Chapter 13: Multiple Regression

Section 13.1. Simple Linear Regression: A Reprise

```
plot(brisbane$kilometers, brisbane$minutes, pch = 19, xlab =
     'Kilometers the Vehicle Travels on the Daily Client Visits',
      ylab = 'Minutes the Vehicle is on the Road')
```

```
slr <- lm(minutes ~ kilometers, data = brisbane)
```

```
plot(brisbane$kilometers, resid(slr), pch = 19, abline(h = 0),
    xlab = 'Kilometers the Vehicle Travels on the Daily Client
    Visits', ylab = 'Residuals')
```

```
options(scipen = 999)
```

```
slr <- lm(minutes ~ kilometers, data = brisbane)
```

```
summary(slr)
```

Section 13.5. The 2 Independent Variable Case

```r
plot(brisbane$visits, brisbane$minutes, pch = 19, xlab = 'Number
    of Scheduled Daily Client Visits', ylab = 'Minutes the
    Vehicle is on the Road')
```

```r
cor(brisbane$kilometers, brisbane$visits)
```

```r
plot(brisbane$kilometers, brisbane$visits, pch = 19, xlab =
    'Kilometers the Vehicle Travels on the Daily Visits', ylab =
    'Number of Scheduled Daily Client Visits')
```

```r
pairs(brisbane, pch = 19, lower.panel = NULL)
```

Example 1

```r
mr <- lm(minutes ~ kilometers + visits, data = brisbane)

summary(mr)

ss_y <- sum((brisbane$minutes - mean(brisbane$minutes))^2)

ss_res <- sum((resid(mr))^2)

r_squared <- (ss_y - ss_res) / ss_y
```

Example 2

```
confint(mr, level = 0.95)
```

Section 13.6. Assumptions: What Are They? Can We Validate Them?

```
plot(fitted(mr), resid(mr), abline(h = 0), pch = 19,
        xlab = 'Predicted Value of y', ylab =  'Residuals')
```

Example 3

```
qf(0.05, 2, 15, lower.tail = FALSE)
```

Example 4

```
pf(33.5328, 2, 15, lower.tail = FALSE)
```

```
F215 <- rf(100000, 2, 15)
```

```
hist(F215, xlim = c(0, 10), ylim = c(0, 1), breaks = 150,
     freq = FALSE, xlab = 'F', ylab = 'Probability',
     main = 'Histogram of F with df_numerator=2 and
     df_denominator=15', col = 'blue')
```

Example 5

```
qt(0.025, 15, lower.tail = FALSE)
```

```
qt(0.025, 15)
```

Example 6

```
pt(7.538, 15, lower.tail = FALSE) + pt(-7.538, 15)
```

Example 7

```
qt(0.025, 15, lower.tail = FALSE)

qt(0.025, 15)
```

Example 8

```
pt(3.474, 15, lower.tail = FALSE) + pt(-3.474, 15)
```

Example 9

```
fitted(mr)
```

Example 10

```
newvalues <- data.frame(kilometers = c(155, 110), visits = c(12,
14))

predict(mr, newvalues)
```

Section 13.11. Independent Variable Selection: The Best Subsets Method

```
1+choose(8,1)+choose(8,2)+choose(8,3)+choose(8,4)+choose(8,5)
        +choose(8,6)+choose(8,7)+choose(8,8

2^8


head(mtcars,3)

firstmodel <- lm(y ~ x1 + x2 + x3 + x5 + x6, data = mtcarsnvn)

summary(firstmodel)

install.packages("leaps")

library(leaps)

install.packages("car")

library(car)

bestsubsets <- regsubsets(y ~ x1 + x2 + x3 + x5 + x6,
        data = mtcarsnvn, nbest = 1)

subsets(bestsubsets, statistic = "adjr2")

secondmodel <- lm(y ~ x1 + x5, data = mtcarsnvn)

summary(secondmodel)
```

Summary

```
pairs(mtcarsnvn, pch = 19, lower.panel = NULL)

install.packages("leaps")

library(leaps)

install.packages("car")

library(car)

bestsubsets <- regsubsets(y ~ x1 + x2 + x3 + x5 + x6,
        data = mtcarsnvn, nbest = 1)

subsets(bestsubsets, statistic = "adjr2"

secondmodel <- lm(y ~ x1 + x5, data = mtcarsnvn)

summary(secondmodel)

plot(fitted(secondmodel), resid(secondmodel), abline(h = 0),
    pch = 19, xlab = 'Predicted Value of y', ylab = 'Residuals')

newvalues <- data.frame(x1 = c(4, 6, 8), x5 = c(5.00, 3.35, 1.75))

predict(secondmodel, newvalues)

min(mtcarsnvn$x5)

max(mtcarsnvn$x5)

table(mtcarsnvn$x1)
```

End-of-Chapter 13 Exercises

Exercise 2

```
pf(52.277, 4, 11, lower.tail = FALSE)
```

Exercise 3

```
2 * pt(-0.2718, 11)

2 * pt(-2.5875, 11)

2 * pt(3.9340, 11, lower.tail = FALSE)

2 * pt(1.8232, 11, lower.tail = FALSE)
```

Exercise 4

```
pf(21.1331, 3, 6, lower.tail = FALSE)

2 * pt(-0.5737, 6)

2 * pt(5.362, 6, lower.tail = FALSE)

2 * pt(3.439, 6, lower.tail = FALSE)

2 * pt(0.823, 6, lower.tail = FALSE)
```

Exercise 5

```
tail(mtcars, 2)
```

```
pairs(mtcars[, c(1, 2, 6)], pch = 19, lower.panel = NULL)

reg_eq_mileage <- lm(mpg ~ cyl + wt, data = mtcars)

predicted <- fitted(reg_eq_mileage)

tail(predicted, 2)

newvalues <- data.frame(cyl = c(4, 8), wt = c(5, 2))

predict(reg_eq_mileage, newvalues)
```

R Functions

- confint(,level=) Provides confidence interval estimates for all independent vari- ables. The first argument is the name of the model object; the level of desired confidence is specified with the second argument level=.

- lm($y \sim x_1 + x_2 + ... + x_k$, data=) Provides the intercept term b_0 and the regression coefficients $b_1, b_2,...,b_k$ for an estimated multiple regression equation of the form $\hat{y}=b_0 +b_1x_1 +b_2x_2 +...+b_kx_k$.

- pairs() Produces a matrix of pairwise scatterplots of all variables simultaneously. The most important (and essential) argument to include is the name of the data object (in this case, it is brisbane).

- regsubsets($y \sim x_1 +x_2 +...+x_k$, data =, nbest =) A regression procedure that finds and reports the best-subset of predictor (independent) variables in terms of some pre-specified criterion such as the highest adjusted-r^2. In R, the regsubsets() routine first requires the installation of the leaps package. See Section 13.11.

. subsets() Plots the output from the regsubsets() best-subsets procedure in terms of the best-subset of predictor variables (in terms of some pre-specified number of variables) as well as a summarization statistic such as the adjusted-r^2 . In R, subsets() first requires the installation of the car package. See Section 13.11.