R Textbook Companion for
Statistics for Business and Economics
by Anderson, Sweeney, and Williams[1]

Created by
Shivam Agrawal
B.Tech.
Computer Science and Engineering
Techno India NJR Institute of Technology
Cross-Checked by
R TBC Team

May 26, 2020

Book Description

Title: Statistics for Business and Economics

Author: Anderson, Sweeney, and Williams

Publisher: Cengage Learning, USA

Edition: 11

Year: 2011

ISBN: 9780538471886

R numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means an R code whose theory is explained in Section 2.3 of the book.

Contents

List of R Codes

7

Chapter 2

Descriptive Statistics Tabular and Graphical Presentations

R code Exa 2.1a Summarizing Categorical Data Part 1

```
1                                           # Page no. :
                                               33

2
3  # Frequency Distribution (Categorical Data)
4
5  # Dataset
6
7  soft_drink_names <- c("Coke Classic", "Diet Coke", "
      Pepsi", "Diet Coke", "Coke Classic","Coke Classic
      ",
8                       "Dr. Pepper", "Diet Coke", "
                         Pepsi", "Pepsi" , "Coke
                         Classic", "Dr. Pepper",
9                       "Sprite", "Coke Classic", "Diet
                         Coke", "Coke Classic", "
                         Coke Classic", "Sprite",
10                      "Coke Classic", "Diet Coke", "
                         Coke Classic", "Diet Coke",
                         "Coke Classic", "Sprite",
```

```
11                          "Pepsi", "Coke Classic", "Coke
                              Classic", "Coke Classic", "
                              Pepsi", "Coke Classic",
12                          "Sprite", "Dr. Pepper", "Pepsi"
                              , "Diet Coke", "Pepsi", "
                              Coke Classic",
13                          "Coke Classic", "Coke Classic",
                              "Pepsi", "Dr. Pepper", "
                              Coke Classic", "Diet Coke",
14                          "Pepsi", "Pepsi", "Pepsi", "
                              Pepsi", "Coke Classic", "Dr.
                              Pepper", "Pepsi", "Sprite")
15
16 soft_drink_table <- data.frame(table(soft_drink_
     names))
17
18                                       # Page no. : 34
19
20 FD <- data.frame(Soft_drinks = soft_drink_table$soft
     _drink_names,
21                    Frequency = soft_drink_table$Freq)
                         # Frequency Distribution
22
23
24 RF <-  FD$Frequency / sum(FD$Frequency)   # Relative
      Frequency
25
26 FD <- cbind(FD,Relative_frequency = RF)
27
28 PF <- FD$Relative_frequency * 100    # Percentage
     Frequency
29
30 FD <- cbind(FD, Percentage_frequency = PF)
31
32 View(FD)    # Viewing the Frequency Distribution
     Table
33
34 # Total values of Frequency Distribution
```

11

```
35
36  total_freq <- sum(FD$Frequency)
37
38  total_rel_freq <- sum(FD$Relative_frequency)
39
40  total_per_freq <- sum(FD$Percentage_frequency)
41
42  cat("Total value for frequency is", total_freq, "\n"
        )
43  cat("Total value for relative frequency is", total_
        rel_freq, "\n")
44  cat("Total value for percentage frequency is", total
        _per_freq)
```

R code Exa 2.1b Summarizing Categorical Data Part 2

```
1                                          # Page no. :
                                               35
2
3  # Bar Charts and Pie Charts
4
5  # Dataset
6
7  soft_drink_names <- c("Coke Classic", "Diet Coke", "
       Pepsi", "Diet Coke", "Coke Classic","Coke Classic
       ",
8                          "Dr. Pepper", "Diet Coke", "
                            Pepsi", "Pepsi" , "Coke
                            Classic", "Dr. Pepper",
9                          "Sprite", "Coke Classic", "
                            Diet Coke", "Coke Classic",
                            "Coke Classic", "Sprite",
10                         "Coke Classic", "Diet Coke", "
                            Coke Classic", "Diet Coke",
                            "Coke Classic", "Sprite",
```

12

```
11                              "Pepsi", "Coke Classic", "Coke
                                    Classic", "Coke Classic",
                                    "Pepsi", "Coke Classic",
12                              "Sprite", "Dr. Pepper", "Pepsi
                                    ", "Diet Coke", "Pepsi", "
                                    Coke Classic",
13                              "Coke Classic", "Coke Classic"
                                    , "Pepsi", "Dr. Pepper", "
                                    Coke Classic", "Diet Coke",
14                              "Pepsi", "Pepsi", "Pepsi", "
                                    Pepsi", "Coke Classic", "Dr
                                    . Pepper", "Pepsi", "Sprite
                                    ")
15
16 soft_drink_table <- data.frame(table(soft_drink_
      names))
17
18 # Install Library if not installed
19
20 # install.packages("ggplot2")
21
22 # Import Library
23
24 library(ggplot2)
25
26 # Bar Chart
27
28 ggplot(soft_drink_table,aes(soft_drink_names,Freq,
      fill = soft_drink_names))+geom_bar(stat = "
      identity")+
29 labs(title="Bar chart" , x = "Soft Drink", y = "
      Frequency")+ylim(0,20)
30
31 # Pie Chart
32
33 soft_drink_purchase_slices <- soft_drink_table$Freq
34 soft_drink_names_labels <- soft_drink_table$soft_
      drink_names
```

```
35  soft_drink_purchase_pct <- (soft_drink_table$Freq/
       sum(soft_drink_table$Freq)) * 100
36
37  soft_drink_names_labels <- paste(soft_drink_names_
       labels,soft_drink_purchase_pct)
38  soft_drink_names_labels <- paste(soft_drink_names_
       labels, "%", sep = "")
39
40  pie(soft_drink_purchase_slices, labels = soft_drink_
       names_labels,
41      col = rainbow(length(soft_drink_names_labels)),
42      main = "Pie Chart for Soft Drink Purchase")
```

R code Exa 2.2a Summarizing Quantitative Data

```
1                                            # Page no. :
                                               39−40
2
3  # Frequency  Distribution  (Quantitative  Data)
4
5  audit_data <- c
       (12,15,20,22,14,14,15,27,21,18,19,18,22,33,16,18,17,23,28,13)

6
7  no_of_classes <- 5
8
9  width <- (max(audit_data) - min(audit_data)) / no_of
       _classes
10
11 width <- ceiling(width)   # Rounding  up  of  the  value
12
13 breaks <- seq(10,34,by = width)
14
15 class_range <- cut(audit_data, breaks, right=T)
16
```

14

```r
17  frequency <- table(class_range)
18
19  frequency_distribution <- data.frame(frequency)
20
21  frequency_distribution <- data.frame(class_range =
        frequency_distribution$class_range,
22                                        frequency =
                                            frequency_
                                            distribution
                                            $Freq)
23
24  # Note that :- Book answer will differ with my
        answer though number of classes and width of
25  # each class is same as in the book!!!
26
27                                              # Page no. :
                                                    41
28
29  relative_frequency <- round(frequency_distribution$
        frequency / sum(frequency_distribution$frequency)
30                                          ,2)   # Rounding
                                                of data to 2
                                                digits
31
32  percentage_frequency <- relative_frequency * 100
33
34  audit_data_FD <- cbind(frequency_distribution ,
35                                  relative_frequency ,
                                    percentage_frequency
                                    )
36
37  View(audit_data_FD)
38
39                                          # Page no. :
                                                41-42
40
41  # Dot Plot and Histogram
42
```

15

```
43  # Install Library if not installed
44
45  # install.packages("ggplot2")
46
47  # Import Library
48
49  library(ggplot2)
50
51  # Dot Plot
52
53  dotchart(audit_data, main = "Dot Plot for the Audit
       Time Data", xlab = "Audit Time (days)",
54              cex = 0.5)    # cex is for scaling
55
56  # Note that: Book dot plot is different from my dot
       plot.
57
58  # Histogram
59
60  ggplot(audit_data_FD, aes(class_range, frequency,
       fill = class_range))+
61    geom_histogram(stat = "identity")+labs(title="
         Histogram for the Audit Time Data" ,
62                                            x = "Audit
                                               Time (
                                               days)",
                                             y = "
                                               Frequency
                                               ")+ylim
                                               (0,8)
```

R code Exa 2.3a Scatter Plot and Tradeline

```
1                                            # Page no. :
                                               57−58
```

16

```r
 2
 3  # Dataset
 4
 5  week <- c(1,2,3,4,5,6,7,8,9,10)
 6  x <- c(2,5,1,3,4,1,5,3,4,2)
 7  y <- c(50,57,41,54,54,38,63,48,59,46)
 8  data <- data.frame(week,x,y)
 9
10  # Install Library if not installed
11
12  # install.packages("ggplot2")
13
14  # Import Library
15
16  library(ggplot2)
17
18  # Scatter Plot
19
20  ggplot(data, aes(x,y)) + geom_point() + geom_smooth(
        method = "lm", se = F) +
21  labs(title = "Scatter Plot and Tradeline for the
        Stereo and Sound Equipment Store",
22                                                      x
                                                        =
                                                        "
                                                        Number

                                                        of

                                                        Commercials
                                                        "
                                                        ,
23                                                      y

                                                        =
```

R code Exa 2.4a Cumulative Distributions

```
1                                              # Page no. :
                                                 44
2
3  # Cumulative Distributions
4
5  audit_time <- c("Less than or equal to 14", "Less
       than or equal to 19", "Less than or equal to 24",
6                  "Less than or equal to 29", "Less
                      than or equal to 34")
7  frequency <- c(4,8,5,2,1)    # Refer to table no. 2.5
        page no. 40
8  cumulative_freq <- cumsum(frequency)
9
10 DF <- data.frame(audit_time, frequency, cumulative_
     freq)
11
12 CRF <-  DF$cumulative_freq / sum(DF$frequency)    #
       Cumulative Relative Frequency
13
14 CPF <- CRF * 100    # Cumulative Percentage Frequency
```

```
15
16 DF <- cbind(DF, CRF, CPF)
17 View(DF)
18
19 breaks <- seq(9, 34, by = 5)
20 cumfreq0 <- c(0, DF$cumulative_freq)
21
22 DF2 <- data.frame(breaks, cumfreq0)
23
24 # Install Library if not installed
25
26 # install.packages("ggplot2")
27
28 # Import Library
29
30 library(ggplot2)
31
32 ggplot(DF2, aes(breaks, cumfreq0, group = 1)) + geom
       _point() + geom_line() +
33   xlim(c(0,35)) + labs(title = "Ogive For the Audit
        Time Data", x ="Audit Time (Days)",
34      y ="Cumulative Frequency")
```

R code Exa 2.5a Exploratory Data Analysis The Stem and Leaf Display

```
1                                               # Page no. :
                                                  49
2
3 # Exploratory Data Analysis : The Stem − and − Leaf
     Display
4
5 data <- c(112, 72, 69, 97, 107, 73, 92, 76, 86, 73,
     126, 128, 118, 127, 124, 82, 104, 132, 134, 83,
6            92, 108, 96, 100, 92, 115, 76, 91, 102,
               81, 95, 141, 81, 80, 106, 84, 119, 113,
```

19

```
                     98, 75,
7               68, 98, 115, 106, 95, 100, 85, 94, 106,
                   119)
8
9  stem(data)
10
11                                               # Page no. :
                                                     51
12
13 data2 <- c(1565, 1852, 1644, 1766, 1888, 1912, 2044,
       1812, 1790, 1679, 2008, 1852, 1967, 1954,
14              1733)
15 stem(data2)    # Answer is varing from the book
```

Chapter 3

Descriptive Statistics Numerical Measures

R code Exa 3.1a Measures of Location Mean Part 1

```
1                                           # Page  no . :
                                              87
2
3  # Mean
4
5  x <- c(46, 54, 42, 46, 32)
6
7  sample_mean <- mean(x)
8
9  cat("Sample  mean  for  x  is  ",sample_mean)
```

R code Exa 3.1b Measures of Location Mean Part 2

```
1                                           # Page  no . :
                                              88
2
```

```
3  # Dataset
4
5  graduate <- c(1,2,3,4,5,6,7,8,9,10,11,12)
6  salary <- c
       (3450,3550,3650,3480,3355,3310,3490,3730,3540,3925,3520,3480)

7

8

9  dataset <- data.frame(graduate, salary)
10
11 # Mean
12
13 mean_salary <- mean(dataset$salary)
14
15 cat("Mean monthly starting salary of 12 business
       school graduates is",mean_salary)
```

R code Exa 3.1c Measures of Location Median Part 1

```
1                                              # Page no.
                                                    : 89
2
3  # Median
4
5  x <- c(32, 42, 46, 46, 54)
6
7  median <- median(x)
8
9  cat("Median value for x is ", median)
```

R code Exa 3.1d Measures of Location Median Part 2

```
1                                                         # Page no
                                                              . : 89
2
3  salary <- c
       (3450 ,3550 ,3650 ,3480 ,3355 ,3310 ,3490 ,3730 ,3540 ,3925 ,3520 ,3480)

4
5  # Median
6
7  median <- median ( salary )
8
9  cat ( " The median of salary is " , median )
```

R code Exa 3.1e Measures of Location Percentiles and Quartiles

```
1                                                        # Page no .
                                                              : 90 −91
2
3  salary <- c
       (3450 ,3550 ,3650 ,3480 ,3355 ,3310 ,3490 ,3730 ,3540 ,3925 ,3520 ,3480)

4
5  # 85 th and 50 th Percentiles
6
7  solution <- quantile ( salary , probs = c (0.85 , 0.5))
8
9  cat ( " Value for 85 th and 50 th percentile are " ,
       solution [1] , " , " , solution [2])
10
11 # Note that : 85 th percentile value is different from
       the book
12
13
14 # 25 th , 50 th , 75 th Percentiles ( First , Second , Third
       Quartiles )
```

```
15
16 values <- quantile(salary, probs = c(0.25, 0.5,
       0.75))
17
18 cat("Value for first, second, third quartiles are ",
       values[1], ",", values[2], ",", values[3] )
19
20 # Note that: First and Second Quartile values are
       different from the book
```

R code Exa 3.2a Measures of Variability Range and IQR

```
 1                                                    # Page no. :
                                                        96−97
 2
 3 salary <- c
       (3450,3550,3650,3480,3355,3310,3490,3730,3540,3925,3520,3480)
 4
 5 # Range
 6
 7 range <- range(salary)
 8 diff <- range[2] - range[1]
 9
10 cat("Range is",diff)
11
12 # Inter−Quartile Range
13
14 IQR <- IQR(salary)
15
16 cat("IQR is ",IQR)
17
18 # Note that : IQR value of Book is different.
```

R code Exa 3.2b Measures of Variability Variance

```
1                                           # Page no. :
                                              97−98
2
3 students <- c(46, 54, 42, 46, 32)
4
5 # Variance
6
7 variance <- var(students)
8
9 cat("Variance of students is ",variance)
```

R code Exa 3.2c Measures of Variability Standard Deviation

```
1                                           # Page no. :
                                              98−99
2
3 salary <- c
     (3450,3550,3650,3480,3355,3310,3490,3730,3540,3925,3520,3480)
4
5 # Variance
6
7 variance <- var(salary)
8
9 # Standard Deviation
10
11 sd <- sd(salary)
12
13 cat("Variance of salary is ",variance)
14 cat("Standard Deviation is ",sd)
```

R code Exa 3.3a Z Score

```
1                                                  # Page
                                                     no. :
                                                     104
2
3  # Z-score
4
5  students <- c(46,54,42,46,32)
6
7  deviation <- students - mean(students)
8
9  sample_variance <- var(students)
10
11 dataset <- data.frame(students,deviation)
12
13 z <- c()
14
15 for(i in 1:length(dataset$students)){
16   z[i] <- deviation[i]/sqrt(sample_variance)
17 }
18
19 dataset <- cbind(dataset, zScore = z)
20
21 View(dataset)
```

R code Exa 3.4a Boxplot

```
1                                              # Page no.
                                                  : 110
2
```

```
3 salary <- c
      (3450,3550,3650,3480,3355,3310,3490,3730,3540,3925,3520,3480)

4
5 # Install Library if not installed
6
7 #install.packages(" ggplot2")
8
9 # Import Library
10
11 library(ggplot2)
12
13 # Boxplot
14
15 ggplot(data.frame(salary), aes(x = "", y = salary))
      + geom_boxplot(fill = "purple")+
16    ggtitle("Boxplot for Salary") + ylab("Salary")
```

R code Exa 3.5a Covariance and Correlation Coefficient

```
1                                            # Page no. :
                                               115−116 and 119
2
3 # Dataset
4
5 x <- c(2,5,1,3,4,1,5,3,4,2)
6 y <- c(50,57,41,54,54,38,63,48,59,46)
7
8 dataset <- data.frame(x,y)
9
10 # Coveriance
11
12 coveriance <- cov(dataset$x,dataset$y)
13
14 cat("Value of coveriance is ",coveriance)
```

27

```
15
16 # Correlation  Coefficient
17
18 correlation <- cor(dataset$x,dataset$y)
19
20 cat("Value of correlation coefficient is ",
       correlation)
```

R code Exa 3.5b Sample Correlation Coefficient

```
1                                          # Page no.
                                             : 121
2
3 # Data
4
5 x <- c(5,10,15)
6 y <- c(10,30,50)
7
8 # Sample  Correlation  Coefficient
9
10 corr <- cor(x, y)
11
12 cat("Sample correlation coefficient of x and y is ",
        corr)
```

R code Exa 3.6a Weighted Mean

```
1                                          # Page no. :
                                             124 - 125
2
3 # Data
4
5 purchase <- c(1,2,3,4,5)
```

```
6  cost <- c(3.00,3.40,2.80,2.90,3.25)
7  pound <- c(1200,500,2750,1000,800)
8
9  dataset <- data.frame(purchase,cost,pound)
10
11 # Weighted Mean
12
13 mean <- weighted.mean(dataset$cost,dataset$pound)
14
15 cat("Weigted mean for the dataset is",mean)
```

R code Exa 3.6b Grouped Data Mean and Sample Variance

```
1                                            # Page no.
                                               : 126
                                               − 127
2
3  # Data
4
5  audit <- c("10−14", "15−19","20−24","25−29","30−34")
6  midpoint <- c(12,17,22,27,32)
7  frequency <- c(4,8,5,2,1)
8
9  dataset <- data.frame(audit, midpoint, frequency)
10
11 # Mean for Grouped Data
12
13 mean <- weighted.mean(dataset$midpoint,dataset$
      frequency)
14
15 cat("Mean for grouped data is",mean)
16
17 # Sample Variance for Grouped Data
18
19 var <- sum(dataset$frequency*((dataset$midpoint -
```

29

```r
    mean)**2)) / (sum(dataset$frequency) - 1)
20
21 # Note that : Grouped sample variance has no inbuild
       function
22
23 cat("Sample variance for grouped data is",var)
```

Chapter 4

Introduction to Probability

R code Exa 4.1a Combinations

```
1                                                    # Page no. :
                                                         154
2
3  # Combinations
4
5  # Eg. 1
6
7  N <- 5
8  n <- 2
9
10 combinations <- choose(n = N, k = n)
11
12 cat("The total combinations are",combinations)
13
14 # Eg. 2
15
16 N <- 53
17 n <- 6
18
19 combinations <-  choose(n = N, k = n)
20
```

```
21  cat("The total combinations are",combinations)
```

R code Exa 4.1b Permutations

```
 1                                              # Page no.
                                                  : 155
 2
 3  # Permutations
 4
 5  N <- 5
 6  n <- 2
 7
 8  permutations <- choose(n = N, k = n)*factorial(n)
 9
10  cat("The total permutations are",permutations)
```

R code Exa 4.1c Assigning Probabilities

```
 1                                      # Page no. : 155
                                            - 156
 2
 3  # Assigning Probabilities
 4
 5  x <- c(0,1,2,3,4)
 6  y <- c(2,5,6,4,3)
 7
 8  DF <- data.frame(x,y)
 9
10  y_sum <- sum(DF$y)
11
12  prob <- DF$y / y_sum
13
14  DF <- cbind(DF,prob)
```

```
15
16  View(DF)
```

R code Exa 4.1d Probabilities Assigning Example

```
1                                           # Page no. :
                                            157 - 158
2
3  # Probabilities  Assigning  Example
4
5  x <- c(2,2,2,3,3,3,4,4,4)
6  y <- c(6,7,8,6,7,8,6,7,8)
7
8  z = list()
9
10 for(i in 1:length(x))
11 {
12    z[i] <- list(c(x[i],y[i]))
13 }
14
15 past_project <- c(6,6,2,4,8,2,2,4,6)
16
17 DF <- data.frame(x,y,I(z),past_project)
18
19 past_project_sum <- sum(DF$past_project)
20
21 p <- DF$past_project / past_project_sum
22
23 DF <- cbind(DF,p)
24
25 total_probability <- sum(DF$p)
26
27 cat("Total probability for the Sample Point is",
       total_probability)
28
```

```
29  View ( DF )
```

R code Exa 4.2a Probability of an Event

```
1                                            # Page no. :
                                                  161
2
3  # Probability  of  an  Event
4
5  # C  denotes  the  event  that  is  completed  in  10  months
        or  less
6
7  C <- c( list (c(2,6) ), list (c(2,7) ), list (c(2,8) ),
        list (c(3,7) ), list (c(3,8) ), list (c(4,6) ))
8  prob <- c(0.15,0.15,0.05,0.10,0.20,0.05)
9
10 dataset <- data.frame (I(C), prob )
11
12 event <- sum( dataset$prob )
13
14 # P(C) = P(2,6) + P(2,7) + P(2,8) + P(3,7) + P(3,8)
        + P(4,6)
15
16 cat(" Probability  of  an  event  P(C)  is", event )
17
18 # L  denotes  the  event  that  is  completed  in  less  than
        10  months
19
20 L <- c( list (c(2,6) ), list (c(2,7) ), list (c(3,7) ))
21 prob <- c(0.15,0.15,0.10)
22
23 dataset <- data.frame (I(L), prob )
24
25 # P(L) = P(2,6) + P(2,7) + P(3,7)
26
```

34

```
27  event2 <- sum(dataset$prob)
28
29  cat("Probability of an event P(L) is",event2)
30
31  # M denotes the event that is completed in more than
         10 months
32
33  M <- c(list(c(3,8)), list(c(4,7)), list(c(4,8)))
34  prob <- c(0.05,0.10,0.15)
35
36  dataset <- data.frame(I(M),prob)
37
38  # P(M) = P(3,8) + P(4,7) + P(4,8)
39
40  event3 <- sum(dataset$prob)
41
42  cat("Probability of an event P(L) is",event3)
```

R code Exa 4.3a Probability Computation using Complement

```
1                                            # Page no. :
                                                165
2
3  # Probability Computation using Complement
4
5  p_comp_A <- 0.80
6
7  p_A <- 1 - p_comp_A
8
9  cat("Probability for A is",p_A)
```

R code Exa 4.3b Intersection and Union of Events

35

```
 1                                                    # Pahe no. : 167
 2
 3  # Intersection and Union of Events
 4
 5  a <- 5
 6  n <- 50
 7  c <- 6
 8  d <- 2
 9
10  p_L <- a / n
11
12  p_D <- c / n
13
14  p_L_and_D <- d / n
15
16
17  cat("Probability for L intersection D is",p_L_and_D)
18
19  p_L_or_D <- p_L + p_D - p_L_and_D
20
21  cat("Probability for L union D is",p_L_or_D)
```

R code Exa 4.3c Addition Law

```
 1                                                    # Page no. :
                                                        168
 2
 3  # Addition Law
 4
 5  p_S <- 0.30
 6  p_W <- 0.20
 7  p_S_and_W <- 0.12
 8
 9  p_S_or_W <- p_S + p_W - p_S_and_W
10
```

```
11 cat(" Probability  after  applying  addition  law  is  ",p_
       S_or_W)
```

R code Exa 4.4a Conditional Probability

```
 1                                              # Page no. :
                                                171 - 174
 2
 3 # Conditional  Probability
 4
 5 position <- c(" Promoted ", " Not  Promoted ")
 6 gender <- c(" Men "," Women ")
 7 number1 <- c(288, 36)    # Promoted
 8 number2 <- c(672, 204)   # Not  Promoted
 9
10 DF <- data.frame( position, gender, number1, number2)
11
12 table <- round(prop.table(DF[,3:4]), 2)
13 table <- as.matrix(table)
14 table
15
16 position <- c(" Promoted ", " Not  Promoted ")
17 men <- c(table[1,1][[1]], table[1,2][[1]])
18 women <- c(table[2,1][[1]], table[2,2][[1]])
19
20 DF2 <- data.frame( position, men, women)
21 View(DF2)
22
23 columnSums <- apply(DF2[,2:3],2, sum)
24 rowSums <- apply(DF2[,2:3], 1, sum)
25
26 cat(" Total  Probability  For  having  Men  is ",
       columnSums[1])
27 cat(" Total  Probability  For  having  Women  is ",
       columnSums[2])
```

```
28  cat("Total  Probability  For  Promotion", rowSums[1])
29  cat("Total  Probability  For  Not  a  Promotion", rowSums
       [2])
30
31  cond1 <- DF2$men[1]  / columnSums[1]
32
33  cat("Conditional  Probability  for  Men  and  getting
        Promoted  given  the  Probability
34        of  Total  Men  is", cond1)
35
36  cond2 <- DF2$women[1] / columnSums[2]
37
38  cat("Conditional  Probability  for  Women  and  getting
        Promoted  given  the  Probability
39        of  Total  Women  is", cond2)
```

R code Exa 4.4b Multiplication Law

```
 1                                               # Page no. :
                                                  174 - 175
 2
 3  # Multiplication  Law
 4
 5  # P(S | D) = 0.75
 6
 7  x <- 0.75
 8
 9  # Event  S = 0.84
10
11  y <- 0.84
12
13  x_and_y <- x * y
14
15  cat("After  applying  multiplication  law  we  get", x_
       and_y)
```

```
16
17  # Multiplication Law for Independent Events
18
19  A <- 0.80
20
21  B <- 0.80
22
23  A_and_B <- A * B
24
25  cat("After applying multiplication law for
        independent events A and B we get", A_and_B)
```

R code Exa 4.5a Bayes Theorem Tabular Approach

```
1                                      # Page no. : 182
2
3  # Tabular Approach for Bayes' Theorem
4
5  events <- c("A1","A2")
6  prior_probabilities <- c(0.65,0.35)
7
8  conditional_probabilities <- c(0.02,0.05)
9
10 joint_probabilities <- prior_probabilities *
      conditional_probabilities
11
12 total_joint_probability <- sum(joint_probabilities)
13
14 posterior_probabilities <- joint_probabilities /
      total_joint_probability
15
16 DF <- data.frame(events,prior_probabilities,
      conditional_probabilities
17                   ,joint_probabilities,posterior_
                        probabilities)
```

```
18
19  View(DF)
20
21  cat("Total posterior probability is",sum(DF$
       posterior_probabilities))
```

Chapter 5

Discrete Probability Distribution

R code Exa 5.1a Discrete Probability Distribution Graph Expected value
Variance and Standard Deviation

```
1                                                    # Page no. : 198
                                                      - 199
2
3 # Discrete Probability Distribution :
4
5 x <- c(0,1,2,3,4,5)
6 prob_of_x <- c(0.18,0.39,0.24,0.14,0.04,0.01)
7
8 dataset <- data.frame(x, prob_of_x)
9
10 # Install Library if not installed
11
12 # install.packages("ggplot2")
13
14 # Import Library
15
16 library(ggplot2)
17
```

```
18  ggplot(dataset, aes(x = x, y = prob_of_x)) + geom_
       bar(stat = "identity", fill = "blue") + labs(
19    title = "Graphical representation of the
          Probability Distribution",
20    x = "Number of Automobiles", y = "Probability")
21
22                                      # Page no. : 203
23
24  # Expected value
25
26  exp_values <- dataset$x * dataset$prob_of_x
27
28  dataset <- data.frame(cbind(dataset, exp_values))
29
30  expected_value <- sum(dataset$exp_values)
31  cat("Expected value for the given problem is",
       expected_value)
32
33                                      # Page no. : 204 -
                                           205
34
35  # Variance and Standard Deviation
36
37  deviation_of_x <- (dataset$x - expected_value)
38
39  deviation_square <- (deviation_of_x) ** 2
40  variance <- sum(dataset$prob_of_x * deviation_square
       )
41  standard_deviation <- sqrt(variance)
42
43  cat("Variance is",variance)
44  cat("Standard Deviation is",standard_deviation)
```

R code Exa 5.2a Binomial Probability Distribution

```
1                                              # Page  no.  :  211
2
3  # Binomial  Probability  Distribution
4
5  no_of_trials  <-  3
6  no_of_successes  <-  2
7  BPD  <-  choose(n = no_of_trials, k = no_of_successes)
8
9  cat("Answer  is",BPD)
10
11  no_of_successes  <-  3
12  BPD  <-  choose(n = no_of_trials, k = no_of_successes)
13  cat("Answer  is",BPD)
```

R code Exa 5.2b Binomial Probability Distribution Eg2

```
1                                                      # Page  no
                                                           .  :
                                                          211
2
3  # Binomial  Probability  Distribution  Eg-2
4
5  # Data
6  customer1  <-  c("Purchase","Purchase","No Purchase")
7  customer2  <-  c("Purchase","No Purchase","Purchase")
8  customer3  <-  c("No Purchase","Purchase","Purchase")
9
10  customer  <-  data.frame(customer1,customer2,customer3
     )
11
12  len  <-  nrow(customer)   # Trial
13
14  x  <-  1     # For  purchase
15  y  <-  0     # For  no  purchase
16
```

```r
17  p <- 0.30
18  q <- 1 - p
19  outcome <- c()
20
21  # Install Library if not installed
22
23  install.packages("Rlab")
24
25  # Import Library
26
27  library(Rlab)   # For dbern
28
29  for(i in 1:len)
30  {
31    if(customer1[i] == "Purchase" && customer2[i] == "
        Purchase" && customer3[i] == "Purchase")
32    {
33      outcome[i] = dbern(x,p) * dbern(x,p) * dbern(x,p
          )
34    } else if(customer1[i] == "Purchase" && customer2[
        i] == "Purchase" && customer3[i] == "No
        Purchase")
35    {
36      outcome[i] = dbern(x,p) * dbern(x,p) * dbern(y,p
          )
37    } else if(customer1[i] == "Purchase" && customer2[
        i] == "No Purchase" && customer3[i] == "
        Purchase")
38    {
39      outcome[i] = dbern(x,p) * dbern(y,p) * dbern(x,p
          )
40    } else if(customer1[i] == "Purchase" && customer2[
        i] == "No Purchase" && customer3[i] == "No
        Purchase")
41    {
42      outcome[i] = dbern(x,p) * dbern(y,p) * dbern(y,p
          )
43    } else if(customer1[i] == "No Purchase" &&
```

```r
        customer2[i] == "Purchase" && customer3[i] == "
          Purchase")
44    {
45      outcome[i] = dbern(y,p) * dbern(x,p) * dbern(x,p
          )
46    } else if(customer1[i] == "No Purchase" &&
          customer2[i] == "Purchase" && customer3[i] == "
          No Purchase")
47    {
48      outcome[i] = dbern(y,p) * dbern(x,p) * dbern(y,p
          )
49    } else if(customer1[i] == "No Purchase" &&
          customer2[i] == "No Purchase" && customer3[i]
          == "Purchase")
50    {
51      outcome[i] = dbern(y,p) * dbern(y,p) * dbern(x,p
          )
52    } else
53    {
54      outcome[i] = dbern(y,p) * dbern(y,p) * dbern(y,p
          )
55    }
56  }
57
58  customer <- cbind(customer,outcome)
59
60  View(customer)
61
62                                                # Page no. :
                                                   212 -
                                                   213
63
64  # Binomial Probability Function
65
66  x <- c(0,1,2,3)
67  fun <- c()
68
69  for (i in 0:length(x)) {
```

45

```r
70   fun[i] <- dbinom(x[i],len,p)
71 }
72
73 dataset <- data.frame(x, fun)
74 View(dataset)
75
76 # Install Library if not installed
77
78 # install.packages("ggplot2")
79
80 # Import Library
81
82 library(ggplot2)
83
84 ggplot(dataset, aes(x = x, y = fun)) + geom_bar(stat
       = "identity", fill = "blue") + labs(
85   title = "Graphical representation of the
        Probability Distribution",
86   x = "Number of Customers", y = "Probability")
87
88                                            # Page no. :
                                               214 -
                                               215
89
90 # Expected Value, Variance and Standard Deviation
     for Binomial Probability Distribution
91
92 expected_value <- len * p
93 variance <- len * p * q
94 standard_deviation <- sqrt(variance)
95
96 cat("Expected value is",expected_value)
97 cat("Variance is",variance)
98 cat("Standard deviation is",standard_deviation)
```

R code Exa 5.3a Poisson Probability Distribution

```
1                                           # Page no. :
                                                218
2
3  # Poisson  Probability  Distribution
4
5  x = 5
6  expected_value <- 10
7  e <- 2.72   # Exponential  value
8
9  PPD <- dpois(x, expected_value)  # Poisson
       Probability  Distribution
10
11 cat("Answer  is",PPD)
```

R code Exa 5.4a Hypergeometric Probability Distribution

```
1                                       # Page no. : 222
                                            - 223
2
3  # Hypergeometric  Probability  Distribution
4
5  # Probability  for  1  defective  item
6
7  N <- 12
8  n <- 3
9  r <- 5
10 x <- 1
11
12 HPD <- dhyper(x = x,m = r,n = N-r,k = n)
13
14 cat("Answer  is",HPD)
15
16 # Probability  for  atleast  one  defective  item
```

47

```
17
18  HPD <- dhyper(x = 0,m = r,n = N-r,k = n)  #
        Probability for no defective item
19
20  cat("Answer is",1 - HPD)  # Probability for atleast
        one defective item
21
22  # Expected value, Variance and Standard Deviation
23
24  expected_value <- n * (r / N)
25  variance <- expected_value * (1 -(r/N)) * ((N-n)/(N
        -1))
26  standard_deviation <- sqrt(variance)
27
28  cat("Expected value",expected_value)
29  cat("Variance",variance)
30  cat("standard deviation",standard_deviation)
```

R code Exa 5.5a Expected Value and Variance

```
1                                          # Page no. :
                                               203 − 204
2
3  # Expected Value and Variance
4
5  x <- c(0,1,2,3,4,5)
6  prob_of_x <- c(0.18,0.39,0.24,0.14,0.04,0.01)
7  expected_value <- x * prob_of_x
8
9  DF <- data.frame(x, prob_of_x, expected_value)
10
11  expected_mean <- sum(DF$expected_value)
12
13  cat("Expected mean is",expected_mean)
14
```

```r
15  deviation <- DF$x - expected_mean
16  sq_deviation <- deviation ** 2
17  expected_value2 <- DF$prob_of_x * sq_deviation
18
19  DF <- cbind(DF, deviation, sq_deviation, expected_
      value2)
20  View(DF)
21
22  expected_variance <- sum(DF$expected_value2)
23
24  cat("Expected variance is", expected_variance)
25
26  expected_SD <- sqrt(expected_variance)
27
28  cat("Expected SD is", expected_SD)
```

Chapter 6

Continuous Probability Distribution

R code Exa 6.1a Uniform Probability Distribution

```
1                                          # Page no . : 234 −
                                             236
2
3 # Uniform Probability Distribution
4
5 a <- 120
6 b <- 140
7 fun_over_x <- 1/20
8
9 # Since uniform probability is symmetric we can
      split it into left and right parts which
10 # are symmetric in nature
11
12 c <- 130   # (120+140)/2 = 130
13
14 # Probability of uniform probability distribution
      is the area of the figure (rectangle)
15
16 # Area for the left symmetric part of the figure
```

```
17
18  area <- punif(c,a,b)
19  area_full <- 2 * area
20
21  cat(" Probability is ",area_full)
22
23  # Expected value, Variance and standard Deviation
24
25  expected_value <- (a + b) / 2
26  variance <- (b - a) ** 2 / 12
27  standard_deviation <- sqrt(variance)
28
29  cat(" Expected value is ",expected_value)
30  cat(" Variance is ",variance)
31  cat(" Standard deviation is ",standard_deviation)
```

R code Exa 6.2a Normal Probability Distribution

```
1                                              # Page no. : 246
                                                 − 247
2
3  # Normal Probability Distribution
4
5  mean <- 36500
6  sigma <- 5000
7  x <- 40000
8  probability <- pnorm(40000, mean=36500, sd=5000,
      lower.tail= F)
9
10  cat(" The probability of x exceed 40000 is ",
      probability)
11
12                                             # Page no. : 247
13
14  probability <- 0.10
```

```
15  z_value <- round(qnorm(probability), 2)    # Round it
        to 2 decimal place
16  x <- (sigma * z_value) + mean
17
18  cat("Value of x for not more than 10% of area is
        selected is",x)
```

R code Exa 6.3a Normal Approximation of Binomial Probabilities

```
1                                        # Page no. : 251 −
                                                252
2
3  # Normal Approximation of Binomial Probabilities
4
5  n <- 100
6  p <- 0.1
7  q <- 1 - p
8
9  mu <- n * p
10 sigma <- sqrt(mu * q)
11
12 # P(x = 12)  ==>  P(11.5 <= x <= 12.5)
13
14 x1 <- 12.5
15 x2 <- 11.5
16 z_value1 <- (x1 - mu) / sigma
17 z_value2 <- (x2 - mu) / sigma
18
19 area1 <- pnorm(z_value1)
20 area2 <- pnorm(z_value2)
21
22 diff <- area1 - area2
23
24 cat("The normal approximation to the probability of
        12 successes in 100 trials is ",diff)
```

```
25
26  # Probability  for  13
27
28  x <- 13.5
29  z <- (x - mu) / sigma
30  ans <- pnorm(z)
31
32  cat("Answer is", ans)
```

R code Exa 6.4a Exponential Probability Distribution

```
 1                                          # Page no. :
                                               255
 2
 3  # Exponential  Probability  Distribution
 4
 5  mu <- 15
 6  x1 <- 6
 7  x2 <- 18
 8
 9  # P(x <= 6)
10
11  EPD <- pexp(x1,1/mu)
12
13  # P(x <= 18)
14
15  EPD2 <- pexp(x2, 1/mu)
16
17  diff <- EPD2 - EPD
18
19  cat("The probability that loading a truck will take
        between 6 and 18 minutes is ",diff)
20
21  SD <- mu
22  sigma <- SD ** 2
```

```
23
24 cat("Variance is", sigma)
```

Chapter 7

Sampling and Sampling Distribution

R code Exa 7.1a Point Estimator

```
1                                                # Page no
                                                 . :
                                                 274
2
3 # Sample Mean and Sample Standard Deviation
4
5 annual_salary <- c
     (49094.30,53263.90,49643.50,49894.90,47621.60,55924.00,49092.30,5
6                      55109.70,45922.60,57268.40,55688.80,51564.70,5618
7                      51932.60,52973.00,45120.90,51753.00,54391.80,5016
8                      50979.40,55860.90,57309.10)
9
10 program <- c("Yes","Yes","Yes","Yes","No","Yes","Yes
      ","Yes","Yes","Yes","Yes","No","Yes","No","No","
      Yes",
11                "No","Yes","Yes","Yes","Yes","Yes","No","
```

```
                     No","No","No","No","Yes","Yes","No")
12
13 dataset <- data.frame(annual_salary, program)
14
15 sample_mean <- mean(dataset$annual_salary)
16 sample_sd <- sd(dataset$annual_salary)
17
18 cat("Sample mean of the data is", sample_mean)
19 cat("Sample satndard deviation is", sample_sd)
20
21 # Note that : Book SD is different from our SD
22
23 # Sample Proportion
24
25 n <- nrow(dataset)
26 x <- 19
27
28 sample_proportion <- x / n
29
30 cat("Sample Proportion is",sample_proportion)
```

R code Exa 7.2a Sampling Distribution

```
1                                              # Page no. 277
                                                  − 278
2
3 # Sampling Distribution
4
5 mean_annual_salary <- c("49500.00−49999.99","
      50000.00−50499.99","50500.00−50999.99","
      51000.00−51499.99",
6                            "51500.00−51999.99","
                             52000.00−52499.99","
                             52500.00−52999.99","
                             53000.00−53499.99",
```

56

```
7                                    "53500.00 - 53999.99")
8  frequency <- c(2,16,52,101,133,110,54,26,6)
9  relative_frequency <- c
       (.004,.032,.104,.202,.266,.220,.108,.052,.012)
10
11 DF <- data.frame(mean_annual_salary,frequency,
       relative_frequency)
12
13 library(ggplot2)
14
15 ggplot(DF,aes(mean_annual_salary,relative_frequency)
       ) +
16   geom_histogram(stat = "identity", fill = "purple")
          + labs(title = "Relative Frequency Histogram",
17                                       x = " Mean Salary",
                                          y = "frequency"
                                          )
```

R code Exa 7.3a Sampling Distribution of Sample Mean

```
1                                        # Page no. :
                                          281
2
3  # Sampling Distribution of xbar
4
5  sigma <- 4000
6  N <- 2500
7  n <- 30
8  x <- n / N
9
10 if(x > 0.05)    # Condition to include finite
      population factor or not (< 5%)
11 {
12   standard_error <- sqrt((N-n)/(N-1)) * (sigma /
       sqrt(n))
```

```
13 } else{
14     standard_error <- sigma / sqrt(n)
15 }
16
17 cat("Standard deviation of sample mean is",standard_
       error)
18
19                                             # Page no. : 284
20
21 # To find probability that xbar is between 51300 and
       52300
22
23 xbar1 <- 52300
24 xbar2 <- 51300
25 mu <- 51800
26
27 z1 <- (xbar1 - mu) / standard_error
28 z2 <- (xbar2 - mu) / standard_error
29
30 p1 <- pnorm(z1, lower.tail = T)
31 p2 <- pnorm(z2, lower.tail = T)
32
33 diff <- p1 - p2
34
35 cat("Probability that xbar is between 51300 and
       52300 is", diff)
```

R code Exa 7.3b Relationship between Sample Size and Sampling Distribution of xbar

```
1                                             # Page no. :
                                                285 −
                                                286
2
3 # Relationship between Sample Size and Sampling
```

58

```
         Distribution of xbar
4
5  sigma <- 4000
6  n <- 100
7  population_mean <- 51800
8
9  standard_error <- sigma / sqrt(n)
10
11 xbar1 <- 52300
12 xbar2 <- 51300
13
14 z1 <- (xbar1 - population_mean) / standard_error
15 z2 <- (xbar2 - population_mean) / standard_error
16
17 p1 <- pnorm(z1, lower.tail = T)
18 p2 <- pnorm(z2, lower.tail = T)
19
20 diff <- p1 - p2
21
22 cat("Probability that xbar is between 51300 and
       52300 with increased sample size is", diff)
```

R code Exa 7.4a Sampling Distribution of Sample Proportion

```
1                                              # Page no.
                                                 : 290
2
3  # Sampling Distribution of pbar
4
5  population_proportion <- 0.60
6  n <- 30
7  N <- 2500
8
9  x <- n / N
10
```

59

```
11  if(x > 0.05)    # Condition to include finite
        population or not ( < 5%)
12  {
13    standard_deviation <- sqrt((N-n)(N-1)) * sqrt((
        population_proportion *
14                                                              (1

                                                                -

                                                                population
                                                                -
                                                                proportion
                                                                )
                                                                )

                                                                /

                                                                n
                                                                )

15  } else{
16    standard_deviation <- sqrt((population_proportion
        * (1 - population_proportion)) / n)
17  }
18
19  cat("Standard deviation for sample proportion is",
        standard_deviation)
```

R code Exa 7.4b Practical value of the Sampling Distribution of Sample Proportion

```
1                                          # Page no. : 291
2
3  # Practical value of the Sampling Distribution of
       pbar
```

```
4
5 population_proportion <- 0.60
6 standard_error <- 0.0894
7 sample_proportion <- 0.65
8
9 z_value <- (sample_proportion - population_
      proportion) / standard_error
10
11 prob1 <- pnorm(sample_proportion, population_
      proportion, standard_error, lower.tail =T)
12
13 sample_proportion2 <- 0.55
14
15 z_value <- (sample_proportion2 - population_
      proportion) / standard_error
16
17 prob2 <- pnorm(sample_proportion2, population_
      proportion, standard_error, lower.tail =T)
18
19 final_prob <- prob1 - prob2
20
21 cat("The final probability is",final_prob)
```

R code Exa 7.4c Practical value of the Sampling Distribution of Sample Proportion Eg2

```
1                                           # Page no. : 292 -
                                                293
2
3 # Practical value of the Sampling Distribution of
      pbar Eg-2
4
5 population_proportion <- 0.60
6 n <- 100
7
```

61

```
 8  standard_error <- sqrt(population_proportion*(1 -
      population_proportion)/(n))
 9
10  sample_proportion <- 0.65
11
12  z_value <- (sample_proportion - population_
      proportion) / standard_error
13
14  prob1 <- pnorm(sample_proportion, population_
      proportion, standard_error, lower.tail =T)
15
16  sample_proportion2 <- 0.55
17
18  z_value <- (sample_proportion2 - population_
      proportion) / standard_error
19
20  prob2 <- pnorm(sample_proportion2, population_
      proportion, standard_error, lower.tail =T)
21
22  final_prob <- prob1 - prob2
23
24  cat("The final probability is",final_prob)
```

Chapter 8

Interval Estimation

R code Exa 8.1a Population Mean Sigma Known

```r
1                                          # Page no. :
                                             310 - 314
2 # Population Mean Sigma known
3
4 pop_sd <- 20
5 sample_size <- 100
6 sample_mean <- 82
7
8 standard_error <- pop_sd / sqrt(sample_size)
9
10 # 95% confidence interval
11
12 margin_of_error <- qnorm(0.975)*standard_error    #
      95% confidence interval --> 1 - 0.025 = 0.975
13
14 IE <- sample_mean + c(-margin_of_error, margin_of_
      error)
15
16 cat("The margin of error is given by", margin_of_
      error)
17 cat("The 95% interval estimate is given by", IE)
```

```
18
19  # 90% confidence interval
20
21  margin_of_error <- qnorm(0.95)*standard_error    #
       90% confidence interval --> 1 - 0.05 = 0.95
22
23  IE <- sample_mean + c(-margin_of_error, margin_of_
       error)
24
25  cat("The margin of error is given by", margin_of_
       error)
26  cat("The 90% interval estimate is given by", IE)
27
28  # 99% confidence interval
29
30  margin_of_error <- qnorm(0.995)*standard_error    #
       99% confidence interval --> 1 - 0.005 = 0.995
31
32  IE <- sample_mean + c(-margin_of_error, margin_of_
       error)
33
34  cat("The margin of error is given by", margin_of_
       error)
35  cat("The 99% interval estimate is given by", IE)
```

R code Exa 8.2a Population Mean Sigma Unknown

```
1                                          # Page no. :
                                              319 - 320
2
3
4  # Population Mean Sigma Unknown
5
6  cradit_card_balances <- c
       (9430,7535,4078,5604,5179,4416,10676,1627,10112,6567,13627,18719,
```

64

```
7                                    10544 ,13659 ,7061 ,6245 ,13021 ,9719 ,2200 ,1074

8                                    7917 ,11346 ,12806 ,4972 ,11356 ,7117 ,9465 ,1926

9                                    6845 ,10493 ,615 ,13627 ,12557 ,6232 ,9691 ,11448

10                                   12851 ,5337 ,8372 ,7445 ,11032 ,6525 ,5239 ,6195 ,

11
12  test <- t.test(cradit_card_balances)
13  IE1 <- test$conf.int[1]
14  IE2 <- test$conf.int[2]
15
16  cat("The 95% interval estimate is given by", IE1, "
        to", IE2)
```

R code Exa 8.2b Population Mean Sigma Unknown Eg2

```
1                                       # Page no. :
                                        321 − 322
2
3  # Population Mean Sigma Unknown Eg−2
4
5  data <- c
       (52,44,55,44,45,59,50,54,62,46,54,42,60,62,43,42,48,55,57,56)

6
7
8  hist(data, col = "blue", main = "Histogram of
       Training Times", xlab = "Traing Times (days)",
9         ylab = "Frequency")
10
11 test <- t.test(data)
12 IE1 <- test$conf.int[1]
```

```
13  IE2 <- test$conf.int[2]
14
15  cat("The 95% interval estimate is given by", IE1, "
        to", IE2)
```

R code Exa 8.3a Determining the Sample Size

```
1                                              # Page no. :
                                                  326 - 327
2
3  # Determining the Sample Size
4
5  margin_of_error <- 2
6  z_value <- 1.96    # 95% Level of Confidence Interval
7
8  sample_standard_deviation <- 9.65
9
10 sample_size <- ((z_value)**2) * ((sample_standard_
        deviation)**2) / (margin_of_error)**2
11
12 cat("Sample size is",ceiling(sample_size))
13 # If Sample Size is not integer then we round up to
        next higher integer
```

R code Exa 8.4a Population Proportion

```
1                                              # Page no. :
                                                    329
2
3  # Population Proportion
4
5  N <- 900
6  n <- 396
```

```
 7
 8  p <- n / N
 9  q <- 1 - p
10
11  # Confidence Interval is 95%
12
13  z_value <- qnorm(0.975)   # 95% confidence interval
       --> 1 - 0.025 = 0.975
14
15  margin_of_error <- z_value * sqrt((p*q)/N)
16  IE <- p + c(-margin_of_error, margin_of_error)
17
18  cat("The margin of error is given by", margin_of_
        error)
19  cat("The 95% interval estimate is given by", IE)
```

R code Exa 8.4b Determining the Sample Size

```
 1                                          # Page no. :
                                               330
 2
 3  # Determining the Sample Size
 4
 5  margin_of_error <- 0.025
 6  z_value <- 1.96   # 95% Level of Confidence Interval
 7
 8  p <- 0.44
 9  q <- 1 - p
10
11  sample_size <- ((z_value)**2 * p * q) / (margin_of_
        error)**2
12
13  cat("Sample size is",ceiling(sample_size))
14  # If Sample Size is not integer then we round up to
        next higher integer
```

```r
15
16                                              # Page no. :
                                                   331
17
18 margin_of_error <- 0.025
19 z_value <- 1.96    # 95% Level of Confidence Interval
20
21 p <- 0.50
22 q <- 1 - p
23
24 sample_size <- ((z_value)**2 * p * q) / (margin_of_
      error)**2
25
26 cat("Sample size is",ceiling(sample_size))
27 # If Sample Size is not integer then we round up to
      next higher integer
```

Chapter 9

Hypothesis Testing

R code Exa 9.1a Population Mean Sigma Known One Tailed Test

```
1                                              # Page no . :
                                                    359 − 360
2
3  # Population Mean Sigma Known One Tailed Test
4
5  sigma <- 0.18
6  n <- 36
7  xbar <- 2.92
8  mu <- 3
9
10 z_value <- (xbar - mu) / (sigma / sqrt(n))
11 z_value <- round(z_value ,2)
12
13 alpha <- 0.01
14
15 # P Value Approch (Lower−Tail Test)
16
17 pval <- pnorm(z_value)
18
19 if(pval > alpha)
20 {
```

```
21    cat ("Since p−value ",pval ,"is greater than 0.01 ,
          therefore we will accept null hypothesis")
22  } else {
23    cat ("Since p−value ",pval ,"is less than 0.01 ,
          therefore we will reject null hypothesis and
          accept
24        alternative hypothesis.")
25  }
26
27  # Critical Value Approch (Lower−Tail Test)
28
29  z_alpha <- qnorm(1 - (alpha))    # Area of 0.01 to
          the left (1 − 0.01 = 0.99)
30  critical_approch <- -z_alpha
31
32  if(z_value <= critical_approch)
33  {
34    cat ("Since z−value", z_value ,"is less then or
          equal to", critical_approch , "therefore we
          reject
35        the null hypothesis and accept the alternative
              hypothesis .")
36  } else{
37    cat ("Since z−value", z_value ,"is more than",
          critical_approch ,"therefore we accept the null
          hypothesis .")
38  }
```

R code Exa 9.1b Population Mean Sigma Known Two Tailed Test

```
1                                              # Page no. :
                                                   362 −
                                                   364
2
3  # Population Mean Sigma Known Two Tailed Test
```

```r
4
5  sample_size <- 50
6  sample_mean <- 297.6
7  population_mean <- 295
8  significance_level <- 0.05    # alpha
9  population_sd <- 12
10
11                                           # Page no. :
                                                 364
12
13 ## Critical value approch
14
15 z_value <- (sample_mean - population_mean) / (
      population_sd / sqrt(sample_size))
16
17 z_half_alpha <- qnorm(1 - (significance_level/2))
18 critical_value_1 <- -z_half_alpha
19 critical_value_2 <- z_half_alpha
20
21 if(z_value >= critical_value_2 || z_value <=
      critical_value_1)
22 {
23   cat("Since z-value", z_value ,"does not lie in the
         range", critical_value_1 ,"and", critical_
      value_2
24       ,"therefore we reject the null hypothesis and
            accept the alternative hypothesis.")
25 } else{
26   cat("Since z-value", z_value ,"lies in the range",
         critical_value_1 ,"and", critical_value_2
27       ,"therefore we accept the null hypothesis.")
28 }
29
30                                           # Page no. :
                                                 363 - 364
31
32 ## P-value approch
33
```

```
34  area_under_curve <- 1 - pnorm(z_value)
35  pval <- 2 * area_under_curve    # P-value
36  if(pval > 0.05)
37  {
38    cat("Since p-value ",pval ,"is greater than 0.05,
         therefore we will accept null hypothesis")
39  } else {
40    cat("Since p-value ",pval ,"is less than 0.05,
         therefore we will reject null hypothesis and
         accept
41       alternative hypothesis.")
42  }
```

R code Exa 9.1c Relationship between Interval Estimation and Hypothesis Testing

```
1                                              # Page no. :
                                                 366 - 367
2
3  # Relationship between Interval Estimation and
       Hypothesis Testing
4
5  mu <- 295
6  alpha <- 0.05
7  n <- 50
8  xbar <- 297.6
9  sigma <- 12
10
11 z_value <- 1.96     # alpha = 0.05 so alpha/2 = 0.05/
       2 = 0.025 (z-value is for 0.025)
12
13 # 95% Confidence Interval
14
15 margin_of_error <- z_value * (sigma / sqrt(n))
16
```

72

```
17  IE <- xbar + c(-margin_of_error, margin_of_error)  #
        Interval Estimate
18
19  cat("Interval estimate for 95% Confidence Interval
        is",IE)
20
21  if(mu >= IE[2] || mu <= IE[1])
22  {
23    cat("Null Hypothesis is rejected..")
24  } else{
25    cat("Null Hypothesis cannot be rejected..")
26  }
```

R code Exa 9.2a Population Mean Sigma Unknown One Tailed Test

```
1                                              # Page no. :
                                                   371 -
                                                   372
2
3   # Population Mean Sigma Unknown One Tailed Test
4
5   mu0 <- 7
6   alpha <- 0.05
7   xbar <- 7.25
8   s <- 1.052
9   n <- 60
10
11  t_value <- (xbar - mu0) / (s / sqrt(n))
12
13  df <- n - 1
14
15  # Upper Tail Test
16
17  pval <- pt(t_value,df = df,lower.tail = F)   # Book
        answer is   0.354
```

73

```
18
19  if(pval > alpha)
20  {
21    cat("Since p-value ",pval ,"is greater than 0.05,
         therefore we will accept null hypothesis")
22  } else {
23    cat("Since p-value ",pval ,"is less than 0.05,
         therefore we will reject null hypothesis and
         accept
24        alternative hypothesis.")
25  }
```

R code Exa 9.2b Population Mean Sigma Unknown Two Tailed Test

```
1                                        # Page no. :
                                          372 - 373

2
3  # Population Mean Sigma Unknown Two Tailed Test
4
5  mu0 <- 40
6  alpha <- 0.05
7  xbar <- 37.4
8  s <- 11.79
9  n <- 25
10
11 t_value <- (xbar - mu0) / (s / sqrt(n))
12
13 df <- n - 1   # Degree of Freedom
14
15 # Two Tail Test
16 # P - value Approach
17
18 pval <- 2 *(1 - pt(t_value,df = df,lower.tail = F))
        # Book answer is   0.2822
19
```

74

```
20  if(pval > alpha)
21  {
22      cat("Since p-value ",pval ,"is greater than 0.05,
            therefore we will accept null hypothesis")
23  } else {
24      cat("Since p-value ",pval ,"is less than 0.05,
            therefore we will reject null hypothesis and
            accept
25          alternative hypothesis.")
26  }
27
28  # Critical Value Approach
29
30  t_half_alpha <- qt(1 - (alpha/2),df)
31  critical_value_1 <- -t_half_alpha
32  critical_value_2 <- t_half_alpha
33
34  if(t_value >= critical_value_2 || t_value <=
        critical_value_1)
35  {
36      cat("Since t-value", t_value ,"does not lie in the
            range", critical_value_1 ,"and", critical_
        value_2
37          ,"therefore we reject the null hypothesis and
                accept the alternative hypothesis.")
38  } else{
39      cat("Since t-value", t_value ,"lies in the range",
            critical_value_1 ,"and", critical_value_2
40          ,"therefore we accept the null hypothesis.")
41  }
```

R code Exa 9.3a Population Proportion

```
1                                              # Page no. :
                                                   377 - 378
```

```r
2
3  # Population Proportion
4
5  p0 <- 0.20
6  alpha <- 0.05
7  n <- 400
8  x <- 100
9
10 p_bar <- x / n
11
12 z_value <- (p_bar - p0) / sqrt((p0*(1 - p0)) / n)
13
14 # Upper Tail Test
15
16 # P-value Approach
17
18 pval <- pnorm(z_value,lower.tail = F)
19 if(pval > alpha)
20 {
21   cat("Since p-value ",pval ,"is greater than 0.05,
         therefore we will accept null hypothesis")
22 } else {
23   cat("Since p-value ",pval ,"is less than 0.05,
         therefore we will reject null hypothesis and
         accept
24       alternative hypothesis .")
25 }
26
27 # Critical Value Approach
28
29 z_alpha <- qnorm(1 - alpha)
30 critical_value <- z_alpha
31
32 if(z_value >= critical_value)
33 {
34   cat("Since z-value", z_value ,"is greater then or
         equal to", critical_value , "therefore we
         reject
```

```
35          the null hypothesis and accept the alternative
              hypothesis.")
36 } else{
37   cat("Since z-value", z_value ,"is less than",
        critical_value ,"therefore we accept the null
        hypothesis.")
38 }
```

R code Exa 9.4a Calculating the Probability of Type Second Errors

```
1                                          # Page no. :
                                             382 - 383
2
3 # Calculating the Probability of Type Second Errors
4
5 mu0 <- 120
6 alpha <- 0.05
7 z_value <- 1.645
8 n <- 36
9 sigma <- 12
10
11 xbar <- mu0 - z_value * (sigma / sqrt(n))
12
13 mu <- 112
14 z <- (xbar - mu) / (sigma / sqrt(n))
15
16 # Upper Tail Test
17
18 beta_value <- pnorm(z, lower.tail = F)
19
20 cat("The type 2nd error is",beta_value)
21
22                                          # Page no. : 384
23
24 xbar <- 116.71
```

```
25  mu <- 115
26  sigma <- 12
27  n <- 36
28
29  z <- (xbar - mu) / (sigma / sqrt(n))
30
31  # Upper Tail Test
32
33  beta_value <- pnorm(z, lower.tail = F)
34
35  cat("The type 2nd error is",beta_value)
```

R code Exa 9.5a Determining the Sample Size

```
 1                                    # Page no. :
                                          389
 2
 3  # Determining the Sample Size
 4
 5  alpha <- 0.05
 6  beta <- 0.10
 7  z_alpha <- 1.645
 8  z_beta <- 1.28
 9  mu0 <- 120
10  mua <- 115
11  sigma <- 12
12
13  n <- ((z_alpha + z_beta)**2) * (sigma)**2 / (mu0 -
        mua)**2    # Sample Size
14  n <- ceiling(n)
15  cat("Sampling Size is",n)
```

Chapter 10

Inference About Means and Proportions With Two Populations

R code Exa 10.1a Inference about the Difference between the two Population Means Sigma 1 and Sigma 2 known

```
1                                              # Page  no .  :
                                                      410
2
3  # Inference  about  the  Difference  between  the  two
       Population  Means  Sigma  1  and  Sigma  2  known
4
5  sigma1 <- 9
6  sigma2 <- 10
7  sample_size1 <- 36
8  sample_size2 <- 49
9  sample_mean1 <- 40
10 sample_mean2 <- 35
11
12 point_estimate <- sample_mean1 - sample_mean2
13
14 z_value <- qnorm(0.975)    # alpha/2 = 0.05/2 = 0.025
```

```
       = 1- 0.025 = 0.975
15
16  standard_error <- sqrt(((((sigma1)^2)/(sample_size1))
        + (((sigma2)^2)/(sample_size2)))
17
18  IE1 <- point_estimate + z_value*standard_error
19  IE2 <- point_estimate - z_value*standard_error
20
21  cat("The interval estimation for the given
        information at 95% confidence level is ",IE2 ,"to
        "
22      , IE1)
```

R code Exa 10.1b Hypothesis Tests About Difference between two Means

```
1                                         # Page no. : 410
                                             - 412
2
3   # Hypothesis Tests About Difference between two
       Means
4
5   sigma1 <- 10
6   sigma2 <- 10
7   alpha <- 0.05
8   n1 <- 30
9   n2 <- 40
10  xbar1 <- 82
11  xbar2 <- 78
12  D0 <- 0
13
14  z_value <- ((xbar1 - xbar2) - D0) / sqrt(((sigma1)**
       2/n1) + ((sigma2)**2/n2))
15
16  # P-value Approach
17
```

80

```
18  # Two Tail Test
19
20  pval <- 2 * pnorm(z_value, lower.tail = F)
21
22  if(pval <= alpha)
23  {
24    cat("Since P-Value",pval,"is less than or equal to
          0.05 therefore we can reject Null Hypothesis")
25  } else {
26    cat("Since P-Value",pval,"is more than 0.05
        therefore we cannot reject Null Hypothesis")
27  }
28
29  # Critical Value Approach
30
31  z_half_alpha <- qnorm(0.975)    # alpha/2 = 0.05/2 =
        0.025 = 1- 0.025 = 0.975
32  critical_value_1 <- -z_half_alpha
33  critical_value_2 <- z_half_alpha
34
35  if(z_value >= critical_value_2 || z_value <=
        critical_value_1)
36  {
37    cat("Since Z-value",z_value,"does not lie in the
          range",critical_value_1,"to",critical_value_2,
38        "therefore we can reject Null Hypothesis")
39  } else {
40    cat("Since Z-value",z_value,"lie in the range",
          critical_value_1,"to",critical_value_2,
41        "therefore we cannot reject Null Hypothesis")
42  }
```

R code Exa 10.2a Inference about the Difference between the two Population Means Sigma 1 and Sigma 2 Unknown

```
1                                                       # Page no. :
                                                         415 − 417
2
3  # Inference about the Difference between the two
       Population Means Sigma 1 and Sigma 2 Unknown
4
5  s1 <- 150
6  s2 <- 125
7  n1 <- 28
8  n2 <- 22
9  xbar1 <- 1025
10 xbar2 <- 910
11
12 point_estimate <- xbar1 - xbar2
13
14 numerator <- (((((s1)**2 /n1) + ((s2)**2 /n2))**2)
15 denomenator <- ((1 /(n1 -1)) * (((s1)**2 / n1)**2))
       + ((1 /(n2 -1)) * (((s2)**2 / n2)**2))
16
17 df <- numerator / denomenator   # Degree of Freedom
18
19 t_value <- qt(0.975,df)    # alpha/2 = 0.05/2 = 0.025
       = 1− 0.025 = 0.975
20
21 standard_error <- sqrt((((s1)^2)/(n1)) + (((s2)^2)/(
    n2)))
22
23 IE1 <- point_estimate + t_value*standard_error
24 IE2 <- point_estimate - t_value*standard_error
25
26 cat("The interval estimation for the given
       information at 95% confidence level is ",IE2 ,
27     "to", IE1)
```

R code Exa 10.2b Hypothesis Tests About Difference between two Means

```
1                                              # Page no. :
                                                  418 −
                                                  419
2
3  # Hypothesis Tests About Difference between two
       Means
4
5  x <- c(300, 280, 344, 385, 372, 360, 288, 321, 376,
       290, 301, 283)
6  y <- c(274, 220, 308, 336, 198, 300, 315, 258, 318,
       310, 332, 263)
7
8  DF <- data.frame(x,y)
9
10 test <- t.test(DF$x, DF$y, paired = F, alternative =
        "greater")
11 test
12
13 # Upper Tail Test
14
15 if(test$p.value <= 0.05)    # 95% Confidence Level
16 {
17   cat("Since P−Value",test$p.value,"is less than or
        equal to 0.05 therefore we can reject
18       Null Hypothesis")
19 } else {
20   cat("Since P−Value",test$p.value,"is more than
        0.05 therefore we cannot reject Null Hypothesis
        ")
21 }
```

R code Exa 10.3a Inference About the Difference Between Two Population Means Matched samples

83

```
1                                          # Page no. :
                                             424 − 425
2
3  # Inference About the Difference Between Two
       Population Means Matched samples
4
5  workers <- c(1,2,3,4,5,6)
6  method_1 <- c(6.0,5.0,7.0,6.2,6.0,6.4)
7  method_2 <- c(5.4,5.2,6.5,5.9,6.0,5.8)
8  diff <- method_1 - method_2
9
10 dataFrame <- data.frame(workers, method_1, method_2,
       diff)
11
12 test <- t.test(dataFrame$method_1, dataFrame$method_
       2, paired = T)
13 test
14
15 # Two Tail Test
16
17 if(test$p.value <= 0.05)
18 {
19   cat("Since P−Value",test$p.value,"is less than or
         equal to 0.05 therefore we can
20       reject Null Hypothesis")
21 } else {
22   cat("Since P−Value",test$p.value,"is more than
         0.05 therefore we cannot reject Null Hypothesis
         ")
23 }
24
25 # Interval Estimate
26
27 IE1 <- test$conf.int[1]
28 IE2 <- test$conf.int[2]
29
30 cat("The interval estimation for the given
       information at 95% confidence level is ",IE1 ,"to
```

84

```
            ",
31      IE2)
```

R code Exa 10.4a Inference About the Difference Between Two Population Proportions

```
1                                            # Page no. :
                                                431
2
3  # Inference About the Difference Between Two
       Population Proportions
4
5  n1 <- 250
6  n2 <- 300
7  x1 <- 35
8  x2 <- 27
9  alpha <- 0.1
10
11 pbar1 <- x1 / n1
12 pbar2 <- x2 / n2
13
14 qbar1 <- 1 - pbar1
15 qbar2 <- 1 - pbar2
16
17 # Interval Estimation
18
19 diff_prop <- pbar1 - pbar2
20
21 z_half_alpha <- qnorm(0.95)   # alpha / 2 = 0.1 /
       2 = 0.05 = 0.95 (1 - 0.05)
22
23 margin_of_error <- z_half_alpha * sqrt((pbar1 *
       qbar1)/n1 + (pbar2 * qbar2)/n2 )
24
25 IE1 <- diff_prop + margin_of_error
```

```
26  IE2 <- diff_prop - margin_of_error
27
28  cat("The interval estimation for the given
        information at 90% confidence level is ",IE2 ,"to
        ", IE1)
```

R code Exa 10.4b Hypothesis Tests About Difference between two Proportions

```
1                                           # Page no. :
                                            432 − 433
2
3   # Hypothesis Tests About Difference between two
        Proportions
4
5   pbar1 <- 0.14
6   pbar2 <- 0.09
7   n1 <- 250
8   n2 <- 300
9   alpha <- 0.10    # Significance Level
10
11  pbar <- ((n1 * pbar1) + (n2 * pbar2)) / (n1 + n2)
        # Pooled Estrimator
12
13  z_value <- (pbar1 - pbar2) / sqrt((pbar*(1 - pbar))*
        ((1/n1)+(1/n2)))
14
15  # Two Tail Test
16
17  pval <- 2 * pnorm(z_value, lower.tail = F)
18
19  if(pval <= alpha)
20  {
21    cat("Since P−Value",pval,"is less than or equal to
            0.10 therefore we can reject Null Hypothesis")
```

86

```
22  } else {
23      cat(" Since P−Value",pval ," is  more  than  0.10
              therefore  we  cannot  reject  Null  Hypothesis")
24  }
```

Chapter 11

Inferences About Population Variances

R code Exa 11.1a Inferences About Population Variance

```
1                                              # Page no. :
                                                453
2
3  # Inferences About the Population Variance
4
5  n <- 20
6  variance <- 0.0025
7  df <- 19    # Degrees of Freedom
8
9  chisq_1 <- qchisq(0.975,df)
10 chisq_2 <- qchisq(0.025,df)
11
12 # Interval Estimation
13
14 IE1 <- sqrt((df * variance) / chisq_1)
15 IE2 <- sqrt((df * variance) / chisq_2)
16
17 cat("Interval Estimation at 95% confidence interval
       for population standard deviation is",IE1,
```

```
18        "to",IE2)
```

R code Exa 11.1b Hypothesis Testing

```
1                                                    # Page no. :
                                                         455 − 456
2
3  # Hypothesis  Testing
4
5  n <- 24
6  alpha <- 0.05    # Significance  Level
7  sigma0_sq <- 4
8  variance <- 4.9
9
10 chisq_value <- ((n - 1) * variance) / sigma0_sq
11
12 df <- n - 1    # Degree  of  Freedom
13
14 # Upper  Tail  Test
15
16 # P−value  Approach
17
18 pval <- pchisq(chisq_value, df, lower.tail = F)
19
20 if(pval <= alpha)
21 {
22   cat("Since  pval",pval,"is  less  than  or  equal  to
        0.05  therefore  we  can  reject  Null  Hypothesis")
23 } else{
24   cat("Since  pval",pval,"is  greater  than  0.05
        therefore  we  cannot  reject  Null  Hypothesis")
25 }
26
27 # Critical  Value  Approach
28
```

89

```r
29  chisqvalue <- qchisq(0.95,df)    # Chi-square value
       for 0.95 (1 - 0.05)
30
31  if(chisq_value >= chisqvalue)
32  {
33    cat("Since Chi-square value",chisq_value,"is
         greater than or equal to Chi-square value",
         chisqvalue,
34       "therefore we will reject Null Hypothesis")
35  } else {
36    cat("Since Chi-square value",chisq_value,"is less
         than Chi-square value",chisqvalue,
37       "therefore we cannot reject Null Hypothesis")
38  }
39
40
41                                        # Page no. :
                                            456 -
                                            457
42
43  n <- 30
44  alpha <- 0.05    # Significance Level
45  sigma0_sq <- 100
46  variance <- 162
47
48  chisq_value <- ((n - 1) * variance) / sigma0_sq
49
50  df <- n - 1    # Degree of Freedom
51
52  # Two Tail Test
53
54  # P-value Approach
55
56  pval <- 2 * pchisq(chisq_value, df, lower.tail = F)
57
58  if(pval <= alpha)
59  {
60    cat("Since pval",pval,"is less than or equal to
```

```
              0.05  therefore  we  can  reject  Null  Hypothesis")
61  } else{
62    cat(" Since  pval", pval ," is  greater  than  0.05
          therefore  we  cannot  reject  Null  Hypothesis")
63  }
```

R code Exa 11.2a Inferences About Two Population Variances

```
 1                                        # Page no. :
                                            462 - 464
 2
 3  # Inferences  About  Two  Population  Variances
 4
 5  alpha <- 0.10    # Significance  Level
 6  n1 <- 26
 7  n2 <- 16
 8  sv1 <- 48
 9  sv2 <- 20
10
11  f_value <- (sv1) / (sv2)
12
13  df1 <- n1 - 1    # Degrees  of  Freedom  1
14  df2 <- n2 - 1    # Degrees  of  Freedom  2
15
16  # Two  Tail  Test
17
18  # P-value  Approach
19
20  pval <- 2 * pf(f_value ,df1 ,df2 ,lower.tail = F)
21
22  if(pval <= alpha)
23  {
24    cat(" Since  pval", pval ," is  less  than  or  equal  to
          0.10  therefore  we  can  reject  Null  Hypothesis")
25  } else{
```

91

```
26    cat(" Since pval", pval ,"is greater than 0.10
         therefore we cannot reject Null Hypothesis")
27 }
28
29 # Critical Value Approach
30
31 half_alpha <- alpha / 2
32 fval <- qf(0.95, df1, df2)    # half_alpha = 0.05  =  1
        - 0.05 = 0.95
33
34 if(f_value >= fval)
35 {
36    cat(" Since F value", f_value ,"is greater than or
         equal to F value", fval ,
37        " therefore we will reject Null Hypothesis")
38 } else {
39    cat(" Since F value", f_value ,"is less than F value"
        , fval ,
40        " therefore we cannot reject Null Hypothesis")
41 }
42
43                                          # Page no . :
                                             464
44
45 alpha <- 0.05    # Significance Level
46 n1 <- 41
47 n2 <- 31
48 sv1 <- 120
49 sv2 <- 80
50
51 f_value <- (sv1) / (sv2)
52
53 df1 <- n1 - 1    # Degrees of Freedom 1
54 df2 <- n2 - 1    # Degrees of Freedom 2
55
56 # Upper Tail Test
57
58 # P-value Approach
```

92

```
59
60  pval <- pf(f_value,df1,df2,lower.tail = F)
61
62  if(pval <= alpha)
63  {
64      cat("Since pval",pval,"is less than or equal to
              0.10 therefore we can reject Null Hypothesis")
65  } else{
66      cat("Since pval",pval,"is greater than 0.10
              therefore we cannot reject Null Hypothesis")
67  }
68
69  # Critical Value Approach
70
71  fval <- qf(0.90,df1,df2)    # alpha = 0.05  =  1 - 2
          * 0.05 = 0.90
72
73  if(f_value >= fval)
74  {
75      cat("Since F value",f_value,"is greater than or
              equal to F value",fval,
76          "therefore we will reject Null Hypothesis")
77  } else {
78      cat("Since F value",f_value,"is less than F value"
              ,fval,
79          "therefore we cannot reject Null Hypothesis")
80  }
```

Chapter 12

Tests of Goodness of Fit and Independence

R code Exa 12.1a Goodness of Fit Test A Multinomial Population

```
1                                                # Page no. :
                                                  474 − 476
2
3  # Goodness of Fit Test A Multinomial Population
4
5  category <- c(" Company A" ," Company B" , " Company C" )
6  prop <- c(0.30, 0.50, 0.20)
7  freq1 <- c(48, 98, 54)
8  freq2 <- c(60, 100, 40)
9  diff <- freq1 - freq2
10 sq_diff <- (diff)**2
11 answer <- sq_diff / freq2
12 DF <- data.frame(category, prop, freq1, freq2, diff,
       sq_diff, answer)
13
14 total_observe_freq <- sum(DF$freq1)
15 chisq_value <- sum(DF$answer)
16
17 alpha <- 0.05
```

```
18  df <- nrow(DF) - 1    # Degrees of Freedom
19
20  # Upper Tail Test
21
22  # P-value Approach
23
24  pval <- pchisq(chisq_value,df,lower.tail = F)
25
26  if(pval <= alpha)
27  {
28    cat("Since pval",pval,"is less than or equal to
          0.05 therefore we can reject Null Hypothesis")
29  } else{
30    cat("Since pval",pval,"is greater than 0.05
          therefore we cannot reject Null Hypothesis")
31  }
32
33  # Critical Value Approach
34
35  chisqValue <- qchisq(0.95,df)    # 1 - alpha = 1 -
          0.05 = 0.95
36
37  if(chisq_value >= chisqValue)
38  {
39    cat("Since Chi-square value",chisq_value,"is
          greater than or equal to Chi-square value",
          chisqValue,
40       "therefore we will reject Null Hypothesis")
41  } else {
42    cat("Since Chi-square value",chisq_value,"is less
          than Chi-square value",chisqValue,
43       "therefore we cannot reject Null Hypothesis")
44  }
```

R code Exa 12.2a Tests of Independence

```
1                                                    # Page no . :
                                                        480 − 482
2
3  # Tests of Independence
4
5  gender <- c("Male", "Female")
6  light <- c(20,30)
7  regular <- c(40,30)
8  dark <- c(20,10)
9  total <- c(80,70)
10
11 DF <- data.frame(gender, light, regular, dark, total
      )
12
13 test <- chisq.test(DF[,2:4])
14 test
15
16 # Upper Tail Test
17
18 if(test$p.value <= 0.05)
19 {
20    cat("Since pval",test$p.value,"is less than or
         equal to 0.05 therefore we can reject
21          Null Hypothesis")
22 } else{
23    cat("Since pval",test$p.value,"is greater than
         0.05 therefore we cannot reject Null
24          Hypothesis")
25 }
```

R code Exa 12.3a Goodness of Fit Test Poisson Distribution

```
1                                                    # Page no . :
                                                        488 − 490
2
```

```r
3  # Goodness of Fit Test Poisson Distribution
4
5  x <- c(0,1,2,3,4,5,6,7,8,9)
6  observed_freq <- c(2,8,10,12,18,22,22,16,12,6)
7  y <- x * observed_freq
8  e <- 2.72
9
10 mu <- sum(y) / sum(observed_freq)
11 fun_of_x <- round(((mu)**x) * (e)**(-mu) / factorial
      (x), 4)    # Function of x
12
13 expected_freq <- round(sum(observed_freq)*fun_of_x,
      2)
14
15 diff <- observed_freq - expected_freq
16
17 sq_diff <- round((diff)**2,2)
18
19 answer <- round(sq_diff / expected_freq,2)
20
21 DF <- data.frame(x,observed_freq,expected_freq,diff,
      sq_diff,answer)
22
23 total_observe_freq <- sum(DF$observed_freq)
24 total_expexted_freq <- sum(DF$expected_freq)
25
26 chisq_value <- sum(DF$answer)
27
28 df <- nrow(DF) - 2   # Degrees of Freedom (k - p - 1
       where k = 10, p = 1)
29
30 alpha <- 0.05
31
32 # Upper Tail Test
33
34 # P-value Approach
35
36 pval <- pchisq(chisq_value,df,lower.tail = F)
```

```
37
38  if ( pval <= alpha )
39  {
40      cat ( " Since  pval " , pval , " is  less  than  or  equal  to
            0.05  therefore  we  can  reject  Null  Hypothesis " )
41  } else {
42      cat ( " Since  pval " , pval , " is  greater  than  0.05
            therefore  we  cannot  reject  Null  Hypothesis " )
43  }
44
45  # In  Book  it  is  taken  9  categories  but  we  consider
        10  so  p−value  may  vary  but  answeris  correct
```

R code Exa 12.3b Goodness of Fit Test Normal Distribution

```
1                                              # Page no. :
                                                 491 − 494
2
3  # Goodness  of  Fit  Test  Normal  Distribution
4
5  data <- c
        (71 ,66 ,61 ,65 ,54 ,93 ,60 ,86 ,70 ,70 ,73 ,73 ,55 ,63 ,56 ,62 ,76 ,54 ,82 ,79 ,76 ,6

6              61 ,61 ,64 ,65 ,62 ,90 ,69 ,76 ,79 ,77 ,54 ,64 ,74 ,65 ,65 ,61 ,56 ,63 ,80 ,5

7
8  xbar <- mean ( data )
9  s <- sd ( data )
10
11  cat ( " Value  of  mean  is " , xbar )
12  cat ( " Value  of  standard  variance  is " , s )
13
14  percentage <- c (0.1 ,0.2 ,0.3 ,0.4 ,0.5 ,0.6 ,0.7 ,0.8 ,0.9)
15  z <- c ()
16  test_score <- c ()
```

98

```
17
18  for (i in 1:length(percentage)) {
19   z[i] <- round(qnorm(percentage[i]),2)
20   test_score[i] <- round(xbar + (z[i] * round(s,2)),
        2)
21  }
22
23  DF <- data.frame(percentage, z, test_score)
24
25  interval <- c("Less than 55.10", "55.10 to 59.68", "
        59.68 to 63.01", "63.01 to 65.82",
26                "65.82 to 68.42", "68.42 to 71.02", "
                    71.02 to 73.83", "73.83 to 77.16",
27                "77.16 to 81.74", "81.74 and over")
28  observed_freq <- c(5,5,9,6,2,5,2,5,5,6)
29  expected_freq <- c(5,5,5,5,5,5,5,5,5,5)
30
31  diff <- observed_freq - expected_freq
32  sq_diff <- (diff)**2
33
34  answer <- (sq_diff) / expected_freq
35
36  dataset <- data.frame(interval, observed_freq,
        expected_freq, diff, sq_diff, answer)
37
38  chisq <- sum(dataset$answer)
39
40  alpha <- 0.10   # Significance Level
41
42  df <- nrow(dataset) - 3   # Degrees of Freedom (k -
        p - 1 where k = 10, p = 2)
43
44  # Upper Tail Test
45
46  # P-value Approach
47
48  pval <- pchisq(chisq,df,lower.tail = F)
49
```

```r
50  if(pval <= alpha)
51  {
52    cat("Since pval",pval,"is less than or equal to
         0.05 therefore we can reject Null Hypothesis")
53  } else{
54    cat("Since pval",pval,"is greater than 0.05
         therefore we cannot reject Null Hypothesis")
55  }
```

Chapter 13

Experimental Design and Analysis of Variance

R code Exa 13.1a Analysis of Variance and the Completely Randomized Design

```
1                                              # Page no. :
                                                   510 - 512
2
3 # Analysis of Variance and the Completely Randomized
      Design
4
5 method_A <- c(58,64,55,66,67)
6 method_B <- c(58,69,71,64,68)
7 method_C <- c(48,57,59,47,49)
8
9 DF <- data.frame(method_A,method_B,method_C)
10
11 mean1 <- sum(DF$method_A) / nrow(DF)    # Sample Mean
      1
12 mean2 <- sum(DF$method_B) / nrow(DF)    # Sample Mean
      2
13 mean3 <- sum(DF$method_C) / nrow(DF)    # Sample Mean
      3
```

```r
14
15  variance1 <- sum((DF$method_A - mean1)**2) / (nrow(
       DF)-1)    # Sample Variance 1
16  variance2 <- sum((DF$method_B - mean2)**2) / (nrow(
       DF)-1)    # Sample Variance 2
17  variance3 <- sum((DF$method_C - mean3)**2) / (nrow(
       DF)-1)    # Sample Variance 3
18
19  sd1 <- sqrt(variance1)    # Sample Standard Variance
       1
20  sd2 <- sqrt(variance2)    # Sample Standard Variance
       2
21  sd3 <- sqrt(variance3)    # Sample Standard Variance
       3
22
23  sample_mean <- (mean1 + mean2 + mean3) / 3    #
       Overall Sample Mean
24
25  variance <- ((mean1 - sample_mean)**2 + (mean2 -
       sample_mean)**2 + (mean3 - sample_mean)**2)  /
       (3 - 1)
26  # Sample Varince for Overall Sample Mean (3 --->
       Methods)
27
28  sigma_sq <- nrow(DF) * variance    # Between-
       treatment Estimate of Sigma Square
29
30  estimate_sigma_sq <- (variance1 + variance2 +
       variance3) / 3    # Within-treatment Estimate of
       Sigma Square
31
32  ratio <- sigma_sq / estimate_sigma_sq
33
34  cat("Ratio of Between-treatment Estimate of Sigma
       Square by Within-treatment Estimate
35       of Sigma Square is",ratio)
```

R code Exa 13.2a Analysis of Variance and the Completely Randomized Design

```
1                                          # Page no. : 518
2
3  # Analysis of Variance and the Completely Randomized
       Design
4
5  method_A <- c(58,64,55,66,67)
6  method_B <- c(58,69,71,64,68)
7  method_C <- c(48,57,59,47,49)
8
9  DF <- data.frame(method_A,method_B,method_C)
10
11 k <- ncol(DF)    # Number of Treatments
12 n <- nrow(DF)    # Number of Observations for each
       Treatment
13 N <- n * k    # Total Observations
14
15 df_numerator <- k - 1    # Degrees of Freedom for
       Numerator
16 df_denomenator <- N - k    # Degrees of Freedom for
       Denomenator
17
18 alpha <- 0.05
19
20 x <- c(t(as.matrix(DF)))
21 f <- c("method_A", "method_B", "method_C")
22 tm <- gl(k, 1, n*k, factor(f))
23 result <- anova(lm(x ~ tm))    # Similar to aov(x ~
       tm)
24
25 result
26
```

103

```r
27  # Upper Tail Test
28
29  # Critical Value Approach
30
31  fval <- qf(0.95,df_numerator, df_denomenator)   # 1
        - alpha = 1 - 0.05 = 0.95
32  fval <- round(fval, 2)
33
34  fvalue <- result$'F value'[1]
35
36  if(fvalue >= fval)
37  {
38    cat("Since F value",fvalue,"is greater than or
          equal to F value",fval,
39       "therefore we will reject Null Hypothesis")
40  } else {
41    cat("Since fvalue value",fvalue,"is less than
          fvalue value",fval,
42       "therefore we cannot reject Null Hypothesis")
43  }
44
45  # P-value Approach
46
47  pval <- pf(fvalue,df_numerator,df_denomenator,lower.
        tail = F)
48
49  if(pval <= alpha)
50  {
51    cat("Since pval",pval,"is less than or equal to
          0.05 therefore we can reject Null Hypothesis")
52  } else{
53    cat("Since pval",pval,"is greater than 0.05
          therefore we cannot reject Null Hypothesis")
54  }
```

R code Exa 13.3a Multiple Comparison Procedures Fishers LSD

```
1                                           # Page no . : 525
                                              − 526
2
3  # Multiple Comparison Procedures : Fisher 's LSD
4
5  method_A <- c (58 ,64 ,55 ,66 ,67)
6  method_B <- c (58 ,69 ,71 ,64 ,68)
7  method_C <- c (48 ,57 ,59 ,47 ,49)
8
9  DF <- data.frame ( method_A , method_B , method_C )
10
11 k <- ncol ( DF )    # Number of Treatments
12 n <- nrow ( DF )    # Number of Observations for each
      Treatment
13 N <- n * k    # Total Observations
14
15 x <- c (t( as.matrix ( DF )))
16 f <- c (" method_A " , " method_B " , " method_C ")
17 tm <- gl (k , 1 , n*k , factor (f))
18 result <- anova ( lm (x ~ tm ))    # Similar to aov (x ~
      tm )
19
20 # Approach − 1
21
22 # Fisher 's LSD Procedure for Method A and Method B
23
24 t_value1 <- ( mean ( DF $method_A ) - mean ( DF $method_B ))
      / sqrt ( result $'Mean Sq '[2] * ((1 / n)
25
```

105

```r
26
27 t_value1 <- round(t_value1, 2)
28
29 df <- N - k    # Degrees of Freedom
30
31 # Two Tail Test
32
33 # P-value Approach
34
35 alpha <- 0.05
36
37 pval <- 2 * pt(t_value1, df, lower.tail = T)
38
39 if(pval <= alpha)
40 {
41    cat("Since pval",pval,"is less than or equal to
        0.05 therefore we can reject Null Hypothesis")
42 } else{
43    cat("Since pval",pval,"is greater than 0.05
        therefore we cannot reject Null Hypothesis")
44 }
45
46 tval <- qt(0.975,df)     # alpha/2  = 0.05 / 2  =
      0.025 = (1 - 0.025) = 0.975
47 tval <- round(tval, 3)
48
49 LSD <- tval * sqrt(result$'Mean Sq'[2] * ((1 / n) +
      (1 / n)))
50 LSD <- round(LSD, 2)
51
52 # Approach - 2
53
54 # Fisher's LSD Procedure for Method A and Method C
55
56 diff_A_C <- mean(DF$method_A)- mean(DF$method_C)
```

```
57
58  if ( diff _A_C > LSD )
59  {
60    cat ( " Since  the  value  of  difference " , diff _A_C , " is
         greater  than  LSD " , LSD , " therefore  we  will  reject
61       Null  Hypothesis " )
62  } else {
63    cat ( " Since  the  value  of  difference " , diff _A_C , " is
         less  than  LSD " , LSD , " therefore  we  cannot  reject
64       Null  Hypothesis " )
65  }
66
67  # Fisher 's LSD Procedure  for  Method B and Method C
68
69  diff _B_C <- mean ( DF $method _B ) - mean ( DF $method _C )
70
71  if ( diff _B_C > LSD )
72  {
73    cat ( " Since  the  value  of  difference " , diff _B_C , " is
         greater  than  LSD " , LSD , " therefore  we  will  reject
74       Null  Hypothesis " )
75  } else {
76    cat ( " Since  the  value  of  difference " , diff _B_C , " is
         less  than  LSD " , LSD , " therefore  we  cannot  reject
77       Null  Hypothesis " )
78  }
```

R code Exa 13.4a Randomized Block Design

```
1                                              # Page  no . :
                                                    534
2
3  # Randomized  Block  Design
4
5  blocks <- c ( " Controller  1 " , " Controller  2 " , "
```

107

```
          Controller 3", "Controller 4", "Controller 5", "
          Controller 6")
 6  system_A <- c(15, 14, 10, 13, 16, 13)
 7  system_B <- c(15, 14, 11, 12, 13, 13)
 8  system_C <- c(18, 14, 15, 17, 16, 13)
 9
10  DF <- data.frame(blocks, system_A, system_B, system_
       C)
11
12  k <- ncol(DF) - 1   # Number of Treatments (blocks
        is not the treatement)
13  b <- nrow(DF)   # Number of blocks
14  N <- k * b   # Total sample Size
15
16  x <- c(t(as.matrix(DF[,-1])))
17  f <- c("system_A", "system_B", "system_C")
18  tm <- gl(k, 1, N, factor(f))
19  blk <- gl(b, k, N)
20  result <- anova(lm(x ~ tm + blk))   # Similar to aov
       (x ~ tm)
21
22  result
23
24  f_value <- result$'F value'[1]
25
26  # Upper Tail Test
27
28  # P-value Approach
29
30  pval <- result$'Pr(>F)'[1]
31
32  if(pval <= 0.05)
33  {
34    cat("Since pval",pval,"is less than or equal to
         0.05 therefore we can reject Null Hypothesis")
35  } else{
36    cat("Since pval",pval,"is greater than 0.05
         therefore we cannot reject Null Hypothesis")
```

```
37  }
```

R code Exa 13.5a Factorial Design

```
 1                                          # Page no. : 542
 2
 3  # Factorial Experiment
 4
 5  program <- c("Three−hour review", "Three−hour review
       ", "One−day program", "One−day program", "10−week
        course",
 6                "10−week course")
 7  business <- c(500, 580, 460, 540, 560, 600)
 8  engineering <- c(540, 460, 560, 620, 600, 580)
 9  arts_and_science <- c(480, 400, 420, 480, 480, 410)
10
11  DF <- data.frame(program, business, engineering,
       arts_and_science)
12
13  a <- 3    # Number of levels in Factor A (Unique
       Programs)
14  b <- 3    # Number of levels in Factor B (Columns
       Except program)
15  r <- 2    # Number of Replications (Each Program has
       2 Replications)
16  N <- a * b * r   # Total Observations
17
18  x <- c(t(as.matrix(DF[,-1])))
19  f1 <- c("Three−hour review", "One−day program", "10−
       week course")
20  f2 <- c("business", "engineering", "arts and science
       ")
21  tm2 <- gl(a, 1, N, factor(f1))
22  tm1 <- gl(b, r * a, N, factor(f2))
23  result <- anova(lm(x ~ tm1 * tm2))   # Similar to
```

```
      aov(x ~ tm)
24
25  result
26
27  alpha <- 0.05
28
29  # Upper Tail Test
30
31  pval1 <- result$'Pr(>F)'[1]
32  pval2 <- result$'Pr(>F)'[2]
33  pval3 <- result$'Pr(>F)'[3]
34
35  if(pval1 <= alpha)
36  {
37    cat("Since pval",pval1,"is less than or equal to
         0.05 therefore we can reject Null Hypothesis
38         for Undergraduation.")
39  } else{
40    cat("Since pval",pval1,"is greater than 0.05
         therefore we cannot reject Null Hypothesis
41         for Undergraduation.")
42  }
43
44  if(pval2 <= alpha)
45  {
46    cat("Since pval",pval2,"is less than or equal to
         0.05 therefore we can reject Null Hypothesis
47         for programs.")
48  } else{
49    cat("Since pval",pval2,"is greater than 0.05
         therefore we cannot reject Null Hypothesis
50         for programs.")
51  }
52
53  if(pval3 <= alpha)
54  {
55    cat("Since pval",pval3,"is less than or equal to
         0.05 therefore we can reject Null Hypothesis
```

```
56            for interaction.")
57 } else{
58    cat("Since pval",pval3,"is greater than 0.05
          therefore we cannot reject Null Hypothesis
59            for interaction.")
60 }
```

Chapter 14

Simple Linear Regression

R code Exa 14.1a Least Squares Method

```
1                                              # Page no. : 565
                                                    − 566
2
3 # Least Squares Method
4
5 restaurant <- c(1,2,3,4,5,6,7,8,9,10)
6 student_population <- c(2,6,8,8,12,16,20,20,22,26)
7 quartely_sales <- c
      (58,105,88,118,117,137,157,169,149,202)
8
9 DF <- data.frame(restaurant, student_population,
      quartely_sales)
10
11 # Install Library if not installed
12
13 # install.packages("ggplot2")
14
15 # Import Library
16
17 library(ggplot2)
18
```

```
19  ggplot ( DF , aes ( student_population , quartely_sales )) +
        geom_point () +
20    labs ( title = " Scatter  Plot  between  Student
          Population  and  Quartely  Sales " , x = " Student
21          Population  (1000 s ) " ,  y = " Quartely  Sales  ( $
            1000 s ) " )
22
23                                    # Page no . : 567 −
                                            569
24
25  regressor  <- lm ( quartely_sales  ~  student_population ,
        data  = DF )
26  res  <- summary ( regressor )
27
28  res
29
30  b1  <- res $ coefficients [[2]]
31
32  b0  <- res $ coefficients [[1]]
33
34  cat ( " Estimated  Regression  Equation  is  y_cap  = " , b0 , " +
      " , b1 , " x " )
35
36  ggplot ( DF , aes ( student_population ,  quartely_sales )) +
        geom_point () +
37    geom_smooth ( method = 'lm ' ,  se  = F )  + labs ( title  = "
          Scatter  Plot  between  Student  Population
38  and  Quartely  Sales " , x  = " Student  Population  (1000 s ) "
      ,  y = " Quartely  Sales  ( $ 1000 s ) " )
```

R code Exa 14.2a Coefficient of Determination

```
1                                          # Page no . :
                                              576 −
                                              580
```

113

```r
2
3  # Coefficient of Determination
4
5  restaurant <- c(1,2,3,4,5,6,7,8,9,10)
6  student_population <- c(2,6,8,8,12,16,20,20,22,26)
7  quartely_sales <- c
       (58,105,88,118,117,137,157,169,149,202)
8
9  DF <- data.frame(restaurant, student_population,
      quartely_sales)
10
11 regressor <- lm(quartely_sales ~ student_population,
       data = DF)
12 res <- summary(regressor)
13
14 table <- anova(regressor)
15
16 SSE <- table$'Sum Sq'[2]    # Sum of Squares due to
      Error
17
18 cat("Value of SSE is",SSE)
19
20 SSR <- table$'Sum Sq'[1]    # Sum of Squares due to
      Regression
21
22 cat("Value of SSR is",SSR)
23
24 SST <- SSE + SSR    # Total Sum of Squares
25
26 cat("Value of SST is",SST)
27
28 r_sq <- res$r.squared    # Coefficient of
      Determination
29
30 corrcoeff <- sqrt(r_sq)    # Correlation Coefficient
31
32 cat("Value of Coefficient of Determination is",r_sq)
33 cat("Value of correlation Coefficient is",corrcoeff)
```

114

R code Exa 14.3a Test of Significance

```
 1                                    # Page no. :
                                       485 - 489
 2
 3 # Test of Significance
 4
 5 restaurant <- c(1,2,3,4,5,6,7,8,9,10)
 6 student_population <- c(2,6,8,8,12,16,20,20,22,26)
 7 quartely_sales <- c
       (58,105,88,118,117,137,157,169,149,202)
 8
 9 DF <- data.frame(restaurant, student_population,
       quartely_sales)
10
11 regressor <- lm(quartely_sales ~ student_population,
        data = DF)
12 res <- summary(regressor)
13
14 standard_error_MSE <- res$sigma
15
16 cat("Value of square root of MSE is",standard_error_
       MSE)
17
18 b1 <- res$coefficients[2]
19
20 tval <- res$coefficients[6]
21
22
23 # T Test
24
25 # Two Tail Test
26
27 # P-value Approach
```

```r
28
29  pval <- round(res$coefficients[8],3)
30
31  if(pval >= 0.01)
32  {
33    cat("Since pval",pval,"is greater than or equal to
          0.01 therefore we cannot reject the Null
        Hypothesis")
34  } else{
35    cat("Since pval",pval,"is less than 0.01 therefore
          we can reject the Null Hypothesis")
36
37  }
38
39  # F Test
40
41  test <- anova(regressor)
42
43  fval <- test$`F value`
44
45  pval <- round(test$`Pr(>F)`[1],3)    # P value is
      extremely small ie negligible to 0
46
47  if(pval >= 0.01)
48  {
49    cat("Since pval",pval,"is greater than or equal to
          0.01 therefore we cannot reject the Null
        Hypothesis")
50  } else{
51    cat("Since pval",pval,"is less than 0.01 therefore
          we can reject the Null Hypothesis")
52
53  }
54
55  # Confidence Interval
56
57  confidence <- confint(regressor, "student_population
      ", level = 0.99)
```

```
58  IE1 <- confidence[1]
59  IE2 <- confidence[2]
60
61  cat("The 99 %  confidence interval is ",IE1, "to",
       IE2)
```

R code Exa 14.4a Using the Estimated Regression Equation for Estimation and Prediction

```
1                                          # Page no. : 595 -
                                                597
2
3  # Using the Estimated Regression Equation for
       Estimation and Prediction
4
5  restaurant <- c(1,2,3,4,5,6,7,8,9,10)
6  student_population <- c(2,6,8,8,12,16,20,20,22,26)
7  quartely_sales <- c
       (58,105,88,118,117,137,157,169,149,202)
8
9  DF <- data.frame(restaurant, student_population,
       quartely_sales)
10
11 regressor <- lm(quartely_sales ~ student_population,
        data = DF)
12 res <- summary(regressor)
13
14 pred <- predict(regressor, data.frame(student_
       population=10), interval="confidence")
15
16 PE <- pred[1]
17 IE1 <- pred[2]
18 IE2 <- pred[3]
19
20 cat("Point estimate is ", PE)
```

```
21  cat("Confidence Interval is ",IE1, "to", IE2)
22
23  pred2 <- predict(regressor, data.frame(student_
        population=10), interval="predict")
24
25  IE1 <- pred2[2]
26  IE2 <- pred2[3]
27
28  cat("Prediction Confidence Interval is ",IE1, "to",
        IE2)
```

R code Exa 14.5a Residual Analysis Validating Model Assumptions

```
1                                               # Page no. : 605
                                                     - 609
2
3  # Residual Analysis : Validating Model Assumptions
4
5  x <- c(2,6,8,8,12,16,20,20,22,26)    # Student
        Population
6  y <- c(58,105,88,118,117,137,157,169,149,202)    #
        Sales
7
8  estimated_sales <- 60 + (5 * x)    # Regression
        Equation  = 60 + 5 x
9
10 residuals <- y - estimated_sales
11
12 DF <- data.frame(x, y, estimated_sales, residuals)
13
14 # Install Library if not installed
15
16 # install.packages("ggplot2")
17
18 # Import Library
```

118

```
19
20  library(ggplot2)
21
22  ggplot(DF,aes(x, residuals)) + geom_point() + geom_
        hline(yintercept = 0, linetype=2) +
23  labs(title = "Residual Plot", x = "X", y = "Residual
        ")
24
25  ggplot(DF,aes(estimated_sales, residuals)) + geom_
        point() + geom_hline(yintercept = 0,
26             linetype=2) + labs(title = "Plot between
                 Estimated Sales and Residuals", x =
27                                  "Estimated Sales", y =
                                      "Residual")
```

R code Exa 14.5b Standardized Residuals

```
1                                        # Page no. : 610 −
                                             612
2
3  # Standardized Residuals
4
5  i <- c(1,2,3,4,5,6,7,8,9,10)
6  x <- c(2,6,8,8,12,16,20,20,22,26)    # Student
        Population
7  n <- 10
8  s <- 13.829    # Standard error
9  y <- c(58,105,88,118,117,137,157,169,149,202)    #
        Sales
10
11 estimated_sales <- 60 + (5 * x)    # Regression
        Equation  = 60 + 5 x
12
13 x_deviation <- x - mean(x)
14
```

```r
15  x_deviation_sq <- (x_deviation)**2
16
17  z <- round(x_deviation_sq / sum(x_deviation_sq),4)
18
19  h <- round((1 / n) + z, 4)
20
21  s_i <- round(s * sqrt(1 -h), 4)    # Standard
        Deviation for Residual i
22
23  residuals <- y - estimated_sales
24
25  standard_residuals <- round((residuals) / (s_i), 4)
26
27  DF <- data.frame(x, y, estimated_sales, x_deviation,
        x_deviation_sq, z, h, s_i, residuals, standard_
      residuals)
28
29  # Install Library if not installed
30
31  # install.packages("ggplot2")
32
33  # Import Library
34
35  library(ggplot2)
36
37  ggplot(DF,aes(x, standard_residuals)) + geom_point()
        + geom_hline(yintercept = 0, linetype=2) +
38    labs(title = "Scatter Plot between x and
          Standard Residuals", x = "X",
39        y = "Standard Residuals")
40
41  normal_scores <- round(qqnorm(1:10)$x, 2)
42  standard_residuals <- sort(standard_residuals,
      decreasing = F)
43
44  table <- data.frame(normal_scores, standard_
      residuals)
45
```

```
46  ggplot(table,aes(normal_scores, standard_residuals))
        + geom_point() +
47    geom_smooth(method = "lm", se = F) +
48  labs(title = "Scatter Plot between Normal Scores and
        Standard Residuals", x = "Normal Scores",
49        y = "Standard Residuals")
```

R code Exa 14.6a Detecting Outliers

```
1                                            # Page no. : 615
2
3  # Detecting Outliers
4
5  x <- c(1,1,2,3,3,3,4,4,5,6)
6  y <- c(45,55,50,75,40,45,30,35,25,15)
7
8  DF <- data.frame(x,y)
9
10 ggplot(DF,aes(x, y)) + geom_point()+ geom_smooth(
        method = "lm", se = F) +
11 labs(title = "Scatter Plot between  x and Y Showing
        Outlier", x = "X", y = "Y")
```

R code Exa 14.6b Detecting Influential Observations

```
1                                            # Page no. :
                                              617 − 618
2
3  # Detecting Influential Observations
4
5  x <- c(10, 10, 15, 20, 20, 25, 70)
6  y <- c(135, 130, 120, 115, 120, 110, 100)
7
```

```r
 8  DF <- data.frame(x, y)
 9
10  # Install Library if not installed
11
12  # install.packages("ggplot2")
13
14  # Import Library
15
16  library(ggplot2)
17
18  ggplot(DF, aes(x, y)) + geom_point() + labs(title =
        "Scatter Plot between x and y", x = "X",
19                                                    y = "Y")
20
21  point <- x[7]   # From Scatter Plot
22
23  h <- (1 / nrow(DF)) + (((point - mean(DF$x))**2) / (
        sum((DF$x - mean(DF$x))**2)))
24
25  cat("Leverage at point 7 is", h)
```

Chapter 15

Multiple Regression

R code Exa 15.1a Least Squares Method

```
1                                      # Page no. :
                                         646 − 647
2
3 # Least Squares Method
4
5 driving_assignment <- c(1,2,3,4,5,6,7,8,9,10)
6 x <- c(100,50,100,100,50,80,75,65,90,90)    # Miles
      Traveled
7 y <- c(9.3,4.8,8.9,6.5,4.2,6.2,7.4,6.0,7.6,6.1)    #
      Travel Time (hours)
8
9 DF <- data.frame(driving_assignment, x ,y)
10
11 # Install Library if not installed
12
13 # install.packages("ggpolt2")
14
15 # Import Library
16
17 library(ggplot2)
18
```

```r
19  ggplot(DF,aes(x, y)) + geom_point() +labs(title = "
        Scatter Plot between Miles Traveled
20     and Travel Time", x = "Miles Traveled", y = "
        Travel Time in Hours")
21
22                                    # Page no. : 647
                                         - 648
23
24  model <- lm(y ~ x, data = DF)
25  summ <- summary(model)
26  summ
27
28  b0 <- model$coefficients[1]
29  b1 <- model$coefficients[2]
30
31  cat("Linear Regression Equation is y_cap =",b0,"+",
        b1,"x1")
32
33  # F Test
34
35  fval <- summ$fstatistic[1]
36
37  # Upper Tail Test
38
39  # P-value Approach
40
41  alpha <- 0.05
42
43  pval <- summ$coefficients[8]   # P-value
44
45  if(pval >= alpha)
46  {
47    cat("Since pval",pval,"is greater than or equal to
          0.01 therefore we cannot reject the Null
        Hypothesis")
48  } else{
49    cat("Since pval",pval,"is less than 0.05 therefore
          we can reject the Null Hypothesis")
```

```
50
51  }
```

R code Exa 15.2a Two Independent Variables

```
1                                          # Page no. : 648
                                           − 649
2
3  # Two Independent Variables
4
5  driving_assignment <- c (1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10)
6  x1 <- c (100 ,50 ,100 ,100 ,50 ,80 ,75 ,65 ,90 ,90)    # Miles
       Traveled
7  x2 <- c (4 ,3 ,4 ,2 ,2 ,2 ,3 ,4 ,3 ,2)    # Number of
       Deliveries
8  y <- c (9.3 ,4.8 ,8.9 ,6.5 ,4.2 ,6.2 ,7.4 ,6.0 ,7.6 ,6.1)    #
       Travel Time ( hours )
9
10 DF <- data . frame ( driving_assignment , x1 , x2 , y )
11
12 model <- lm ( y ~ x1 + x2 , data = DF )
13 summary ( model )
14
15 b0 <- model $ coefficients [1]
16 b1 <- model $ coefficients [2]
17 b2 <- model $ coefficients [3]
18
19 cat (" Multiple Regression Equation is y_cap =" ,b0 ," +"
       ,b1 ," x1 +" ,b2 ," x2 ")
```

R code Exa 15.3a Multiple Coefficient of Determination

```
1                                           # Page no. :
                                                654 − 655
2
3  # Multiple Coefficient of Determination
4
5  driving_assignment <- c(1,2,3,4,5,6,7,8,9,10)
6  x1 <- c(100,50,100,100,50,80,75,65,90,90)    # Miles
       Traveled
7  x2 <- c(4,3,4,2,2,2,3,4,3,2)    # Number of
       Deliveries
8  y <- c(9.3,4.8,8.9,6.5,4.2,6.2,7.4,6.0,7.6,6.1)    #
       Travel Time (hours)
9
10 DF <- data.frame(driving_assignment, x1, x2, y)
11
12 model <- lm(y ~ x1 + x2, data = DF)
13 summ <- summary(model)
14
15 mean_y <- mean(DF$y)    # Mean of Travel time
16
17 predicted_travel_time <- round(predict(model), 2)
18
19 DF <- cbind(DF, predicted_travel_time)
20
21 SSR <- sum((DF$predicted_travel_time - mean_y)**2)
        # Sum of Squares due to Regression
22
23 cat("Value of SSR is",SSR)
24
25 SSE <- sum((DF$y - DF$predicted_travel_time)**2)    #
       Sum of Squares due to Error
26
27 cat("Value of SSE", SSE)
28
29 SST <- SSR + SSE    # Total Sum of Squares
30
31 cat("Value of SST", SST)
32
```

126

```
33  r_sq <- summ$r.squared  # Multiple Coefficient of
        Determination

34
35  cat("Value of Multiple Coefficient of Determination
        is", r_sq)

36
37  adj_r_sq <- summ$adj.r.squared   # Adjusted Multiple
         Coefficient of Determination

38
39  cat("Value of Adjusted Multiple Coefficient of
        Determination is", adj_r_sq)
```

R code Exa 15.4a Testing of significance

```
1                                      # Page no. :
                                          660 - 662

2
3  # Testing for Significance

4
5  driving_assignment <- c(1,2,3,4,5,6,7,8,9,10)
6  x1 <- c(100,50,100,100,50,80,75,65,90,90)   # Miles
        Traveled
7  x2 <- c(4,3,4,2,2,2,3,4,3,2)    # Number of
        Deliveries
8  y <- c(9.3,4.8,8.9,6.5,4.2,6.2,7.4,6.0,7.6,6.1)    #
        Travel Time (hours)

9
10  DF <- data.frame(driving_assignment, x1, x2, y)

11
12  model <- lm(y ~ x1 + x2, data = DF)
13  summ <- summary(model)

14
15  SSR <- 21.6252     # Sum of Squares due to Regression

16
17  SSE <- 2.2952     # Sum of Squares due to Error
```

```r
18
19  n <- nrow(DF)    # Total Observations
20
21  p <- 2    # Number of Independent Variables
22
23  MSR <- SSR / p    # Mean Square due to Regression
24
25  cat("Value of MSR is",MSR)
26
27  MSE <- SSE / (n - p - 1)    # Mean Square due to
        Error
28
29  # F Test
30
31  fval <- summ$fstatistic[1]
32
33  alpha <- 0.01
34
35  # Upper Tail Test
36
37  # P-value Approach
38
39  pval <- summ$coefficients[11]
40
41  if(pval >= alpha)
42  {
43    cat("Since pval",pval,"is greater than or equal to
          0.01 therefore we cannot reject the Null
        Hypothesis")
44  } else{
45    cat("Since pval",pval,"is less than 0.01 therefore
          we can reject the Null Hypothesis")
46
47  }
48
49  s <- summ$sigma    # Standard Error of the Estimate
50
51  cat("Value of Standard Error of the Estimate is",s)
```

```r
52
53 coeff <- as.data.frame(summ$coefficients)
54
55 b1 <- coeff$Estimate[2]
56 b2 <- coeff$Estimate[3]
57
58 s1 <- coeff$'Std. Error'[2]   # Standard Error of
       the x1 (Miles Traveled)
59 s2 <- coeff$'Std. Error'[3]   # Standard Error of
       the x2 (Number of Deliveries)
60
61 df <- n - p - 1   # Degrees of Freedom
62
63 # T Test
64
65 tval1 <- round(b1 / s1, 3)
66 tval2 <- round(b2 / s2, 3)
67
68 # Upper Tail Test
69
70 # P-value Approach
71
72 pval1 <- round(pt(tval1, df, lower.tail = F),5)
73 pval2 <- round(pt(tval2, df, lower.tail = F),5)
74
75
76 if(pval1 >= alpha && pval2 >= alpha)
77 {
78   cat("Since pval1",pval1,"and pval2",pval2,"is
       greater than or equal to 0.01 therefore we
       cannot reject the
79        Null Hypothesis")
80 } else{
81   cat("Since pval1",pval1,"and pval2",pval2,"is less
         than 0.01 therefore we can reject the Null
       Hypothesis")
82
83 }
```

129

R code Exa 15.5a Categorical Independent Variables

```
1                                              # Page  no . :
                                                  668 − 671
2
3  # Categorical  Independent  Variables
4
5  service <- c(1:10)
6  month <- c(2,6,8,3,2,7,9,8,4,6)
7  repair <- c("electrical","mechanical","electrical","
      mechanical","electrical","electrical",
8                 "mechanical","mechanical","electrical","
                     electrical")
9  time <- c(2.9, 3.0, 4.8, 1.8, 2.9, 4.9, 4.2, 4.8,
      4.4, 4.5)
10
11 DF <- data.frame(service, month, repair, time)
12
13 # With  one  Independent  variable
14
15 regressor <- lm(time ~ month, data = DF)
16 res <- summary(regressor)
17
18 res
19
20 b0 <- res$coefficients[1]
21 b1 <- res$coefficients[2]
22
23 cat("Equation  is  y = ", b0 ,"+",b1,"x1")
24
25 # With  Categorical  Variable
26
27 DF$repair <- factor(DF$repair,
28                       levels = c('electrical', '
```

```
                         mechanical '),
29                       labels = c(1, 0))
30
31  regressor <- lm(time ~ month + repair, data = DF)
32  res <- summary(regressor)
33
34  res
35
36  b0 <- res$coefficients[1]
37  b1 <- res$coefficients[2]
38  b2 <- res$coefficients[3]
39
40  # For Electrical Repair
41
42  bterm <- b0 + b2 * 1
43
44  cat("Equation for electrical is  y = ", bterm ,"+",
       b1,"x1" )
45
46  # For Mechanical Repair
47
48  bterm2 <- b0 + b2 * 0
49
50  cat("Equation for mechanical is  y = ", bterm2 ,"+",
       b1,"x1" )
```

R code Exa 15.6a Residual Analysis

```
1                                         # Page no. :
                                            676 − 679
2
3  # Residual Analysis
4
5  miles <- c(100, 50, 100, 100, 50, 80, 75, 65, 90,
       90)
```

```r
6  deliveries <- c(4, 3, 4, 2, 2, 2, 3, 4, 3, 2)
7  time <- c(9.3, 4.8, 8.9, 6.5, 4.2, 6.2, 7.4, 6.0,
      7.6, 6.1)
8
9  DF <- data.frame(miles, deliveries, time)
10
11 regressor <- lm(time ~ miles + deliveries, data = DF
      )
12 res <- summary(regressor)
13
14 predict <- predict(regressor)
15 residuals <- DF$time - predict
16 std_residuals <- rstandard(regressor)
17
18 DF <- cbind(DF, predict, residuals, std_residuals)
19
20 View(DF)
21
22 # Install Library if not installed
23
24 # install.packages("ggplot2")
25
26 # Import Library
27
28 library(ggplot2)
29
30 ggplot(DF,aes(predict, std_residuals)) + geom_point
      () + geom_hline(yintercept = 0,
31        linetype=2) + labs(title = "Plot between
            predicted values and standardizes residuals
            ", x =
32                                        "Prediction", y
                                          = "
                                          Standardization
                                          Residual")
33
34
35 leverage <- hatvalues(regressor)
```

```
36  cook_dist <- cooks.distance(regressor)
37
38  DF <- cbind(DF, leverage, cook_dist)
39
40  View(DF)
```

R code Exa 15.6b Influential Observations

```
1                                          # Page no. :
                                           679 - 680
2
3  # Influential  Observations
4
5  x <- c(1, 1, 2, 3, 4, 4, 5, 15)
6  y <- c(18, 21, 22, 21, 23, 24, 26, 39)
7
8  DF <- data.frame(x, y)
9
10 regressor <- lm(y ~ x, data = DF)
11 res <- summary(regressor)
12
13 lev <- hatvalues(regressor)
14 DF <- cbind(DF, lev)
15
16 View(DF)
17
18
19 # Install  Library  if  not  installed
20
21 # install.packages("ggplot2")
22
23 # Import  Library
24
25 library(ggplot2)
26
```

```
27 ggplot(DF,aes(x, y)) + geom_point() + geom_smooth(
     method = "lm", se = F) +
28   labs(title = "Plot between x and y", x = "X", y =
       "Y")
29
30
31 cat("Equation is y = ",res$coefficients[1],"+",res$
     coefficients[2],"x1")
32
33
34 # Removing Influential Observation
35
36 x <- c(1, 1, 2, 3, 4, 4, 5)
37 y <- c(18, 21, 22, 21, 23, 24, 26)
38
39 DF <- data.frame(x, y)
40
41 regressor <- lm(y ~ x, data = DF)
42 res <- summary(regressor)
43
44 cat("Equation is y = ",res$coefficients[1],"+",res$
     coefficients[2],"x1")
```

R code Exa 15.7a Logistic Regression

```
1                                              # Page no. :
                                                 684 − 686
2
3 # Logistic Regression
4
5 customer <- c(1,2,3,4,5,6,7,8,9,10)
6 spending <- c
     (2.291,3.215,2.135,3.924,2.528,2.473,2.384,7.076,1.182,3.345)

7 card <- c(1,1,1,0,1,0,0,0,1,0)
```

134

```
 8  coupon <- c(0,0,0,0,0,1,0,0,1,0)
 9
10  DF <- data.frame(customer, spending, card, coupon)
11
12  regressor <- glm(coupon ~ spending + card, data = DF
       )
13  summary(regressor)
14
15  # Book answer is different
```

Chapter 16

Regression Analysis Model Building

R code Exa 16.1a General Linear Model

```
1                                          # Page no. : 714
                                               - 717
2
3 # General Linear Model
4
5 x <- c(41,106,76,10,22,12,85,111,40,51,9,12,6,56,19)
        # Months Employed
6 y <- c
      (275,296,317,376,162,150,367,308,189,235,83,112,67,325,189)
        # Scales Sold
7
8 DF <- data.frame(x,y)
9
10 # Install Library if not installed
11
12 # install.packages("ggplot2")
13
14 # Import Library
15
```

```r
16  library(ggplot2)
17
18  ggplot(DF, aes(x,y)) + geom_point() + labs(title = "
        Scatter Plot between Months
19      Employed and Scales sold", x = "Months Employed",
          y = "Scales Sold")
20
21  model <- lm(y ~ x, data = DF)
22  summary(model)
23
24  b1 <- model$coefficients[1]
25  b2 <- model$coefficients[2]
26
27  cat("Regression Equation is sales =",b1,"+",b2,"
        months")
28  # Book Answer is sales = 111 + 2.38 months   (Data
        is wrongly used in book)
29
30  predicted_scales_sold <- predict(model)
31
32  DF <- cbind(DF, predicted_scales_sold)
33
34  # Standard Residuals
35
36  standard_residuals <- rstandard(model)
37
38  DF <- cbind(DF, standard_residuals)
39
40  ggplot(DF,aes(predicted_scales_sold, standard_
        residuals)) + geom_point() +
41      geom_hline(yintercept = 0, linetype=2) +
42  labs(title = "Scatter Plot between Predicted scales
        sold and Standard Residuals",
43      x = "Predicted scales sold", y = "Standard
            Residuals")
44
45                                      # Page no. :
                                          716 - 717
```

```
46
47  x2 <- x**2
48
49  DF2 <- data.frame(x,x2,y)
50
51  model2 <- lm(y ~ x + x2, data = DF2)
52  summary(model2)
53
54  b1 <- model2$coefficients[1]
55  b2 <- model2$coefficients[2]
56  b3 <- model2$coefficients[3]
57
58  cat("Regression Equation is sales =",b1,"+",b2,"
       months","+",b3,"monthsq")
59  # Book Answer is sales = 45.3 + 6.34 months − 0.0345
        monthsq  (Data is wrongly used in book)
60
61  predicted_scales_sold2 <- predict(model2)
62
63  DF2 <- cbind(DF2, predicted_scales_sold2)
64
65  # Standard Residuals
66
67  standard_residuals2 <- rstandard(model2)
68
69  DF2 <- cbind(DF2, standard_residuals2)
70
71  ggplot(DF2,aes(predicted_scales_sold2, standard_
       residuals2)) + geom_point() +
72    geom_hline(yintercept = 0, linetype=2) +
73    labs(title = "Scatter Plot between Predicted
         scales sold and Standard Residuals",
74        x = "Predicted scales sold", y = "Standard
             Residuals")
```

R code Exa 16.1b Interaction

```
1                                          # Page no . : 718
                                                  − 721
2
3  # Interaction
4
5  price <- c(2.00, 2.50, 3.00, 2.00, 2.50, 3.00, 2.00,
       2.50, 3.00, 2.00, 2.50, 3.00,
6              2.00, 2.50, 3.00, 2.00, 2.50, 3.00, 2.00,
                   2.50, 3.00, 2.00, 2.50, 3.00)
7  advertising <- c(50, 50, 50, 50, 50, 50, 50, 50, 50,
       50, 50, 50, 100, 100, 100, 100, 100,
8                     100, 100, 100, 100, 100, 100, 100)
9  sales <- c(478, 373, 335, 473, 358, 329, 456, 360,
       322, 437, 365, 342, 810, 653, 345, 832, 641,
10              372, 800, 620, 390, 790, 670, 393)
11
12 DF <- data.frame(price, advertising, sales)
13
14 # Install Library if not install
15
16 # install.packages("dplyr")
17
18 # Import Library
19
20 library(dplyr)
21
22 DF %>% group_by(price,advertising) %>% summarize(
       Average=mean(sales))
23
24 model <- lm(sales ~ price + advertising + (price *
       advertising), data = DF)
25 res <- summary(model)
26
27 b0 <- res$coefficients[1]
28 b1 <- res$coefficients[2]
29 b2 <- res$coefficients[3]
```

```
30  b3 <- res$coefficients[4]
31
32  cat("Equation is sales = ", b0, "+", b1, "Price +",
        b2, "AdvExp +", b3, "PriAdv")
```

R code Exa 16.1c Transformations Involving the Dependent Variables

```
1                                              # Page no. : 721
                                                    - 724
2
3   # Transformations Involving the Dependent Variables
4
5   x <- c
        (2289,2113,2180,2448,2026,2702,2657,2106,3226,3213,3607,2888)
            # Weight
6   y <- c
        (28.7,29.2,34.2,27.9,33.3,26.4,23.9,30.5,18.1,19.5,14.3,20.9)
            # Miles Per Gallon
7
8   DF <- data.frame(x,y)
9
10  # Install Library if not install
11
12  # install.packages("ggplot2")
13
14  # Import Library
15
16  library(ggplot2)
17
18  ggplot(DF, aes(x,y)) + geom_point() + labs(title = "
        Scatter Plot between Weight and Miles
19                                                          Per
                                                        Gallon
                                                        ", x =
                                                        "
```

140

Weight
", y =
"
Miles
Per
Gallon
")

```r
20
21  model <- lm(y ~ x, data = DF)
22  summary(model)
23
24  b1 <- model$coefficients[1]
25  b2 <- model$coefficients[2]
26
27  cat("Regression Equation is sales =",b1,"+",b2,"
       months")
28
29  # Prediction
30
31  predicted_miles_per_gallon <- predict(model)
32
33  DF <- cbind(DF, predicted_miles_per_gallon)
34
35  # Standard Residuals
36
37  standard_residuals <- rstandard(model)
38
39  DF <- cbind(DF, standard_residuals)
40
41
42  ggplot(DF,aes(predicted_miles_per_gallon, standard_
       residuals)) + geom_point() +
43    geom_hline(yintercept = 0, linetype=2) +
44    labs(title = "Scatter Plot between Predicted Miles
         per Gallon and Standard Residuals",
45        x = "Predicted Miles per Gallon", y = "
             Standard Residuals")
46
```

```
47  # Log  Transformation
48
49  #log_x <- log(x)
50  log_y <- log(y)
51
52  DF2 <- data.frame(x, log_y)
53
54  options(scipen = 999)    # To display not in
        scientific notation
55  model <- lm(log_y ~ x, data = DF2)
56  summary(model)
57
58  b1 <- model$coefficients[1]
59  b2 <- model$coefficients[2]
60
61  cat("Regression Equation is sales =",b1,"+",b2,"
        months")
62
63  # Prediction
64
65  predicted_miles_per_gallon <- predict(model)
66
67  DF2 <- cbind(DF2, predicted_miles_per_gallon)
68
69  # Standard Residuals
70
71  standard_residuals <- rstandard(model)
72
73  DF2 <- cbind(DF2, standard_residuals)
74
75
76  ggplot(DF2,aes(predicted_miles_per_gallon, standard_
        residuals)) + geom_point() +
77    geom_hline(yintercept = 0, linetype=2) + labs(
          title = "Scatter Plot between Log
78  Transformation of Predicted Miles per Gallon and
        Standard Residuals",
79          x = "Predicted Miles per Gallon", y = "
```

R code Exa 16.2a Analysis of a Large Problem

```
1                                          # Page no . : 736
                                              − 738
2
3  # Analysis of a Large Problem
4
5  Sales <- c (3669.88 , 3473.95 , 2295.10 , 4675.56 ,
      6125.96 , 2134.94 , 5031.66 , 3367.45 , 6519.45 ,
6            4876.37 , 2468.27 , 2533.31 , 2408.11 ,
                2337.38 , 4586.95 , 2729.24 , 3289.40 ,
                2800.78 ,
7            3264.20 , 3453.62 , 1741.45 , 2035.75 ,
                1578.00 , 4167.44 , 2799.97)
8  Time <- c (43.10 , 108.13 , 13.82 , 186.18 , 161.79 ,
      8.94 , 365.04 , 220.32 , 127.64 , 105.69 , 57.72 ,
      23.58 ,
9            13.82 , 13.82 , 86.99 , 165.85 , 116.26 ,
                42.28 , 52.84 , 165.04 , 10.57 , 13.82 ,
                8.13 , 58.44 ,
10           21.14)
11 Poten <- c (74065.1 , 58117.3 , 21118.5 , 68521.3 ,
      57805.1 , 37806.9 , 50935.3 , 35602.1 , 46176.8 ,
      42053.2 ,
12            36829.7 , 33612.7 , 21412.8 , 20416.9 ,
                36272.0 , 23093.3 , 26878.6 , 39572.0 ,
                51866.1 , 58749.8 ,
13            23990.8 , 25694.9 , 23736.3 , 34314.3 ,
                22809.5)
14 AdvExp <- c (4582.9 , 5539.8 , 2950.4 , 2243.1 , 7747.1 ,
      402.4 , 3140.6 , 2086.2 , 8846.2 , 5673.1 , 2761.8 ,
15            1991.8 , 1971.5 , 1737.4 , 10694.2 , 8618.6 ,
                7747.9 , 4565.8 , 6022.7 , 3721.1 ,
```

143

```
                      861.0 , 3571.5 ,
16                2845.5 , 5060.1 , 3552.0)
17 Share <- c(2.51 , 5.51 , 10.91 , 8.27 , 9.15 , 0.15 ,
     8.54 , 7.07 , 12.54 , 8.85 , 5.38 , 5.43 , 8.48 , 7.80 ,
18             10.34 , 5.15 , 6.64 , 5.45 , 6.31 , 6.35 ,
                  7.37 , 8.39 , 5.15 , 12.88 , 9.14)
19 Change <- c(0.34 , 0.15 , -0.72 , 0.17 , 0.50 , 0.15 ,
     0.55 , -0.49 , 1.24 , 0.31 , 0.37 , -0.65 , 0.64 , 1.01 ,
20             0.11 , 0.04 , 0.68 , 0.66 , -0.10 , -0.03 ,
                  -1.63 , -0.43 , 0.04 , 0.22 , -0.74)
21 Accounts <- c(74.86 , 107.32 , 96.75 , 195.12 , 180.44 ,
     104.88 , 256.10 , 126.83 , 203.25 , 119.51 , 116.26 ,
22                142.28 , 89.43 , 84.55 , 119.51 ,80.49 ,
                   136.58 , 78.86 , 136.58 , 138.21 ,
                   75.61 , 102.44 ,
23             76.42 , 136.58 , 88.62)
24 Work <- c(15.05 , 19.97 , 17.34 , 13.40 , 17.64 , 16.22 ,
     18.80 , 19.86 , 17.42 , 21.41 , 16.32 , 14.51 ,
25             19.35 , 20.02 , 15.26 , 15.87 , 7.81 , 16.00 ,
                  17.44 , 17.98 , 20.99 , 21.66 , 21.46 ,
                  24.78 ,
26         24.96)
27 Rating <- c(4.9 , 5.1 , 2.9 , 3.4 , 4.6 , 4.5 , 4.6 , 2.3 ,
     4.9 , 2.8 , 3.1 , 4.2 , 4.3 , 4.2 , 5.5 , 3.6 , 3.4 ,
28             4.2 , 3.6 , 3.1 , 1.6 , 3.4 , 2.7 , 2.8 , 3.9)
29
30 DF <- data.frame(Sales , Time , Poten , AdvExp , Share ,
     Change , Accounts , Work , Rating)
31 View(DF)
32
33 library(corrplot)
34
35 matrix <- round(cor(DF), 3)
36 matrix
37
38 regressor <- lm(Sales ~ Time + Poten + AdvExp +
     Share + Change + Accounts + Work + Rating ,
39                 data = DF)
```

```
40  summary(regressor)
41
42  cat("Equation is Sales = ",regressor$coefficients
        [1],"+",regressor$coefficients[2],"Time +",
43      regressor$coefficients[3],"Poten +",regressor$
            coefficients[4],"AdvExp +",regressor$
            coefficients[5],
44      "Share +",regressor$coefficients[6],"Change +",
            regressor$coefficients[7],"Accounts +",
45      regressor$coefficients[8],"Work +",regressor$
            coefficients[8],"Rating")
46
47  # Book Answer is different
48
49  regressor2 <- lm(Sales ~ Poten + AdvExp + Share,
        data = DF)
50  summary(regressor2)
51
52  cat("Equation is Sales = ",regressor2$coefficients
        [1],"+",regressor2$coefficients[3],"Poten +",
53      regressor2$coefficients[4],"AdvExp +",regressor2
            $coefficients[5],
54      "Share")
55
56  # Book Answer is different
```

R code Exa 16.3a Multiple Regression Approach to Experimental Design

```
1                                      # Page no. : 745 −
                                          747
2
3  # Multiple Regression Approach to Experimental
      Design
4
5  A <- c(58,64,55,66,67)
```

```r
 6  B <- c(58,69,71,64,68)
 7  C <- c(48,57,59,47,49)
 8
 9  DF <- data.frame(A,B,C)
10
11  newA <- c(1,1,1,1,1,0,0,0,0,0,0,0,0,0,0)
12  newB <- c(0,0,0,0,0,1,1,1,1,1,0,0,0,0,0)
13  y <- c(58,64,55,66,67,58,69,71,64,68,48,57,59,47,49)
14
15  DF2 <- data.frame(newA, newB, y)
16
17  regressor <- lm(y ~ newA + newB, data = DF2)
18  summary(regressor)
19
20  cat("Equation is y = ",regressor$coefficients[1],"+"
        ,regressor$coefficients[2],"A +",
21          regressor$coefficients[3],"B")
```

Chapter 17

Index Numbers

R code Exa 17.1a Price Relatives

```
1                                          # Page  no .  :  765
2
3 # Price  Relatives
4
5 year <- c
     (1990 ,1991 ,1992 ,1993 ,1994 ,1995 ,1996 ,1997 ,1998 ,1999 ,2000 ,2001 ,2002
6
7 price_per_gallon <- c(1.30 ,
     1.10 ,1.09 ,1.07 ,1.08 ,1.11 ,1.22 ,1.20 ,1.03 ,1.14 ,1.48 ,1.42 ,1.34 ,1.56 ,
8
9 DF <- data.frame(year , price_per_gallon)
10
11 base_year <- 1990
12 base_year_price <- DF$price_per_gallon[DF$year ==
     base_year]
13
14 price_relative <- round(((DF$price_per_gallon) / (
     base_year_price)) * 100, 2)
15
```

```
16  DF <- cbind(DF, price_relative)
17
18  View(DF)
```

R code Exa 17.2a Aggregate Price Indexes

```
 1                                              # Page no. :
                                                  766
 2
 3  # Aggregate Price Indexes
 4
 5  item <- c("Gallon of gasoline","Quart of oil","Tire"
        ,"Insurance policy")
 6  year_1990 <- c(1.30,2.10,130.00,820.00)
 7  year_2008 <- c(3.25,8.00,140.00,1030.00)
 8  quantity <- c(1000,15,2,1)
 9
10  DF <- data.frame(item,year_1990,year_2008,quantity)
11
12  base_year <- 1990
13
14  sum_of_1990_items <- sum(DF$year_1990)
15
16  sum_of_2008_items <- sum(DF$year_2008)
17
18  aggregate_index_2008 <- (sum_of_2008_items / sum_of_
        1990_items) * 100
19
20  cat("The unweighted aggregate index for year 2008 is
        ",aggregate_index_2008)
21
22
23  sum_of_1990_items <- sum(DF$year_1990 * DF$quantity)
            # Weighted Sum
24
```

```
25  sum_of_2008_items <- sum(DF$year_2008 * DF$quantity)
        # Weighted Sum
26
27  aggregate_index_2008 <- (sum_of_2008_items / sum_of_
    1990_items) * 100
28
29  cat("The weighted aggregate index for year 2008 is",
    aggregate_index_2008)
```

R code Exa 17.3a Computing an Aggregate Price Index from Price Relatives

```
1                                              # Page no. : 769
                                                  − 770
2
3   # Computing an Aggregate Price Index from Price
    Relatives
4
5   item <- c("Gallon of gasoline","Quart of oil","Tire"
    ,"Insurance policy")
6   year_1990 <- c(1.30,2.10,130.00,820.00)
7   year_2008 <- c(3.25,8.00,140.00,1030.00)
8   quantity <- c(1000,15,2,1)
9
10  DF <- data.frame(item,year_1990,year_2008,quantity)
11
12  base_year <- 1990
13
14  price_relative <- (DF$year_2008 / DF$year_1990) *
    100
15
16  weight <- DF$year_1990 * DF$quantity
17
18  weight_price_relative <- price_relative * weight
19
```

149

```
20  DF <- cbind(DF, price_relative, weight, weight_price
        _relative)
21
22  aggregate_2008 <- sum(DF$weight_price_relative) /
        sum(DF$weight)
23
24  cat("Aggregate Price Index for year 2008 is",
        aggregate_2008)
```

R code Exa 17.4a Deflating a Series by Price Indexes

```
1                                               # Page no. :
                                                       774
2
3  # Deflating a Series by Price Indexes
4
5  year <- c(2004,2005,2006,2007,2008)
6  hourly_wage <- c(15.69,16.12,16.76,17.45,18.07)
7  CPI <- c(188.9,195.3,201.6,207.3,215.3)
8
9  DF <- data.frame(year, hourly_wage, CPI)
10
11 # Install Library if not installed
12
13 # install.packages("ggplot2")
14
15 # Import Library
16
17 library(ggplot2)
18
19 ggplot(DF, aes(year, hourly_wage)) + geom_line() +
        geom_point() +
20  labs(title = "Year V/S Hourly Wage Graph", x = "
        Year", y = "Hourly Wage")
21
```

```
22
23 deflated_hourly_wage <- (DF$hourly_wage / DF$CPI) *
       100
24
25 DF <- cbind(DF, deflated_hourly_wage)
26
27 ggplot(DF, aes(year, deflated_hourly_wage)) + geom_
       line() + geom_point() +
28   ylim(c(7.75,9.0)) +
29   labs(title = "Year V/S Real Hourly Wage Graph", x
         = "Year", y = "Real Hourly Wage")
```

Chapter 18

Time Series Analysis and Forecasting

R code Exa 18.1a Time Series Patterns

```
1                                          # Page no. : 786
                                                  - 787
2
3 # Time Series Patterns
4
5 week <- c(1,2,3,4,5,6,7,8,9,10,11,12)
6 sales <- c(17,21,19,23,18,16,20,18,22,20,15,22)
7
8 DF <- data.frame(week,sales)
9
10 # Install Library if not installed
11
12 # install.packages("gpplot2")
13
14 # Import Library
15
16 library(ggplot2)
17
18 ggplot(DF,aes(week,sales)) + geom_line() + geom_
```

```
      point () + ylim(c(0 ,25)) + labs(title =
19                              "Week V/S Sales Time Series
                                Plot", x = "Weeks", y = "
                                Sales")
```

R code Exa 18.1b Time Series Patterns Eg2

```
 1                                      # Page no. : 787
                                              − 788
 2
 3 # Time Series Patterns Eg−2
 4
 5 week <- c
      (1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11 ,12 ,13 ,14 ,15 ,16 ,17 ,18 ,19 ,20 ,21 ,22)
 6 sales <- c
      (17 ,21 ,19 ,23 ,18 ,16 ,20 ,18 ,22 ,20 ,15 ,22 ,31 ,34 ,31 ,33 ,28 ,32 ,30 ,29 ,34 ,3
 7
 8 DF <- data.frame(week,sales)
 9
10 # Install Library if not installed
11
12 # install.packages("gpplot2")
13
14 # Import Library
15
16 library(ggplot2)
17
18 ggplot(DF,aes(week,sales)) + geom_line () + geom_
      point () + ylim(c(0 ,40)) +
19 labs(title = "Week V/S Sales Time Series Plot", x =
      "Weeks", y = "Sales")
```

R code Exa 18.1c Time Series Patterns Eg3

```
1                                        # Page no. :
                                           788 - 789
2
3 # Time Series Patterns Eg-3
4
5 year <- c(1,2,3,4,5,6,7,8,9,10)
6 sales <- c
      (21.6,22.9,25.5,21.9,23.9,27.5,31.5,29.7,28.6,31.4)

7
8 DF <- data.frame(year,sales)
9
10 # Install Library if not installed
11
12 # install.packages("gpplot2")
13
14 # Import Library
15
16 library(ggplot2)
17
18 ggplot(DF,aes(year,sales)) + geom_line() + geom_
      point() + ylim(c(20,34)) +
19  labs(title = "Years V/S Sales Time Series Plot", x
      = "Years", y = "Sales")
```

R code Exa 18.1d Time Series Patterns Eg4

```
1                                        # Page no. : 789
                                           - 790
2
```

```
3  # Time Series Patterns Eg−4
4
5  year <- c(1,2,3,4,5,6,7,8,9,10)
6  revenue <- c
      (23.1,21.3,27.4,34.6,33.8,43.2,59.5,64.4,74.2,99.3)

7
8  DF <- data.frame(year,revenue)
9
10 # Install Library if not installed
11
12 # install.packages("gpplot2")
13
14 # Import Library
15
16 library(ggplot2)
17
18 ggplot(DF,aes(year,revenue)) + geom_line() + geom_
      point() + ylim(c(0,120)) +
19  labs(title = "Years V/S Revenue Time Series Plot",
       x = "Years", y = "Revenue")
```

R code Exa 18.1e Time Series Patterns Eg5

```
1                                    # Page no. : 790
                                          − 791
2
3  # Time Series Patterns Eg−5
4
5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
     "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
6                   "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                      "Y4 Q2", "Y4 Q3", "Y4 Q4", "Y5 Q1
                      ", "Y5 Q2",
7                   "Y5 Q3", "Y5 Q4")  # Years and
```

```
                              Quarters
 8  sales <- c(125, 153, 106, 88, 118, 161, 133, 102,
        138, 144, 113, 80, 109, 137, 125, 109, 130, 165,
 9               128, 96)
10
11  DF <- data.frame(year_quart, sales)
12
13  # Install Library if not installed
14
15  # install.packages("gpplot2")
16
17  # Import Library
18
19  library(ggplot2)
20
21  ggplot(DF,aes(year_quart,sales, group = 1)) + geom_
        line() + geom_point() + ylim(c(0,180)) +
22      labs(title = "Years/quart V/S Sales Time Series
          Plot", x = "Year/Quarter", y = "Sales")
```

R code Exa 18.1f Time Series Patterns Eg6

```
 1                                          # Page no. :
                                              791 - 792
 2
 3  # Time Series Patterns Eg-6
 4
 5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
        "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
 6                 "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                     "Y4 Q2", "Y4 Q3", "Y4 Q4")
 7                  # Years and Quarters
 8  sales <- c(4.8, 4.1, 6.0, 6.5, 5.8, 5.2, 6.8, 7.4,
        6.0, 5.6, 7.5, 7.8, 6.3, 5.9, 8.0, 8.4)
 9
```

```
10 DF <- data.frame(year_quart, sales)
11
12 # Install Library if not installed
13
14 # install.packages("gpplot2")
15
16 # Import Library
17
18 library(ggplot2)
19
20 ggplot(DF,aes(year_quart,sales, group = 1)) + geom_
       line() + geom_point() + ylim(c(0.0,9.0)) +
21    labs(title = "Years/quart V/S Sales Time Series
         Plot", x = "Year/Quarter", y = "Sales")
```

R code Exa 18.2a Forecast Accuracy

```
1                                              # Page no. :
                                                  793 − 795
2
3 # Forecast Accuracy
4
5 week <- c(1,2,3,4,5,6,7,8,9,10,11,12)
6 sales <- c(17,21,19,23,18,16,20,18,22,20,15,22)
7 forcast <- c(NA,17,21,19,23,18,16,20,18,22,20,15)
8
9 DF <- data.frame(week,sales,forcast)
10
11 forcast_error <- DF$sales - DF$forcast
12
13 absolute_forecast_error <- abs(forcast_error)
14
15 square_absolute_forcast_error <- absolute_forecast_
       error**2
16
```

```
17  percent_error <- round((forcast_error / DF$sales) *
        100, 2)

18

19  absolute_percent_error <- abs(percent_error)

20

21  DF <- cbind(DF,forcast_error,absolute_forecast_error
        ,square_absolute_forcast_error,

22                  percent_error,absolute_percent_error)

23

24  View(DF)

25

26  total_forcast_error <- sum(DF$forcast_error, na.rm =
        T)

27

28  total_absolute_forcast_error <- sum(DF$absolute_
        forecast_error, na.rm = T)

29

30  total_sq_abs_error <- sum(DF$square_absolute_forcast
        _error, na.rm = T)

31

32  total_percent_error <- sum(DF$percent_error, na.rm =
        T)

33

34  total_absolute_percent_error <- sum(DF$absolute_
        percent_error, na.rm = T)

35

36  # Native Value

37

38  MAE <- round(total_absolute_forcast_error / (nrow(DF
        )-1), 2)    # Not including 1st row

39  # Mean Absolute Error

40

41  MSE <- round(total_sq_abs_error / (nrow(DF)-1), 2)
            # Not including 1st row

42  # Mean Square Error

43

44  MAPE <- round(total_absolute_percent_error / (nrow(
        DF)-1), 2)    # Not including 1st row
```

```r
45  # Mean Absolute Percent Error
46
47  cat("Value of MAE is",MAE)
48  cat("Value of MSE is",MSE)
49  cat("Value of MAPE is",MAPE)
50
51  # Average of Past Values
52
53  forcast2 <- c(NA
        ,17.00 ,19.00 ,19.00 ,20.00 ,19.60 ,19.00 ,19.14 ,19.00 ,19.33 ,19.40 ,19.00

54
55  DF2 <- data.frame(week,sales,forcast2)
56
57  forcast_error2 <- DF2$sales - DF2$forcast2
58
59  absolute_forecast_error2 <- abs(forcast_error2)
60
61  square_absolute_forcast_error2 <- absolute_forecast_
        error2**2
62
63  percent_error2 <- round((forcast_error2 / DF2$sales)
         * 100, 2)
64
65  absolute_percent_error2 <- abs(percent_error2)
66
67  DF2 <- cbind(DF2,forcast_error2,absolute_forecast_
        error2,square_absolute_forcast_error2,
68                  percent_error2,absolute_percent_error2)
69
70  View(DF2)
71
72  total_forcast_error2 <- sum(DF2$forcast_error2, na.
        rm = T)
73
74  total_absolute_forcast_error2 <- sum(DF2$absolute_
        forecast_error2, na.rm = T)
75
```

```
76  total_sq_abs_error2 <- sum(DF2$square_absolute_
        forcast_error2, na.rm = T)
77
78  total_percent_error2 <- sum(DF2$percent_error2, na.
        rm = T)
79
80  total_absolute_percent_error2 <- sum(DF2$absolute_
        percent_error2, na.rm = T)
81
82  MAE2 <- round(total_absolute_forcast_error2 / (nrow(
        DF2)-1), 2)    # Not including 1st row
83  # Mean Absolute Error
84
85  MSE2 <- round(total_sq_abs_error2 / (nrow(DF2)-1),
        2)    # Not including 1st row
86  # Mean Square Error
87
88  MAPE2 <- round(total_absolute_percent_error2 / (nrow
        (DF2)-1), 2)    # Not including 1st row
89  # Mean Absolute Percent Error
90
91  cat("Value of MAE is",MAE2)
92  cat("Value of MSE is",MSE2)
93  cat("Value of MAPE is",MAPE2)
```

R code Exa 18.3a Moving Averages

```
1                                        # Page no. : 798
                                                - 799
2
3  # Moving Averages
4
5  week <- c(1,2,3,4,5,6,7,8,9,10,11,12)
6  sales <- c(17,21,19,23,18,16,20,18,22,20,15,22)
7  forcast <- c(NA,NA,NA,19,21,20,19,18,18,20,20,19)
```

160

```
 8
 9  DF <- data.frame(week,sales,forcast)
10
11  forcast_error <- DF$sales - DF$forcast
12
13  absolute_forecast_error <- abs(forcast_error)
14
15  square_absolute_forcast_error <- absolute_forecast_
       error**2
16
17  percent_error <- round((forcast_error / DF$sales) *
       100, 2)
18
19  absolute_percent_error <- abs(percent_error)
20
21  DF <- cbind(DF,forcast_error,absolute_forecast_error
       ,square_absolute_forcast_error,
22                percent_error,absolute_percent_error)
23
24  View(DF)
25
26  # Install Library if not installed
27
28  # install.packages("ggplot2")
29
30  # Import Library
31
32  library(ggplot2)
33
34  ggplot(DF,aes(week)) + geom_line(aes(y = forcast),
       color = "red")+
35    geom_line(aes(y = sales), color = "blue") + geom_
         point(aes(y = forcast)) +
36    geom_point(aes(y = sales))+
37    ylim(c(0,25)) +
38    labs(title = "Week V/S Sales Time Series Plot", x
         = "Week", y = "Sales")
```

R code Exa 18.3b Exponential Smoothing

```
1                                           # Page no . : 802
                                                 - 803
2
3  # Exponential Smoothing
4
5  week <- c(1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11 ,12)
6  sales <- c(17 ,21 ,19 ,23 ,18 ,16 ,20 ,18 ,22 ,20 ,15 ,22)
7  forcast <- c(NA ,17.00 ,17.80 , 18.04 , 19.03 , 18.83 ,
       18.26 , 18.61 , 18.49 , 19.19 , 19.35 , 18.48)
8  forcast_error <- sales - forcast
9  forcast_error_sq <- (forcast_error)**2
10
11 DF <- data.frame(week , sales , forcast , forcast_error
       , forcast_error_sq)
12
13 View(DF)
14
15 # Install Library if not installed
16
17 # install.packages(" ggplot2")
18
19 # Import Library
20
21 library(ggplot2)
22
23   ggplot(DF ,aes(week)) + geom_line(aes(y = forcast) ,
         color = "red")+
24     geom_line(aes(y = sales), color = "blue") + geom_
           point(aes(y = forcast)) +
25     geom_point(aes(y = sales))+
26      ylim(c(0 ,25)) +
27    labs(title = "Week V/S Sales Time Series Plot", x
```

```
                = " Week " , y = " S a l e s " )
```

R code Exa 18.4a Trend Projection

```
 1                                          # Page no. :
                                              807 − 808
 2
 3  # Trend Projection
 4
 5  year <- c (1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10)
 6  sales <- c
        (21.6 ,22.9 ,25.5 ,21.9 ,23.9 ,27.5 ,31.5 ,29.7 ,28.6 ,31.4)

 7
 8  DF <- data.frame (year ,sales)
 9
10  # Install Library if not installed
11
12  # install.packages (" ggplot2 ")
13
14  # Import Library
15
16  library (ggplot2)
17
18  ggplot (DF ,aes (year ,sales)) + geom_line () + geom_
        point () + ylim (c (20 ,34)) +
19   labs (title = " Years V/S Sales Time Series Plot " , x
        = " Years " , y = " Sales ")
20
21  ggplot (DF ,aes (year ,sales)) + geom_line () + geom_
        point () + ylim (c (20 ,34)) +
22   geom_smooth (method = " lm " , se = F) +
23   labs (title = " Years V/S Sales Time Series and
         Linear Function Plot " , x = " Years " , y = " Sales ")
```

R code Exa 18.4b Trend Projection Part2

```
1                                           # Page no . : 809 −
                                              812
2
3 # Trend Projection Part −2
4
5 year <- c (1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10)
6 sales <- c
      (21.6 ,22.9 ,25.5 ,21.9 ,23.9 ,27.5 ,31.5 ,29.7 ,28.6 ,31.4)

7 forcast <- c
      (21.5 ,22.6 ,23.7 ,24.8 ,25.9 ,27.0 ,28.1 ,29.2 ,30.3 ,31.4)

8
9 DF <- data.frame ( year , sales , forcast )
10
11 regressor <- lm ( sales ~ year , data = DF )
12
13 cat (" Linear Trend Equation is T =" , regressor$
      coefficients [1] ,"+" , regressor$coefficients [2] ," t"
      )
14
15 res <- anova ( regressor )
16
17 cat ("MSE is " , res$'Sum Sq '[2]/10)
18
19 cat ("MSE is " , res$'Mean Sq '[2])
20
21 regressor
22 res
23
24 # Install Library if not installed
25
```

```
26  # install.packages(" ggplot2 ")
27
28  # Import  Library
29
30  library ( ggplot2 )
31
32  ggplot ( DF , aes ( year , sales )) + geom_line () + geom_
       point () + ylim ( c (20 ,34)) +
33    geom_smooth ( method = " lm " , se = F ) +
34    labs ( title = " Years  V/S  Sales  Time  Series  Plot " , x
          = " Years " , y = " Sales ")
```

R code Exa 18.4c Nonlinear Trend Regression

```
 1                                             # Page  no .  :  814 −
                                                      816
 2
 3  # Nonlinear  Trend  Regression
 4
 5  year <- c (1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10)
 6  revenue <- c
       (23.1 ,21.3 ,27.4 ,34.6 ,33.8 ,43.2 ,59.5 ,64.4 ,74.2 ,99.3)

 7  year_sq <- year **2
 8
 9  DF <- data . frame ( year , revenue , year_sq )
10
11  # Install  Library  if  not  installed
12
13  # install.packages(" gpplot2 ")
14
15  # Import  Library
16
17  library ( ggplot2 )
18
```

```
19  ggplot(DF,aes(year,revenue)) + geom_line() + geom_
        point() + ylim(c(0,120)) +
20    labs(title = "Years V/S Revenue Time Series Plot",
            x = "Year", y = "Revenue")
21
22  regressor <- lm(revenue ~ year + year_sq, data = DF)
23  summary(regressor)
24  anova(regressor)
25
26  ggplot(DF,aes(year,revenue)) + geom_line() + geom_
        point() + ylim(c(0,120)) + geom_smooth(method =
27                                                         "
                                                          lm
                                                          "
                                                          ,
                                                          se
                                                          =
                                                          F
                                                          )
                                                          +
28    labs(title = "Years V/S Revenue Time Series Plot",
            x = "Year", y = "Revenue")
```

R code Exa 18.5a Seasonality Without Trend

```
1                                      # Page no. : 821
                                            - 822
2
3  # Seasonality Without Trend
4
```

```r
5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
      "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
6                      "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                        "Y4 Q2", "Y4 Q3", "Y4 Q4", "Y5 Q1
                        ", "Y5 Q2",
7                      "Y5 Q3", "Y5 Q4")  # Years and
                        Quarters
8  sales <- c(125, 153, 106, 88, 118, 161, 133, 102,
      138, 144, 113, 80, 109, 137, 125, 109, 130, 165,
9              128, 96)
10
11 DF <- data.frame(year_quart, sales)
12
13 # Install Library if not installed
14
15 # install.packages("gpplot2")
16
17 # Import Library
18
19 library(ggplot2)
20
21 ggplot(DF,aes(year_quart,sales, group = 1)) + geom_
      line() + geom_point() + ylim(c(0,180)) +
22   labs(title = "Years/quart V/S Sales Time Series
        Plot", x = "Year/Quarter", y = "Sales")
23
24
25 q1 <- c(1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0)
26 q2 <- c(0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0)
27 q3 <- c(0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0)
28
29 DF <- cbind(DF,q1,q2,q3)
30
31 regressor <- lm(sales ~ q1 + q2 + q3, data = DF)
32 summary(regressor)
33
34 b0 <- regressor$coefficients[1]
35 b1 <- regressor$coefficients[2]
```

```
36  b2 <- regressor$coefficients [3]
37  b3 <- regressor$coefficients [4]
38
39  cat ("Equation is sales = ",b0,"+",b1,"Qtr1 +",b2,"
        Qtr2 +",b3,"Qtr3")
```

R code Exa 18.5b Seasonality andTrend

```
1                                    # Page no. : 823 −
                                          824
2
3  # Seasonality and Trend
4
5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
       "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
6                   "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                        "Y4 Q2", "Y4 Q3", "Y4 Q4")
7  # Years and Quarters
8  sales <- c(4.8, 4.1, 6.0, 6.5, 5.8, 5.2, 6.8, 7.4,
       6.0, 5.6, 7.5, 7.8, 6.3, 5.9, 8.0, 8.4)
9
10  DF <- data.frame(year_quart, sales)
11
12  # Install Library if not installed
13
14  # install.packages("gpplot2")
15
16  # Import Library
17
18  library(ggplot2)
19
20  ggplot(DF,aes(year_quart,sales, group = 1)) + geom_
       line() + geom_point() + ylim(c(0.0,9.0)) +
21    labs(title = "Years/quart V/S Sales Time Series
        Plot", x = "Year/Quarter", y = "Sales")
```

```
22
23  q1 <- c(1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0)
24  q2 <- c(0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0)
25  q3 <- c(0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0)
26  period <- c(1:16)
27
28  DF <- cbind(DF,q1,q2,q3,period)
29
30  regressor <- lm(sales ~ q1 + q2 + q3 + period, data
        = DF)
31  summary(regressor)
32
33  b0 <- regressor$coefficients[1]
34  b1 <- regressor$coefficients[2]
35  b2 <- regressor$coefficients[3]
36  b3 <- regressor$coefficients[4]
37  b4 <- regressor$coefficients[5]
38
39  cat("Equation is sales = ",b0,"+",b1,"Qtr1 +",b2,"
        Qtr2 +",b3,"Qtr3 +",b4,"t")
```

R code Exa 18.6a Time Series Decomposition

```
1                                        # Page no. :
                                          832 - 833
2
3  # Time Series Decomposition
4
5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
       "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
6                      "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                         "Y4 Q2", "Y4 Q3", "Y4 Q4")
7  # Years and Quarters
8  sales <- c(4.8, 4.1, 6.0, 6.5, 5.8, 5.2, 6.8, 7.4,
       6.0, 5.6, 7.5, 7.8, 6.3, 5.9, 8.0, 8.4)
```

```
 9  moving_avg <- c(NA
        ,5.350,5.600,5.875,6.075,6.300,6.350,6.450,6.625,6.725,6.800,6.875

10                      7.15,NA,NA)
11  centered_avg <- c(NA,NA
        ,5.475,5.738,5.975,6.188,6.325,6.400,6.538,6.675,6.763,6.838,6.938

12                      NA,NA)
13
14  DF <- data.frame(year_quart, sales,moving_avg,
        centered_avg)
15
16  # Install Library if not installed
17
18  # install.packages("gpplot2")
19
20  # Import Library
21
22  library(ggplot2)
23
24  ggplot(DF,aes(year_quart, group = 1)) + geom_line(
        aes(y = sales),color = "red") +
25      geom_point(aes(y = sales)) + geom_line(aes(y =
            centered_avg),color = "blue") +
26      geom_point(aes(y = centered_avg))+
27      ylim(c(0.0,9.0)) +
28      labs(title = "Years/quart V/S Sales Time Series
            Plot", x = "Year/Quarter", y = "Sales")
```

R code Exa 18.6b Deseasonalizing the Time Series

```
1                                      # Page no. : 835 -
                                          836
2
3  # Deseasonalizing the Time Series
```

170

```r
4
5  year_quart <- c("Y1 Q1", "Y1 Q2", "Y1 Q3", "Y1 Q4",
      "Y2 Q1", "Y2 Q2", "Y2 Q3", "Y2 Q4", "Y3 Q1",
6                   "Y3 Q2", "Y3 Q3", "Y3 Q4", "Y4 Q1",
                        "Y4 Q2", "Y4 Q3", "Y4 Q4")
7  # Years and Quarters
8  period <- c(1:16)
9  sales <- c(4.8, 4.1, 6.0, 6.5, 5.8, 5.2, 6.8, 7.4,
      6.0, 5.6, 7.5, 7.8, 6.3, 5.9, 8.0, 8.4)
10 index <- c
      (0.93,0.84,1.09,1.14,0.93,0.84,1.09,1.14,0.93,0.84,1.09,1.14,0.93

11 deseasonalized_sales <- c
      (5.16,4.88,5.50,5.70,6.24,6.19,6.24,6.49,6.45,6.67,6.88,6.84,6.77

12                                7.02,7.34,7.37)
13
14 DF <- data.frame(year_quart,period,sales,index,
      deseasonalized_sales)
15
16 # Install Library if not installed
17
18 # install.packages("gpplot2")
19
20 # Import Library
21
22 library(ggplot2)
23
24 ggplot(DF,aes(year_quart,deseasonalized_sales, group
      = 1)) + geom_line() + geom_point() +
25   ylim(c(0.0,9.0)) +
26   labs(title = "Years/quart V/S Deseasonalized Sales
          Time Series Plot", x = "Year/Quarter",
27       y = "Deseasonalized Sales")
28
29 regressor <- lm(deseasonalized_sales ~ period, data
      = DF)
30 summary(regressor)
```

```
31  anova(regressor)
32
33  b0 <- regressor$coefficients[1]
34  b1 <- regressor$coefficients[2]
35
36  cat("Equation is deseasonalized sales = ",b0,"+",b1,
        "Period")
```

Chapter 19

Non Parametric Tests

R code Exa 19.1a Rank Correlation

```
1                                        # Page no . : 887
                                              − 889
2
3  # Rank Correlation
4
5  sales_person <- c('A','B','C','D','E','F','G','H','I
       ','J')
6  x <- c(2,4,7,1,6,3,10,9,8,5)     # Ranking of
       Potential
7  y <- c(400,360,300,295,280,350,200,260,220,385)     #
       Two Years Sales
8  z <- c(1,3,5,6,7,4,10,8,9,2)     # Ranking According
       to y
9
10 DF <- data.frame(sales_person,x,y,z)
11
12 d <- DF$x - DF$z
13 d_sq <- d**2
14
15 DF <- cbind(DF,d,d_sq)
16
```

```
17  total_d_sq <- sum(DF$d_sq)
18
19  n <- nrow(DF)
20
21  r_s <- 1 - ((6 * total_d_sq)/(n * (n**2 + 1)))    #
        Spearman Rank-Correlation Coefficient
22
23  mean_rs <- 0
24
25  sigma_rs <- sqrt(1 / (n - 1))
26
27  z_val <- (r_s - mean_rs) / sigma_rs
28
29  alpha <- 0.05
30
31  # Two Tail Test
32
33  # P-value Approach
34
35  pval <- 2 * pnorm(z_val,lower.tail =  F)
36
37  if(pval >= alpha)
38  {
39    cat("Since pval",pval,"is greater than or equal to
          0.05 therefore we cannot reject the Null
        Hypothesis")
40  } else{
41    cat("Since pval",pval,"is less than 0.05 therefore
          we can reject the Null Hypothesis")
42  }
```

R code Exa 19.2a Sign Test

```
1                                               # Page no. : 858
                                                    - 861
```

174

```r
2
3  # Sign  Test
4
5  store <- c(56, 19, 36, 128, 12, 63, 39, 84, 102, 44)
6  sales <- c(485, 562, 415, 860, 426, 474, 662, 380,
       515, 721)
7
8  DF <- data.frame(store, sales)
9
10 median <- 450
11 sign <- c()
12
13 for (i in 1:nrow(DF)) {
14   if(DF$sales[i] >= median)
15   {
16     sign[i] = "+"
17   }else
18   {
19     sign[i] <- "-"
20   }
21 }
22
23 DF <- cbind(DF,sign)
24 View(DF)
25
26 positive <- table(DF$sign)[[2]]
27 n <- nrow(DF)
28
29 test <- binom.test(positive, n)
30 test
31
32 if(test$p.value >= 0.05)
33 {
34   cat("We cannot reject null hypothesis")
35 } else
36 {
37   cat("We can reject null hypothesis")
38 }
```

```
39
40
41  N <- 60
42  n1 <- 22    # "+ sign
43  n2 <- 38    # "- sign
44  med <- 236000
45
46  mean <- 0.50 * N
47  sd <- sqrt(0.25 * N)
48
49  CF <- 22.5    # Correction Factor
50
51  p <- pnorm(CF, mean = mean, sd = sd)
52
53  if(p >= 0.05)
54  {
55    cat("We cannot reject null hypothesis")
56  } else
57  {
58    cat("We can reject null hypothesis")
59  }
```

R code Exa 19.3a Wilcoxon Signed Rank Test

```
1                                    # Page no. : 866 -
                                            868
2
3  # Wilcoxon Signed - Rank Test
4
5  worker <- c(1:11)
6  A <- c(10.2, 9.6, 9.2, 10.6, 9.9, 10.2, 10.6, 10.0,
       11.2, 10.7, 10.6)
7  B <- c(9.5, 9.8, 8.8, 10.1, 10.3, 9.3, 10.5, 10.0,
       10.6, 10.2, 9.8)
8
```

```
 9  DF <- data.frame(worker, A, B)
10
11  options(warn = -1)
12
13  test <- wilcox.test(DF$A, DF$B, paired = T)
14  test
15
16  if(test$p.value >= 0.05)
17  {
18     cat("We cannot reject null hypothesis")
19  } else
20  {
21     cat("We can reject null hypothesis")
22  }
```

R code Exa 19.4a Mann Whitney Wilcoxon Test

```
 1                                    # Page no. : 873 −
                                             875
 2
 3  # Mann − Whitney − Wilcoxon − Test
 4
 5  college <- c(1:4)
 6  m1 <- c(15,3,23,8)
 7
 8  high <- c(1:5)
 9  m2 <- c(18,20,32,9,25)
10
11  test <- wilcox.test(m1, m2, correct = F)
12  test
13
14  if(test$p.value >= 0.05)
15  {
16     cat("We cannot reject null hypothesis")
17  } else
```

```
18 {
19    cat ("We can reject null hypothesis")
20 }
```

R code Exa 19.4b Mann Whitney Wilcoxon Test Eg2

```
1                                         # Page no. :
                                            876-878
2
3 # Mann - Whitney - Wilcoxon - Test Eg - 2
4
5 account1 <- c(1:12)
6 account2 <- c(1:10)
7
8 balance1 <- c(1095, 955, 1200, 1195, 925, 950, 805,
      945, 875, 1055, 1025, 975)
9 balance2 <- c(885, 850, 915, 950, 800, 750, 865,
      1000, 1050, 935)
10
11 test <- wilcox.test(balance1, balance2, correct = F)
12 test
13
14 if(test$p.value >= 0.05)
15 {
16    cat ("We cannot reject null hypothesis")
17 } else
18 {
19    cat ("We can reject null hypothesis")
20 }
```

R code Exa 19.5a Kruskal Wallis Test

```r
1                                    # Page no. : 883 -
                                            884
2
3  # Kruskal - Wallis Test
4
5  A <- c(25, 70, 60, 85, 95, 90, 80)
6  B <- c(60, 20, 30, 15, 40, 35)
7  C <- c(50, 70, 60, 80, 90, 70, 75)
8
9  x <- list(A,B,C)
10
11 test <- kruskal.test(x)
12 test
13
14 if(test$p.value >= 0.05)
15 {
16   cat("We cannot reject null hypothesis")
17 } else
18 {
19   cat("We can reject null hypothesis")
20 }
```

Chapter 20

Statistical Process Control

R code Exa 20.1a Sample Mean Chart Process Mean and SD Unknown

```
1                                        # Page no. : 913 −
                                              919
2
3  # Sample Mean Chart : Process Mean and SD Unknown
4
5  o1 <- c(3.5056, 3.4882, 3.4897, 3.5153, 3.5059,
       3.4977, 3.4910, 3.4991, 3.5099, 3.4880, 3.4881,
6         3.5043, 3.5043, 3.5004, 3.4846, 3.5145,
               3.5004, 3.4959, 3.4878, 3.4969)
7  o2 <- c(3.5086, 3.5085, 3.4898, 3.5120, 3.5113,
       3.4961, 3.4913, 3.4853, 3.5162, 3.5015, 3.4887,
8         3.4867, 3.4769, 3.5030, 3.4938, 3.4832,
               3.5042, 3.4823, 3.4864, 3.5144)
9  o3 <- c(3.5144, 3.4884, 3.4995, 3.4989, 3.5011,
       3.5050, 3.4976, 3.4830, 3.5228, 3.5094, 3.5141,
10        3.4946, 3.4944, 3.5082, 3.5065, 3.5188,
               3.4954, 3.4964, 3.4960, 3.5053)
11 o4 <- c(3.5009, 3.5250, 3.5130, 3.4900, 3.4773,
       3.5014, 3.4831, 3.5083, 3.4958, 3.5102, 3.5175,
12        3.5018, 3.5014, 3.5045, 3.5089, 3.4935,
               3.5020, 3.5082, 3.5070, 3.4985)
```

```
13  o5 <- c(3.5030, 3.5031, 3.4969, 3.4837, 3.4801,
        3.5060, 3.5044, 3.5094, 3.5004, 3.5146, 3.4863,
14          3.4784, 3.4904, 3.5234, 3.5011, 3.4989,
              3.4889, 3.4871, 3.4984, 3.4885)
15
16  DF <- data.frame(o1, o2, o3, o4, o5)
17
18  sample_mean <- rowMeans(DF)
19  sample_range <- c(0.0135, 0.0368, 0.0233, 0.0316,
        0.0340, 0.0099, 0.0213, 0.0264, 0.0270, 0.0266,
20                      0.0312, 0.0259, 0.0274, 0.0230,
                          0.0243, 0.0356, 0.0153, 0.0259,
                            0.0206, 0.0259)
21
22  DF <- cbind(DF, sample_mean, sample_range)
23  View(DF)
24
25  AR <- mean(DF$sample_range)   # Average Range
26  OM <- mean(DF$sample_mean)  # Overall Mean
27  n <- 5    # Sample Observations
28  d2 <- 2.362
29  A2 <- 3 / (d2 * sqrt(n))
30
31  UCL <- OM + (A2 * AR)    # Upper Control Limit
32  LCL <- OM - (A2 * AR)    # Lower Control Limit
33
34  cat("UCL is ",UCL)
35  cat("LCL is",LCL)
36
37  library(qicharts2)
38
39  qic(DF$sample_mean, xlab = "Sample Number", ylab = "
        Sample Mean", title = "Sample Mean Chart")
40
41  d3 <- 0.864
42
43  D1 <- 1 + (3 * (d3 / d2))
44  D2 <- 1 - (3 * (d3 / d2))
```

```
45
46  UCL2 <- AR * D1
47  LCL2 <- AR * D2
48
49  cat("UCL is ",UCL2)
50  cat("LCL is",LCL2)     # Book answer is different
51
52  qic(DF$sample_range, xlab = "Sample Number", ylab =
        "Sample Range", title = "R Chart")
53
54  p <- 0.03
55  n <- 200
56
57  sigma <- sqrt((p * (1 - p)) / n)
58
59  UCL3 <- p + 3 * sigma
60  LCL3 <- p - 3 * sigma
61
62  cat("UCL is ",UCL3)
63  cat("LCL is",LCL3)
64
65  # Data for P chart is not available in Book
66
67  UCL4 <- n * p + 3 * sqrt(n * p * (1 - p))
68  LCL4 <- n * p - 3 * sqrt(n * p * (1 - p))
69
70  cat("UCL is ",UCL4)
71  cat("LCL is",LCL4)
```

Chapter 21

Decision Analysis

R code Exa 21.1a Problem Formulation

```
1                                          # Page  no .  :
                                               940 − 941
2
3 # Problem  Formulation
4
5 decision <- factor(c("Small  Complex,  d1",  "Medium
     Complex,  d2",  "Large  Complex,  d3"))
6 demand1 <- c(8,  14,  20)
7 demand2 <- c(7,  5,  -9)
8
9 DF <- data.frame(decision,  demand1,  demand2)
10
11 # Install  Library  if  not  install
12
13 install.packages("rpart")
14 install.packages("rpart.plot")
15
16 # Import  Library
17
18 library(rpart)
19 library(rpart.plot)
```

```
20
21  ans <- rpart(decision ~ ., data = DF, method = "
        class" )
22  rpart.plot(ans)
23
24  # Decision Tree is Different from Book
```