

R Textbook Companion for
An Introduction to Statistical Methods and
Data Analysis
by R Lyman Ott and Michael Longnecker¹

Created by
Himanshu Gupta
B.E.
Information Technology
SGSITS INDORE
Cross-Checked by
R TBC Team

May 25, 2020

¹Funded by a grant from the National Mission on Education through ICT
- <http://spoken-tutorial.org/NMEICT-Intro>. This Textbook Companion and R
codes written in it can be downloaded from the "Textbook Companion Project"
section at the website - <https://r.fossee.in>.

Book Description

Title: An Introduction to Statistical Methods and Data Analysis

Author: R Lyman Ott and Michael Longnecker

Publisher: Cengage Learning, Canada

Edition: 6

Year: 2010

ISBN: 0-495-01758-2

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	4
3 Data Description	5
4 Probability and Probability Distributions	14
5 Inferences about Population Central Values	25
6 Inferences Comparing Two Population Central Values	38
7 Inferences about Population Variances	46
8 Inferences about More Than Two Population Central Values	52
9 Multiple Comparisons	57
10 Categorical Data	68
11 Linear Regression and Correlation	81
12 Multiple Regression and the General Linear Model	88
13 Further Regression Topics	103
14 Analysis of Variance for Completely Randomized Designs	104
15 Analysis of Variance for Blocked Designs	108

List of R Codes

Exa 3.1	Mode	5
Exa 3.2	Median	5
Exa 3.3	Mode and median	6
Exa 3.4	Median for interval data	7
Exa 3.5	Mean	8
Exa 3.6	Mean for interval data	8
Exa 3.7	Range	9
Exa 3.8	Percentile	9
Exa 3.9	Sample varaince	10
Exa 3.10	Variance and standard deviation	10
Exa 3.12	Approximate value	11
Exa 3.13	Crime study	11
Exa 3.14	Outliers	12
Exa 3.15	Boxplot	13
Exa 3.16	Correlation coefficient value	13
Exa 4.1	Venn diagram	14
Exa 4.2	Intersection probability	14
Exa 4.3	Book club	15
Exa 4.4	Circuit boards	15
Exa 4.7	Probability numerical	16
Exa 4.8	Number of trials	17
Exa 4.9	Probability of unemployed	17
Exa 4.10	Mean and standard deviation	18
Exa 4.11	Economic estimate	18
Exa 4.12	Mice in trap	18
Exa 4.13	Drug effect	19
Exa 4.15	Probability in normal distribution	19
Exa 4.16	Probability in normal distribution	20

Exa 4.17	Probability in normal distribution	20
Exa 4.18	Percentile of normal distribution	20
Exa 4.19	Income tax	20
Exa 4.20	Random sample	21
Exa 4.21	Sample function	21
Exa 4.22	Probability numerical	21
Exa 4.24	Blood pressure test	22
Exa 4.25	Finding probability	22
Exa 4.26	License probability	22
Exa 4.27	Normal quantile	23
Exa 4.28	Correlation coefficient	23
Exa 5.1	Calculating a Confidence Interval From a Normal Dis- tribution	25
Exa 5.2	Calculating a Confidence Interval From a Normal Dis- tribution	25
Exa 5.3	cost of textbooks	26
Exa 5.4	federal agency	26
Exa 5.5	Hypothesis Testing	27
Exa 5.6	Cholesterol levels	27
Exa 5.7	municipal employees	28
Exa 5.8	power for test	29
Exa 5.9	power for test	30
Exa 5.10	Suppose that the consumer testing agency thinks . . .	30
Exa 5.11	cereal manufacturer produces cereal	31
Exa 5.12	research hypothesis validity	31
Exa 5.13	research hypothesis validity	32
Exa 5.14	support the research hypothesis	33
Exa 5.15	research hypothesis validity	33
Exa 5.17	confidence interval	34
Exa 5.18	confidence interval	35
Exa 5.19	CPS personnel	35
Exa 5.20	confidence interval	36
Exa 5.21	large scale approximation	36
Exa 5.23	large sample approximation to the sign test	37
Exa 6.1	confidence interval for independent sample	38
Exa 6.2	confidence interval for independent sample	39
Exa 6.3	research hypothesis	39
Exa 6.4	research hypothesis	41

Exa 6.5	Many states are considering lowering the blood alcohol	41
Exa 6.6	Environmental engineers	42
Exa 6.7	Insurance adjusters are concerned about the high . . .	43
Exa 6.8	perform a paired t test	43
Exa 6.9	A city park department compared a new formulation	44
Exa 6.10	one sided test	45
Exa 6.11	Sample Size	45
Exa 7.1	The 99 confidence interval for mean	46
Exa 7.2	The 95 confidence interval for standard deviation . . .	47
Exa 7.4	the upper percentile for the F distribution	48
Exa 7.5	test the equality of the population variances	48
Exa 7.6	confidence interval	49
Exa 7.7	confidence interval	49
Exa 7.8	comparing the variability in power	50
Exa 7.9	Three different additives that are marketed	50
Exa 8.1	A large body of evidence	52
Exa 8.2	clinical psychologist	53
Exa 8.3	clerics knowledge of mental illness	53
Exa 8.4	dissolved oxygen contents at four distances from mouth	54
Exa 8.7	rank sum test	55
Exa 9.1	contrasts orthogonal	57
Exa 9.2	contrasts orthogonal	58
Exa 9.3	control weeds in crops	59
Exa 9.5	Test each of the four contrasts for significance	61
Exa 9.6	control the experimentwise error rate	62
Exa 9.7	five different weed agents	62
Exa 9.9	confidence interval for mean	63
Exa 9.11	confidence interval	63
Exa 9.13	Compare the two biological treatments	64
Exa 9.14	Scheffs procedure to determine	65
Exa 10.1	Estimate the proportion of all patients with the speci- fied type of cancer	68
Exa 10.2	water department	69
Exa 10.3	confidence interval for pie would be	70
Exa 10.4	designer of the new operating system	70
Exa 10.5	percentage of binge drinkers at the university	71
Exa 10.6	confidence interval	72
Exa 10.7	confidence interval	72

Exa 10.8	clinical trial is conducted to compare two drug therapies	73
Exa 10.9	Previous experience with the breeding of a particular herd of cattle	73
Exa 10.10	null hypothesis	74
Exa 10.11	Environmental engineers	75
Exa 10.12	random sample of 216 patients	76
Exa 10.13	test to determine the severity of the disease	77
Exa 10.14	test of homogeneity of distributions	77
Exa 10.15	Consider both a population in which	78
Exa 10.16	level of stress	79
Exa 10.17	The pharmaceutical study	79
Exa 11.2	least squares estimates of slope and intercept	81
Exa 11.3	the least squares estimates of the intercept	81
Exa 11.4	Forest growth retardation	82
Exa 11.7	confidence interval for the slope	83
Exa 11.8	use the F test for testing	83
Exa 11.10	The manufacturer of a new brand of thermal panes	83
Exa 11.11	Conduct a test for lack of fit	84
Exa 11.12	An engineer is interested in calibrating a flow meter	85
Exa 11.13	In a study of the reproductive success of grasshoppers	85
Exa 11.14	compute SSTotal	86
Exa 11.15	The personnel director of a small company	86
Exa 11.16	Perform t tests for the null hypothesis	87
Exa 12.2	An industrial engineer is designing a simulation model to generate	88
Exa 12.3	Check your findings by substituting values	89
Exa 12.5	An experiment was conducted to investigate	89
Exa 12.6	A kinesiologist is investigating measures of the physical fitness	90
Exa 12.8	The following SPSS computer output is obtained from the data	91
Exa 12.9	Using the sum of squares in the ANOVA table	92
Exa 12.11	The following SAS output is provided for fitting the model	93
Exa 12.12	A large city bank studies the relation of average account size	95
Exa 12.13	confidence interval	96
Exa 12.14	Locate the estimated partial slope	96

Exa 12.15	conclusion of the test compatible with the confidence interval	97
Exa 12.16	Locate the t statistic	97
Exa 12.17	A state fisheries commission wants to estimate the number of bass	98
Exa 12.22	logistic regression	99
Exa 12.26	matrix operations	99
Exa 12.27	inverse of matrix	100
Exa 12.28	Obtain the least squares estimates for the prediction equation	100
Exa 12.29	Compute SSR _{residual} for the data	101
Exa 12.30	Calculate SSR _{regression} and SST _{total} for the data . . .	101
Exa 12.31	Calculate the estimated standard error	102
Exa 14.1	assignment of paints to the highway sections	104
Exa 14.2	sum of squares for error	104
Exa 14.11	confidence interval	105
Exa 14.12	locate significant differences among display panels . .	106
Exa 14.13	number of replications is	107
Exa 15.7	Assess whether taking into account the two extraneous sources of variation	108
Exa 15.11	determine which pairs of treatments have significantly different means	108

Chapter 3

Data Description

R code Exa 3.1 Mode

```
1 # Page No. 78
2
3 selling_price<-c( 26.6, 25.3 ,23.8 ,24.0, 27.5,
4                 21.1 ,25.9 ,22.6, 23.8, 25.1,
5                 22.6 ,27.5 ,26.8 ,23.4, 27.5,
6                 20.8 ,20.4, 22.4 ,27.5, 23.7,
7                 22.2 ,23.8, 23.2, 28.7,27.5)
8 modal_selling_price<-table(selling_price)
9
10 print("Modal SP")
11 names(modal_selling_price)[which(modal_selling_price
    ==max(modal_selling_price))]
```

R code Exa 3.2 Median

```
1 # Page No. 79
2
3 percentage<-c(95 ,86 ,78 ,90, 62, 73 ,89 ,92 ,84
    ,76)
```

```

4 new_list<-sort(percentage)
5 n<-length(percentage)
6
7 if(n%%2==0) {
8   a<-new_list[n/2]
9   print(a)
10  b<-new_list[(n/2)+1]
11  print(b)
12  print("median is")
13  print((a+b)/2)
14 } else {
15   m<-(n+1)/2
16   print("median is")
17   print(new_list[m])
18 }

```

R code Exa 3.3 Mode and median

```

1 # Page No. 80
2
3 productivity<-c(4.4 ,4.9, 4.2 ,4.4, 4.8 ,4.9, 4.8,
4               4.5, 4.3 ,4.8 ,4.7 ,4.4, 4.2)
5 new_list<-sort(productivity)
6 n<-length(productivity)
7 if(n%%2==0) {
8   a<-new_list[n/2]
9   print(a)
10  b<-new_list[(n/2)+1]
11  print(b)
12  print("median is")
13  print((a+b)/2)
14 } else {
15   m<-(n+1)/2
16   print("output is")

```

```

17   print(new_list[m])
18 }
19 print("mode is ")
20 modal_productivity<-table(productivity)
21 names(modal_productivity)[which(modal_productivity==
    max(modal_productivity))]

```

R code Exa 3.4 Median for interval data

```

1 # Page No. 80
2
3 Median_calculate <- function(frequencies, intervals,
    sep = NULL, trim = NULL) {
4   if (!is.null(sep)) {
5     if (is.null(trim)) pattern <- ""
6     else if (trim == "cut") pattern <- "
        \\[[\\]]|\\(|\\)"
7     else pattern <- trim
8     intervals <- sapply(strsplit(gsub(pattern, "",
        intervals), sep), as.numeric)
9   }
10
11   Midpoints <- rowMeans(intervals)
12   cf <- cumsum(frequencies)
13   Midrow <- findInterval(max(cf)/2, cf) + 1
14   L <- intervals[1, Midrow]
15   h <- diff(intervals[, Midrow])
16   f <- frequencies[Midrow]
17   cf2 <- cf[Midrow - 1]
18   n_2 <- max(cf)/2
19   unname(L + (n_2 - cf2)/f * h)
20 }
21
22 mydataframe <- structure(list(Class_interval = c("
    16.25-18.75", "18.75-21.25", "21.25-23.75", "

```

```

23.75-26.25", "26.25-28.75", "28.75-31.25", "
31.25-33.75", "33.75-36.25", "36.25-38.75", "
38.75-41.25", "41.25-43.75"), freq = c(2L, 7L, 7
L, 14L, 17L, 24L, 11L, 11L, 3L, 3L, 1L)), .Names =
c("class_interval", "freq"), class = "data.frame
", row.names = c(NA, -11L))
23 print(mydataframe)
24
25 Median_calculate(mydataframe$freq, mydataframe$class
_interval, sep = "-")

```

R code Exa 3.5 Mean

```

1 # Page No. 82
2
3 accounts_due<-c(55.20, 4.88 ,271.95,
4                18.06, 180.29, 365.29,
5                28.16, 399.11 ,807.80,
6                44.14, 97.47, 9.98,
7                61.61, 56.89 ,82.73)
8 n<-sum(accounts_due)
9 d<-length(accounts_due)
10
11 mean_accountsdue<-n/d
12 print(mean_accountsdue)
13 total_overdueamount=150*mean_accountsdue
14 print(total_overdueamount)

```

R code Exa 3.6 Mean for interval data

```

1 # Page No. 83
2

```

```

3 ClassInterval <- c("16.25-18.75", "18.75-21.25", "
    21.25-23.75", "23.75-26.25", "26.25-28.75", "
    28.75-31.25", "31.25-33.75", "33.75-36.25", "
    36.25-38.75", "38.75-41.25", "41.25-43.75")
4 freq <- c(2,7,7,14,17,24,11,11,3,3,1)
5 mid_interval<- c
    (17.5,20.0,22.5,25.0,27.5,30.0,32.5,35.0,37.5,40.0,42.5)

6
7 fmi<-freq*mid_interval
8 List<- data.frame(ClassInterval, freq, mid_interval,
    fmi)
9 print(List)
10 print("mean is")
11 print(sum(fmi)/sum(freq))

```

R code Exa 3.7 Range

```

1 # Page No. 86
2
3 accounts_due<-c(55.20, 4.88 ,271.95,
4                18.06, 180.29, 365.29,
5                28.16, 399.11 ,807.80,
6                44.14, 97.47, 9.98,
7                61.61, 56.89 ,82.73)
8
9 range(accounts_due)
10 diff=max(accounts_due)-min(accounts_due)
11 print(diff)

```

R code Exa 3.8 Percentile

```

1 # Page No. 90

```

```

2
3 L <- 33.75
4 n <- 100
5 cfb <- 82
6 f90 <- 11
7 w <- 2.5
8
9 P <- L + (w/f90) * (0.9 * n - cfb)
10 print(P)

```

R code Exa 3.9 Sample variance

```

1 # Page No. 92
2
3 y <- c(5,4,3,1,3)
4
5 mean_y <- sum(y)/length(y)
6 sample_variance <- (sum((y-mean_y)^2/(length(y)-1)))
7 print(sample_variance)

```

R code Exa 3.10 Variance and standard deviation

```

1 # Page No. 93
2
3 ClassInterval <- c("16.25-18.75", "18.75-21.25", "
    21.25-23.75", "23.75-26.25", "26.25-28.75", "
    28.75-31.25", "31.25-33.75", "33.75-36.25", "
    36.25-38.75", "38.75-41.25", "41.25-43.75")
4 freq <- c(2,7,7,14,17,24,11,11,3,3,1)
5 mid_interval <- c
    (17.5,20.0,22.5,25.0,27.5,30.0,32.5,35.0,37.5,40.0,42.5)

6 fmi <- freq * mid_interval

```

```

7
8 mean_y<-sum(fmi)/sum(freq)
9 sample_variance <-(sum(freq*((mid_interval-mean_y)
  ^2)/(sum(freq)-1)))
10 print(sample_variance)
11 standard_deviation <-sqrt(sample_variance)
12 print(standard_deviation)

```

R code Exa 3.12 Approximate value

```

1 # Page No. 95
2
3 y<-c(26 ,28 ,30, 37 ,33 ,30,
4      29 ,39 ,49 ,31, 38 ,36,
5      33 ,24, 34, 40 ,29, 41,
6      40, 29, 35 ,44 ,32, 45,
7      35 ,26, 42, 36 ,37 ,35)
8
9 mean_y<-sum(y)/length(y)
10 sample_variance <-(sum((y-mean_y)^2/(length(y)-1)))
11 standard_deviation <-sqrt(sum((y-mean_y)^2/(length(y)
  )-1)))
12 s=(max(y)-min(y))/4;
13
14 print(mean_y)
15 print(sample_variance)
16 print(standard_deviation)
17 print(s)

```

R code Exa 3.13 Crime study

```

1 # Page No. 97
2

```



```

3 crime_rate=c
  (876,578,718,388,562,971,698,298,673,537,642,856,376,508,529,393,
4
5 median(crime_rate)
6 lower_quartile=quantile(crime_rate,0.25)
7 lower_quartile
8 upper_quartile=quantile(crime_rate,0.75)
9 upper_quartile
10 IQR(crime_rate)
11
12 # The answer provided in the textbook is wrong.

```

R code Exa 3.14 Outliers

```

1 # Page No. 100
2
3 crime_rate=c
  (876,578,718,388,562,971,698,298,673,537,642,856,376,508,529,393,
4
5 lower_quartile= 464.5
6 upper_quartile=718.5
7
8 iqr=IQR(crime_rate)
9 lower_inner_fence= lower_quartile - (1.5*iqr)
10 upper_inner_fence= upper_quartile + (1.5*iqr)
11 lower_outer_fence= lower_quartile - (3*iqr)
12 upper_outer_fence= upper_quartile +(3*iqr)
13 print(lower_inner_fence)
14 print(upper_inner_fence)
15 print(lower_outer_fence)
16 print(upper_outer_fence)
17
18 # The answer provided in the textbook is wrong.

```

R code Exa 3.15 Boxplot

```
1 # Page No. 101
2
3 crime_rate=c
  (876,578,718,388,562,971,698,298,673,537,642,856,376,508,529,393,
4
5 boxplot(crime_rate, horizontal = TRUE, axes = FALSE,
  staplewex = 1)
6 text(x=fivenum(crime_rate), labels =fivenum(crime_
  rate), y=1.25)
```

R code Exa 3.16 Correlation coefficient value

```
1 # Page No. 107
2
3 x<-c(20,23,29,27,30,34,35,37,40,43)
4 y<-c
  (1.32,1.67,2.17,2.70,2.75,2.87,3.65,2.86,3.61,4.25)
5 z<-(x-mean(x))*(y-mean(y))
6
7 A<-sum(z)
8 p<-sum((x-mean(x))*(x-mean(x)))
9 q<-sum((y-mean(y))*(y-mean(y)))
10 B<-sqrt(p*q)
11 coefficient<-A/B
12 print((coefficient))
```

Chapter 4

Probability and Probability Distributions

R code Exa 4.1 Venn diagram

```
1 # Page No. 149
2
3 p_A<-0.5
4 p_B<-0.2
5 p_A_inter_B<-0.05
6
7 p_A_Comp<-1 - p_A
8 print(p_A_Comp)
9 p_B_Comp<-1 - p_B
10 print(p_B_Comp)
11 print(p_A_inter_B)
12 p_A_un_B<-p_A+p_B-p_A_inter_B
13 print(p_A_un_B)
```

R code Exa 4.2 Intersection probability

```

1 # Page No. 151
2
3 p_A<-0.6
4 p_B_by_A<-5/9
5 p_A_inter_B<-p_A*p_B_by_A
6 print(p_A_inter_B)

```

R code Exa 4.3 Book club

```

1 # Page No. 153
2
3 p_l=0.50
4 p_m=0.30
5 p_h=0.20
6 p_0_l=0.60
7 p_0_m=0.15
8 p_0_h=0.05
9
10 p=(p_l*p_0_l)/((p_l*p_0_l)+(p_m*p_0_m)+(p_h*p_0_h))
11 print(p)

```

R code Exa 4.4 Circuit boards

```

1 # Page No. 154
2
3 p_d1=0.028
4 p_d2=0.012
5 p_d3=0.032
6 p_d4=0.928
7 p_a4_d1=0.02
8 p_a4_d2=0.09
9 p_a4_d3=0.10
10 p_a4_d4=0.95

```

```

11
12 p_nd_or_d1=(p_d1*p_a4_d1)/((p_d1*p_a4_d1)+(p_d2*p_a4
    _d2)+(p_d3*p_a4_d3)+(p_d4*p_a4_d4))
13 p_nd_or_d2=(p_d2*p_a4_d2)/((p_d1*p_a4_d1)+(p_d2*p_a4
    _d2)+(p_d3*p_a4_d3)+(p_d4*p_a4_d4))
14 p_nd_or_d3=(p_d3*p_a4_d3)/((p_d1*p_a4_d1)+(p_d2*p_a4
    _d2)+(p_d3*p_a4_d3)+(p_d4*p_a4_d4))
15 p_nd_or_d4=(p_d4*p_a4_d4)/((p_d1*p_a4_d1)+(p_d2*p_a4
    _d2)+(p_d3*p_a4_d3)+(p_d4*p_a4_d4))
16
17 print(p_nd_or_d1)
18 print(p_nd_or_d2)
19 print(p_nd_or_d3)
20 print(p_nd_or_d4)

```

R code Exa 4.7 Probability numerical

```

1 # Page No. 162
2
3 n<-20
4 z<-0.85
5
6 y<-18
7 p_18sds<-(factorial(n))/(factorial(y)*factorial(n-y)
    )*(z^y)*(1-z)^(n-y)
8 y<-19
9 p_19sds<-(factorial(n))/(factorial(y)*factorial(n-y)
    )*(z^y)*(1-z)^(n-y)
10 y<-20
11 p_20sds<-(factorial(n))/(factorial(y)*factorial(n-y)
    )*(z^y)*(1-z)^(n-y)
12 t_p<-p_18sds+p_19sds+p_20sds
13 print(t_p)

```

R code Exa 4.8 Number of trials

```
1 # Page No. 162
2
3 n<-5
4 z<-0.9
5 y<-5
6
7 p_15<-(factorial(n))/(factorial(y)*factorial(n-y))*
  z^y*(1-z)^(n-y)
8 print(p_15)
```

R code Exa 4.9 Probability of unemployed

```
1 # Page No. 163
2
3 n<-5
4 z<-0.9
5 y<-4
6
7 p_1_unemp<-(factorial(n))/(factorial(y)*factorial(n-
  y))*(z^y)*(1-z)^(n-y)
8 p_few_unemp=((factorial(n))/(factorial(4)*factorial(
  n-4))*(z^4)*(1-z)^(n-4))+((factorial(n))/(
  factorial(5)*factorial(n-5))*(z^5)*(1-z)^(n-5))
9
10 print(p_1_unemp)
11 print(p_few_unemp)
```

R code Exa 4.10 Mean and standard deviation

```
1 # Page No. 164
2
3 n<-20
4 z<-0.85
5
6 m<-n*z
7 s_d=sqrt(n*z*(1-z))
8 print(m)
9 print(s_d)
```

R code Exa 4.11 Economic estimate

```
1 # Page No. 165
2
3 n<-1218
4 z<-0.5
5
6 m<-n*z
7 print(m)
8 s_d=sqrt(n*z*(1-z))
9 print(s_d)
10
11 o_v_y=516
12 o_v_y>3*s_d
13 o_v_y<m
```

R code Exa 4.12 Mice in trap

```
1 # Page No. 167
2
3 U<-2.3
```

```

4 y<-4
5
6 p_4<-((U^y)*(exp(1)^-U))/factorial(y)
7 p_most_4=((U^0)*(exp(1)^-U))/factorial(0))+(((U^1)*
  (exp(1)^-U))/factorial(1))+(((U^2)*(exp(1)^-U))/
  factorial(2))+(((U^3)*(exp(1)^-U))/factorial(3))
  +(((U^4)*(exp(1)^-U))/factorial(4))
8 p_more_4=1-p_most_4
9
10 print(p_4)
11 print(p_most_4)
12 print(p_more_4)

```

R code Exa 4.13 Drug effect

```

1 # Page No. 167
2
3 n<-1000
4 z<-0.001
5 U<-1
6 y<-0
7
8 m<-n*z
9 p_se<-((U^y)*(exp(1)^-U))/factorial(y)
10
11 print(m)
12 print(p_se)

```

R code Exa 4.15 Probability in normal distribution

```

1 # Page No. 174
2
3 pnorm(23,mean =20,sd=2)

```

R code Exa 4.16 Probability in normal distribution

```
1 # Page No. 174
2
3 pnorm(16, mean =20, sd=2)
```

R code Exa 4.17 Probability in normal distribution

```
1 # Page No. 175
2
3 pnorm(60, mean =70, sd=13)
4 pnorm(90, mean =70, sd=13, lower.tail = FALSE)
5 pnorm(90, mean =70, sd=13) - pnorm(60, mean =70, sd=13)
```

R code Exa 4.18 Percentile of normal distribution

```
1 # Page No. 177
2
3 qnorm(0.10, 70, 13)
```

R code Exa 4.19 Income tax

```
1 # Page No. 178
2
3 mu=530
4 sigma=205
```

```

5 z_75=0.67
6 y=mu+sigma*z_75
7 print(y)

```

R code Exa 4.20 Random sample

```

1 # Page No. 179
2
3 ct<-c("c1","c2","c3","c4","c5","c6","c7","c8","c9","
      c10")
4 t(combn(ct, 2))
5 t_p<-nrow(t(combn(ct, 2)))
6 p_s2c<-1/t_p
7 print(p_s2c)

```

R code Exa 4.21 Sample function

```

1 # Page No. 180
2
3 st<-c(0:849)
4 sample(st, 20)

```

R code Exa 4.22 Probability numerical

```

1 # Page No. 182
2
3 pop<-c(2, 3, 4, 5, 6, 7, 8, 9, 10, 11)
4
5 combn(pop, 2)
6 samps<-combn(pop, 2)

```

```
7 xbars <- colMeans(samps)
8 table(xbars)
9 prop.table(table(xbars))
```

R code Exa 4.24 Blood pressure test

```
1 # Page No. 189
2
3 pnorm(150, mean = 160, sd = 20)
4 sd <- 20 / sqrt(5)
5 pnorm(150, 160, sd = 8.94)
6 st_dv = 20
7 n = ((-2.326 * st_dv) / (150 - 160))^2
8 print(n)
```

R code Exa 4.25 Finding probability

```
1 # Page No. 192
2
3 n <- 1000
4 z <- 0.5
5
6 m <- n * z
7 s <- sqrt(n * z * (1 - z))
8 pnorm(460, mean = m, sd = s)
```

R code Exa 4.26 License probability

```
1 # Page No. 194
2
```

```

3 n<-100
4 z<-0.2
5
6 m<-n*z
7 s<-sqrt(n*z*(1-z))
8 pnorm(14.5, mean=m, sd=s, lower.tail=FALSE)

```

R code Exa 4.27 Normal quantile

```

1 # Page No. 195
2
3 Ch_Rd=c
    (133,137,148,149,152,167,174,179,189,192,201,209,210,211,218,238,
4 n_q=c
    (-1.96,-1.44,-1.15,-.935,-.755,-.598,-.454,-.319,-.189,-.063,.063
5
6 plot(n_q,Ch_Rd)
7 md=lm(Ch_Rd~n_q)
8 summary(md)

```

R code Exa 4.28 Correlation coefficient

```

1 # Page No. 197
2
3 y=c
    (133,137,148,149,152,167,174,179,189,192,201,209,210,211,218,238,
4 x=c
    (-1.868,-1.403,-1.128,-.919,-.744,-.589,-.448,-.315,-.187,-.062,.
5

```

6 `cor(y, x)`

Chapter 5

Inferences about Population Central Values

R code Exa 5.1 Calculating a Confidence Interval From a Normal Distribution

```
1 # Calculating a Confidence Interval From a Normal
  Distribution
2 n<-50
3 a<-2.8
4 s<-0.6
5 # we will use a 95% confidence level and wish to
  find the confidence interval
6 margin <- qnorm(0.975)*s/sqrt(n)
7 left_i <- a-margin
8 right_i <- a+margin
9 print("Confidence interval is")
10 print(left_i)
11 print(right_i)
```

R code Exa 5.2 Calculating a Confidence Interval From a Normal Distribution

```

1 # Calculating a Confidence Interval From a Normal
  Distribution
2 n<-50
3 a<-27.3
4 s<-12.1
5 # we will use a 99% confidence level and wish to
  find the confidence interval
6 margin <- qnorm(0.995)*s/sqrt(n)
7 left_i <- a-margin
8 right_i <- a+margin
9 print("Confidence interval is")
10 print(left_i)
11 print(right_i)

```

R code Exa 5.3 cost of textbooks

```

1 # the 95% confidence level would imply the 97.5th
  percentile of the normal distribution at the
  upper tail
2 zstar <-qnorm(.975)
3 # standard deviation
4 sd <- 125
5 # level of accuracy
6 E <- 25
7 sample_size<- zstar^2 * sd * sd/ E^2
8 print(ceiling(sample_size))
9 # A sample size of 97 or larger is recommended to
  obtain an estimate of the mean textbook

```

R code Exa 5.4 federal agency

```

1 # the 99% confidence level would imply the 99.5th
  percentile of the normal distribution at the
  upper tail
2 zstar <- qnorm(0.995)
3 width_interval<-0.50
4 E<-width_interval/2
5 sd<-0.75
6 sample_size<- zstar^2 * sd * sd/ E^2
7 print(ceiling(sample_size))
8 # the federal agency must obtain a random sample of
  60 cereal cartons to estimate .

```

R code Exa 5.5 Hypothesis Testing

```

1 # Hypothesis Testing or one-tailed test
2
3 ybar = 573           # sample mean
4 mu0 = 520           # hypothesized value
5 sigma = 124         # population standard
  deviation
6 n = 36              # sample size
7 z = (ybar- mu0)/(sigma/sqrt(n))
8 print(z) # test statistic
9
10 # We then compute the critical value at .025
  significance level.
11 # For alpha= .025, reject the null hypothesis if
  lies more than 1.96
12 alpha = .025
13 z.alpha = qnorm(1-alpha)
14 print(z.alpha)

```

R code Exa 5.6 Cholesterol levels


```

1 # two tailed test
2 ybar = 178.2          # sample mean
3 mu0 = 190             # hypothesized value
4 sigma = 45.3          # population standard
   deviation
5 n = 100               # sample size
6 # We compute the critical value at .025
   significance level.
7 alpha = .05
8 z.half.alpha = qnorm(1-alpha/2)
9 # critical values
10 lr=mu0-(z.half.alpha*sigma)/sqrt(n)
11 ur=mu0+(z.half.alpha*sigma)/sqrt(n)
12 paste0(" lower rejection = ",lr)
13 paste0(" upper rejection = ",ur)
14 z = (ybar- mu0)/(sigma/sqrt(n))
15 print(z) # test statistic
16
17 print("The test statistic doesnot lies between the
   critical values(i.e. |z|>critical value). Hence,
   at .025 significance level, we reject the null
   hypothesis")

```

R code Exa 5.7 municipal employees

```

1
2
3 ybar = 390            # sample mean
4 mu0 = 380             # hypothesized value
5 sigma = 35.2          # population standard
   deviation
6 n = 50               # sample size
7 z = (ybar- mu0)/(sigma/sqrt(n))
8 print(z) # test statistic
9

```

```

10 # We then compute the critical value at .01
    significance level.
11 # For alpha= .01, reject the null hypothesis if
    lies more than 2.33
12 alpha = .01
13 # critical value
14 z.alpha = qnorm(1-alpha)
15 print(z.alpha)
16 print("the observed value of z < critical value, so
    we might be tempted to accept the null
    hypothesis")
17 # but Beta is not computed so there is
    insufficient evidence to reject the null
    hypothesis.
18 # To reach a conclusion about whether to accept or
    reject H0, beta should be calculated.

```

R code Exa 5.8 power for test

```

1
2
3 ybar = 380           # sample mean
4 mu0 = 395           # hypothesized value
5 sigma = 35.2        # population standard
    deviation
6 n = 50              # sample size
7 z = abs((ybar- mu0)/(sigma/sqrt(n)))
8 # test statistic
9 print(z)
10 # We then compute the critical value at .01
    significance level.
11 alpha = .01
12 # critical value
13 z.alpha = qnorm(1-alpha)
14 print(z.alpha)

```

```

15 # computing Beta for hypothesized value
16 Beta_onetailedtest<-pnorm(z.alpha-z)
17 print(Beta_onetailedtest)
18 # power for test
19 powerfortest<-1-Beta_onetailedtest
20 print(powerfortest)

```

R code Exa 5.9 power for test

```

1
2
3 ybar = 31.2           # sample mean
4 mu0 = 33              # hypothesized value
5 sigma = 8.4           # population standard deviation
6 n = 35                # sample size
7 z = (ybar - mu0)/(sigma/sqrt(n))
8 print(z) # test statistic
9
10 # We then compute the critical value at .05
    significance level.
11 # For alpha = .05, we will reject the null
    hypothesis if z <= -1.645
12 alpha = .05
13 z.alpha = qnorm(1-alpha)
14 # the observed value of z is not less than -z.
    alpha, the test statistic does not fall in the
    rejection region.

```

R code Exa 5.10 Suppose that the consumer testing agency thinks

```

1 ybar=33
2 sigma=8.4
3 n=35

```

```

4 alpha = .05
5 # critical value
6 z.alpha = qnorm(1-alpha)
7 z1=function(mu0){
8   z=abs((ybar- mu0)/(sigma/sqrt(n)))
9   b=pnorm(z.alpha-z)
10  return(b)
11 }
12 muo=c(33,32,31,30,29,28,27,26,25)
13 beta=c(z1(33),z1(32),z1(31),z1(30),z1(29),z1(28),z1
        (27),z1(26),z1(25))
14 pwr=1-beta
15 rbind(muo,beta,pwr)

```

R code Exa 5.11 cereal manufacturer produces cereal

```

1 sd<-.225
2 z_foralpha=qnorm(1-0.05)
3 z_forbeta=qnorm(1-0.01)
4 zstar<-z_foralpha+z_forbeta
5 E<-16.37 - 16.27
6 sample_size<- zstar^2 * sd * sd/ E^2
7 print(ceiling(sample_size))
8 # the manufacturer must obtain a random sample of n
   = 80 boxes to conduct this test

```

R code Exa 5.12 research hypothesis validity

```

1
2 ybar = 390           # sample mean
3 mu0 = 380           # hypothesized value
4 sigma = 35.2        # population standard deviation
5 n = 50              # sample size

```

```

6 z = (ybar- mu0)/(sigma/sqrt(n))
7 # test statistic
8
9 p_value=1-pnorm(z)
10 print(p_value)
11 alpha=0.01
12 if(p_value>alpha){
13     print("we fail to reject H0")
14     print(" data do not support the research hypothesis.
15         ")
16 }else{
17     print("reject H0")
18 }

```

R code Exa 5.13 research hypothesis validity

```

1
2 ybar = 31.2           # sample mean
3 mu0 = 33              # hypothesized value
4 sigma = 8.4          # population standard deviation
5 n = 35                # sample size
6 z = (ybar- mu0)/(sigma/sqrt(n))
7 # test statistic
8
9 p_value=pnorm(z)
10 print(p_value)
11 alpha=0.05
12 if(p_value>alpha){
13     print("we fail to reject H0")
14     print(" data do not support the research
15         hypothesis(insufficient evidence).")
16 }else{
17     print("reject H0")
18 }

```

R code Exa 5.14 support the research hypothesis

```
1
2 ybar = 178.2           # sample mean
3 mu0 = 190              # hypothesized value
4 sigma = 45.3           # population standard deviation
5 n = 100                # sample size
6 z = (ybar - mu0)/(sigma/sqrt(n))
7 print(z)
8 k=abs(z)
9 # test statistic
10 # formula based on level of significance
11 p_value=2*(1-pnorm(k))
12 print(p_value)
13 # mentioned p value in book is wrong
14 alpha=0.01
15 if(p_value>alpha){
16   print("we fail to reject H0")
17   print(" data do not support the research
        hypothesis(insufficient evidence).")
18 }else{
19   print(" there is very little evidence in the data
        to support the research hypothesis hence we
        will reject H0")
20 }
```

R code Exa 5.15 research hypothesis validity

```
1 y<-c(.593 ,.142, .329, .691 ,.231 ,.793 ,.519 ,.392,
      .418 )
2 ybar = mean(y)
3 mu0 = 0.3
```

```

4 sigma = sd(y)
5 n = 9 # sample size
6 z = abs((ybar- mu0)/(sigma/sqrt(n)))
7 print(z)
8 p_value=2*(1-pnorm(z))
9 print(p_value)
10 alpha=0.01
11 if(p_value>alpha){
12   print("we fail to reject H0")
13   print(" data do not support the research
        hypothesis(insufficient evidence).")
14 }else{
15   print("reject H0")
16 }

```

R code Exa 5.17 confidence interval

```

1 confidence_interval <- function(vector, interval) {
2   # Standard deviation of sample
3   vec_sd <- sd(vector)
4   # Sample size
5   n <- length(vector)
6   # Mean of sample
7   vec_mean <- mean(vector)
8   # Error according to t distribution
9   error <- qt((interval + 1)/2, df = n - 1) * vec_sd
        / sqrt(n)
10  # Confidence interval as a vector
11  ans <- c("lower" = vec_mean - error, "upper" = vec
        _mean + error)
12  return(ans)
13 }
14 vector <- c(2.7, 2.4, 1.9, 2.6, 2.4, 1.9, 2.3,
15            2.2, 2.5 ,2.3 ,1.8, 2.5, 2.0 ,2.2 )
16 confidence_interval(vector, 0.95)

```

R code Exa 5.18 confidence interval

```
1 confidence_interval <- function(vector, interval) {
2   # Standard deviation of sample
3   vec_sd <- sd(vector)
4   # Sample size
5   n <- length(vector)
6   # Mean of sample
7   vec_mean <- mean(vector)
8   # Error according to t distribution
9   error <- qt((interval + 1)/2, df = n - 1) * vec_sd
10    / sqrt(n)
11   # Confidence interval as a vector
12   ans <- c("lower" = vec_mean - error, "upper" = vec
13     _mean + error)
14   return(ans)
15 }
16 vector <- c( 29, 30, 53, 75, 89, 34, 21, 12, 58, 84,
17   92, 117, 115, 119, 109, 115, 134, 253, 289, 287
18   )
19 confidence_interval(vector, 0.95)
```

R code Exa 5.19 CPS personnel

```
1
2 ybar = 105.75           # sample mean
3 mu0 = 75                # hypothesized value
4 sigma = 82.429          # population standard deviation
5 n = 20                  # sample size
6 # test statistic
7 t = abs((ybar - mu0)/(sigma/sqrt(n)))
```



```

8  print(t)
9
10 # formula based on level of significance
11 m=33
12 B=1000
13 p_value= m/B
14 print(p_value)
15 alpha=0.05
16 # our p value< alpha , therefore
17 print("we conclude that there is sufficient evidence
      that the mean cotanine level exceeds 75 in the
      population of children under CPS supervision")

```

R code Exa 5.20 confidence interval

```

1 x<-c(14.2, 5.3 ,2.9, 4.2,1.2, 4.3, 1.1, 2.6 ,6.7
      ,7.8 ,25.9, 43.8, 2.7,
2      5.6, 7.8 ,3.9, 4.7, 6.5, 29.5 ,2.1 ,34.8 ,3.6
      ,5.8, 4.5, 6.7 )
3 bootmed = apply(matrix(sample(x, rep=TRUE, 10^4*
      length(x)), nrow=10^4), 1, median)
4 # The 95% confidence interval for the population
      median is given by
5 print("Confidence interval is")
6 quantile(bootmed, c(.025, 0.975))

```

R code Exa 5.21 large scale approximation

```

1 # large scale approximation
2 n<-25
3 alpha<-0.05
4 z.half.alpha=qnorm(1-alpha/2)
5 C_alpha2_n=(n/2)-z.half.alpha*sqrt(n/4)

```

```
6 print(C_alpha2_n)
```

R code Exa 5.23 large sample approximation to the sign test

```
1 # Large-Sample Approximation
2 n=25
3 B=13
4 # test statistic
5 Bst=(B-(n/2))/(sqrt(n/4))
6 print(Bst)
7 # critical value
8 alpha=0.05
9 z.alpha=qnorm(1-alpha/2)
10 print(z.alpha)
11 print("e BST is not greater than z.alpha, we fail to
      reject H0")
12 pvalue=1-pnorm(Bst)
13 print(pvalue)
```

Chapter 6

Inferences Comparing Two Population Central Values

R code Exa 6.1 confidence interval for independent sample

```
1 # confidence interval for independent sample
2 fresh=c(10.2, 10.6,10.5 ,10.7,10.3, 10.2,10.8,
          10.0,9.8 ,10.6 )
3 stored=c( 9.8, 9.7,
4           9.6, 9.5,
5           10.1, 9.6,
6           10.2, 9.8,
7           10.1 ,9.9)
8 n1=length(fresh)
9 n2= length(stored)
10 y1bar = mean(fresh)
11 y2bar=mean(stored)
12 s1=sd(fresh)
13 s2=sd(stored)
14 # common standard deviation
15 sp=sqrt(((n1-1)*s1*s1+(n2-1)*s2*s2)/(n1+n2-2))
16
17 # the t-percentile based on df for 95% confidence
    interval
```

```

18 tstar=qt( .975, df=18)
19 margin=tstar*sp*sqrt((1/n1)+(1/n2))
20 left_i=(y1bar-y2bar)-margin
21 right_i=(y1bar-y2bar)+margin
22 print("confidence interval is")
23 print(left_i)
24 print(right_i)

```

R code Exa 6.2 confidence interval for independent sample

```

1 # confidence interval for independent sample
2
3 n1= 10
4 n2= 9
5 y1bar = 8.27
6 y2bar=6.78
7 s1=2.956
8 s2=2.565
9 # common standard deviation
10 sp=sqrt(((n1-1)*s1*s1+(n2-1)*s2*s2)/(n1+n2-2))
11
12 # the t-percentile based on df for 95% confidence
    interval
13 tstar=qt( .975, df=18)
14 margin=tstar*sp*sqrt((1/n1)+(1/n2))
15 left_i=(y1bar-y2bar)-margin
16 right_i=(y1bar-y2bar)+margin
17 print("confidence interval is")
18 print(left_i)
19 print(right_i)

```

R code Exa 6.3 research hypothesis

```

1 # confidence interval for independent sample
2
3 n1= 12
4 n2= 12
5 y1bar = 26.58
6 y2bar=39.67
7 s1=14.36
8 s2=13.86
9 # solving part c
10 # common standard deviation
11 sp=sqrt(((n1-1)*s1*s1+(n2-1)*s2*s2)/(n1+n2-2))
12
13 # the t-percentile based on df for 95% confidence
    interval
14 tstar=qt( .975, df=18)
15 margin=tstar*sp*sqrt((1/n1)+(1/n2))
16 left_i=(y1bar-y2bar)-margin
17 right_i=(y1bar-y2bar)+margin
18 print("confidence interval is")
19 print(left_i)
20 print(right_i)
21
22 # solving part a and b
23 t=(y1bar-y2bar)/((sp)*sqrt((1/n1)+(1/n2)))
24 print(t)
25 # critical value
26 alpha= 0.05
27 df=n1+n2-2
28 t.alpha=qt(0.05, df=22)
29 if(t<=t.alpha){
30     print(" We will reject H0")
31 }else{
32     print("we will fail to reject H0 (no significant
        evidence")
33 }

```

R code Exa 6.4 research hypothesis

```
1 n1= 33
2 n2= 12
3 y1bar = 25.2
4 y2bar=33.9
5 s1=8.6
6 s2=17.4
7
8
9 t=(y1bar-y2bar)/(sqrt((s1*s1/n1)+(s2*s2/n2)))
10 print(t)
11 # To compute the rejection and p-value, we need to
    compute the approximate df
12 c=((s1*s1)/n1)/(((s1*s1)/n1)+((s2*s2)/n2))
13 print(c)
14 df=((n1-1)*(n2-1))/((1-c)^2*(n1-1)+(c*c)*(n2-1))
15 print(df)
16 # crtitical value
17 alpha= 0.05
18
19 t.alpha=qt(0.05, df=13)
20 if(t<=t.alpha){
21   print(" We will reject H0")
22 }else{
23   print("we fail to reject H0 (no significant
        evidence")
24 }
```

R code Exa 6.5 Many states are considering lowering the blood alcohol

```
1 library(DescTools)
```

```

2 x = c(0.90, 0.37, 1.63, 0.83, 0.95, 0.78, 0.86,
        0.61, 0.38, 1.97)
3 y = c(1.46, 1.45, 1.76 ,1.44, 1.11 ,3.07 ,0.98 ,1.27
        ,2.56 ,1.32)
4
5 cbind(c(x,y),rank(c(x,y)))
6
7 a <- wilcox.test(x,y,correct=FALSE,conf.int = TRUE)
8 n1 <- length(x)
9 a$statistic <- a$statistic + n1*(n1+1)/2
10 names(a$statistic) <- "T.W"
11 a
12 # T<83 so we reject H0 and conclude there is
    significant evidence that the placebo population
    has smaller reaction times than the population of
    alcohol consumers
13 # p value calculated in book is wrong
14 # confidence interval for delta (-1.08, -0.25)
15
16 # 95% confidence interval for the placebo
    population median
17 MedianCI(x,conf.level = 0.95,na.rm = FALSE, method =
    "exact",R = 10000)
18 # # 95% confidence interval for the alcohol
    population median
19 MedianCI(y,conf.level = 0.95,na.rm = FALSE, method =
    "exact",R = 10000)

```

R code Exa 6.6 Environmental engineers

```

1 x=c
    (11.0,11.2,11.2,11.2,11.4,11.5,11.6,11.7,11.8,11.9,11.9,12.1,10.2

2 y=rank(x)
3 cbind(x,y)

```

```

4 rep(table(y), table(y))
5 n1=12
6 n2=12
7 mut=n1*(n1+n2+1)/2
8 s=((n1*n2)/12)*((n1+n2+1)-(48/((n1+n2)*(n1+n2-1))))
9 sigmat=sqrt(s)
10 T=216 # sum of ranks of before clean up values
11 Z=(T-mut)/sigmat
12 Z
13 # This value exceeds 1.645, so we reject H0 and
    conclude that the distribution of before-cleanup
    measurements is shifted to the right of the
    corresponding distribution of after-cleanup
    measurements
14 # part b
15 si=(n1*n2*(n1+n2+1))/n1
16 sqrt(si)

```

R code Exa 6.7 Insurance adjusters are concerned about the high

```

1 garage1=c
    (17.6,20.2,19.5,11.3,13.0,16.3,15.3,16.2,12.2,14.8,21.3,22.1,16.9
2 garage2=c
    (17.3,19.1,18.4,11.5,12.7,15.8,14.9,15.3,12.0,14.2,21.0,21.0,16.1
3 t.test(garage1,garage2)

```

R code Exa 6.8 perform a paired t test

```

1 garage1=c
    (17.6,20.2,19.5,11.3,13.0,16.3,15.3,16.2,12.2,14.8,21.3,22.1,16.9

```



```

2 garage2=c
  (17.3,19.1,18.4,11.5,12.7,15.8,14.9,15.3,12.0,14.2,21.0,21.0,16.1

3 t.test(garage1,garage2,paired = TRUE)
4 tvalue=qt(1-0.05,14)
5 # t>tvale we reject H0 and conclude that mean
  repair estimate for garage I is greater than that
  for garage II

```

R code Exa 6.9 A city park department compared a new formulation

```

1 library(DescTools)
2 brandA=c
  (211.4,204.4,202.0,201.9,202.4,202.0,202.4,207.1,203.6,216.0,208.0

3 brandB=c
  (186.3,205.7,184.4,203.6,180.4,202.0,181.5,186.7,205.7,189.1,183.0

4 difference=brandA-brandB
5
6 y=rank(replace(abs(difference),abs(difference)==0,NA
  ),na='keep');
7 cbind(difference,y)
8 # sum of positive and negative ranks are
9 Tminus=1+2+3+4+5+6
10 Tplus= 7 + 8 + 9 + 10 + 11 + 12 + 13 + 14 +15 + 16
  +17.5 +18 + 19
11 T=min(Tminus,Tplus)
12 T
13 # T<53, we reject H0 and conclude that brand A
  fertilizer tends to produce more grass than brand
  B
14 difference=difference[-6]
15
16

```

```
17 MedianCI(difference, conf.level = 0.95, na.rm = FALSE,
    method = "exact", R = 999)
```

R code Exa 6.10 one sided test

```
1 # one sided test
2 sigma=2.4
3 delta=1.5
4 z_alpha=qnorm(0.05)
5 z_beta=qnorm(0.10)
6 sample_size=2*(sigma^2)*((z_alpha+z_beta)^2)/(delta
    ^2)
7 print(sample_size)
```

R code Exa 6.11 Sample Size

```
1
2 sigma=2.4
3 delta=1.5
4 z_alpha=qnorm(0.05)
5 z_beta=qnorm(0.10)
6 m=3
7 # replace 2 with (m+1)/m i.e 4/3
8 sample_size=(4/3)*(sigma^2)*((z_alpha+z_beta)^2)/(
    delta^2)
9 print(sample_size)
```

Chapter 7

Inferences about Population Variances

R code Exa 7.1 The 99 confidence interval for mean

```
1 weights=c(501.4, 498.0, 498.6 ,499.2, 495.2 ,501.4
            ,509.5 ,494.9 ,498.6, 497.6,
2           505.5 ,505.1 ,499.8 ,502.4, 497.0 ,504.3
            ,499.7 ,497.9 ,496.5, 498.9,
3           504.9 ,503.2 ,503.0 ,502.6 ,496.8 ,498.2,
            500.1 ,497.9 ,502.2, 503.2)
4 n=length(weights)
5 ybar=mean(weights)
6 s=sd(weights)
7 # The upper-tail chi-square value
8 XU=qchisq(.995, df=29)
9 # The lower-tail chi-square value
10 XL=qchisq(.005, df=29)
11 # The 99% confidence interval for standard deviation
12 right_i=sqrt((n-1)*(s^2)/(XL))
13 left_i=sqrt((n-1)*(s^2)/(XU))
14 print(left_i)
15 print(right_i)
16 # The 99% confidence interval for mean
```

```

17 margin <- qnorm(0.995)*s/sqrt(n)
18 left_interval_mean=ybar-margin
19 right_interval_mean=ybar+margin
20 print(left_interval_mean)
21 print(right_interval_mean)

```

R code Exa 7.2 The 95 confidence interval for standard deviation

```

1 readings=c(203.1, 184.5, 206.8 ,211.0 ,218.3, 174.2,
            193.2 ,201.9 ,199.9 ,194.3,
2           199.4, 193.6, 194.6 ,187.2 ,197.8 ,184.3,
            196.1, 196.4 ,197.5 ,187.9)
3 n=length(readings)
4 ybar=mean(readings)
5 s=sd(readings)
6 muo=5
7 # test static
8 X=(n-1)*(s^2)/(muo^2)
9 print(X)
10 #critical value
11 alpha=0.05
12 X.alpha=qchisq(1-0.05,df=19)
13
14 # the null hypothesis, H0 is rejected if the value
    of the X is greater than X.alpha
15 #Since the computed value of the X., 74.61, is
    greater than the
16 # critical value 30.14, there is sufficient evidence
    to reject H0
17 # The upper-tail chi-square value
18 XU=qchisq(.975, df=19)
19 # The lower-tail chi-square value
20 XL=qchisq(1-.975, df=19)
21 # The 95% confidence interval for standard deviation
22 right_i=sqrt((n-1)*(s^2)/(XL))

```

```

23 left_i=sqrt((n-1)*(s^2)/(XU))
24 print(left_i)
25 print(right_i)

```

R code Exa 7.4 the upper percentile for the F distribution

```

1 #the upper .025 percentile for the F distribution
  with df1 = 10 and df2 =7 is
2
3 upper_percentile=qf(1-0.025,10,7)
4 lower_percentile=1/upper_percentile
5 print(lower_percentile)

```

R code Exa 7.5 test the equality of the population variances

```

1
2 y1bar=38.48
3 s1=16.37
4 n1=40
5 y2bar=26.93
6 s2=9.88
7 n2=40
8 # test statistic
9 F=s1^2/s2^2
10 print(F)
11 #critical value
12 alpha=0.05
13 f.alpha=qf(1-alpha/2,39,39)
14 # we reject H0 if F>=f.alpha

```

R code Exa 7.6 confidence interval

```
1 # confidence interval for the ratio of the two
  variances
2
3 y1bar=38.48
4 s1=16.37
5 n1=40
6 y2bar=26.93
7 s2=9.88
8 n2=40
9 alpha=0.05
10 FU=qf(1-alpha/2,39,39)
11 FL=1/FU
12
13 # confidence interval for  $\sigma_1^2/\sigma_2^2$ 
14 left_i=(s1^2/s2^2)*FL
15 right_i=(s1^2/s2^2)*FU
16 print(left_i)
17 print(right_i)
```

R code Exa 7.7 confidence interval

```
1
2 y1bar=20.04
3 s1=0.474
4 n1=10
5 y2bar=9.99
6 s2=0.233
7 n2=16
8 alpha=0.10
9 FU=qf(1-alpha/2,15,9)
10 FL=1/FU
11 # confidence interval for  $\sigma_1/\sigma_2$ 
12 left_i=sqrt((s1^2/s2^2)*FL)
```

```

13 right_i=sqrt((s1^2/s2^2)*FU)
14 print(left_i)
15 print(right_i)

```

R code Exa 7.8 comparing the variability in power

```

1 s=c(8.69, 6.89, 80.22)
2 s_min=min(s)
3 s_max=max(s)
4 #test statistic
5 F=s_max/s_min
6 print(F)
7
8 # critical value
9 alpha=0.05
10 df=8
11 F.alpha=qnorm(alpha/2,8)
12 print(F.alpha)
13
14 # Reject H0 if F >=F.alpha
15 # conclusion : Thus, we reject H0 and conclude that
    the variances are not all equa

```

R code Exa 7.9 Three different additives that are marketed

```

1 # install car package by writing install.packages("
    car") command in console
2 library(car)
3 y=c
    (4.2,2.9,0.2,25.7,6.3,7.2,2.3,9.9,5.3,6.5,0.2,11.3,0.3,17.1,51.0,

```

```
4 additive=c
  (1,1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3,3,3)

5 leveneTest(y,additive)
6 critical_value=qf(1-0.05,2,27)
7 critical_value
8 # L < critical value we fail to reject and conclude
  that there is insufficient evidence of a
  difference in the population variances of the
  percentage increase in mpg for the three
  additives.
```

Chapter 8

Inferences about More Than Two Population Central Values

R code Exa 8.1 A large body of evidence

```
1
2 Group1 <- c(5,17,12,10,4)
3 Group2 <- c(19,10,9,7,5)
4 Group3 <- c(25,15,12,9,8)
5
6 Combined_Groups <- data.frame(cbind(Group1, Group2,
   Group3)) # combines the data into a single data
   set.
7 Combined_Groups # shows spreadsheet like results
8 #summary(Combined_Groups) # min, median, mean, max
9
10 Stacked_Groups <- stack(Combined_Groups)
11 Stacked_Groups #shows the table Stacked_Groups
12
13 Anova_Results <- aov(values ~ ind, data = Stacked_
   Groups)
14 summary(Anova_Results) # shows Anova_Results
15
16
```

```
17 # answer given in book is wrong because sample
    varaince calcaulated for group 1 column in book
    is 33.7 which is wrong
18 # correct sample varaince is 28.3
```

R code Exa 8.2 clinical psychologist

```
1
2 Group1 <- c(96,79,91,85,83,91,82,87)
3 Group2 <- c(77,76,74,73,78,71,80)
4 Group3 <- c(66,73,69,66,77,73,71,70,74)
5
6
7 cols <- list(m=Group1, y=Group2,z=Group3)
8 as.data.frame(lapply(cols, 'length<-', max(sapply(
    cols, length))))
9 cols
10 Stacked_Groups <- stack(cols)
11 Stacked_Groups #shows the table Stacked_Groups
12
13 Anova_Results <- aov(values ~ ind, data = Stacked_
    Groups)
14 summary(Anova_Results) # shows Anova_Results
```

R code Exa 8.3 clerics knowledge of mental illness

```
1 M=c(62,60,60,25,24,23,20,13,12,6)
2 C=c(62,62,24,24,22,20,19,10,8,8)
3 P=c(37,31,15,15,14,14,14,5,3,2)
4
5 Group1=abs(M-median(M))
6 Group2=abs(C-median(C))
7 Group3=abs(P-median(P))
```

```

8
9 Combined_Groups <- data.frame(cbind(Group1, Group2,
    Group3)) # combines the data into a single data
    set.
10 Combined_Groups # shows spreadsheet like results
11 #summary(Combined_Groups) # min, median, mean, max
12
13 Stacked_Groups <- stack(Combined_Groups)
14 Stacked_Groups #shows the table Stacked_Groups
15
16 Anova_Results <- aov(values ~ ind, data = Stacked_
    Groups)
17 summary(Anova_Results) # shows Anova_Results

```

R code Exa 8.4 dissolved oxygen contents at four distances from mouth

```

1 Mean=c(2.2, 4.6 ,21.2 ,31.4)
2 Standard_deviation=c(1.476,2.119,4.733,5.52)
3 s_min=min(Standard_deviation)
4 s_max=max(Standard_deviation)
5 # test statistic
6 F=s_max^2/s_min^2
7 print(F)
8 # The critical value of F > F.alpha
9 # we reject the hypothesis of homogeneity
10 #of the population variances.
11
12 distance_1km=c(1,5,2,1,2,2,4,3,0,2)
13 distance_5km=c(4,8,2,3,8,5,6,4,3,3)
14 distance_10km=c(20,26,24,11,28,20,19,19,21,24)
15 distance_20km=c(37,30,26,24,41,25,36,31,31,33)
16 print(Standard_deviation[1]^2/Mean[1])
17 print(Standard_deviation[2]^2/Mean[2])
18 print(Standard_deviation[3]^2/Mean[3])
19 print(Standard_deviation[4]^2/Mean[4])

```

```

20 i=1
21 while(i<11){
22     distance_1km[i]=sqrt(distance_1km[i]+0.375)
23     i=i+1
24 }
25 i=1
26 while(i<11){
27     distance_5km[i]=sqrt(distance_5km[i]+0.375)
28     i=i+1
29 }
30 i=1
31 while(i<11){
32     distance_10km[i]=sqrt(distance_10km[i]+0.375)
33     i=i+1
34 }
35 i=1
36 while(i<11){
37     distance_20km[i]=sqrt(distance_20km[i]+0.375)
38     i=i+1
39 }
40 combined_group=data.frame(cbind(distance_1km,
    distance_5km,distance_10km,distance_20km))
41 combined_group

```

R code Exa 8.7 rank sum test

```

1 # the rank sum test
2 Methodist=c(62,60,60,25,24,23,20,13,12,6)
3 Catholic=c( 62,62,24,24,22,20,19,10,8,8)
4 Pentecostal=c(37,31,15,15,14,14,14,5,3,2 )
5 n=30
6 data.value <- c( Methodist,Catholic,Pentecostal)
7
8 data.rank <- rank(data.value)
9 data <- data.frame(data.value, data.rank)

```

```
10 print(data)
11 Sumofranks=c(182.5,167.5,115)
12 #test statistic
13 H=(12/(n*(n+1)))*((Sumofranks[1]^2+Sumofranks[2]^2+
    Sumofranks[3]^2)/10)-3*(n+1)
14 print(H)
```

Chapter 9

Multiple Comparisons

R code Exa 9.1 contrasts orthogonal

```
1  sample_size=c(5,4,6,5)
2  #l1=y1bar-y3bar
3  #l2=y2bar-y4bar
4  # We can rewrite the contrasts in the following form
   :
5  #l1=y1bar+0*y2bar-y3bar+0*y4bar
6  #l2=0*y1bar+y2bar+0*y3bar-y4bar
7  # thus we identify a1 = 1, a2 = 0, a3 =-1, a4 = 0
   and b1 =0, b2 = 1, b3 =0, b4 =-1
8  a=c(1,0,-1,0)
9  b=c(0,1,0,-1)
10 test=0
11 i=1
12 while(i<=length(sample_size)){
13   test=test+(a[i]*b[1])/sample_size[i]
14   i=i+1
15 }
16 print(test)
17 if(test==0){
18   print("hence the contrasts are orthogonal.")
19 }else{
```

```

20   print("hence the contrasts are not orthogonal")
21 }

```

R code Exa 9.2 contrasts orthogonal

```

1  sample_size=c(5,4,6,5)
2  #l1=y1bar-y3bar
3  #l2=y1bar+y2bar+y3bar-3*y4bar
4  # We can rewrite the contrasts in the following form
   :
5  #l1=y1bar+0*y2bar-y3bar+0*y4bar
6  #l2=l2=y1bar+y2bar+y3bar-3*y4bar
7  # thus we identify a1 = 1, a2 = 0, a3 = -1, a4 = 0
   and b1 = 0, b2 = 1, b3 = 0, b4 = -1
8  a=c(1,0,-1,0)
9  b=c(1,1,1,-3)
10 test=0
11 i=1
12 while(i<=length(sample_size)){
13   test=test+(a[i]*b[1])/sample_size[i]
14   i=i+1
15 }
16 print(test)
17 if(test==0){
18   print("hence the contrasts are orthogonal.")
19 }else{
20   print("hence the contrasts are not orthogonal")
21 }
22 # part b
23 sample_size=5
24 a=c(1,0,-1,0)
25 b=c(1,1,1,-3)
26 test=0
27 i=1
28 while(i<=4){

```

```

29     test=test+(a[i]*b[1])/sample_size
30     i=i+1
31 }
32 print(test)
33 if(test==0){
34     print("hence the contrasts are orthogonal.")
35 }else{
36     print("hence the contrasts are not orthogonal")
37 }

```

R code Exa 9.3 control weeds in crops

```

1  sample_size=c(6,6,6,6,6)
2
3  # l=y1bar-(y2bar+y3bar+y4bar+y5bar)/4
4
5  # thus we identify a1 = 4, a2 = -1, a3 = -1, a4 = -1
   , a5=-1
6  a=c(4,-1,-1,-1,-1)
7
8  test=0
9  i=1
10 while(i<=length(sample_size)){
11     test=test+a[i]^2
12     i=i+1
13 }
14 print(test)
15 y1bar=1.175
16 y2bar=1.293
17 y3bar=1.328
18 y4bar=1.415
19 y5bar=1.500
20 l=4*y1bar-y2bar-y3bar-y4bar-y5bar
21 print(l)
22 # we can obtain the sum of squares associated with

```



```

    the contrast from
23 SSC1=(sample_size[1]*(1^2))/test
24 print(SSC1)
25
26 a=c(0,1,1,-1,-1)
27 test=0
28 i=1
29 while(i<=length(sample_size)){
30     test=test+a[i]^2
31     i=i+1
32 }
33 l=0*y1bar+y2bar+y3bar-y4bar-y5bar
34 # we can obtain the sum of squares associated with
    the contrast from
35 SSC2=(sample_size[2]*(1^2))/test
36 print(SSC2)
37
38 a=c(0,1,-1,0,0)
39 test=0
40 i=1
41 while(i<=length(sample_size)){
42     test=test+a[i]^2
43     i=i+1
44 }
45 l=0*y1bar+y2bar-y3bar+0*y4bar+0*y5bar
46 # we can obtain the sum of squares associated with
    the contrast from
47 SSC3=(sample_size[2]*(1^2))/test
48 print(SSC3)
49
50 a=c(0,0,0,1,-1)
51 test=0
52 i=1
53 while(i<=length(sample_size)){
54     test=test+a[i]^2
55     i=i+1
56 }
57 l=0*y1bar+0*y2bar+0*y3bar+y4bar-y5bar

```

```

58 # we can obtain the sum of squares associated with
    the contrast from
59 SSC4=(sample_size[2]*(1^2))/test
60 print(SSC4)

```

R code Exa 9.5 Test each of the four contrasts for significance

```

1  si=c(.1204,.1269,.1196,.1249,.1265)
2  # data fom example 9.3
3  Fmax=max(si)^2/min(si)^2
4  print(Fmax)
5  # four test statistic
6  SSC1=.2097 # these value are computed in 9.3
7  SSC2=.1297
8  SSC3=.0037
9  SSC4=.0217
10 MSEError=.0153
11 F1=SSC1/MSEError
12 F2=SSC2/MSEError
13 F3=SSC3/MSEError
14 F4=SSC4/MSEError
15
16 print(F1)
17 print(F2)
18 print(F3)
19 print(F4)
20
21 alpha=0.05
22 df1=1
23 df2=25
24 F_0.05_1_25=qf(1-alpha,df1,df2)
25 print(F_0.05_1_25)
26
27 # we conclude that contrasts l1 and l2 were
    significantly

```

```
28 #different from zero but contrasts l3 and l4 were
    not significantly different from zero.
```

R code Exa 9.6 control the experimentwise error rate

```
1 alpha=0.05
2 m=4 #comparisons
3 alpha_l=alpha/m
4 F_aplha_l_1_25=qf(1-alpha_l,df1 = 1,df2 = 25)
5 print(F_aplha_l_1_25)
6 # We would then reject H0 if SSCi/MSError >=F_alpha_
  l_1_25
7 F1 = 13.71 # computed in 9.5
8 F2 = 8.48
9 F3 = 0.24
10 F4 = 1.42
11 # we would declare contrast l1 and l2 significantly
    different from 0 because their F ratios are
    greater than 7.24.
```

R code Exa 9.7 five different weed agents

```
1 F=5.96 # computed from 9.3
2 alpha=0.05
3 MSError=.0153
4 F_value=qf(1-alpha,df1=4,df2 = 25)
5 # as the F >F_value
6 print(" we reject H0 and conclude that at least one
    of the population means differs from the rest")
7 t.alpha=qt(1-alpha/2,df=25)
8 LSD=t.alpha*(sqrt((2*MSError)/6))
9 print(LSD)
```

R code Exa 9.9 confidence interval for mean

```
1 y1bar=1.175
2 y2bar=1.293
3 y3bar=1.328
4 y4bar=1.415
5 y5bar=1.500
6
7 alpha=0.05
8 tstar=qt(1-alpha/2,df=25)
9 MSError=0.0153
10 LSD=tstar*sqrt((2*MSError)/6)
11 print(LSD)
12
13 # the 95% confidence interval for y3bar-y1bar
14 left_i=(y3bar-y1bar)-LSD
15 right_i=(y3bar-y1bar)+LSD
16 print("Confidence interval is")
17 print(left_i)
18 print(right_i)
```

R code Exa 9.11 confidence interval

```
1 y1bar=1.175
2 y2bar=1.293
3 y3bar=1.328
4 y4bar=1.415
5 y5bar=1.500
6
7 alpha=0.05
8 q.alpha=4.158
9 MSError=0.0153
```

```

10 LSD=q.alpha*sqrt((MSError)/6)
11 print(LSD)
12
13 # the 95% confidence interval for y3bar-y1bar
14 left_i=(y3bar-y1bar)-LSD
15 right_i=(y3bar-y1bar)+LSD
16 print("Confidence interval is")
17 print(left_i)
18 print(right_i)

```

R code Exa 9.13 Compare the two biological treatments

```

1 # install package by writing install.packages("
  DunnettTests") command in console
2 # install package by writing install.packages(
  mvtnorm") command in console
3 library(DunnettTests)
4 library(mvtnorm)
5 alpha=0.05
6 k=4
7 v=25
8 n=6
9
10 cvSUDT(k=4,alpha=0.05,alternative="U",df = 25,corr
  = .5)
11 # max value of critical value is taken
12 critical_value=2.28 # approx
13 sw2=0.0153
14 # test statistic
15 D=critical_value*sqrt((2*sw2)/n)
16 print(D)
17 # conclusion
18 yi=c(1.293,1.328,1.415,1.500)
19 yc=1.175
20 i=1

```

```

21 while (i<5) {
22     if((yi[i]-yc)<D){
23         print("Not greater than control")
24     }
25     else{
26         print("greater than control")
27     }
28     i=i+1
29 }

```

R code Exa 9.14 Scheffs procedure to determine

```

1  sample_size=6
2  sw2=0.0153
3  y1bar=1.175
4  y2bar=1.293
5  y3bar=1.328
6  y4bar=1.415
7  y5bar=1.500
8  t=5
9  alpha=0.05
10 F_value=qf(1-alpha,t-1,25)
11
12 a=c(4,-1,-1,-1,-1)# for control vs agents
13 # test statistic
14 v1=sw2*((a[1]^2+a[2]^2+a[3]^2+a[4]^2+a[5]^2)/sample_
    size)
15 print(v1)
16 S=sqrt(v1)*sqrt((t-1)*F_value)
17 print(S)
18 # critcal value
19 l=abs(4*y1bar-y2bar-y3bar-y4bar-y5bar)
20 print(l)
21 if(S<l){
22     print(" contrasts are significantly different

```

```

        from zero")
23
24 }else{
25     print(" contrasts are not significantly different
        from zero")
26 }
27
28 a=c(0,1,1,-1,-1)# Biological vs. chemical
29 # test statistic
30 vl=sw2*((a[1]^2+a[2]^2+a[3]^2+a[4]^2+a[5]^2)/sample_
    size)
31 print(vl)
32 S=sqrt(vl)*sqrt((t-1)*F_value)
33 print(S)
34 # critical value
35 l=abs(0*y1bar+y2bar+y3bar-y4bar-y5bar)
36 print(l)
37 if(S<l){
38     print(" contrasts are significantly different
        from zero")
39
40 }else{
41     print(" contrasts are not significantly different
        from zero")
42 }
43
44 a=c(0,1,-1,0,0)# Bio1 vs. Bio2
45 # test statistic
46 vl=sw2*((a[1]^2+a[2]^2+a[3]^2+a[4]^2+a[5]^2)/sample_
    size)
47 print(vl)
48 S=sqrt(vl)*sqrt((t-1)*F_value)
49 print(S)
50 # critical value
51 l=abs(0*y1bar+y2bar-y3bar+0*y4bar+0*y5bar)
52 print(l)
53 if(S<l){
54     print(" contrasts are significantly different

```

```

        from zero")
55
56 }else{
57     print(" contrasts are not significantly different
        from zero")
58 }
59
60 a=c(0,0,0,1,-1)# Chm1 vs. Chm2
61 # test statistic
62 vl=sw2*((a[1]^2+a[2]^2+a[3]^2+a[4]^2+a[5]^2)/sample_
    size)
63 print(vl)
64 S=sqrt(vl)*sqrt((t-1)*F_value)
65 print(S)
66 # critical value
67 l=abs(0*y1bar+0*y2bar+0*y3bar+y4bar-y5bar)
68 print(l)
69 if(S<l){
70     print(" contrasts are significantly different
        from zero")
71
72 }else{
73     print(" contrasts are not significantly different
        from zero")
74 }

```

Chapter 10

Categorical Data

R code Exa 10.1 Estimate the proportion of all patients with the specified type of cancer

```
1 # y denote the number of successes in the n sample
   trials ,
2 # sample proportion
3 y=330
4 n=870
5 pie=y/n
6 sigma=sqrt((pie*(1-pie))/n)
7
8 alpha=0.05
9 z.alpha=qnorm(1-alpha)
10 error=z.alpha*sigma
11 # the 90% confidence interval on the proportion of
   cancer
12 #patients who will survive at least 5 years
13 left_i=pie-error
14 right_i=pie+error
15 print(left_i)
16 print(right_i)
```

R code Exa 10.2 water department

```
1 # y denote the number of successes in the n sample
   trials ,
2 # sample proportion
3 y=43
4 n=50
5 pie=y/n
6 sigma=sqrt((pie*(1-pie))/n)
7 alpha=0.025
8 z.alpha=qnorm(1-alpha)
9 error=z.alpha*sigma
10 # 95% confidence interval
11 left_i=pie-error
12 right_i=pie+error
13 print(" Wald  95 % confidence interval")
14 print(left_i)
15 print(right_i)
16
17 # Using the WAC confidence interval, we need to
   compute:
18 ybar = y + 0.5*(z.alpha^2)
19 nbar = n + (z.alpha^2)
20 pie_bar=ybar/nbar
21
22
23 sigma_bar=sqrt((pie_bar*(1-pie_bar))/nbar)
24 error_bar=z.alpha*sigma_bar
25 left=pie_bar-error_bar
26 right=pie_bar+error_bar
27 print(" WAC 95% confidence interval")
28 print(left)
29 print(right)
```

R code Exa 10.3 confidence interval for pie would be

```
1 y=50
2 n=50
3 alpha=.05
4 # If we used the standard estimator of pie
5 pie=y/n
6 # The point estimator would be given by
7 pie_adj=(n+(3/8))/(n+(3/4))
8 print(pie_adj)
9 # A 95% confidence interval for pie would be
10 left_i=(alpha/2)^(1/n)
11 right_i=1
12 print(left_i)
13 print(right_i)
```

R code Exa 10.4 designer of the new operating system

```
1 alpha=0.025
2 pie=0.5
3 E=0.03
4 z.alpha=qnorm(1-alpha)
5 # sample size necessary to achieve this accuracy
6 n=((z.alpha^2)*pie*(1-pie))/E^2
7 print(ceiling(n))
8 # 1,068 programs would need to be tested in order to
   be 95% confident that
9 #the estimate of pie is within .03 of the actual
   value of pie
10
11 pie=0.8
12 n=((z.alpha^2)*pie*(1-pie))/E^2
```

```

13 print(n)
14 # if the designer was fairly certain that the
    actual value of pie was at least .80,
15 #then the required sample size can be greatly
    reduced.

```

R code Exa 10.5 percentage of binge drinkers at the university

```

1 y=1200
2 n=2500
3 pie=y/n
4 sigma=sqrt((pie*(1-pie))/n)
5 pie0=0.44
6 #test statistic
7 z=(pie-pie0)/sigma
8 print(z)
9 # critical value
10 alpha=0.05
11 z.alpha=qnorm(1-alpha)
12 #Because the observed value of z exceeds the
    critical value 1.645, we conclude that the
13 #percentage of students that participate in binge
    drinking exceeds the national percentage of 44%
14 nbar = n + (z.alpha^2)
15 pie_bar=(y+z.alpha)/nbar
16
17 sigma_bar=sqrt((pie_bar*(1-pie_bar))/nbar)
18 error_bar=z.alpha*sigma_bar
19 left=pie_bar-error_bar
20 right=pie_bar+error_bar
21 print(left)
22 print(right)
23 # the percentage of binge drinkers at the university
    is, with 95% confidence, between 46% and 50%

```

R code Exa 10.6 confidence interval

```
1 pie1=413/527
2 pie2=392/608
3 # The sample awareness proportion is higher in
  Wichita, so let's make Wichita region 1.
4 #The estimated standard error is
5 sigma=sqrt(((pie1*(1-pie1))/527)+((pie2*(1-pie2))/
  608))
6 print(sigma)
7 alpha=0.025
8 z.alpha=qnorm(1-alpha)
9 error=z.alpha*sigma
10 # 95% confidence interval
11 left_i=(pie1-pie2)-error
12 right_i=(pie1-pie2)+error
13 print(left_i)
14 print(right_i)
```

R code Exa 10.7 confidence interval

```
1 pie1=94/125
2 pie2=113/175
3 # The sample awareness proportion is higher in
  Wichita, so let's make Wichita region 1.
4 #The estimated standard error is
5 sigma=sqrt(((pie1*(1-pie1))/125)+((pie2*(1-pie2))/
  175))
6
7 #test statistic
8 z=(pie1-pie2)/sigma
9 print(z)
```

```

10 alpha=0.05
11 z.alpha=qnorm(1-alpha)
12 zstar=qnorm(1-alpha/2)
13 # Since z is greater than z.alpha, we reject H0
    and conclude that the observations
14 #support the hypothesis
15 error=zstar*sigma
16 # 95% confidence interval
17 left_i=(pie1-pie2)-error
18 right_i=(pie1-pie2)+error
19 print(left_i)
20 print(right_i)

```

R code Exa 10.8 clinical trial is conducted to compare two drug therapies

```

1
2 data = rbind(c(38,4), c(14,7) )
3
4 print(data)
5 # fisher test
6 fisher.test(data, alternative="greater")
7 alpha=0.025
8 # as pvalue > alpha then we conclude that there is
    not
9 #significant evidence that the proportion of
    patients obtaining a successful outcome
10 #is higher for drug PV than for drug P.

```

R code Exa 10.9 Previous experience with the breeding of a particular herd of cattle

```

1 p_1calf=0.83
2 p_2calf=0.02

```

```

3 p_0calf=0.15
4
5 # event A= dams gives birth to no healthy progeny
  ,1 healthy progeny,2 healthy progeny(n1=1,n2=1,n3
    =1)
6
7 P_eventA=(factorial(3)*(p_0calf^1*p_1calf^1*p_2calf
  ^1))/(factorial(1)*factorial(1)*factorial(1))
8 print(P_eventA)
9 # event B= 3 dams give birth to 1 healthy progeny(
  n1=0,n2=3,n3=0)
10 P_eventB=(factorial(3)*(p_0calf^0*p_1calf^3*p_2calf
  ^0))/(factorial(0)*factorial(3)*factorial(0))
11 print(P_eventB)
12 # the probability of obtaining exactly three
  healthy progeny from three dams
13 p=P_eventA+P_eventB
14 print(p)

```

R code Exa 10.10 null hypothesis

```

1 observed_cell_number = c(120,60,10,10)
2 expected_cell_number=c(100,50,20,30)
3 # test statistic
4 Xsquare = 0
5 i=1
6 while(i<=length(observed_cell_number)){
7   Xsquare=Xsquare+(((observed_cell_number[i]-
    expected_cell_number[i])^2)/expected_cell_
    number[i])
8   i=i+1
9 }
10 print(Xsquare)
11 # critical value
12 alpha = 0.05

```

```

13 X.alpha=qchisq(1-alpha,df=3)
14 # The computed value of xsquare is greater than x.
    alpha, so we reject the null hypotheses

```

R code Exa 10.11 Environmental engineers

```

1 yi=c(0,1,2,3,4,5,6,7)
2 ni=c(6,23,29,31,27,13,8,13)
3 n=sum(ni)
4 ybar=sum(yi*ni)/sum(ni)
5 print(ybar)
6 # ybar value in book is calculated wrong
7 # The Poisson probabilities
8 Pyi=c(dpois(0,ybar),dpois(1,ybar),dpois(2,ybar),
        dpois(3,ybar),dpois(4,ybar),dpois(5,ybar),dpois
        (6,ybar),dpois(7,ybar))
9 print(Pyi)
10 # Expected cell count
11 Ei=n*Pyi
12 print(Ei)
13
14 # test statistic
15 i=1
16 X2=0
17 while(i<9){
18     X2=X2+(((ni[i]-Ei[i])^2)/Ei[i])
19     i=i+1
20 }
21 print(X2) # ans in book is calculate wrong
22 df=6
23
24 pvalue=pchisq(X2,df,lower.tail = FALSE)
25 print(pvalue)
26 # as p-value <=.01 model is Poisson model provides
    an Unacceptable fit to data

```

R code Exa 10.12 random sample of 216 patients

```
1 moderate=c(15,32,18,5)
2 mildly=c(8,29,23,18)
3 severe=c(1,20,25,22)
4 all_ages=c(sum(moderate),sum(mildly),sum(severe))
5 all_servetiles=c(24,81,66,45)
6 grand_total=216
7 # For row 1,the estimated expected number of
  occurrences
8 E11=(sum(moderate)*all_servetiles[1])/grand_total
9 print(E11)
10 E12=(sum(moderate)*all_servetiles[2])/grand_total
11 print(E12)
12 E13=(sum(moderate)*all_servetiles[3])/grand_total
13 print(E13)
14 E14=(sum(moderate)*all_servetiles[4])/grand_total
15 print(E14)
16
17 # For row 2,the estimated expected number of
  occurrences
18 E21=(sum(mildly)*all_servetiles[1])/grand_total
19 print(E21)
20 E22=(sum(mildly)*all_servetiles[2])/grand_total
21 print(E22)
22 E23=(sum(mildly)*all_servetiles[3])/grand_total
23 print(E23)
24 E24=(sum(mildly)*all_servetiles[4])/grand_total
25 print(E24)
26
27 # For row 3,the estimated expected number of
  occurrences
28 E31=(sum(severe)*all_servetiles[1])/grand_total
29 print(E31)
```

```

30 E32=(sum(severe)*all_servetiles[2])/grand_total
31 print(E32)
32 E33=(sum(severe)*all_servetiles[3])/grand_total
33 print(E33)
34 E34=(sum(severe)*all_servetiles[4])/grand_total
35 print(E34)
36
37 dt=data.frame(cbind(E11,E12,E13,E14),cbind(E21,E22,
      E23,E24),cbind(E31,E32,E33,E34))
38 dt

```

R code Exa 10.13 test to determine the severity of the disease

```

1 n = c(15,32,18,5,8,29,23,18,1,20,25,22)
2 E=c(
      7.78,26.25,21.39,14.58,8.67,29.25,23.83,16.25,7.56,25.50,20.78,14

3 # test statistic
4 Xsquare = 0
5 i=1
6 while(i<=length( n)){
7   Xsquare=Xsquare+(((n[i]-E[i])^2)/E[i])
8   i=i+1
9 }
10 print(Xsquare)
11 # critical value
12 alpha = 0.05
13 X.alpha=qchisq(1-alpha,df=6)
14 # The computed value of xsquare is greater than x.
    alpha, so we reject the null hypotheses

```

R code Exa 10.14 test of homogeneity of distributions

```

1 n = c(50,59,161,88,20,40,56,52,188,4,3,5,2,66,6)
2 E=c
    (67.5,67.5,135,37,37,74,74,74,148,3,3,6,18.5,18.5,37)

3 # test statistic
4 Xsquare = 0
5 i=1
6 while(i<=length( n)){
7   Xsquare=Xsquare+(((n[i]-E[i])^2)/E[i])
8   i=i+1
9 }
10 print(Xsquare)
11 # critical value
12 alpha = 0.001
13 X.alpha=qchisq(1-alpha,df=8)
14 # The computed value of xsquare is greater than x.
    alpha, so we reject the null hypotheses

```

R code Exa 10.15 Consider both a population in which

```

1 # odds of a randomly chosen person carrying HIV
2 p=.001/.009
3 # occurrence of a positive test result causes the
    odds to change to
4 p_hiv=.001
5 p_positive_hiv=.95
6 p_nothiv=.999
7 p_positive_nothiv=.02
8 P_occurrencepositive=(p_hiv*p_positive_hiv)/(p_nothiv
    *p_positive_nothiv)
9 print(P_occurrencepositive)
10
11 # The odds of carrying HIV do go up given a positive
    test result, from about
12 #.001 (to 1) to about .0475 (to 1).

```

R code Exa 10.16 level of stress

```
1 low=c(250,750)
2 high=c(400,1600)
3 odds_ratio=(250/750)/(400/1600)
4 print(odds_ratio)
5 # We will next compute a 95% confidence interval for
  the odds ratio
6 a= log(odds_ratio)
7 sigma=sqrt(1/low[1]+1/low[2]+1/high[1]+1/high[2])
8 print(sigma)
9 # The 95% confidence interval for the odds ratio is
  obtained by first computing
10 error=1.96*sigma
11 left_i=a-error
12 right_i=a+error
13
14 print(left_i)
15 print(right_i)
16 print("confidence interval")
17 print(exp(left_i))
18 print(exp(right_i))
```

R code Exa 10.17 The pharmaceutical study

```
1 # for clinic 1
2 r1=c(50,50)
3 c1=c(55,45)
4
5 # for clinic 2
6 r2=c(50,50)
```

```

7  c2=c(55,45)
8
9  # for clinic 3
10 r3=c(50,50)
11 c3=c(74,26)
12 nh=100
13 # The numerator of the CMH statistic
14 N=((40-((r1[1]*c1[1])/nh))+(35-((r2[1]*c2[1])/nh))
    +(43-((r3[1]*c3[1])/nh)))^2
15 print(N)
16
17 D=((r1[1]*r1[2]*c1[1]*c1[2])/(nh^2*(nh-1)))+((r2[1]*
    r2[2]*c2[1]*c2[2])/(nh^2*(nh-1)))+((r3[1]*r3[2]*
    c3[1]*c3[2])/(nh^2*(nh-1)))
18 print(D)
19
20 X2=N/D
21 print(X2)
22 # For df = 1, this result is significant at the p <
    .001 level. the drug-treated groups have
    consistently higher improvement rates than the
    placebo groups.

```

Chapter 11

Linear Regression and Correlation

R code Exa 11.2 least squares estimates of slope and intercept

```
1 # Sales Volume
2 y=c(25,55,50,75,110,138,90,60,10,100)
3 # % of Ingredients Purchased Directly ,
4 x=c(10,18,25,40,50,63,42,30,5,55)
5 # Sxy is the sum of x deviations times y deviations
   and Sxx is the sum of x deviations squared.
6
7 Sxx=(x-mean(x))^2
8 Sxy=(x-mean(x))*(y-mean(y))
9
10 # least-squares estimates of slope and intercept
11 Beta1=sum(Sxy)/sum(Sxx)
12 Beta0=mean(y)-Beta1*mean(x)
13 print(Beta1)
14 print(Beta0)
```

R code Exa 11.3 the least squares estimates of the intercept

```
1 xbar=31.80
2 ybar=2.785
3 Sxx=485.60
4 Syy=7.36
5 Sxy=55.810
6 Beta1=Sxy/Sxx
7 Beta0=ybar-Beta1*xbar
8 print(Beta1)
9 print(Beta0)
```

R code Exa 11.4 Forest growth retardation

```
1 soil_ph <- c
  (3.3,3.4,3.4,3.5,3.6,3.6,3.7,3.7,3.8,3.8,3.9,4.0,4.1,4.2,4.3,4.4,
2 grow_ret <- c
  (17.78,21.59,23.84,15.13,23.45,20.87,17.78,20.09,17.78,12.46,14.9
3
4 # Apply the lm() function.
5 relation <- lm(grow_ret~soil_ph)
6
7 print(summary(relation))
8 anova(relation)
9 # The regression equation : GrowthRet = 47.475 -
  7.86 SoilpH
10 # The estimated intercept (constant)
11 beta0=47.475
12 # estimated slope (Soil pH)
13 beta1=-7.859
14 # least square prediction
15 y=47.475-7.859*4
16 print(y)
```

R code Exa 11.7 confidence interval for the slope

```
1 beta1=-7.859 # caclulated in 11.4
2 error=1.090
3 alpha=0.05/2
4 df=18
5 t.alpha=qt(1-alpha,df)
6 print(t.alpha)
7 # corresponding confidence interval for the true
  value of beta1
8 left_i=beta1-t.alpha*error
9 right_i=beta1+t.alpha*error
10 print("confidence interval")
11 print(left_i)
12 print(right_i)
```

R code Exa 11.8 use the F test for testing

```
1 F_statistic=52.01 # computed in example 11.4
2 F_statistic
3 t_statistic=-7.212
4 # both are calculated in 11.4
5 p=t_statistic^2
6 # t^2 equals the F value, to within round-off error
  .
7 p
```

R code Exa 11.10 The manufacturer of a new brand of thermal panes


```

1
2 heat_loss <- c(86,80,77 , 78,84,75 ,33,38,43)
3 temperature <- c(20,20,20,40,40,40,60,60,60)
4 plot(temperature,heat_loss)
5 # Apply the lm() function.
6 relation <- lm(heat_loss~temperature)
7
8 print(summary(relation))
9 # linear regression model : y= 109-1.07*x
10 anova(relation)
11 # y and y-ycap for the nine observations
12 cbind( temperature,heat_loss,predict(relation),resid
      (relation))
13 plot(predict(relation),resid(relation))

```

R code Exa 11.11 Conduct a test for lack of fit

```

1
2 heat_loss <- c(86,80,77 , 78,84,75 ,33,38,43)
3 temperature <- c(81,81,81,79,79,79,38,38,38)
4
5 # Apply the lm() function.
6 relation <- lm(heat_loss~temperature)
7
8 aov(relation)
9
10 SSPexp=134
11 SSresidual=894.5 #calculated in 11.10
12 SSLack=SSresidual-SSPexp
13
14 MSPexp=SSPexp/6
15 MSlack=SSLack/1
16 # test statistic
17 F=MSlack/MSPexp
18 print(F)

```

```

19 alpha=0.05
20
21 fvalue=qf(1-alpha,df1 = 1,df2 = 6)
22 if(F>fvalue){
23   print(" we reject H0 and conclude that there is
          significant lack of fit for a linear regression
          model")
24 }else{
25   print("we do not reject H0 and conclude that
          there is no significant lack of fit for a
          linear regression model")
26 }

```

R code Exa 11.12 An engineer is interested in calibrating a flow meter

```

1
2 x <- c(1,2,3,4,5,6,7,8,9,10)
3 y <- c(1.4,2.3,3.1,4.2,5.1,5.8,6.8,7.6,8.7,9.5)
4 xbar=mean(x)
5 # Apply the lm() function.
6 relation <- lm(y~x)
7 print(summary(relation))
8
9 # linear regression model : y=.9012*x+.4934
10 mod=lm(x~y)
11 predict(mod,data.frame(y=4),interval = "prediction"
          ,level = 0.95)
12 # the 95% prediction limits for x are 3.65 to 4.13

```

R code Exa 11.13 In a study of the reproductive success of grasshoppers


```

9 cor(y,x)
10 # For all 12 observations , the output shows a
    correlation coefficient of .646
11 # after subsetting x>=37
12
13 y1 <- c( 57, 46, 50, 59, 61, 52)
14 x1<- c( 37 ,37 ,38 ,40 ,43 ,49)
15 # Apply the lm() function.
16 relation <- lm(y1~x1)
17 print(summary(relation))
18 cor(y1,x1)
19 # For subset having x greater than or equal to 37,
    the output shows a correlation coefficient of
    .188

```

R code Exa 11.16 Perform t tests for the null hypothesis

```

1 y <- c(41, 39, 47, 51, 43, 40, 57, 46, 50, 59, 61,
    52)
2
3 x<- c( 24 ,30 ,33 ,35 ,36 ,36 ,37 ,37 ,38 ,40 ,43
    ,49)
4 cor.test(y,x)

```

Chapter 12

Multiple Regression and the General Linear Model

R code Exa 12.2 An industrial engineer is designing a simulation model to generate

```
1 # model :  $y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + e$ 
2 #x1 = 1 if system 2 is used, x1 = 0 otherwise
3 #x2 = 1 if system 3 is used, x2 = 0 otherwise
4 #x3 = 1 if system 4 is used, x3 = 0 otherwise
5 u1=7
6 u2=9
7 u3=6
8 u4=15
9 b0=u1
10 b1=u2-u1
11 b2=u3-u1
12 b3=u4-u1
13 print(b0)
14 print(b1)
15 print(b2)
16 print(b3)
```

R code Exa 12.3 Check your findings by substituting values

```
1 u1=7
2 u2=9
3 u3=6
4 u4=15
5 x=(u3-u1)-(u2-u1)
6 y=(u3-u1)-(u4-u1)
7 print(x)
8 print(y)
```

R code Exa 12.5 An experiment was conducted to investigate

```
1 y=c(4.3,5.5,6.8,8.0,4.0,5.2,6.6,7.5,2.0,4.0,5.7,6.5)
2 x1=c(4,5,6,7,4,5,6,7,4,5,6,7)
3 x2=c
   (.20,.20,.20,.20,.30,.30,.30,.30,.40,.40,.40,.40)
4
5 cbind(sum(y),sum(x1),sum(x2),sum(y*x1),sum(y*x2),sum
   (x1*x2),sum(x1^2),sum(x2^2))
6 # three normal equations for this model
7 # 66.1=12*beta0+66*beta1+3.6*beta2
8 # 383.3=66*beta0+378*beta1+19.8*beta2
9 # 19.19=3.6*beta0+19.8*beta1+1.16*beta2
10 relation = lm(y~x1+x2)
11 print(summary(relation))
12 anova(relation)
13 # linear regression model : y=0.667+1.316*x1-8.000*
   x2
14 x1=6.5
15 x2=.35
16 y=0.667+1.316*x1-8.000*x2
```

17 `print(y)`

R code Exa 12.6 A kinesiologist is investigating measures of the physical fitness

```
1 y=c(1.5 ,2.1, 1.8, 2.2, 2.2, 2.0, 2.1, 1.9, 2.8,
      1.9, 2.0, 2.7, 2.4, 2.3, 2.0, 1.7, 2.3, 0.9, 1.2,
      1.9, 0.8, 2.2, 2.3, 1.7, 1.6, 1.6, 2.8, 2.7,
      1.3 ,2.1, 2.5, 1.5, 2.4, 2.3, 1.9, 1.5, 2.4,
      2.3, 1.7 ,2.0 ,1.9, 2.3, 2.1, 2.2, 1.8, 2.1, 2.2,
      1.3, 2.5, 2.2, 1.4, 2.2 ,2.5 ,1.8)
2 x1=c(139.8, 143.3, 154.2 ,176.6 ,154.3, 185.4,
      177.9, 158.8, 159.8, 123.9, 164.2, 146.3, 172.6
      ,147.5 ,163.0, 159.8 ,162.7, 133.3 ,142.8 ,146.6
      ,141.6, 158.9, 151.9, 153.3 ,144.6, 133.3, 153.6,
      158.6, 108.4, 157.4 ,141.7, 151.1, 149.5, 144.3
      ,166.6 ,153.6 ,144.1, 148.7, 159.9, 162.8 ,145.7
      ,156.7, 162.3, 164.7, 134.4, 160.1, 143.0,
      141.6, 152.0, 187.1, 122.9, 157.1 ,155.1 ,133.6)
3 x2=c(19.1, 21.1, 21.2, 23.2, 22.4 ,22.1, 21.6, 19.0
      ,20.9 ,22.0 ,19.5 ,19.8 ,20.7 ,21.0 ,21.2 ,20.4
      ,20.0 ,21.1 ,22.6 ,23.0 ,22.1, 22.8 ,21.8 ,20.0
      ,22.9 ,22.9 ,19.4 ,21.0 ,21.1 ,20.1 ,19.8 ,21.8
      ,20.5 ,21.0, 21.4 ,20.8 ,20.3 ,19.1 ,19.6 ,21.3
      ,20.0, 19.2 ,22.1 ,19.1, 20.9 ,21.1 ,20.5 ,21.7
      ,20.8 ,21.5, 22.6 ,23.4 ,20.8, 22.5)
4 x3=c(18.1, 15.3, 15.3, 17.7, 17.1, 16.4, 17.3, 16.8,
      15.5 ,13.8, 17.0, 13.8, 16.8, 15.3, 14.2, 16.8,
      16.6, 17.5, 18.0 ,15.7 ,19.1, 13.4, 13.6, 16.1,
      15.8 ,18.2 ,13.3, 14.9, 16.7 ,15.7, 13.5, 18.8
      ,14.9 ,17.2 ,17.4, 16.4, 13.3, 15.4, 17.4, 16.2,
      18.6, 16.4, 19.0, 17.1, 15.6, 14.2, 17.1, 14.5
      ,17.3, 14.6, 18.6, 14.2, 16.0, 15.4)
5 x4=c(133.6 ,144.6, 164.6, 139.4 ,127.3, 137.3, 144.0
      ,141.4, 127.7 ,124.2 ,135.7 ,116.1 ,109.0,
```

```

131.0, 143.3 ,156.6 ,120.1, 131.8, 149.4 ,106.9,
135.6 ,164.6 ,162.6 ,134.8 ,154.0 ,120.7 ,151.9,
133.6 ,142.8 ,168.2 ,120.5, 135.6, 119.5 ,119.0
,150.8, 144.0 , 124.7 ,154.4 ,136.7 ,152.4
,133.6, 113.2 ,81.6 ,134.8, 130.4 ,162.1, 144.7
,163.1, 137.1 ,156.0 ,127.2 ,121.4 ,155.3, 140.4)
6
7 relation = lm(y~x1+x2+x3+x4)
8 print(summary(relation))

```

R code Exa 12.8 The following SPSS computer output is obtained from the data

```

1 y=c(1.5 ,2.1, 1.8, 2.2, 2.2, 2.0, 2.1, 1.9, 2.8,
      1.9, 2.0, 2.7, 2.4, 2.3, 2.0, 1.7, 2.3, 0.9, 1.2,
      1.9, 0.8, 2.2, 2.3, 1.7, 1.6, 1.6, 2.8, 2.7,
      1.3 ,2.1, 2.5, 1.5, 2.4, 2.3, 1.9, 1.5, 2.4,
      2.3, 1.7 ,2.0 ,1.9, 2.3, 2.1, 2.2, 1.8, 2.1, 2.2,
      1.3, 2.5, 2.2, 1.4, 2.2 ,2.5 ,1.8)
2 x1=c(139.8, 143.3, 154.2 ,176.6 ,154.3, 185.4,
      177.9, 158.8, 159.8, 123.9, 164.2, 146.3, 172.6
      ,147.5 ,163.0, 159.8 ,162.7, 133.3 ,142.8 ,146.6
      ,141.6, 158.9, 151.9, 153.3 ,144.6, 133.3, 153.6,
      158.6, 108.4, 157.4 ,141.7, 151.1, 149.5, 144.3
      ,166.6 ,153.6 ,144.1, 148.7, 159.9, 162.8 ,145.7
      ,156.7, 162.3, 164.7, 134.4, 160.1, 143.0,
      141.6, 152.0, 187.1, 122.9, 157.1 ,155.1 ,133.6)
3 x2=c(19.1, 21.1, 21.2, 23.2, 22.4 ,22.1, 21.6, 19.0
      ,20.9 ,22.0 ,19.5 ,19.8 ,20.7 ,21.0 ,21.2 ,20.4
      ,20.0 ,21.1 ,22.6 ,23.0 ,22.1, 22.8 ,21.8 ,20.0
      ,22.9 ,22.9 ,19.4 ,21.0 ,21.1 ,20.1 ,19.8 ,21.8
      ,20.5 ,21.0, 21.4 ,20.8 ,20.3 ,19.1 ,19.6 ,21.3
      ,20.0, 19.2 ,22.1 ,19.1, 20.9 ,21.1 ,20.5 ,21.7
      ,20.8 ,21.5, 22.6 ,23.4 ,20.8, 22.5)
4 x3=c(18.1, 15.3, 15.3, 17.7, 17.1, 16.4, 17.3, 16.8,

```



```

        15.5 ,13.8, 17.0, 13.8, 16.8, 15.3, 14.2, 16.8,
        16.6, 17.5, 18.0 ,15.7 ,19.1, 13.4, 13.6, 16.1,
        15.8 ,18.2 ,13.3, 14.9, 16.7 ,15.7, 13.5, 18.8
        ,14.9 ,17.2 ,17.4, 16.4, 13.3, 15.4, 17.4, 16.2,
        18.6, 16.4, 19.0, 17.1, 15.6, 14.2, 17.1, 14.5
        ,17.3, 14.6, 18.6, 14.2, 16.0, 15.4)
5  x4=c(133.6 ,144.6, 164.6, 139.4 ,127.3, 137.3, 144.0
        ,141.4, 127.7 ,124.2 ,135.7 ,116.1 ,109.0,
        131.0, 143.3 ,156.6 ,120.1, 131.8, 149.4 ,106.9,
        135.6 ,164.6 ,162.6 ,134.8 ,154.0 ,120.7 ,151.9,
        133.6 ,142.8 ,168.2 ,120.5, 135.6, 119.5 ,119.0
        ,150.8, 144.0 , 124.7 ,154.4 ,136.7 ,152.4
        ,133.6, 113.2 ,81.6 ,134.8, 130.4 ,162.1, 144.7
        ,163.1, 137.1 ,156.0 ,127.2 ,121.4 ,155.3, 140.4)
6
7  relation = lm(y~x1+x2+x3+x4)
8
9  anova(relation)
10 SSresidual=4.3938
11 df=49 # by looking at table given in question
12 # Std. Error of the Estimate
13 SE=sqrt(SSresidual/df)
14 print(SE)

```

R code Exa 12.9 Using the sum of squares in the ANOVA table

```

1  y=c(1.5 ,2.1, 1.8, 2.2, 2.2, 2.0, 2.1, 1.9, 2.8,
        1.9, 2.0, 2.7, 2.4, 2.3, 2.0, 1.7, 2.3, 0.9, 1.2,
        1.9, 0.8, 2.2, 2.3, 1.7, 1.6, 1.6, 2.8, 2.7,
        1.3 ,2.1, 2.5, 1.5, 2.4, 2.3, 1.9, 1.5, 2.4,
        2.3, 1.7 ,2.0 ,1.9, 2.3, 2.1, 2.2, 1.8, 2.1, 2.2,
        1.3, 2.5, 2.2, 1.4, 2.2 ,2.5 ,1.8)
2  x1=c(139.8, 143.3, 154.2 ,176.6 ,154.3, 185.4,
        177.9, 158.8, 159.8, 123.9, 164.2, 146.3, 172.6
        ,147.5 ,163.0, 159.8 ,162.7, 133.3 ,142.8 ,146.6

```

```

,141.6, 158.9, 151.9, 153.3 ,144.6, 133.3, 153.6,
 158.6, 108.4, 157.4 ,141.7, 151.1, 149.5, 144.3
,166.6 ,153.6 ,144.1, 148.7, 159.9, 162.8 ,145.7
,156.7, 162.3, 164.7, 134.4, 160.1, 143.0,
141.6, 152.0, 187.1, 122.9, 157.1 ,155.1 ,133.6)
3 x2=c(19.1, 21.1, 21.2, 23.2, 22.4 ,22.1, 21.6, 19.0
,20.9 ,22.0 ,19.5 ,19.8 ,20.7 ,21.0 ,21.2 ,20.4
,20.0 ,21.1 ,22.6 ,23.0 ,22.1, 22.8 ,21.8 ,20.0
,22.9 ,22.9 ,19.4 ,21.0 ,21.1 ,20.1 ,19.8 ,21.8
,20.5 ,21.0, 21.4 ,20.8 ,20.3 ,19.1 ,19.6 ,21.3
,20.0, 19.2 ,22.1 ,19.1, 20.9 ,21.1 ,20.5 ,21.7
,20.8 ,21.5, 22.6 ,23.4 ,20.8, 22.5)
4 x3=c(18.1, 15.3, 15.3, 17.7, 17.1, 16.4, 17.3, 16.8,
15.5 ,13.8, 17.0, 13.8, 16.8, 15.3, 14.2, 16.8,
16.6, 17.5, 18.0 ,15.7 ,19.1, 13.4, 13.6, 16.1,
15.8 ,18.2 ,13.3, 14.9, 16.7 ,15.7, 13.5, 18.8
,14.9 ,17.2 ,17.4, 16.4, 13.3, 15.4, 17.4, 16.2,
18.6, 16.4, 19.0, 17.1, 15.6, 14.2, 17.1, 14.5
,17.3, 14.6, 18.6, 14.2, 16.0, 15.4)
5 x4=c(133.6 ,144.6, 164.6, 139.4 ,127.3, 137.3, 144.0
,141.4, 127.7 ,124.2 ,135.7 ,116.1 ,109.0,
131.0, 143.3 ,156.6 ,120.1, 131.8, 149.4 ,106.9,
135.6 ,164.6 ,162.6 ,134.8 ,154.0 ,120.7 ,151.9,
133.6 ,142.8 ,168.2 ,120.5, 135.6, 119.5 ,119.0
,150.8, 144.0 , 124.7 ,154.4 ,136.7 ,152.4
,133.6, 113.2 ,81.6 ,134.8, 130.4 ,162.1, 144.7
,163.1, 137.1 ,156.0 ,127.2 ,121.4 ,155.3, 140.4)
6
7 relation = lm(y~x1+x2+x3+x4)
8
9 anova(relation)
10
11 # coefficient of determination
12 summary(relation)$r.squared

```

R code Exa 12.11 The following SAS output is provided for fitting the model

```

1 y=c(1.5 ,2.1, 1.8, 2.2, 2.2, 2.0, 2.1, 1.9, 2.8,
      1.9, 2.0, 2.7, 2.4, 2.3, 2.0, 1.7, 2.3, 0.9, 1.2,
      1.9, 0.8, 2.2, 2.3, 1.7, 1.6, 1.6, 2.8, 2.7,
      1.3 ,2.1, 2.5, 1.5, 2.4, 2.3, 1.9, 1.5, 2.4,
      2.3, 1.7 ,2.0 ,1.9, 2.3, 2.1, 2.2, 1.8, 2.1, 2.2,
      1.3, 2.5, 2.2, 1.4, 2.2 ,2.5 ,1.8)
2 x1=c(139.8, 143.3, 154.2 ,176.6 ,154.3, 185.4,
      177.9, 158.8, 159.8, 123.9, 164.2, 146.3, 172.6
      ,147.5 ,163.0, 159.8 ,162.7, 133.3 ,142.8 ,146.6
      ,141.6, 158.9, 151.9, 153.3 ,144.6, 133.3, 153.6,
      158.6, 108.4, 157.4 ,141.7, 151.1, 149.5, 144.3
      ,166.6 ,153.6 ,144.1, 148.7, 159.9, 162.8 ,145.7
      ,156.7, 162.3, 164.7, 134.4, 160.1, 143.0,
      141.6, 152.0, 187.1, 122.9, 157.1 ,155.1 ,133.6)
3 x2=c(19.1, 21.1, 21.2, 23.2, 22.4 ,22.1, 21.6, 19.0
      ,20.9 ,22.0 ,19.5 ,19.8 ,20.7 ,21.0 ,21.2 ,20.4
      ,20.0 ,21.1 ,22.6 ,23.0 ,22.1, 22.8 ,21.8 ,20.0
      ,22.9 ,22.9 ,19.4 ,21.0 ,21.1 ,20.1 ,19.8 ,21.8
      ,20.5 ,21.0, 21.4 ,20.8 ,20.3 ,19.1 ,19.6 ,21.3
      ,20.0, 19.2 ,22.1 ,19.1, 20.9 ,21.1 ,20.5 ,21.7
      ,20.8 ,21.5, 22.6 ,23.4 ,20.8, 22.5)
4 x3=c(18.1, 15.3, 15.3, 17.7, 17.1, 16.4, 17.3, 16.8,
      15.5 ,13.8, 17.0, 13.8, 16.8, 15.3, 14.2, 16.8,
      16.6, 17.5, 18.0 ,15.7 ,19.1, 13.4, 13.6, 16.1,
      15.8 ,18.2 ,13.3, 14.9, 16.7 ,15.7, 13.5, 18.8
      ,14.9 ,17.2 ,17.4, 16.4, 13.3, 15.4, 17.4, 16.2,
      18.6, 16.4, 19.0, 17.1, 15.6, 14.2, 17.1, 14.5
      ,17.3, 14.6, 18.6, 14.2, 16.0, 15.4)
5 x4=c(133.6 ,144.6, 164.6, 139.4 ,127.3, 137.3, 144.0
      ,141.4, 127.7 ,124.2 ,135.7 ,116.1 ,109.0,
      131.0, 143.3 ,156.6 ,120.1, 131.8, 149.4 ,106.9,
      135.6 ,164.6 ,162.6 ,134.8 ,154.0 ,120.7 ,151.9,
      133.6 ,142.8 ,168.2 ,120.5, 135.6, 119.5 ,119.0
      ,150.8, 144.0 , 124.7 ,154.4 ,136.7 ,152.4
      ,133.6, 113.2 ,81.6 ,134.8, 130.4 ,162.1, 144.7

```

```

        ,163.1, 137.1 ,156.0 ,127.2 ,121.4 ,155.3, 140.4)
6
7 relation = lm(y~x1+x2+x3+x4)
8 print(summary(relation))
9 anova(relation)
10 SSregression=1.8028+0.6973+2.4205+1.1856
11 print(SSregression)
12 df=4
13 MSregression=SSregression/df
14 print(MSregression)
15 MSresidual=0.08967
16
17 # test statistic
18 F=MSregression/MSresidual
19 print(F)
20 fvalue=qf(1-0.01,4,49)
21 # F>fvalue, therefore there is strong evidence in
    the data to reject the null hypothesis to reject
    the null hypothesis

```

R code Exa 12.12 A large city bank studies the relation of average account size

```

1 # linear regression model :  $y = 0.15085 - 0.00288x_1 - 0.00759x_2 + 0.26528x_3$ 
2 SStotal=3.328
3 SSresidual=0.674
4 n=21
5 R2=(SStotal-SSresidual)/SStotal
6 print(R2)
7
8 #test statistic
9 F=(R2/3)/((1-R2)/(n-4))
10 print(F)
11 alpha=0.05

```

```

12 fvalue=qf(1-alpha,df1=3,df2 = 17)
13 print(fvalue)
14 # Because the computed F statistic , , is greater
    than fvalue , we reject H0 and
15 #conclude that one or more of the x values has some
    predictive power

```

R code Exa 12.13 confidence interval

```

1 beta1=.2652
2 se=.1012
3 n=21
4 k=3
5 df=n-(k+1)
6 alpha=0.025
7 tvalue=qt(1-alpha,df)
8 print(tvalue)
9 left_i=beta1-tvalue*se
10 right_i=beta1+tvalue*se
11 print(" 95% confidence interval")
12 print(left_i)
13 print(right_i)

```

R code Exa 12.14 Locate the estimated partial slope

```

1 beta1=.01291
2 se=.00283
3 n=54
4 k=4
5 df=n-(k+1)
6 alpha=0.05
7 tvalue=qt(1-alpha,df)
8 print(tvalue)

```

```

9 left_i=beta1-tvalue*se
10 right_i=beta1+tvalue*se
11 print(" 95% confidence interval")
12 print(left_i)
13 print(right_i)

```

R code Exa 12.15 conclusion of the test compatible with the confidence interval

```

1 beta1=.01291
2 se=.00283
3 # test statistic
4 t=beta1/se
5 print(t)
6 n=54
7 k=4
8 df=n-(k+1)
9 alpha=0.05
10 tvalue=qt(1-alpha,df)
11 #the computed value of the test statistic > tvalue
    . we conclude there is significant evidence to
    reject H0

```

R code Exa 12.16 Locate the t statistic

```

1 # test statistic
2 t=.26528/.10127
3 print(t)
4 df=17
5 t1value=qt(1-0.01,df)
6 t2value=qt(1-0.005,df)
7 print(t1value)
8 print(t2value)

```

```

9   # Thus, H0 would be rejected at the alpha= .01
    level but not at the alpha= .005 level
10  pvalue =pt(-t, df)
11  print(pvalue)

```

R code Exa 12.17 A state fisheries commission wants to estimate the number of bass

```

1  catch=c
    (3.6,.8,2.5,2.9,1.4,.9,3.2,2.7,2.2,5.9,3.3,2.9,3.6,2.4,.9,2.0,1.9
2  residence=c
    (92.2,86.7,80.2,87.2,64.9,90.1,60.7,50.9,86.1,90.0,80.4,75.0,70.0
3  size=c
    (.21,.30,.31,.40,.44,.56,.78,1.21,.34,.40,.52,.66,.78,.91,1.10,1.
4  access=c(0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1)
5  structures=c
    (81,26,52,64,40,22,80,60,30,90,74,50,61,40,22,50,37,61,39,53)

6  relation = lm(catch~residence+size+access+structures
    )
7  print(summary(relation))
8  anova(relation)
9
10 # for reduced model
11 print(" for reduced model")
12 relation = lm(catch~residence+size)
13 print(summary(relation))
14 anova(relation)
15 # complete linear regression model :   -2.78 + .0268
    x1 + .504x2 + .743x3 + .0511x4
16 # reduced model :  -.87 + .0394x1 + .828x2
17 # test statistic

```

```

18 SSregression_complete=24.0624
19 SSregression_reduced=2.913
20 Ssresidual_complete=2.2756
21 n=20
22 a=(SSregression_complete-SSregression_reduced)/2
23 b=(Ssresidual_complete)/(n-5)
24 F=a/b
25 print(F)
26 fvalue=qf(1-.01,2,15)
27 print(fvalue)
28 # The value of the test statistic is much larger
    than the tabled value, so we have conclusive
    evidence that the access and structure variables
    add predictive value

```

R code Exa 12.22 logistic regression

```

1
2 CK <- c(20, 60, 100, 140, 180, 220, 260, 300, 340,
    380, 420, 460, 500)
3 y <- cbind( c(2, 13, 30, 30, 21, 19, 18, 13, 19, 15,
    7, 8, 35) , c(88, 26, 8, 5, 0, 1, 1, 1, 0, 0, 0,
    0, 0))
4 model1=glm(y~CK, family = binomial)
5 summary(model1)
6 cbind(CK,PRED=predict(model1,type = "response"))
7 # probability that a patient had a heart attack
    when the CK level in the patient was 140 is .868

```

R code Exa 12.26 matrix operations


```

1 C <- matrix(c(-2,4,6,2,4,1),2,3)
2 D <- matrix(c(3,9,7,2,5,1,4,-2,8),3,3)
3 E <- matrix(c(4,8,1,-1,6,-6,0,4,7),3,3)
4 # C+D cannot be computed because of different
   dimensions.
5
6 x=D+E
7 y=D-E
8 z=C%*%D
9 p=t(E)
10 x
11 y
12 z
13 p

```

R code Exa 12.27 inverse of matrix

```

1 B <- matrix(c(7,9,3,5),2,2)
2 C <- matrix(c(3,2,3,2,8,1,4,-2,8),3,3)
3 determinant_B=det(B)
4 determinant_C=det(C)
5 determinant_B
6 determinant_C
7 solve(B)
8 solve(C)
9 # B* inverse B
10 x=zapsmall(solve(B)%*%B)
11 x
12 # C* inverse C
13 y=zapsmall(solve(C)%*%C)
14 y

```

R code Exa 12.28 Obtain the least squares estimates for the prediction equation

```
1 Y <- matrix(c(25,19,33,23),4,1)
2 X <- matrix(c(1,1,1,1,-2,-2,2,2,5,-5,5,-5),4,3)
3 transpose_X=t(X)
4 transpose_X_X=transpose_X%*%X
5 transpose_X_Y=transpose_X%*%Y
6 inverse_transpose_X_X=solve(transpose_X_X)
7
8 beta=inverse_transpose_X_X%*%transpose_X_Y
9 beta
10 # prediction equation is
11 y=25.0+1.5*x1+0.8*x2
```

R code Exa 12.29 Compute SSResidual for the data

```
1 Y <- matrix(c(25,19,33,23),4,1)
2 X <- matrix(c(1,1,1,1,-2,-2,2,2,5,-5,5,-5),4,3)
3 transpose_X=t(X)
4 transpose_Y=t(Y)
5 transpose_X_X=transpose_X%*%X
6 transpose_X_Y=transpose_X%*%Y
7 inverse_transpose_X_X=solve(transpose_X_X)
8
9 beta=inverse_transpose_X_X%*%transpose_X_Y
10 transpose_Y_Y=transpose_Y%*%Y
11 transpose_Y_Y
12 SSresidual=transpose_Y_Y-t(beta)%*%transpose_X_Y
13 SSresidual
```

R code Exa 12.30 Calculate SSRegression and SSTotal for the data

```

1 Y <- matrix(c(25,19,33,23),4,1)
2 X <- matrix(c(1,1,1,1,-2,-2,2,2,5,-5,5,-5),4,3)
3 transpose_X=t(X)
4 transpose_Y=t(Y)
5 transpose_X_X=transpose_X%*%X
6 transpose_X_Y=transpose_X%*%Y
7 inverse_transpose_X_X=solve(transpose_X_X)
8
9 beta=inverse_transpose_X_X%*%transpose_X_Y
10 transpose_Y_Y=transpose_Y%*%Y
11 SSresidual=transpose_Y_Y-t(beta)%*%transpose_X_Y
12 SSregression=t(beta)%*%transpose_X_Y-(sum(Y)^2/4)
13 SSregression
14 SStotal=SSregression+SSresidual
15 SStotal

```

R code Exa 12.31 Calculate the estimated standard error

```

1 ## prediction equation is
2 #y=25.0+1.5*x1+0.8*x2
3 Y <- matrix(c(25,19,33,23),4,1)
4 X <- matrix(c(1,1,1,1,-2,-2,2,2,5,-5,5,-5),4,3)
5 transpose_X=t(X)
6 transpose_X_X=transpose_X%*%X
7 inverse_transpose_X_X=solve(transpose_X_X)
8 inverse_transpose_X_X
9 # for estimated standard error use inverse_
   trasnpose_X_X matrix
10 s_beta0=2*sqrt(0.25)
11 s_beta0
12 s_beta1=2*sqrt(0.0625)
13 s_beta2=2*sqrt(0.01)
14 s_beta1
15 s_beta2

```

Chapter 13

Further Regression Topics

Chapter 14

Analysis of Variance for Completely Randomized Designs

R code Exa 14.1 assignment of paints to the highway sections

```
1 # we obtain a random permutation of the numbers 1
  to 16.
2 x=sample(1:16)
3 # We thus obtain the assignment of paints to the
  highway sections
4 values=c("P1","P1","P1","P1","P2","P2","P2","P2","P3
  ","P3","P3","P3","P4","P4","P4","P4")
5 cbind(x,values)
```

R code Exa 14.2 sum of squares for error

```
1 paint_data=c(28, 35, 27, 21,21, 36, 25 ,18,26, 38,
  27, 17,16, 25, 22, 18)
2 ybar=sum(paint_data)/length(paint_data)
```

```

3  print(ybar)
4  # total sum of squares
5  TSS=0
6  i=1
7  while(i<=length(paint_data)){
8      TSS=TSS+(paint_data[i]-ybar)^2
9      i=i+1
10 }
11 print(TSS)
12 # between treatment sum of squares
13
14 yi=c(mean(paint_data[1:4]),mean(paint_data[5:8]),
        mean(paint_data[9:12]),mean(paint_data[13:16]))
15
16 SST=0
17 j=1
18 while(j<=length(paint_data)/4){
19     SST=SST+4*((yi[j]-ybar)^2)
20     j=j+1
21 }
22 print(SST)
23 # sum of squares for error
24 SSE=TSS-SST
25 print(SSE)

```

R code Exa 14.11 confidence interval

```

1  y1bar=25.1
2  y2bar=23.5

```

```

3 y3bar=37.8
4
5 MSE=10.278
6 sigma=sqrt(MSE)
7
8 # critical value
9 alpha = 0.025
10 z.alpha=qt(1-alpha,df=15)
11
12 # For panels 2 and 3, we have nt = 10 observations
    per panel, thus confidence interval will be
13 n=10
14 error=sigma*z.alpha*sqrt(2/n)
15 # thus confidence interval will be
16 left_i=(y3bar-y2bar)-error
17 right_i=(y3bar-y2bar)+error
18 print("confidence interval is")
19 print(left_i)
20 print(right_i)

```

R code Exa 14.12 locate significant differences among display panels

```

1 alpha=0.05
2 q.alpha=qtukey(1-alpha,3,15)
3 n=10
4 sw2=10.28 # calculated in 14.10
5 W=q.alpha*(sqrt(sw2/n))
6 W
7 sample_means=c(25.1,23.5,37.8)
8 # by ordering sample mean from lowest to highest ,
    we rank display panels by 2 1 3
9 # if difference between means > W then we declare
    them to be significantly different from each
    other

```

R code Exa 14.13 number of replications is

```
1 sigma=(70-40)/4
2
3 alpha=0.025
4 z.alpha=qnorm(1-alpha)
5
6 E=4 #given
7 # number of replications is
8 n=((sigma^2)*(z.alpha^2))/E^2
9 print(n)
```

Chapter 15

Analysis of Variance for Blocked Designs

R code Exa 15.7 Assess whether taking into account the two extraneous sources of variation

```
1 MSR=11128.14
2 MSC=544.44
3 t=5
4 MSE=2887.29
5 # relative efficiency of this Latin square design
   relative to a completely randomized design is
6 re=(MSR+MSC+(t-1)*MSE)/((t+1)*MSE)
7 re
```

R code Exa 15.11 determine which pairs of treatments have significantly different means

```
1 t=9
2 v=16
3 r=3
4 sw2=2.847
5 qvalue=qtukey(1-0.05,t,v)
6 W=qvalue*sqrt(sw2/r)
7 W
8 # any pair of treatment means having a difference
   between corresponding
9 #sample means exceeding 4.9 would be declared
   significantly different
```
