0282, 0.477 respectively.) The model has a large F-statistic (104.3)with a near-zero p-value so the null hypothesis can be rejected.

**(f) Display the least squares line on the scatterplot obtained in (d). Draw the population regression line on the plot, in a different color. Use the legend() command to create an appropriate legend.**

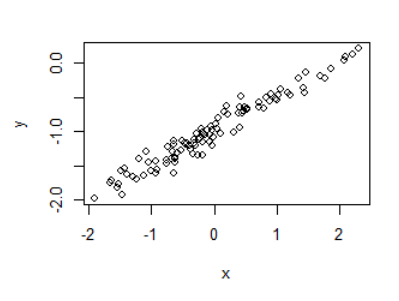
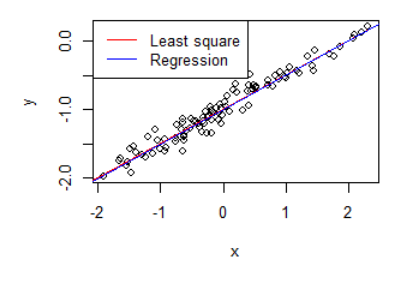


**(g) Now fit a polynomial regression model that predicts y using x and x2. Is there evidence that the quadratic term improves the model fit? Explain your answer.**

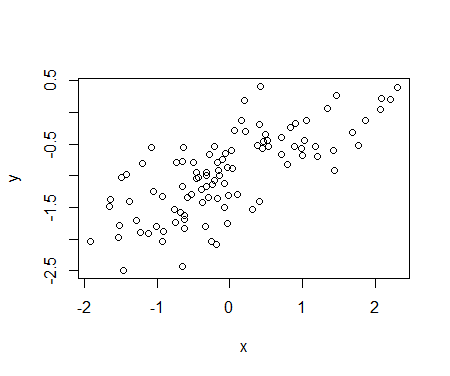
The coefficient for “x^2” is not significant as its p-value is higher than 0.05. So, there is no evidence that the quadratic term improves the model fit even though the R2 is slightly higher (about 50%) and RSE slightly lower than the linear model.

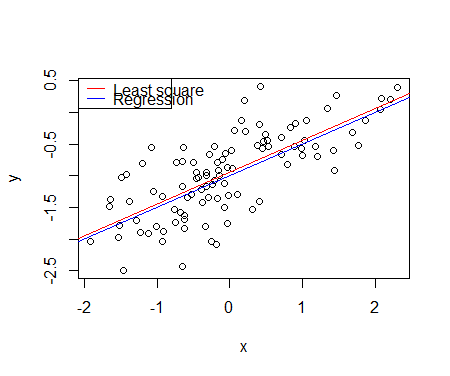
**(h) Repeat (a)–(f) after modifying the data generation process in such a way that there is less noise in the data. The model (3.39) should remain the same. You can do this by decreasing the variance of the normal distribution used to generate the error term**

**ϵ in (b). Describe your results.**



We reduced the noise by decreasing the variance of the normal distribution used to generate the error term ε. We may see that the coefficients are very close to the previous ones, but now, as the relationship is nearly linear, we have a much higher R2(94%) and much lower RSE. Moreover, the two lines overlap each other as we have very little noise.

 **(i) Repeat (a)–(f) after modifying the data generation process in such a way that there is more noise in the data. The model (3.39) should remain the same. You can do this by increasing the variance of the normal distribution used to generate the error term ϵ in (b). Describe your results.**



We increased the noise by increasing the variance of the normal distribution used to generate the error term ε. We may see that the coefficients are very close to the previous ones again, but now, as the relationship is not quite linear, we have a much lower R2(53%) and much higher RSE. Moreover, the two lines are wider apart but are still really close to each other as we have a large data set.

**(j) What are the confidence intervals for β0 and β1 based on the original data set, the noisier data set, and the less noisy data set? Comment on your results.**

|  |  |  |  |
| --- | --- | --- | --- |
| Fit3 | confint | 2.5 % | 97.5 % |
| **(Intercept)** | -1.1137 | 0.9228 |
| **x** | 0.3843 | 0.5698 |
| Fit5 | **confint** | **2.5 %** | **97.5 %** |
| **(Intercept)** | -1.0088 | -0.9640 |
| **x** | 0.4764 | 0.5234 |

|  |  |  |  |
| --- | --- | --- | --- |
| Fit6 | confint | 2.5 % | 97.5 % |
| **(Intercept)** | -1.0352 | -0.8559 |
| **x** | 0.4055 | 0.5935 |

All intervals seem to be centered on approximately 0.5. As the noise increases, the confidence intervals widen. With less noise, there is more predictability in the data set.

**14. This problem focuses on the collinearity problem.**

**(a) Perform the following commands in R:**

**> set.seed(1)**

**> x1=runif(100)**

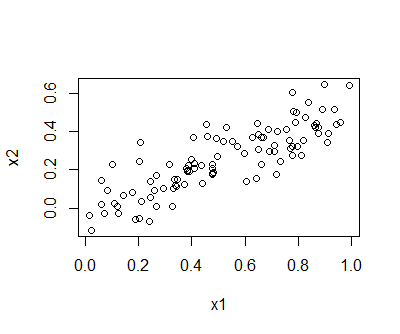
**> x2=0.5\*x1+rnorm(100)/10**

**> y=2+2\*x1+0.3\*x2+rnorm(100)**

**The last line corresponds to creating a linear model in which y is a function of x1 and x2. Write out the form of the linear model. What are the regression coefficients?**

The form of the linear model is Y=2+2X1+0.3X2+ε with ε a N(0,1) random variable. The regression coefficients are respectively 2, 2 and 0.3.

**(b) What is the correlation between x1 and x2? Create a scatterplot displaying the relationship between the variables.**

The correlation between x1 and x2 is 0.8351. The variables seem highly correlated.

**(c) Using this data, fit a least squares regression to predict y using x1 and x2. Describe the results obtained. What are ˆ β0, ˆ β1, and ˆ β2? How do these relate to the true β0, β1, and β2? Can you reject the null hypothesis H0 : β1 = 0? How about the null hypothesis H0 : β2 = 0?**

The coefficients β^0, β^1 and β^2 are respectively 2.1305, 1.4396 and 1.0097. Only β^0 is close to β0. As the p-value is less than 0.05 we may reject H0 for β1, however we may not reject H0 for β2 as the p-value is higher than 0.05.

**(d) Now fit a least squares regression to predict y using only x1. Comment on your results. Can you reject the null hypothesis H0 : β1 = 0?**

The coefficient for “x1” in this last model is very different from the one with “x1” and “x2” as predictors. In this case “x1” is highly significant as its p-value is very low, so we may reject H0.

**(e) Now fit a least squares regression to predict y using only x2. Comment on your results. Can you reject the null hypothesis H0 : β1 = 0?**

The coefficient for “x2” in this last model is very different from the one with “x1” and “x2” as predictors. In this case “x2” is highly significant as its p-value is very low, so we may again reject H0.

**(f) Do the results obtained in (c)–(e) contradict each other? Explain your answer.**

No, the results do not contradict each other. As the predictors “x1” and “x2” are highly correlated, there is a collinearity, in this case it can be difficult to determine how each predictor is associated with the response. Since collinearity reduces the accuracy of the estimates of the regression coefficients, it causes the standard error for β^1 to grow (we have a standard error of 0.7212 and 1.1337 for “x1” and “x2” respectively in the model with two predictors and only of 0.3963 and 0.6330 for “x1” and “x2” respectively in the models with only one predictor). Consequently, we may fail to reject H0 in the presence of collinearity. The importance of the “x2” variable has been masked due to the presence of collinearity.

**(g) Now suppose we obtain one additional observation, which was unfortunately mismeasured.**

**> x1=c(x1, 0.1)**

**> x2=c(x2, 0.8)**

**> y=c(y,6)**

**Re-fit the linear models from (c) to (e) using this new data. What effect does this new observation have on the each of the models? In each model, is this observation an outlier? A high-leverage point? Both? Explain your answers.**

In the model with two predictors, the last point is a high-leverage point. In the model with “x1” as sole predictor, the last point is an outlier. In the model with “x2” as an only predictor, the last point is a high leverage point.

river?

Discussion

After reviewing the datasets and using graphical analysis, we found many relationships between the variables. In the Auto dataset, by using lm() function, we can find that mpg has negative relationship with horsepower. Also, mpg has positive relationship with acceleration. There were many relationships between the variables and we can find out them by using corr() function. We also check the interaction terms by using sign “\*” and the interaction term, displacement \* weight was helpful in model. In the simulation dataset, we can compare the models. By comparing the model using two variables and the model using one variable, we can know about the relationships between y and x1, x2 respectively.

By investigating the datasets, we can find many interesting things. Especially, by creating correlation matrix and graphs, we can understand datasets better.

Appendix (R

**R codes**

#ch3 lab--------------------------------------------------------

library(MASS)

library(ISLR)

data(Boston)

lm.fit=lm(medv~stat ,data=Boston )

plot(lstat,medv);abline(lm.fit)

abline(lm.fit,lwd=3)

abline(lm.fit,lwd=3,col="red")

plot(lstat,medv,col="red")

plot(lstat,medv,pch=20)

plot(lstat,medv,pch="+")

plot(1:20,1:20,pch=1:20)

lm.fit=lm(medv~stat+age ,data=Boston )

summary(lm.fit)

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

summary(lm.fit5)

lm.fit=lm(Sales~.+ Income :Advertising +Price :Age ,data=Carseats )

Summary(lm.fit)

LoadLibraries=function (){

+ library(ISLR)

+ library(MASS)

+ print("The libraries have been loaded.")

+ }

#ex8----------------------------------------------------------

library(ISLR)

data(Auto)

##a------------------------------------------------------------------

fit <- lm(mpg ~ horsepower, data = Auto)

summary(fit)

predict(fit, data.frame(horsepower = 98), interval = "confidence")

predict(fit, data.frame(horsepower = 98), interval = "prediction")

##b------------------------------------------------------------------

plot(Auto$horsepower, Auto$mpg, main = "Scatterplot of mpg vs. horsepower", xlab = "horsepower", ylab = "mpg", col = "blue")

abline(fit, col = "red")

##c------------------------------------------------------------------

par(mfrow = c(2, 2))

plot(fit)

#ex9----------------------------------------------------------

##a-----------------------------------------------------------------

pairs(Auto)

##b-----------------------------------------------------------------

names(Auto)

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

##c-----------------------------------------------------------------

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

summary(fit2)

##d-----------------------------------------------------------------

par(mfrow = c(2, 2))

plot(fit2)

##e-----------------------------------------------------------------

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

summary(fit3)

##f-----------------------------------------------------------------

par(mfrow = c(2, 2))

plot(log(Auto$horsepower), Auto$mpg)

plot(sqrt(Auto$horsepower), Auto$mpg)

plot((Auto$horsepower)^2, Auto$mpg)

#ex13----------------------------------------------------------

##a-----------------------------------------------------------------

set.seed(1)

x <- rnorm(100)

##b-----------------------------------------------------------------

eps <- rnorm(100, sd = sqrt(0.25))

##c-----------------------------------------------------------------

y <- -1 + 0.5 \* x + eps

length(y)

##d-----------------------------------------------------------------

plot(x, y)

##e-----------------------------------------------------------------

fit3 <- lm(y ~ x)

summary(fit3)

##f-----------------------------------------------------------------

plot(x, y)

abline(fit3, col = "red")

abline(-1, 0.5, col = "blue")

legend("topleft", c("Least square", "Regression"), col = c("red", "blue"), lty = c(1, 1))

##g----------------------------------------------------------------

fit4 <- lm(y ~ x + I(x^2))

summary(fit4)

##h-----------------------------------------------------------------

set.seed(1)

eps <- rnorm(100, sd = 0.125)

x <- rnorm(100)

y <- -1 + 0.5 \* x + eps

plot(x, y)

fit5 <- lm(y ~ x)

summary(fit5)

abline(fit5, col = "red")

abline(-1, 0.5, col = "blue")

legend("topleft", c("Least square", "Regression"), col = c("red", "blue"), lty = c(1, 1))

##i-----------------------------------------------------------------

set.seed(1)

eps <- rnorm(100, sd = 0.5)

x <- rnorm(100)

y <- -1 + 0.5 \* x + eps

plot(x, y)

fit6 <- lm(y ~ x)

summary(fit6)

abline(fit6, col = "red")

abline(-1, 0.5, col = "blue")

legend("topleft", c("Least square", "Regression"), col = c("red", "blue"), lty = c(1, 1))

##j-----------------------------------------------------------------

confint(fit3) ; confint(fit5) ; confint(fit6)

#ex14----------------------------------------------------------

##a--------------------------------------------------------------

set.seed(1)

x1 <- runif(100)

x2 <- 0.5 \* x1 + rnorm(100)/10

y <- 2 + 2 \* x1 + 0.3 \* x2 + rnorm(100)

##b--------------------------------------------------------------

cor(x1, x2)

plot(x1, x2)

##c--------------------------------------------------------------

fit1 <- lm(y ~ x1 + x2)

summary(fit1)

##d--------------------------------------------------------------

fit2 <- lm(y ~ x1)

summary(fit2)

##e--------------------------------------------------------------

fit3 <- lm(y ~ x2)

summary(fit3)

##g--------------------------------------------------------------

x1 <- c(x1, 0.1)

x2 <- c(x2, 0.8)

y <- c(y, 6)

fit4 <- lm(y ~ x1 + x2)

fit5 <- lm(y ~ x1)

fit6 <- lm(y ~ x2)

summary(fit4) ; summary(fit5) ; summary(fit6)

plot(fit4) ; plot(fit4) ; plot(fit4)