R Textbook Companion for Probability and Statistics for Engineering and the Sciences by Jay L Devore[1]

Created by
Angela Paul
B.Tech.
Computer Science and Engineering
Government Engineering College, Thrissur
Cross-Checked by
R TBC Team

May 29, 2020

[1]Funded 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: Probability and Statistics for Engineering and the Sciences

Author: Jay L Devore

Publisher: Richard Stratton, Boston, USA

Edition: 8

Year: 2012

ISBN: 9780538733526

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

Chapter 1

Overview and descriptive statistics

R code Exa 1.1 Branches of statistics

```
#Ex1.1 , Page 4

library(lattice)

data<-c
   (6.1,12.6,34.7,1.6,18.8,2.2,3.0,2.2,5.6,3.8,2.2,3.1,1.3,1.1,14.1,

stem(data,scale=2)

hist(data,main="Histogram for charity fundraising
   percentage data",xlab="FundRaising",col="grey",
   xlim=c(0,100),ylim=c(0,40))
```

R code Exa 1.5 Collecting data

```r
#Ex1.5, Page 11

Adhesive_Type<-c(1,1,2,2)
Conductor_material<-c(1,2,1,2)
Observed_bond_strength<-c(82,75,84,78,77,87,80,90)
mat1<-matrix(Observed_bond_strength,nrow=4,ncol=2)
Average<-c(sum(mat1[1,])/2,sum(mat1[2,])/2,sum(mat1[3,])/2,sum(mat1[4,])/2)
mat1<-cbind(Adhesive_Type,Conductor_material,mat1,Average)
colnames(mat1)[3]<-"Observed bond strength"
print(mat1)

n<-length(Adhesive_Type)/2
plot(Conductor_material,Average,main="Average bond strengths",xlab="Conducting material",ylab="Average strength")
segments(Conductor_material[1],Average[1],Conductor_material[2],Average[2])
segments(Conductor_material[3],Average[3],Conductor_material[4],Average[4])
```

R code Exa 1.8 Dotplots

```r
#Ex1.8, Page 15

library(graphics)

data<-c
  (10.8,6.9,8.0,8.8,7.3,3.6,4.1,6.0,4.4,8.3,8.1,8.0,5.9,5.9,7.6,8.9

#To create a stacked dotplot
stripchart(data,method="stack",offset=0.5,pch=20,at=0,main="Dotplot of given data")
```

R code Exa 1.9 Histograms

```
#Ex1.9, Page 17

Hits_per_game<-0:27
No_of_games<-c(20,72,209,527,1048,1457,1988,2256,2403,2256,1967,1509,1230,834,56
sum1<-sum(No_of_games)
Relative_frequency<-round(No_of_games/sum1,digits=4)
df1<-data.frame(Hits_per_game,No_of_games,Relative_frequency)
print(df1)

#Proportion of games with atmost 2 hits
f1<-0
for(i in Hits_per_game){
  if(i<=2){
    f1<-f1+df1[i+1,3]
  }
}
print(paste("Proportion of games with atmost two hits:",f1))

#Proportion of games with between 5 and 10 hits
f2<-0
for(j in Hits_per_game){
  if(j>=5 && j<=10){
    f2<-f2+df1[j+1,3]
  }
}
print(paste("Proportion of games with between 5 and 10 hits:",f2))

```

```
par(mfrow=c(1,2))
plot(Hits_per_game,Relative_frequency,type="h",xlim=c(0,20),main="Histogram of hits/game")
barplot(Relative_frequency,width=1,space=0,xlim=c(0,25),main="Histogram of hits/game",xlab="Hits/game",ylab="Relative frequency")
```

R code Exa 1.10 Histograms

```
#Ex1.10, Page 18
#Answers may vary since different class intervals have been taken

#Please install and load the following packages: 1) HistogramTools 2)ash
library(HistogramTools)
library(ash)

data<-c(2.97,4.00,5.20,5.56,5.94,5.98,6.35,6.62,6.72,6.78,6.80,6.85,6.94


#Class intervals can also be created using classIntervals() from ClassInt library:
#classIntervals(data,style="pretty",intervalClosure="left") which creates different intervals as compared to those of textbook

#To create class intervals
ci<-seq(1,19,2)

freq<-table(cut(data,ci,right=FALSE))
rel_freq<-freq/length(data)
print(cbind(freq,rel_freq))

```

```
#To display histogram
PlotRelativeFrequency(hist(data,breaks = ci,right=
    FALSE,plot=FALSE),xlab="BTUIN",main="Histogram of
     energy consumption data",ylim=c(0,0.30))
```

R code Exa 1.11 Histograms

```
#Ex1.11, Page 20

data<-c
    (11.5,12.1,9.9,9.3,7.8,6.2,6.6,7.0,13.4,17.1,9.3,5.6,5.7,5.4,5.2,
#Unequal class widths
ci<-c(2,4,6,8,12,20,30)

#To find frequency, relative frequency and density
    of data manually
freq<-table(cut(data,ci,right=FALSE))
rel_freq<-freq/length(data)
density<-rel_freq/diff(ci)

print(cbind(freq,rel_freq,density))

#To create histogram of the bond strength data
hist(data,breaks=ci,freq=FALSE,right=FALSE,main="
    Histogram for bond strength data",xlab="Bond
    Strength")
```

R code Exa 1.13 Qualitative data

```
#Ex1.13, Page 23
#Answers may vary slightly due to rounding off of
    values
```

```
Rating<-c("A","B","C","D","F","Don't know")
Frequency<-c(478,893,680,178,100,172)
Relative_frequency<-Frequency/sum(Frequency)
df1<-data.frame(Rating,Frequency,Relative_frequency)
print(df1)

print(paste("Total frequency:",sum(Frequency)))
print(paste("Total relative frequency:",sum(Relative
    _frequency)))

barplot(df1$Relative_frequency,space=1,names.arg=df1
    $Rating,ylim=c(0,0.4),xlab="Rating",ylab="
    Relative Frequency",main="Chart of relative
    frequency vs. rating")
```

R code Exa 1.14 Mean

```
#Ex1.14, Page 29
#Steam and leaf plot differs since number of stem
    parts=1

library(aplpack)

x<-c
    (16.1,9.6,24.9,20.4,12.7,21.2,30.2,25.8,18.5,10.3,25.3,14.0,27.1,

sl<-stem.leaf(x,unit=0.1,m=1,style="bare")

s<-sum(x)
mean<-s/length(x)
print(paste("Mean value:",mean))
print(paste("The value",sl$upper,"is an outlier"))
print(paste("Mean value when",sl$upper,"is excluded=
```

```
",(s-45)/(length(x)-1)))
```

R code Exa 1.15 Median

```
#Ex1.15, Page 30
#Answers may vary slightly due to rounding off of
   values

data<-c
   (62.3,62.8,63.6,65.2,65.7,66.4,67.4,68.4,68.8,70.8,75.7,79.0)


dotplot(data,xlab="Duration")

n<-length(data)
n1<-data[n/2]
n2<-data[(n/2)+1]
med<-(n1+n2)/2
print(paste("Median:",med))

max<-max(data)
#If maximum value is omitted
data1<-data[1:length(data)-1]
mean1<-mean(data1)
print(paste("Mean when maximum value,",max,"is
   omitted:",mean1))
```

R code Exa 1.17 Measures of variability for sample data

```
#Ex1.17, Page 36

Car<-1:11
```

```
xi<-c(27.3,27.9,32.9,35.2,44.9,39.9,30.0,29.7,28.5,32.0,37.6)

xibar<-mean(xi)
s<-sd(xi)

print(paste("The size of the representative deviation from the mean,",round(xibar,digits=2),"is roughly",round(s,digits=2),"mpg"))
```

R code Exa 1.18 Measures of variability

```
#Ex1.18, Page 38
#Answers may vary slightly due to rounding off of values

data<-c(154,142,137,133,122,126,135,135,108,120,127,134,122)

print(paste("s:",sd(data)))
```

R code Exa 1.19 Boxplots

```
#Ex1.19, Page 40

obs<-c(40,52,55,60,70,75,85,85,90,90,92,94,94,95,98,100,115,125,125)

#To find five-number summary
print(summary(fivenum(obs)))
```

```
boxplot(obs,horizontal=TRUE,xlab="Depth",main="
    Boxplot of the corrosion data")
```

R code Exa 1.20 Box plots that show outliers

```
#Ex1.20, Page 41

data<-c
    (9.69,13.16,17.09,18.12,23.70,24.07,24.29,26.43,30.75,31.54,35.07

#To find relevant summary quantities
print(summary(fivenum(data)))

c<-seq(0,1600,by=200)

boxplot(data,horizontal=TRUE,xlab="Daily nitrogen
    load",main="Boxplot of daily nitrogen load data
    showing mild and extreme outliers")
```

Chapter 2

Probability

R code Exa 2.8 Relations from set theory

```
#Ex2.8, Page 53

A<-c(0,1,2,3,4)
B<-c(3,4,5,6)
C<-c(1,3,5)
U<-union(A,B)

cat("A'=",setdiff(U,A),"\n")
cat("A U B =",union(A,B),"\n")
cat("A U C =",union(A,C),"\n")
cat("A n B =",intersect(A,B),"\n")
cat("A n C =",intersect(A,C),"\n")
cat("(A n C)'=",setdiff(U,intersect(A,C)))
```

R code Exa 2.11 Properties of probability

```
#Ex2.11, Page 56

```

```
#S={U,D}

p<-seq(0,1,by=0.25)

for(i in p){
  pu<-i
  pd<-1-i
  cat("P(U):",pu,"\n")
  cat("P(D):",pd,"\n\n")
}
cat("p value can represent any fixed value between 0
    and 1")
```

R code Exa 2.13 Properties of probability

```
#Ex2.13, Page 59
#Answers may vary slightly due to rounding off of
   values

#For A to occur, at least one of the individual
   components must fail
#To find P(A')=SSSSS
p<-90/100
pa_dash<-p^5
cat("P(A'):",pa_dash,"\n")

pa<-1-pa_dash
cat("P(A):",1-pa_dash,"\n\n")
cat("Roughly",round(pa*100,digits=0),"% of the
   systems will fail")
```

R code Exa 2.14 Properties of probability

```
#Ex2.14, Page 60

#P(A)=>probability that a household gets internet service
pa<-60/100

#P(B)=>probability that a household gets TV service
pb<-80/100

#P(A n B)=>probability that a household gets both services
pab<-50/100

#To find probability that a household gets at least one of the services
p1<-pa+pb-pab
cat("P(at least one service)=P(A U B) =",p1,"\n")

pa_b<-p1-pa
pb_a<-p1-pb
#To find probability that a household gets exactly one service
p2<-pa_b+pb_a
cat("P(exactly one service)=",p2)
```

R code Exa 2.19 Product rule

```
#Ex2.19, Page 67

#No of appliance dealers
n1<-5

#No of plumbing contractors
n2<-12

```

```
#No of electrical contractors
n3<-9

cat("There are",n1*n2*n3,"ways to first choose an appliance dealer, then a plumbing contractor and finally an electrical contractor")
```

R code Exa 2.21 Permutations and combinations

```
#Ex2.21, Page 68

#No of teaching assistants
n<-10
#No of questions in the first exam
k<-4

#To find number of permutations
p<-factorial(n)/factorial(n-k)
cat("The professor could give",p,"different four-question exam papers without using the same graders")
```

R code Exa 2.22 Permutations and combinations

```
#Ex2.22, Page 69
#Answers may vary slightly due to rounding off of values

#Total no of songs
n<-100

#No of songs by the Beatles
m<-10
```

```
#Probability that the 1st Beatles song heard is the 5th song played
t<-5
p1<-(factorial(n-m)/factorial(n-m-(t-1))*m/(factorial(n)/factorial(n-t)))
cat("P(1st B is 5th song played):",p1,"\n")

#Probability that the 1st Beatles song heard is the 1st,2nd,3rd,4th or 5th song played
p2<-0
for(i in 1:5){
  p2<-p2+(choose(n-i,m-1))/(choose(n,m))
}
cat("P(1st B is 1st,2nd,3rd,4th or 5th song played):",p2,"\n")
```

R code Exa 2.23 Permutations and combinations

```
#Ex2.23, Page 70
#Answers may vary slightly due to rounding off of values

#Total no of printers
n<-25
l<-10
i<-15

#Probability that 6 random printers selected will have 3 laser printers
r1<-6
r<-3
ND3<-(choose(l,r)*choose(i,r))/choose(n,r1)
cat("P(3 laser printers are selected among the 6 selected):",ND3,"\n")
```

```
#To find the probability that at least 3 inkjet
    printers are selected
prob<-0
for(j in 3:6){
  num<-choose(i,j)*choose(l,r1-j)
  denom<-choose(n,r1)
  prob<-prob+(num/denom)
}
cat("P(at least 3 inkjet printers are selected):",
    prob,"\n")
```

R code Exa 2.25 Conditional probability

```
#Ex2.25, Page 74

#Probability that memory card is purchased
pa<-0.6

#Probability that battery is purchased
pb<-0.4

#Probability that both memory card and battery is
    purchased
pab<-0.3

#Probability that an optional card was purchased
    given that individuals purchased extra battery
pa_b<-pab/pb
cat("P(A|B)=",pa_b,"\n")
cat("Of all individuals who purchased an extra
    battery,",pa_b*100,"% purchased an optional
    memory card\n")

```

```
#Probability that an additional battery was
   purchased given that individuals purchased an
   optional memory card
pb_a<-pab/pa
cat("P(B|A)=",pb_a,"\n")

if(pa_b!=pb_a)  print(paste("P(A|B) is not equal to
   P(B|A)"))
```

R code Exa 2.26 Conditional probability

```
#Ex2.26, Page 75
#Answers may vary slightly due to rounding off of
   values

#Package to be installed: VennDiagram
library(VennDiagram)

#A=>Arts, B=>Books, C=>Cinema
pa<-0.14
pb<-0.23
pc<-0.37
pab<-0.08
pbc<-0.13
pac<-0.09
pabc<-0.05

#To illustrate the Venn diagram using the given
   probabilities
draw.triple.venn(area1=pa,area2=pb,area3=pc,n12=pab,
   n23=pbc,n13=pac,n123=pabc,category=c("Arts","
   Books","Cinema"))

cat("With the help of Venn diagram:\n")
#P(A|B)
```

```
pa_b<-pab/pb
print(paste("P(A|B)=",pa_b))

#P(A|B U C)
pa_bUc<-(pab+pac-pabc)/(pb+pc-pbc)
print(paste("P(A|B U C)=",pa_bUc))

#P(A|reads at least once)=P(A|A U B U C)
pa_aUbUc<-pa/(pa+pb+pc-pab-pac-pbc+pabc)
print(paste("P(A|reads at least once)=",pa_aUbUc))

#P(A U B|C)
paUb_c<-(pac+pbc-pabc)/pc
print(paste("P(A U B|C)=",paUb_c))
```

R code Exa 2.30 Bayes Theorem

```
#Ex2.30, Page 78

# % of messages which come into account 1
pa1<-70/100

# % of messages which come into account 2
pa2<-20/100

# % of messages which come into account 3
pa3<-10/100

# % of messages which are spam in account 1
pb_a1<-1/100

# % of messages which are spam in account 2
pb_a2<-2/100

# % of messages which are spam in account 3
```

```
pb_a3<-5/100

#To find the probability that a randomly selected
    message is spam
#Using law of total probability
pb<-pa1*pb_a1+pa2*pb_a2+pa3*pb_a3
print(paste("P(Randomly selected message is spam):",
    pb,"and thus,",pb*100,"% messages will be spam"))
```

R code Exa 2.36 Independence of more than two events

```
#Ex2.36, Page 85
#Answers may vary slightly due to rounding off of
    values

pai<-rep(0.9,times=6)

#To find probability that system lifetime exceeds t0
prob1<-pai[1]*pai[2]*pai[3]+pai[4]*pai[5]*pai[6]-(
    prod(pai))
print(paste("P(system lifetime exceeds t0):",prob1))

#To find probability that system lifetime is at
    least t0
prob2<-(1-(1-pai[1])*(1-pai[2]))^3
print(paste("P(system lifetime is at least t0):",
    prob2))
```

Chapter 3

Discrete random variables and probability distributions

R code Exa 3.7 Probability distributions for discrete random variables

```
#Example 3.7, Page 97
#Storing all probabilities in an array
prob<-c(0.05,0.1,0.15,0.25,0.20,0.15,0.10)
P<-array(c(prob),dim=c(1,7,1))

#P(X<=2)=P(X=0)+P(X=1)+P(X=2)
p1<-P[1,1,1]+P[1,2,1]+P[1,3,1]
print(paste("P(X<=2) =",p1))

#P(X>=3)=1-P(X<=2)
p2<-1-(P[1,1,1]+P[1,2,1]+P[1,3,1])
print(paste("P(X>=3) =",p2))

#P(2<=X<=5)=P(X=2,3,4 or 5)
p3<-P[1,3,1]+P[1,4,1]+P[1,5,1]+P[1,6,1]
print(paste("P(2<=X<=5) =",p3))

#P(2<X<5)=P(X=3,4)
p4<-P[1,4,1]+P[1,5,1]
```

```
print(paste("P(2<X<5) =",p4))
```

R code Exa 3.8 Probability distributions for discrete random variables

```
#Example 3.8, Page 97
#The answers may slightly vary due to rounding off
    of values
p1<-c(1,0,2,2,3,0,4,1,5,2,6,0)
row.names<-c("Lot","Number of defectives")
column.names<-c("","","","","","")
matrix.names<-c("Number of defectives in each lot")
p<-array(c(p1),dim=c(2,6,1),dimnames=list(row.names,
    column.names,matrix.names))
print(p)
#p(0)=P(X=0)
c<-0
for(i in 1:6) {
  d<-p[2,i,1]
  if(d==0){
    c<-c+1
  }
}
d<-c/6
print(paste("p(0)=",d))

#p(1)=P(X=1)
c<-0
for(i in 1:6){
    d<-p[2,i,1]
    if(d==1){
      c<-c+1
  }
}
d<-c/6
print(paste("p(1)=",d))
```

```

#p(2)=P(X=2)
c<-0
for(i in 1:6){
    d<-p[2,i,1]
    if(d==2){
      c<-c+1
  }
}
d<-c/6
print(paste("p(2)=",d))
```

R code Exa 3.9 Probability distributions for discrete random variables

```
#Ex3.9, Page 98
#X=0 if customer purchases laptop computer
#X=1 if customer purchases desktop computer
#p(0)
p0<-80/100
print(paste("Probability that next customer purchases laptop model:",p0))
p1<-20/100
print(paste("Probability that next customer purchases desktop model:",p1))
p1<-c(0.8,0.2,0)
row.names<-c("if x=0","if x=1","if (x!=0 or 1)")
column.names<-c("")
matrix.names<-c("p(x)")
p<-array(c(p1),dim=c(3,1,1),dimnames=list(row.names,column.names,matrix.names))
print(p)
a<- 1
p<- dbinom(a, size = 1, prob = 0.2)
plot(1, p, type = 'h', xlab = 'x', ylab = 'p(x)')
```

R code Exa 3.10 Probability distributions for discrete random variables

```
#Ex3.10, Page 98
#Let Y=>O+ blood type and N=>Other blood types
p1<-c('a','Y','b','Y','c','N','d','N','e','N')
row.names<-c("Blood donors","O+ blood donors")
column.names<-c("","","","","")
matrix.names<-c("Blood donation table")
p<-array(c(p1),dim=c(2,5,1),dimnames=list(row.names,
   column.names,matrix.names))
print(p)
#p(1)=P(X=1)
c<-0
for(i in 1:5) {
  d<-p[2,i,1]
  if(d=="Y"){
    c<-c+1
  }
}
d1<-c/5
print(paste("p(1)=",d1))

#p(2)=P(X=2)
g<-0
e<-0
for(i in 1:5) {
  h<-p[2,i,1]
  if(h=="Y"){
    g<-g+1
  }
  else{
    e<-e+1
  }
}
```

```
h<-g/5
f<-e/4
d2<-h*f
print(paste("p(2)=",d2))

#p(3)=P(X=3)
d3<-(e/5)*((e-1)/4)*(c/3)
print(paste("p(3)=",d3))

#p(4)=P(X=4)
d4<-(e/5)*((e-1)/4)*((e-2)/3)
print(paste("p(4)=",d4))

p2<-c(1,d1,2,d2,3,d3,4,d4)
row.names<-c("y","p(y)")
column.names<-c("","","","")
matrix.names<-c("PMF")
q<-array(c(p2),dim=c(2,4,1),dimnames=list(row.names,
    column.names,matrix.names))
print(q)

py<- 1:4
y<- c(0.4,0.3,0.2,0.1)
plot(py,y,type="h",main="Line graph for pmf",xlab="y
    ",ylab="p(y)")
```

R code Exa 3.13 Cumulative distribution function

```
#Ex3.13, Page 101

y<-c(1,2,4,8,16)
p<-c(0.05,0.1,0.35,0.4,0.1)

#F(1)=P(Y<=1)=p(1)
print(paste("F(1)=",cumsum(p[1])))
```

```
#F(2)=P(Y<=2)=p(2)
print(paste("F(2)=",max(cumsum(p[1:2]))))

#F(4)=P(Y<=4)=p(4)
print(paste("F(4)=",max(cumsum(p[1:3]))))

#F(8)=P(Y<=8)=p(8)
print(paste("F(8)=",max(cumsum(p[1:4]))))

#F(16)=P(X<=16)=p(16)
print(paste("F(16)=",max(cumsum(p[1:5]))))
```

R code Exa 3.15 Cumulative distribution function of discrete random variable

```
#Ex3.15, Page 104
a<-c(0,0.58,1,0.72,2,0.76,3,0.81,4,0.88,5,0.94)
row.names<-c("X","F(X)")
column.names<-c("","","","","","")
matrix.names<-c("Number of days of sick leave taken
   by a randomly selected employee")
b<-array(c(a),dim=c(2,6,1),dimnames=list(row.names,
   column.names,matrix.names))
print(b)

#P(2<=X<=5)=P(X=2,3,4,5)
p<-b[2,6,1]-b[2,2,1]
print(paste("P(2<=X<=5)=",p))

#P(X=3)
q<-b[2,4,1]-b[2,3,1]
print(paste("P(X=3)=",q))
```

R code Exa 3.16 Expected value of X

```
#Ex3.16, Page 107

x<-1:7
w<-c(0.01,0.03,0.13,0.25,0.39,0.17,0.02)

#Mean, m=sum of(xp(x))
m<-weighted.mean(x,w)
print(paste("Population mean =",m))
```

R code Exa 3.17 Expected value of X

```
#Ex3.17, Page 107

x<-0:10
w<-c
    (0.002,0.001,0.002,0.005,0.02,0.04,0.18,0.37,0.25,0.12,0.01)

#Mean, m= sum of(xp(x))
m<-weighted.mean(x,w)
print(paste("Population mean, E(X) =",m))
```

R code Exa 3.23 Expected value of a function

```
#Ex3.23, Page 110

comp<-c(0,0.1,1,0.2,2,0.3,3,0.4)
```

```
row.names<-c("X","p(X)")
column.names<-c("","","","")
matrix.names<-c("X->Number of computers sold")
p<-array(c(comp),dim=c(2,4,1),dimnames=list(row.names,column.names,matrix.names))
print(p)

x<-0:3
w<-c(0.1,0.2,0.3,0.4)
#h(x)<-profit associated with selling x units<-revenue-cost<-800x-900
print(paste("Expected profit:$",weighted.mean(800*x-900,w)))
```

R code Exa 3.24 The variance of X

```
#Ex3.24, Page 111
#Answers may slightly vary due to rounding off of values

w<-c(0.3,0.25,0.15,0.05,0.1,0.15)
x<-c(1,2,3,4,5,6)

#Mean, m= sum of(xp(x))
m<-weighted.mean(x,w)
print(paste("Mean =",m))

#Variance, v= sum of(x-m)p(x)
v<-weighted.mean((x-m)^2,w)
print(paste("Variance =",v))

#Standard deviation, sd= square root(variance)
print(paste("Standard deviation=", sqrt(v)))
```

R code Exa 3.25 The variance of X

```r
#Ex3.25, Page 112

#Alternative formula for variance, v= E(X^2)-(m^2)
x<-1:6
w<-c(0.3,0.25,0.15,0.05,0.1,0.15)

#E(X^2)= sum of (X^2)p(X)
e<-weighted.mean(x^2,w)
print(paste("E(X^2)=",e))

#Variance
mean<-weighted.mean(x,w)
print(paste("Variance through alternative formula =",e-(mean^2)))
```

R code Exa 3.26 Rules of variance

```r
#Ex3.26, Page 113

x<-0:3
w<-c(0.1,0.2,0.3,0.4)

#Mean, m = sum of (xp(x))
m<-weighted.mean(x,w)
print(paste("Mean, E(X)=",m))

#To find E(X^2)
e<-weighted.mean(x^2,w)
print(paste("E(X^2)=",e))

```

```
#To find V(X)
v<-e-(m^2)
print(paste("V(X)=",v))

#Profit function , h(X)=800X-900
#Variance
print(paste("Variance=",(800^2)*v))

#Standard deviation
print(paste("Standard deviation =",sqrt((800^2)*v)))
```

R code Exa 3.29 Binomial experiment

```
#Ex3.29, Page 115
#Number of licensed restaurants
n<-50
#Number of restaurants having at least one serious
   health code violation
f<-15
#Number of restaurants having no serious health code
    violations
s<-35
#P(s on first trial)
p1<-s/n;
print(paste("P(s on first trial)=",p1))

#P(s on second trial)=P(ss)+P(fs)
p2<-(s/n)*((s-1)/(n-1))+(f/n)*(s/(n-1))
print(paste("P(s on second trial)=",p2))
print(paste("Similarly, P(s on ith trial) = 0.7
   for i=3,4,5"))

#P(s on fifth trial|ssss)
p3<-(s-4)/(n-4)
print(paste("P(s on fifth trial|ssss)=",p3))
```

```
#P(s on fifth trial|ffff)
p4<-s/(n-4)
print(paste("P(s on fifth trial|ffff)=",p4))
```

R code Exa 3.30 Binomial experiment

```
#Ex3.30, Page 116
#Number of licensed drivers
n<-500000
#Number of licensed drivers who are insured
s<-400000
#P(s on second  trial after s on first trial)
p1<-(s-1)/(n-1)
print(paste("P(s on 2|s on 1)=",p1))

#P(s on tenth trial after s on ninth trial)
p2<-(s-9)/(n-9)
print(paste("P(s on 10|s on 9)=",p2))

print(paste("The experiment is binomial with n=10
    and p=0.8"))
```

R code Exa 3.31 Binomial random variable and distribution

```
#Ex3.31, Page 118
#Answers may slightly vary due to rounding off of
    values

#n=6, p=0.5

#P(X=3)
print(paste("P(X=3)=",dbinom(3,size=6,prob=0.5)))
```

```
#Probability that at least 3 three prefer S is
print(paste("P(X>=3)=",pbinom(3,size=6,prob=0.5)))

#Probability that at most one prefers S is
print(paste("P(X<=1)=",pbinom(1,size=6,prob=0.5)))
```

R code Exa 3.32.1 Cumulative distribution function

```
#Example 3.32.1, Page 118
#Binomial distribution: n=15, p=2
#The answers may slightly vary due to rounding off
   of values

#Probability that at most 8 copies fail the test
#P(X<=8)
a=pbinom(8,size=15,prob=0.2)
print(paste("The probability that at most 8 copies
   fail the test is", a))
```

R code Exa 3.32.2 Using binomial tables

```
#Example 3.32.2, Page 118
#Binomial distribution: n=15, p=0.2
#The answers may slightly vary due to rounding off
   of values

#Probability that exactly 8 copies fail
#P(X=8)=P(X<=8)-P(X<=7)
a<-0.999
b=pbinom(7,size=15,prob=0.2)
print(paste("The probability that exactly 8 copies
   fail the test", a-b))
```

R code Exa 3.32.3 Using binomial tables

```
#Example 3.32.3, Page 119
#Binomial distribution: n=15, p=0.2
#The answers may slightly vary due to rounding off
   of values

#Probability that at least 8 copies fail
#P(X>=8)=1-P(X<=7)
b<-0.996
c=1-b
print(paste("The probability that at least 8 copies
   fail", c))
```

R code Exa 3.32.4 Using binomial tables

```
#Example 3.32.4, Page 119
#Binomial distribution: n=15, p=0.2
#The answers may slightly vary due to rounding off
   of values

#Probability that between 4 to 7 copies fail
#P(4<=X<=7)=P(X=4)+P(X=5)+P(X=6)+P(X=7)
d=pbinom(7,15,0.2)-pbinom(3,15,0.2)
print(paste("The probability that between 4 to 7
   copies fail is", d))
```

R code Exa 3.33 Using binomial tables

```
#Ex3.33, Page 119
#Answers may vary slightly due to rounding off of
   values

#n=20
#P(X>=5) when p=0.10
p1<-1-pbinom(4,size=20,prob=0.1)
print(paste("P(X>=5 when p=.1)=",p1))

#P(X<=4) when p=0.2
p2<-pbinom(4,size=20,prob=0.2)
print(paste("P(X<=4 when p=.2)=",p2))
```

R code Exa 3.34 Mean and variance of X

```
#Ex3.34,Page 120
n<-10
p<-0.75

#Mean, E(x)=np
e<-n*p
print(paste("Mean=",e))

#Variance, V(X)=npq
v<-n*p*(1-p)
print(paste("Variance=",v))

#Standard deviation=sqrt(V(X))
s<-sqrt(v)
print(paste("Standard deviation=",s))

#P(X= 7or8)=P(X=7)+P(X=8)
prob1=dbinom(0:7,size=10,prob=0.75)
prob2=dbinom(0:8,size=10,prob=0.75)
c<-max(prob1)+max(prob2)
```

```
print(paste("P(X= 7or8)=",c))
```

R code Exa 3.35 Hypergeometric distribution

```
#Ex3.35, Page 123
#Answers may slightly vary due to rounding off of values

N<-20
n<-5
M<-12

#P(X=2)=h(2,5,12,20)
x<-2
h<-(choose(M,x)*choose(N-M,n-x))/choose(N,n)
print(paste("h(2;5,12,20)=",h))
```

R code Exa 3.36 Hypergeometric distribution

```
#Ex3.36, Page 124
#Answers may slightly vary due to rounding off of values

N<-25
n<-10
M<-5

#P(X=2)=h(2,10,5,25)
x<-2
h1<-(choose(M,X)*choose(N-M,n-x))/choose(N,n)
print(paste("P(X=2)=",h1))

#P(X<=2)=P(X=0)+P(X=1)+P(X=2)
```

```
s<-0
for(i in 0:2){
  h2<-(choose(M,i)*choose(N-M,n-i))/choose(N,n)
  s<-s+h2
}
print(paste("P(X<=2)=",s))
```

R code Exa 3.37 Mean and variance of hypergeometric distribution

```
#Ex3.37, Page 125

n<-10
M<-5
N<-25
p<-M/N

#E(X)=np
e<-n*p
print(paste("Mean=",e))

#V(X)=npq
v<-n*p*(1-p)
print(paste("Variance without replacement=",v))

#Estimate of N when M=5, n=10 and x=16
M<-100
n<-40
x<-16
ne<-(M*n)/x
print(paste("Estimate is", ne))
```

R code Exa 3.38 Negative binomial distribution

```
#Ex3.38, Page 126
#Answers may slightly vary due to rounding off of
    values

#P(X=10)=nb(10;5,0.2)
prob1=dnbinom(0:10,size=5,prob=0.2)
print(paste("P(X=10)=",max(prob1)))

#P(X<=10)
s<-pnbinom(0:10,size=5,prob=0.2)
print(paste("P(X<=10)=",max(s)))
```

R code Exa 3.39 Poisson probability distribution

```
#Ex3.39, Page 129
#Answers may slightly vary due to rounding off of
    values

u<-4.5
#Poisson distribution
#P(X=5)
p1<-dpois(5,lambda=u)
print(paste("P(X=5)=",p1))

#P(X<=5)
p2<-ppois(5,lambda=u)
print(paste("P(X<=5)=",p2))
```

R code Exa 3.40 Poisson distribution as a limit

```
#Ex3.40, Page 129
#Answers may vary slightly due to rounding off of
    values
```

```
n<-400
p<-0.005

#P(X=1):b(x;n,p)->p(x;u)
u<-n*p
p1<-dpois(1,lambda=u)
print(paste("P(X=1)=",p1))

#P(X<=3)
p2<-ppois(3,lambda=u)
print(paste("P(X<=3)=",p2))
```

R code Exa 3.41 Mean and variance in Poisson distribution

```
#Ex3.41, Page 130
#Answer may slightly vary due to rounding off of
    value

u<-4.5
#Standard deviation
sd<-sqrt(u)
print(paste("Standard deviation=",sd))
```

R code Exa 3.42 Poisson process

```
#Ex3.42, Page 131
#Answer may slightly vary due to rounding off of
    value

alpha<-6
t<-0.5
l<-alpha*t
```

```
#P(X>=1)=1-P(X=0)
p<-1-dpois(0,lambda=1)
print(paste("P(X>=1)=",p))
```

Chapter 4

Continuous random variables and probability distributions

R code Exa 4.4 Probability density function

```
#Example 4.4, Page 139*
#n=10,M=5,N=25

#f(x)>=0
#Area under density curve
a<-(1/360)*360

#P(90<=X<=180)
f1<-function(x) {(x^0)/360}
p1<-integrate(f1,lower=90,upper=180)
print(paste("Probability that angle is between 90
   and 180 is",p1$value))

#P(0<=X<=90)+P(270<=X<360)
f2<-function(x) {(x^0)/360}
p2<-integrate(f2,lower=0,upper=90)
f3<-function(x) {(x^0)/360}
p3<-integrate(f3,lower=270,upper=360)
p4<-p2$value+p3$value
```

```
print(paste("P(angle of occurrence is within 90 degrees of reference line)=",p4))
```

R code Exa 4.5 Uniform distribution

```
#Example 4.5, Page 141
#The answers may slightly vary due to rounding off of values

#Taking upper limit as 1000 instead of infinity
integrand<-function(x) {.15*exp(-.15*(x-.5))}
#f(x)>=0
#To show that integral from negative infinity to infinity is 1
c<-integrate(integrand,lower=0.5,upper=Inf)
#To access the list of values of integrate by name
print(paste("Value of integral is",c$value))

#To find P(X<5)
d<-integrate(integrand,lower=0.5,upper=5)
print(paste("Thus, probability that headway time is atmost 5 sec:",d$value))
```

R code Exa 4.7 Computing probabilities

```
#Ex4.7, Page 145*

#if x<0,          F(X)=0
#if 0<=x<=2,      F(X)=(x/8)+(3/16)(x^2)
#if x>2,          F(X)=1

load<-function(x){
  f<-(x/8)+(3/16)*(x^2)
```

```
  return(f)
}

#P(1<=X<=1.5)=P(X=1.5)-P(X=1)
p1<-load(1.5)-load(1)
print(paste(”P(1<=X<=1.5)=”,p1))

#P(X>1)=1-P(X<=1)
p2<-1-load(1)
print(paste(”P(load exceeds 1)=”,p2))
```

R code Exa 4.10 Expected values

```
#Ex4.10, Page 148

#if 0<=x<=1        f(x)= (3/2)(1-x^2)
#otherwise         f(x)= 0

f1<-function(x) {(3/2)*x*(1-x^2)}
p1<-integrate(f1,lower=0,upper=1)
print(paste(”E(X)=”,p1$value))
```

R code Exa 4.11 Expected value of a function

```
#Ex4.11, Page 149
#if 0<=x<=1        f(x)=1
#otherwise         f(x)=0

#h(X)=max(1-X,X)
#if 0<=X<1/2       h(X)=1-X
#if 1/2<=X<=1      h(X)=X

#f<-function(x) {max(x,1-x)*f(x)}
```

```
#integrate(f,lower=0,upper=1)

f1<-function(x) {(1-x)*1}
p1<-integrate(f1,lower=0,upper=1/2)

f2<-function(x) {x*1}
p2<-integrate(f2,lower=0.5,upper=1)

p3<-p1$value+p2$value
print(paste("E[h(X)]=",p3))
```

R code Exa 4.12 Variance and standard deviation

```
#Ex4.12, Page 150

#E(X)
e<-3/8

#E(X^2)
e2<-function(x)  {(3/2)*(x^2)*(1-(x^2))}
p1<-integrate(e2,lower=0,upper=1)
print(paste("E(X^2)=",p1$value))

#V(X)
v<-(p1$value)-(e^2)
print(paste("V(X)=",v))

#Standard deviation
sd<-sqrt(v)
print(paste("Standard deviation =",sd))
```

R code Exa 4.13.a Standard normal distribution

```
#Example 4.13.a, Page 154
#The answers may slightly vary due to rounding off
   of values

#P(Z<=1.25)
a<-pnorm(1.25,0,1)
print(paste("The standard normal probability is",a))

#To plot the standard normal distribution bell curve
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.4), xlab
    = "z", ylab="f(z)",lwd=2, col="black")
z = 1.25
pnorm(z)
x = c(-3, seq(-3, z, by=.001), z)
#plot(x)
y = c(0, dnorm(seq(-3, z, by=.001)), 0)
#plot(y)
polygon(x, y, col="red")
```

R code Exa 4.13.b Standard normal distribution

```
#Example 4.13.b, Page 154
#The answers may slightly vary due to rounding off
   of values

#P(Z>1.25)
#Variable to store result(1-P(Z<=1.25))
a=1-pnorm(1.25,0,1)
print(paste("The standard normal probability is", a)
   )


#To plot the standard normal distribution bell curve
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.4), xlab
    = "z", ylab="f(z)",lwd=2, col="black")
```

```
z = 1.25
pnorm(z)
x = c(z, seq(z, 3, by=.001), 3)
#plot(x)
y = c(0, dnorm(seq(z, 3, by=.001)), 0)
#plot(y)
polygon(x, y, col="red")
```

R code Exa 4.13.c Standard normal distribution

```
#Example 4.13.C, Page 154
#The answers may slightly vary due to rounding off
   of values

#P(Z<=-1.25)=P(Z>=1.25)
#Variable to store result(1-P(Z<=1.25))
a=1-pnorm(1.25,0,1)
print(paste("The standard normal probability is", a)
   )

#To plot the standard normal distribution bell curve
x=seq(-3,3,length=200)
y=dnorm(x)
plot(x,y,type="l", lwd=2, col="black")
x=seq(-3,-1.25,length=200)
y=dnorm(x)
polygon(c(-3,x,-1.25),c(0,y,0),col="red")

#To plot the standard normal distribution bell curve
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.4), xlab
    = "z", ylab="f(z)",lwd=2, col="black")
z = -1.25
pnorm(z)
x = c(-3, seq(-3, z, by=.001), z)
#plot(x)
```

```
y = c(0, dnorm(seq(-3, z, by=.001)), 0)
#plot(y)
polygon(x, y, col="red")
```

R code Exa 4.13.d Standard normal distribution

```
#Example 4.13.d, Page 154
#The answers may slightly vary due to rounding off of values

#P(-0.38<=Z<=1.25)
#Variable to store result(1-P(Z<=1.25))
a<-pnorm(1.25,0,1)-pnorm(-0.38,0,1)
print(paste("The standard normal probability is", a))

#To plot the standard normal distribution bell curve
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.4), xlab = "z", ylab="f(z)",lwd=2, col="black")
z1 = 1.25
z2 = -0.38
pnorm(z)
x = c(z2, seq(z2, z1, by=.001), z1)
#plot(x)
y = c(0, dnorm(seq(z2, z1, by=.001)), 0)
#plot(y)
polygon(x, y, col="red")
```

R code Exa 4.14 Percentiles of standard normal distribution

```
#Ex4.14, Page 155
#Answers may slightly vary due to rounding off of values
```

```

print(paste("99th percentile:",qnorm(0.99)))
print(paste("1st percentile:",qnorm(0.01)))

#Blue=> 99th percentile
#Red=>  1st percentile
plot.new()
title(main="Blue: 99th percentile
Red: 1st percentile")
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.5), xlab
    = "z", ylab="f(z)")
prob1 = 0.99
x1= c(-3, seq(-3, qnorm(prob1), by=.001), qnorm(
   prob1))
y1= c(0, dnorm(seq(-3, qnorm(prob1), by=.001)), 0)
polygon(x1, y1, col="blue")

par(new=TRUE)

prob2 = 0.01
x2= c(-3, seq(-3, qnorm(prob2), by=.0001), qnorm(
   prob2))
y2= c(0, dnorm(seq(-3, qnorm(prob2), by=.0001)), 0)
polygon(x2, y2, col="red")
```

R code Exa 4.15 z critical values

```
#Ex4.15, Page 156
#Answers may slightly vary due to rounding off of
   values

#z(.05)
x<-0.05
z<-100*(1-x)

```

```
plot.new()
title(main="Red: z(0.05)
Blue: -z(0.05)")

#z(.05)
print(paste(z,"th percentile:",qnorm(z/100)))
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.5), xlab = "z", ylab="f(z)")
prob = z/100
x = c(-3, seq(-3, qnorm(prob), by=.001), qnorm(prob))
y = c(0, dnorm(seq(-3, qnorm(prob), by=.001)), 0)
polygon(x, y, col="red")

#-z(.05)
print(paste("-z(.05)",-qnorm(z/100)))
x = c(-3, seq(-3, -qnorm(prob), by=.001), -qnorm(prob))
y = c(0, dnorm(seq(-3, -qnorm(prob), by=.001)), 0)
polygon(x, y, col="blue")
```

R code Exa 4.16 Nonstandard normal distributions

```
#Ex4.16, Page 158
#Answers may slightly vary due to rounding off of values

#P(1.00<=X<=1.75)
#Mean=>1.25, Standard deviation=>0.46

title(main="P(1.00<=X<=1.75)")
a<-pnorm(1.00,mean=1.25,sd=0.46)
b<-pnorm(1.75,mean=1.25,sd=0.46)
c<-b-a
print(paste("P(1.00<=X<=1.75)=",c))
```

```
#To plot the standard normal distribution bell curve
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.4), xlab
    = "z", ylab="f(z)",lwd=2, col="black")
z<-1.75
# Cumulative density function of the standard normal
    distribution
pnorm(z)
x<-c(1.00, seq(1.00, z, by=.001), z)
#plot(x)
y<-c(0, dnorm(seq(1.00, z, by=.001)), 0)
#plot(y)
polygon(x, y, col="red")

#P(X>2)
d<-1-pnorm(2,mean=1.25,sd=0.46)
print(paste("P(X>2)=",d))
```

R code Exa 4.17 Nonstandard normal distributions

```
#Ex4.17, Page 158
#Answers may slightly vary due to rounding off of
    values

#P(-1.00<=Z<=1.00)
a<-pnorm(1.00,0,1)-pnorm(-1.00,0,1)
print(paste("P(X is within 1 standard deviation of
    its mean):",a))

#P(-2.00<=Z<=2.00)
b<-pnorm(2.00,0,1)-pnorm(-2.00,0,1)
print(paste("P(X is within 2 standard deviation of
    its mean):",b))

#P(-3.00<=Z<=3.00)
```

```
c<-pnorm(3.00,0,1)-pnorm(-3.00,0,1)
print(paste("P(X is within 3 standard deviation of
    its mean):",c))
```

R code Exa 4.18 Percentiles of arbitrary normal distribution

```
#Ex4.18, Page 159
#Answers may slightly vary due to rounding off of
    values

m<-64
sd<-0.78
#P(X>c)=0.05
p<-0.005
#P(X<=c)=1-P(X>c)
s<-(1-p)*100
print(paste("c is the",s,"th percentile of the
    normal distribution with mean=64 and sd=0.78"))
q<-qnorm(s/100)
print(paste("Therefore",s,"th percentile:",q))

c<-m+q*sd
print(paste("c=",c,"oz"))

#Plotting 99.5th percentile
plot.new()
title(main="99.5th percentile",sub="Shaded area
    =>0.995")
curve(dnorm, xlim = c(-3, 3), ylim = c(0, 0.5), xlab
    = "z", ylab="f(z)")
prob = 0.995
x1= c(-3, seq(-3, qnorm(prob), by=.001), qnorm(prob)
    )
y1= c(0, dnorm(seq(-3, qnorm(prob), by=.001)), 0)
polygon(x1, y1, col="blue")
```

R code Exa 4.20 Approximating binomial distribution

```
#Ex4.20, Page 161
#Answers may vary slightly due to rounding off of
    values

n<-50
p<-25/100
q<-1-p
m<-12.5
sd<-3.06
x<-n*p
y<-sqrt(n*p*q)
z<-n*q
if(x>=10 && z>=10){
  #P(X<=10)
  a<-pnorm(10+0.5,mean=x,sd=y)
  print(paste("P(X<=10)=",a))

  #P(5<=X<=15)
  b<-pnorm(15+0.5,mean=x,sd=y)-pnorm(5-0.5,mean=x,sd
     =y)
  print(paste("P(5<=X<=15)=",b))
}
```

R code Exa 4.21 Exponential distribution

```
#Ex4.21, Page 166
#Answers may slightly vary due to rounding off of
    values

```

```
#E(X)=6MPa=1/lambda
e<-6
lambda<-1/e
#Probability that stress range is at most 10MPa
p<-pexp(10,lambda)
print(paste("P(stress range is at most 10MPa)=",p))

#Probability that stress range is between 5 and 10
    MPa
q<-pexp(10,lambda)-pexp(5,lambda)
print(paste("P(stress range is between 5 and 10MPa)=
    ",q))
```

R code Exa 4.22 Exponential distribution

```
#Ex4.22, Page 167
#Answer may slightly vary due to rounding off of
    value

alpha<-0.5

#P(X>2)=1-P(X<=2)
x<-2
p<-1-pexp(x,alpha)
print(paste("P(X>2)=",p))

t<-1/alpha
print(paste("The expected time between successive
    calls is",t,"days"))
```

R code Exa 4.23 Gamma distribution

```
#Ex4.23, Page 169
```

```
#Answers may slightly vary due to rounding off of values

alpha<-2

#P(3<=X<=5)
a<-pgamma(5,shape=alpha)-pgamma(3,shape=alpha)
print(paste("P(3<=X<=5)=",a))

#P(X>4)=1-P(X<=4)
b<-1-pgamma(4,shape=alpha)
print(paste("P(reaction time is more than 4 sec)=",b))
```

R code Exa 4.24 Gamma distribution

```
#Ex4.24, Page 169
#Answers may slightly vary due to rounding off of values

#alpha
a<-8
#beta
b<-15

#E(X)
e<-a*b
print(paste("E(X)=",e))

#V(X)
v<-a*(b^2)
print(paste("V(X)=",v))

#SD
sd<-sqrt(v)
```

```
print(paste("Standard deviation =",sd))

#P(60<=X<=120)=P(X<=120)-P(X<=60)
p1<-pgamma(120/b,shape=a)-pgamma(60/b,a)
print(paste("P(mouse survives between 60 and 120
    weeks)=",p1))

#P(X>=30)=1-P(X<30)
p2<-1-pgamma(30/b,a)
print(paste("P(mouse survives at least 30 weeks)=",
    p2))
```

R code Exa 4.25 Weibull distribution

```
#Ex4.25, Page 173
#Answers may slightly vary due to rounding off of
    values

alpha<-2
beta<-10

#P(X<=10)
p1<-pweibull(10,alpha,beta)
print(paste("P(X<=10)=",p1))

#P(X<=25)
p2<-pweibull(25,alpha,beta)
print(paste("P(X<=25)=",p2))

#To find c value which separates 5% of all engines
    having largest NOx emissions from remaining 95%
per<-5/100
c<-sqrt((-1)*(10^2)*log(per))
print(paste(c,"is the 95th percentile of the
    emission distribution"))
```

R code Exa 4.26 Weibull distribution

```
#Ex4.26, Page 173
#Answers may slightly vary due to rounding off of
    values

#alpha
a<-1.3
#beta
b<-0.8
#gamma
g<-4

#P(5<=X<=6)
p1<-pweibull(6-g,a,b)-pweibull(5-g,a,b)
print(paste("P(air void volume of a specimen is
    between 5% and 6%)=",p1))
```

R code Exa 4.27 Lognormal distribution

```
#Ex4.27, Page 175
#Answers may slightly vary due to rounding off of
    values

m<-0.353
s<-0.754

e<-exp(m+(s^2)/2)
print(paste("E(X)=",e))

v<-exp(2*m+(s^2))*(exp(s^2)-1)
```

```
print(paste("V(X)=",v))

#P(1<=X<=2)
l1<-log(1,base=exp(1))
l2<-log(2,base=exp(1))
p<-pnorm(l2,m,s)-pnorm(l1,m,s)
print(paste("P(maximum pit depth is between 1 and 2
    mm)=",p))

#To find c such that only 1% of all specimens have a
    maximum pit depth excceding c
per<-1/100
c<-exp(1)^(qnorm(1-per)*s+m)
print(paste(c,"is the 99th percentile of the maximum
    pit depth distribution"))
```

R code Exa 4.28 Beta distribution

```
#Ex4.28, Page 177

#Optimistic time
A<-2

#Pessimistic time
B<-5

#alpha
a<-2
#beta
b<-3

#E(X)
e<-A+(B-A)*(a/(a+b))
print(paste("Mean=",e))

```

```
c<-(1/(B-A))*((gamma(a+b))/(gamma(a)*gamma(b)))
f1<-function(x) {c*(((x-A)/(B-A))^(a-1))*(((B-x)/(B-A))^(b-1))}
p<-integrate(f1,lower=2,upper=3)
print(paste("P(it take atmost 3 days to lay the foundation)=",p$value))
```

R code Exa 4.29 Probability plots

```
#Ex4.29, Page 181

old.par<-par(mfrow=c(1,2))

#To plot first sample
Percentage1<-c(5, 15, 25, 35, 45)
z_percentile1<-c(qnorm(5/100), qnorm(15/100), qnorm(25/100), qnorm(35/100), qnorm(45/100))
Sample_observation1<-c(-1.91, -1.25, -0.75, -0.53, 0.2)
df1<-data.frame(Percentage1, z_percentile1, Sample_observation1)
print(df1)
plot(z_percentile1,Sample_observation1)

#To plot second sample
Percentage2<-c(55, 65, 75, 85, 95)
z_percentile2<-c(qnorm(55/100), qnorm(65/100), qnorm(75/100), qnorm(85/100), qnorm(95/100))
Sample_observation2<-c(0.35, 0.72, 0.87, 1.40, 1.56)
df2<-data.frame(Percentage2, z_percentile2, Sample_observation2)
print(df2)
plot(z_percentile2,Sample_observation2)

par(old.par)
```

R code Exa 4.30 Probability plots

```
#Ex4.30, Page 182

old.par<-par(mfrow=c(1,1))
n<-20
z_percentile<-c(qnorm((1-0.5)/n), qnorm((2-0.5)/n),
   qnorm((3-0.5)/n), qnorm((4-0.5)/n), qnorm((5-0.5)
   /n),
                qnorm((6-0.5)/n), qnorm((7-0.5)/n),
                   qnorm((8-0.5)/n), qnorm((9-0.5)/n
                   ), qnorm((10-0.5)/n),
                qnorm((11-0.5)/n), qnorm((12-0.5)/n)
                   , qnorm((13-0.5)/n), qnorm
                   ((14-0.5)/n), qnorm((15-0.5)/n),
                qnorm((16-0.5)/n), qnorm((17-0.5)/n)
                   , qnorm((18-0.5)/n), qnorm
                   ((19-0.5)/n), qnorm((20-0.5)/n))
Sample_observation<-c(24.46, 25.61, 26.25, 26.42,
   26.66, 27.15, 27.31, 27.54, 27.74, 27.94, 27.98,
   28.04, 28.28, 28.49, 28.50, 28.87, 29.11, 29.13,
   29.50, 30.88)
df<-data.frame(z_percentile, Sample_observation)
print(df)
plot(z_percentile,Sample_observation)

par(old.par)
```

R code Exa 4.31 Beyond normality

```
#Ex4.31, Page 186
```

```

old.par<-par(mfrow=c(1,1))

Percentile<-c(-2.97, -1.82, -1.25, -0.84, -0.51, -0.23, 0.05, 0.33, 0.64, 1.10)
x<-c(282, 501, 741, 851, 1072, 1122, 1202, 1585, 1905, 2138)
lnx<-c(log(282,base=exp(1)),log(501,base=exp(1)), log(741,base=exp(1)), log(851,base=exp(1)), log(1072,base=exp(1)),
         log(1122,base=exp(1)), log(1202,base=exp(1)), log(1585,base=exp(1)), log(1905,base=exp(1)), log(2138,base=exp(1)))
df<-data.frame(Percentile,x,lnx)
print(df)
plot(Percentile,lnx)

par(old.par)
```

Chapter 5

Joint probability distributions and random samples

R code Exa 5.1 Joint probability mass function

```
#Ex5.1, Page 194

m<-matrix(c(0.20,0.05,0.10,0.15,0.20,0.30),nrow=2,
   ncol=3)
rownames(m)<-c("100","250")
colnames(m)<-c("0","100","200")
print(m)

#To find P(Y>=100)
prob<-m[1,2]+m[1,3]+m[2,2]+m[2,3]
print(paste("P(Y>=100)=",prob))
```

R code Exa 5.2 Marginal probability mass function

```
#Ex5.2, Page 195

```

```r
m<-matrix(c(0.20,0.05,0.10,0.15,0.20,0.30),nrow=2,
   ncol=3)
rownames(m)<-c("100","250")
colnames(m)<-c("0","100","200")
print(m)
cat("\n")

px_100<-apply(m,1,sum)[1]
px_250<-apply(m,1,sum)[2]
cat("px(100)=",px_100,"\n")
cat("px(250)=",px_250,"\n\n")

cat("Marginal pmf of X:\n")
cat("p(x)=",px_100,"when x=100,250\n")
cat("p(x)=0 otherwise\n\n")

py_0<-apply(m,2,sum)[1]
py_100<-apply(m,2,sum)[2]
py_200<-apply(m,2,sum)[3]
cat("py(0)=",py_0,"\n")
cat("py(100)=",py_100,"\n")
cat("py(200)=",py_200,"\n\n")

cat("Marginal pmf of Y:\n")
cat("p(y)=",py_0,"when y=0,100\n")
cat("p(y)=",py_200,"when y=200\n")
cat("p(y)=0 otherwise\n\n")

#To find P(Y>=100)
prob<-py_100+py_200
cat("P(Y>=100)=",prob)
```

R code Exa 5.3 Joint probability density function

```r
#Ex5.3, Page 196
```

```
#Answers may ary slightly due to rounding off of
    values

f<-function(x,y) {(6/5)*(x+y^2)}

#To check whether pdf is legitimate
check_val<-integrate(function(y) {sapply(y,function(
    y) {integrate(function(x) f(x,y),0,1)$value})
    },0,1)
if(check_val$value==1) cat("PDF is legitimate\n\n")

#P(0<=X<=1/4,0<=Y<=1/4)
low1<-0
high1<-1/4
low2<-0
high2<-1/4
integral<-integrate(function(y) {sapply(y,function(y
    ) {integrate(function(x) f(x,y),low1,high1)$value
    })},low2,high2)
print(paste("P(0<=X<=1/4,0<=Y<=1/4)=",integral$value
    ))
```

R code Exa 5.4 Marginal probability density function

```
#Ex5.4, Page 197

f<-function(x,y) {(6/5)*(x+y^2)}

#Finding P(1/4<=Y<=3/4) using marginal pdf of Y
xlow<-0
xhigh<-1
ylow<-1/4
yhigh<-3/4
prob<-integrate(function(y) {sapply(y,function(y) {
    integrate(function(x) f(x,y),xlow,xhigh)$value})
```

```
    },ylow,yhigh)
print(paste("P(1/4<=Y<=3/4)=",prob$value))
```

R code Exa 5.5 Marginal probability density function

```
#Ex5.5, Page 198

#Joint pdf
f<-function(x,y) {24*x*y}

val<-Vectorize(function(x) {sapply(x,function(z) {
    integrate(function(y) f(x,y),0,1-z)$value})})
check_val<-integrate(val,0,1)$value
if(check_val==1)  print(paste("PDF is legitimate"))

#Probability that two types of nuts together make up
     50% of the can
#Taking A={(x,y):0<=x<=1,0<=y<=1,x+y<=0.5}
xlow<-0
xhigh<-0.5
integral<-Vectorize(function(x) {sapply(x,function(z
    ) {integrate(function(y) f(x,y),0,0.5-z)$value})
    })
prob<-integrate(integral,xlow,xhigh)
print(paste("P((X,Y)EA) =",prob$value))
```

R code Exa 5.8 Independent random variables

```
#Ex5.8, Page 200
#Answers may vary slightly due to rounding off of
    values

#Expected lifetimes are 1000 and 1200 hours
```

```
lambda1<-(1/1000)
lambda2<-(1/1200)

#P(X1>=1500)
p1<-exp(-(lambda1)*1500)

#P(X2>=1500)
p2<-exp(-(lambda2)*1500)

#Joint PDF
#P(X1>=1500,X2>=1500)
p<-p1*p2
print(paste("P(both component lifetimes are at least
    1500 hours)=",p))
```

R code Exa 5.9 Multinomial distribution

```
#Ex5.9, Page 201
#Answer may slightly vary due to rounding off of
    value

#Respective number and probabilities
n<-c(2,5,3)
p<-c(0.25,0.5,0.25)

#Multinomial PMF
m<-dmultinom(n,prob=p)
print(paste("Multinomial PMF:",m))
```

R code Exa 5.12 Conditional distributions

```
#Ex5.12, Page 203
```

```
#Answers may slightly vary due to rounding off of values

#Function when X=0.8: (1/34)(24+30(y^2))

#P(Y<=0.5) given X=0.8
f1<-function(y) {(1/34)*(24+30*(y^2))}
f<-integrate(f1,lower=0,upper=0.5)
print(paste("P(walk-up facility is busy atmost half the time)=",f$value))

#E(Y) given that X=0.8
f2<-function(y) {y*(1/34)*(24+30*(y^2))}
p<-integrate(f2,lower=0,upper=1)
print(paste("Expected proportion of time that walk-up facility is busy:",p$value))
```

R code Exa 5.14 Expected values

```
#Ex5.14, Page 207

#Joint pdf
f<-function(x,y) {24*x*y}

almonds<-1
cashews<-1.5
peanuts<-0.5

#Total cost of the contents of a can
h<-function(x,y) {almonds*x+cashews*y+(1-x-y)*peanuts}

#Expected total cost
integral<-Vectorize(function(x) {sapply(x,function(z) {integrate(function(y) h(x,y)*f(x,y),0,1-z)$
```

```
   value})})
prob<-integrate(integral,0,1)
print(paste("Expected total cost:$",prob$value))
```

R code Exa 5.15 Covariance

```
#Ex5.15, Page 208

x<-c(100,250)
y<-c(0,100,200)

m1<-matrix(c(0.20,0.05,0.10,0.15,0.20,0.30),nrow=2,
   ncol=3)
rownames(m1)<-c("100","250")
colnames(m1)<-c("0","100","200")
print(m1)
cat("\n")

m2<-matrix(c(0.5,0.5),nrow=1,ncol=2)
rownames(m2)<-c("p(x)")
colnames(m2)<-c("100","250")
print(m2)
cat("\n")

m3<-matrix(c(0.25,0.25,0.5),nrow=1,ncol=3)
rownames(m3)<-c("p(y)")
colnames(m3)<-c("0","100","200")
print(m3)
cat("\n")

x1<-c(0.5,0.5)
j<-1
mu_x<-0

for(i in x){
```

```
    if(j<=length(x1)){
      mu_x<-mu_x+i*x1[j]
      j=j+1
    }
}
cat("mu_X=",mu_x,"\n\n")

y1<-c(0.25,0.25,0.5)
k<-1
mu_y<-0

for(i in y){
  if(k<=length(y1)){
    mu_y<-mu_y+i*y1[k]
    k=k+1
  }
}
cat("mu_Y=",mu_y,"\n\n")

p<-c(0.2,0.1,0.20,0.05,0.15,0.30)
sum<-0
k<-1
for(i in x){
  for(j in y){
    if(k<=length(p)){
      sum=sum+(i-175)*(j-125)*p[k]
      k=k+1
    }
  }
}
print(paste("Cov(X,Y):",sum))
```

R code Exa 5.18 Correlation

```
#Ex5.18, Page 210
```

```
x<-c(-4,4,2,-2)
y<-c(1,-1,2,-2)
xy<-x*y
w<-c(1/4,1/4,1/4,1/4)
m<-weighted.mean(xy,w)
print(paste("E(XY)=",m))

#Cov(XY)=E(XY)-ux.uy
ux<-0
uy<-0
cov<-m-(ux*uy)
print(paste("Cov(XY)=",cov))

print(paste("Complete absence of any linear
    relationship"))

#To scatterplot
df<-data.frame(x,y)
plot(x,y)
```

R code Exa 5.19 Statistics and their distributions

```
#Ex5.19, Page 212
#Answers may vary slightly due to rounding off of
    values

#alpha
a<-2

#beta
b<-5

#Mean
u<-b*gamma(1+(1/a))
```

```
print(paste("Mean, E(X)=",u))

#Variance
v<-(b^2)*{gamma(1+(2/a))-(gamma(1+(1/a)))^2}
print(paste("Variance=",v))

#Standard deviation
s<-sqrt(v)
print(paste("Standard deviation=",s))

#Median
med<-b*(log(2,base=exp(1)))^(1/a)
print(paste("Median=",med))

q<-qweibull(0.99,shape=a,scale=b)
curve(dweibull(x,a,b),from=0,to=q,ylab='density')

if(u>med){
  print(paste("Mean exceeds median because of the
     distribution's positive skew"))
}
```

R code Exa 5.24 Standard error of the mean

```
#Ex5.24, Page 223

n<-25
u<-28000
s<-5000

#E(Mean)
print(paste("E(Mean)=",u))

#E(T0)
t<-n*u
```

```
print(paste("E(T0)=",t))

#Standard deviation of mean
sm<-s/sqrt(n)
print(paste("SD of mean:",sm))

#Standard deviation of T0
st<-s*sqrt(n)
print(paste("SD of T0:",st))

#When sample size increases to n=100
print(paste("When n=100:"))
n<-100
#E(Mean)
print(paste("E(Mean)=",u))

#E(T0)
t<-n*u
print(paste("E(T0)=",t))

#Standard deviation of mean
sm<-s/sqrt(n)
print(paste("SD of mean:",sm))

#Standard deviation of T0
st<-s*sqrt(n)
print(paste("SD of T0:",st))
```

R code Exa 5.25 Normal population distribution

```
#Ex5.25, Page 224
#Answers may slightly vary due to rounding off of
    values

n<-5
```

```
#Mean(min)
m<-1.5

#Standard deviation(min)
sd<-0.35

ut<-n*m
print(paste("uT0:",ut))

vt<-n*(sd^2)
print(paste("vT0:",vt))

st<-sqrt(vt)
print(paste("sT0:",st))

#P(6<=T0<=8)
p1<-pnorm(8,mean=ut,sd=st)-pnorm(6,mean=ut,sd=st)
print(paste("P(6<=T0<=8)=",p1))

#Probability that the sample average time is atmost
    2 min
u<-1.5
s<-sd/sqrt(n)

#P(Mean<=2.0)
p2<-pnorm(2,mean=u,sd=s)
print(paste("P(Mean<=2)=",p2))
```

R code Exa 5.26 Central limit theorem

```
#Ex5.26, Page 225
#Answer may slightly vary due to rounding off of
    value

```

```
n<-50
m<-4
sd<-1.5

ux<-m
sx<-sd/sqrt(n)

#P(3.5<=Mean<=3.8)
p<-pnorm(3.8,mean=ux,sd=sx)-pnorm(3.5,mean=ux,sd=sx)
print(paste("P(3.5<=Mean<=3.8)=",p))
```

R code Exa 5.27 Central limit theorem

```
#Ex5.27, Page 226
#Answer may slightly vary due to rounding off of
    value

n<-100
m<-3.2
sd<-2.4

ux<-m
sx<-sd/sqrt(n)

#P(Mean>4)=1-P(Mean<=4)
p<-1-pnorm(4,mean=ux,sd=sx)
print(paste("P(Mean>4)=",p))
```

R code Exa 5.29 Linear combination

```
#Ex5.29, Page 231
#Answers may vary slightly due to rounding off of
    values
```

```
a1<-3.00
a2<-3.20
a3<-3.40

#Mean
u1<-1000
u2<-500
u3<-300

#Standard deviation
s1<-100
s2<-80
s3<-50

#Y=3.0X1+3.20X2+3.40X3
EY<-a1*u1+a2*u2+a3*u3
VY<-(a1^2)*(s1^2)+(a2^2)*(s2^2)+(a3^2)*(s3^2)
sY<-sqrt(VY)
print(paste("E(Y)=$",EY))
print(paste("V(Y)=$",VY))
print(paste("sY=$",sY))
```

R code Exa 5.30 Difference between two random variables

```
#Ex5.30, Page 232
#Answers may slightly vary due to rounding off of
   values

u1<-22
u2<-26
s1<-1.2
s2<-1.5

#Mean
```

```
e<-u1-u2
print(paste("E(X1-X2)=",e))

#Variance
v<-(s1^2)+(s2^2)
print(paste("V(X1-X2)=",v))

#Standard deviation
sd<-sqrt(v)
print(paste("SD=",sd))
```

R code Exa 5.31 Normal random variables

```
#Ex5.31, Page 232

#Y=3.0X1+3.2X2+3.4X3
u<-5620
s<-429.46

#P(Y>4500)
p<-1-pnorm(4500,mean=u,sd=s)
print(paste("P(revenue exceeds 4500)=",p))
```

Chapter 6

Point estimation

R code Exa 6.1 Some general concepts of point estimation

```
#Ex6.1, Page 241

x<-15
n<-25
p<-x/n
print(paste("Most reasonable estimate is",p))
```

R code Exa 6.2 Some general concepts of point estimation

```
#Ex6.2, Page 241
#Answers may vary slightly due to rounding off of
   values

volt<-c
   (24.46,25.61,26.25,26.42,26.66,27.15,27.31,27.54,27.74,27.94,27.9

#Estimator1
```

```r
x1<-mean(volt)
print(paste("Estimator(1):",x1))

#Estimator2
x2<-median(volt)
print(paste("Estimator(2):",x2))

#Estimator3
x3<-(min(volt)+max(volt))/2
print(paste("Estimator(3):",x3))

#Estimator4
volt<-setdiff(volt,c(volt[1],volt[2],volt[19],volt[20]))
x4<-mean(volt)
print(paste("Estimator(4):",x4))
```

R code Exa 6.3 Some general concepts of point estimation

```r
#Ex6.3, Page 242
#Answers may vary slightly due to rounding off of values

obs<-c(74.33,71.07,73.82,77.42,79.35,82.27,77.75,78.65,77.19,74.69,
        77.25,74.84,60.90,60.75,74.09,65.36,67.84,69.97,68.83,75.09,
        62.54,67.47,72.00,66.51,68.21,64.46,64.34,64.93,67.33,66.08,
        67.31,74.87,69.40,70.83,81.73,82.50,79.87,81.96,79.51,84.12,
        80.61,79.89,79.70,78.74,77.28,79.97,75.09,74.38,77.67,83.73,
        80.39,76.90)
```

```
#Mean
print(paste("Mean=",mean(obs)))

#Variance
print(paste("Point estimate of the population variance:",var(obs)))

#Alternative formula
#To find numerator of s^2
s1<-sum(obs)
s2<-sum(obs^2)
Sxx<-s2-(s1^2)/length(obs)

#Standard deviation
print(paste("Point estimate of the population standard deviation:",sd(obs)))

#Alterative estimator
print(paste("Alternative estimator of population variance:",Sxx/length(obs)))
```

R code Exa 6.9 Standard error of an estimator

```
#Ex6.9, Page 251
#Answers may slightly vary due to rounding off of values

n<-20
sigma<-1.5

#Standard error of mean
sigx<-sigma/sqrt(n)
print(paste("Standard error of mean:",sigx))

```

```
#Estimate of standard deviation
s<-1.462

sx<-s/sqrt(n)
print(paste("Estimated standard error:",sx))
```

R code Exa 6.10 Standard error of an estimator

```
#Ex6.10, Page 251
#Answers may vary slightly due to rounding off of
   values

n<-25
p<-0.6
q<-1-p

#To find estimated standard error
sp1<-sqrt((p*q)/n)
print(paste("Estimated standard error:",sp1))

#pq is largest when
p<-q<-0.5
sp2<-sqrt((p*q)/n)
print(paste("Estimated standard error when pq is
   largest:",sp2))
```

R code Exa 6.11 Standard error of an estimator

```
#Ex6.11, Page 252
#Answers may slightly vary due to rounding off of
   values

```

```
t<-c
   (41.53,18.73,2.99,30.34,12.33,117.52,73.02,223.63,4.00,26.78)

#Mean
m<-mean(t)
print(paste("Mean:",m))

#lambda(l)
l<-1/m
print(paste("Reasonable estimate of lambda:",l))
```

R code Exa 6.13 Method of moments

```
#Ex6.13, Page 257
#Answers may vary slightly due to rounding off of
   values

data<-c
   (152,115,109,94,88,137,152,77,160,165,125,40,128,123,136,101,62,15
n<-length(data)

#Mean
m<-mean(data)
print(paste("Mean:",m))

#Sum of x^2
s<-sum(data^2)/n

#alpha
alpha<-(m^2)/(s-(m^2))
print(paste("Alpha:",alpha))

#beta
```

```
beta<-(s-(m^2))/m
print(paste("Beta:",beta))
```

R code Exa 6.14 Method of moments

```
#Ex6.14, Page 257
#Answers may vary slightly due to rounding off of
   values

goals<-c(0,1,2,3,4,5,6,7,8,9,10)
freq<-c(29,71,82,89,65,45,24,7,4,1,3)

#Mean
m<-weighted.mean(goals,freq)
print(paste("Mean=",m))

#To find sum of x^2
ms<-weighted.mean(goals^2,freq)
print(paste("Sum of x^2/frequency=",ms))

#To find alpha
p_cap<-m/(ms-(m^2))
print(paste("Alpha:",p_cap))

#To find beta
r_cap<-(m^2)/(ms-(m^2)-m)
print(paste("Beta:",r_cap))
```

R code Exa 6.21 The invariance principle

```
#Ex6.21, Page 262
#Answers may slightly vary due to rounding off of
   values
```

```

alpha<-11.9731
beta<-77.0153
m<-beta*gamma(1+(1/alpha))
print(paste("Mean estimate:",m))
```

Chapter 7

Statistical intervals based on a single sample

R code Exa 7.2 Basic properties of confidence intervals

```
#Ex7.2, Page 270
#Answers may slightly vary due to rounding off of
   values

mean<-80
n<-31
sd<-2
error<-qt(0.95,df=n-1)*sd/sqrt(n)

#Left endpoint
l<-mean-error
print(paste("Left endpoint of interval:",l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of interval:",r))
```

R code Exa 7.3 Other levels of confidence

```
#Ex7.3, Page 272
#Answers may slightly vary due to rounding off of
   values

mean<-5.426
n<-40
sd<-0.1

#To find alpha for (100(1-alpha)=90)
alpha<-solve(100,10)

error<-qnorm(1-(alpha/2))*(sd/sqrt(n))

#Left endpoint
l<-mean-error
print(paste("Left endpoint of interval:",l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of interval:",r))
```

R code Exa 7.4 Confidence level and precision and sample size

```
#Ex7.4, Page 273
#Answers may slightly vary due to rounding off of
   values

#For 95% CI
z<-qnorm(1-(0.05/2))

sd<-25
w<-10

```

```
n<-(2*z*sd/w)^2
print(paste("Sample size necessary for CI to have
    width 10 is ",n,"and thus approx. 97"))
```

R code Exa 7.6 Large sample CI for mean

```
#Ex7.6, Page 277-278
#Answers may slightly vary due to rounding off of
    values

data<-c
    (2948,2996,7197,8338,8500,8759,12710,12925,15767,20000,23247,24863
26210,30552,30600,35700,36466,40316,40596,41021,41234,43000,44607,45
45027,45442,46963,47978,49518,52000,53334,54208,56062,57000,57365,60
60265,60803,62851,64404,72140,74594,79308,79500,80000,80000,84000,11
boxplot(data)
print(data)

n<-50

#To find mean
mean<-mean(data)
print(paste("Mean:",mean))

#To find median
med<-median(data)
print(paste("Median:",med))

#To find standard deviation
s<-sd(data)
print(paste("Standard deviation:",s))
```

```
#To find confidence intervals
z<-qnorm(1-(0.05/2))
error<-z*(s/sqrt(n))

#Left endpoint
l<-mean-error
print(paste("Left endpoint of interval:",l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of interval:",r))
```

R code Exa 7.7 Large sample CI for mean

```
#Ex7.7, Page 279

t1<-320
t2<-440

#To find value of s
s<-(t2-t1)/4

w<-5

z<-qnorm(1-(0.05/2))

n<-(z*s/w)^2
print(paste("Sample size:",n))
print(paste("Integer value of sample size:",ceiling(
   n)))
```

R code Exa 7.8 CI for population proportion

```
#Ex7.8, Page 282
#Answer may vary slightly compared to textbook value

n<-48
n1<-16
p<-n1/n

alpha<-0.05
b<-binom.test(16,48,p,conf.level=0.95)
print(b)
```

R code Exa 7.10 Confidence bounds

```
#Ex7.10, Page 283
#Answer may slightly vary due to rounding off of
    value

mean<-17.17
z<-qnorm(0.95)
n<-48
sd<-3.28

#To find lower bound
c<-mean-(z*sd/sqrt(n))
print(paste("Lower confidence bound for mean:",c))
```

R code Exa 7.11 One sample t confidence interval

```
#Ex7.11, Page 288
#Answers may slightly vary due to rounding off of
    values
```

```
data<-c
   (6807.99,7637.06,6663.28,6165.03,6991.41,6992.23,6981.46,7569.75,
         7437.88,6872.39,7663.18,6032.28,6906.04,6617.17,6984.12,7093
         7659.50,7378.61,7295.54,6702.76,7440.17,8053.26,8284.75,7347
         7422.69,7886.87,6316.67,7713.65,7503.33,7674.99)

n<-30

#To find mean
mean<-mean(data)
print(paste("Mean:",mean))

#To find standard deviation
s<-sd(data)
print(paste("Standard deviation:",s))

#To find confidence intervals
t<-qt(1-(0.05/2),df=n-1)
print(paste("Critical value,t:",t))
error<-t*(s/sqrt(n))

#Left endpoint
l<-mean-error
print(paste("Left endpoint of interval:",l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of interval:",r))

#To plot the normal probability plot
qqnorm(data,main="Normal Probability of MOR")
```

R code Exa 7.12 A prediction interval for a single future value

```
#Ex7.12, Page 288
#Answers may slightly vary due to rounding off of
   values

data<-c
   (25.2,21.3,22.8,17.0,29.8,21.0,25.5,16.0,20.9,19.5)

print(data)

n<-10

#To find mean
mean<-mean(data)
print(paste("Mean:",mean))

#To find standard deviation
s<-sd(data)
print(paste("Standard deviation:",s))

#To find confidence intervals
t<-qt(1-(0.05/2),df=n-1)
print(paste("Critical value,t:",t))
error<-t*(s/sqrt(n))

#Left endpoint
l<-mean-error
print(paste("Left endpoint of interval:",l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of interval:",r))
```

R code Exa 7.13 Prediction interval

```
#Ex7.13, Page 290
#Answers may slightly vary due to rounding off of
    values

mean<-21.9
s<-4.134
n<-10

#To find critical value
t<-qt(1-(0.05/2),df=n-1)
print(paste("Critical value,t:",t))

error<-t*s*sqrt(1+(1/n))

#Left endpoint
l<-mean-error
print(paste("Left endpoint of prediction interval:",
    l))

#Right endpoint
r<-mean+error
print(paste("Right endpoint of prediction interval:"
    ,r))
```

R code Exa 7.14 Tolerance intervals

```
#Ex7.14, Page 291
#Answers may vary slightly due to rounding off of
    values

```

```r
data<-c
   (10490,16620,17300,15480,12970,17260,13400,13900,
        13630,13260,14370,11700,15470,17840,14070,14760)

print(data)

n<-16

#To find mean
mean<-mean(data)
print(paste("Mean:",mean))

#To find standard deviation
s<-sd(data)
print(paste("Standard deviation:",s))

#To find tolerance intervals
t1<-2.903
error1<-t1*s
#Left endpoint
l1<-mean-error1
print(paste("Left endpoint of tolerance interval:",
   l1))
#Right endpoint
r1<-mean+error1
print(paste("Right endpoint of tolerance interval:",
   r1))

#To find confidence intervals
t2<-qt(1-(0.05/2),df=n-1)
error2<-t2*(s/sqrt(n))
#Left endpoint
l2<-mean-error2
print(paste("Left endpoint of confidence interval:",
   l2))
#Right endpoint
r2<-mean+error2
print(paste("Right endpoint of confidence interval:"
```

```
,r2))

#To find prediction intervals
t3<-qt(1-(0.05/2),df=n-1)
error3<-t3*s*sqrt(1+(1/n))
#Left endpoint
l3<-mean-error3
print(paste("Left endpoint of prediction interval:",l3))
#Right endpoint
r3<-mean+error3
print(paste("Right endpoint of prediction interval:",r3))
```

R code Exa 7.15 CI for variance and sd of normal population

```
#Ex7.15, Page 296
#Answers may slightly vary due to rounding off of values

data<-c(1470,1510,1690,1740,1900,2000,2030,2100,2190,
          2200,2290,2380,2390,2480,2500,2580,2700)

interval<-  function(data,conf.level=0.95){
  df<-length(data)-1
  l<-qchisq((1 - conf.level)/2, df)
  r<-qchisq((1 - conf.level)/2, df, lower.tail = FALSE)
  v<-var(data)
  c(df* v/r,df*v/l)
}
print(paste("Confidence interval for variance of normal population:"))
print(interval(data))
```

Chapter 8

Test of hypotheses based on a single sample

R code Exa 8.1 Hypotheses and test procedures

```
#Ex8.1, Page 304
#Answers may vary slightly due to rounding off of
   values

#X<-number of crashes with no visible damage
#Reject H0 if x>=8 (x<-observed value of test
   statistic)

#When H0 is true
n<-20
p1<-0.25

#alpha=P(type I error)
alpha<-1-pbinom(7,n,p1)
print(paste("Alpha value:",alpha))

print(paste("When H0 is true, roughly",trunc(100*
   alpha,digits=2),"% of all experiments consisting
   of 20 crashes would result into a Type I error"))
```

```r

#beta=P(type II error)
#Different beta for each different p that exceeds 0.25, thus taking p=0.3
p2<-0.3
beta<-pbinom(7,n,p2)
print(paste("Beta value:",beta))
print(paste("When H0 is false, roughly",trunc(100*beta,digits=2),"% of all experiments would result into a Type II error"))

#Table showing variation of beta values with increase in value of p
for(i in 3:8){
  p<-0.1*i
  a<-pbinom(7,n,p)
  print(paste("p-value:",p))
  print(paste("Beta value:",a))
}
```

R code Exa 8.2 Hypotheses and test procedures

```r
#Ex8.2, Page 305
#Answers may vary slightly due to rounding off of values

m<-75
sd<-9
#Hypotheses H0: mean=75, H1: mean<75

#Experimental data
n<-25
sd<-9
#Standard deviation of normal distribution
sdx<-sd/sqrt(n)
```

```
print(paste("SD of normal distribution:",sdx))

#When H0 is true
mx<-75

#Consider rejection region x<=70.8
#To find alpha and beta

#P(type I error)
alpha<-pnorm(70.8,mx,sdx)
print(paste("Alpha value:",alpha))

#P(type II error)
mx<-c(72,70,67)                    #H0 is not rejected when it is false because mean= eg:72
for(i in mx){
  beta<-1-pnorm(70.8,i,sdx)
  print(paste("Beta value:",beta,"when mean is",i))
}

print(paste("Only",round(alpha,digits=3)*100,"% of all experiments will result into Type I error"))
print(paste("Possibility of Type II error decreases as mean value increases"))
```

R code Exa 8.3 Hypotheses and test procedures

```
#Ex8.3, Page 306
#Answers may vary slightly due to rounding off of values

#R9={9,10,......,20}
n<-20

#To find alpha=P(H0 rejected when p=.25)=P(X>=9 when
```

```
    X follows binomial dist.)
p<-0.25
alpha<-1-pbinom(8,n,p)
print(paste("Alpha value:",alpha))
print(paste("Type I error probability has reduced
   considerably when using R9 rather than R8
   rejection region"))

#To find beta value=P(H0 not rejected)=P(X<=8 when X
    follows binomial dist.)
p1<-0.3       #1st case
beta1<-pbinom(8,n,p1)
print(paste("Beta(0.3)=",beta1))

p2<-0.5       #2nd case
beta2<-pbinom(8,n,p2)
print(paste("Beta(0.5)=",beta2))
print(paste("P(Type II error in R9)>P(Type II error
   in R8)"))
```

R code Exa 8.4 Hypotheses and test procedures

```
#Ex8.4, Page 307
#Answers may vary slightly due to rounding off of
   values

#New rejection region x<=72
sdx<-1.8

#To find alpha=P(H0 rejected when true)=P(X<=72 when
    X follows normal dist)
mx<-75
alpha<-pnorm(72,mx,sdx)
print(paste("Alpha value:",alpha))
print(paste("Type I error probability is greater in
```

```
    the new rejection region"))

#To find beta=P(H0 not rejected)
mx<-c(72,70,67)                    #H0 is not rejected
    when it is false because mean= eg:72
for(i in mx){
  beta<-1-pnorm(72,i,sdx)
  print(paste("Beta(",i,"):",beta))
}
print(paste("Type II error probability is lower in
    the new rejection region"))
```

R code Exa 8.5 Hypotheses and test procedures

```
#Ex8.5, Page 308

#H0: mu=1.5
#Ha: mu>1.5

mu0<-1.5
n<-32
sd<-0.2
sdx<-sd/sqrt(n)
print(paste("Rejection region: Z>=c"))

alpha<-0.05
#where alpha=P(type I error)=P(Z>=c when Z follows N
    (0,1))
c<-qnorm(1-alpha)
print(paste("C value:",c))

#Z>=c
x<-round(mu0+sdx*c,digits=2)
print(paste("xbar>=",x))
print(paste("Beta refers to the prob. that xbar<",x,
```

```
"and can be calculated for any mu>1.5"))
```

R code Exa 8.6 Tests about a population mean

```
#Ex8.6, Page 312
#Answers may vary slightly due to rounding off of
    values

mu0<-130
n<-9
sd<-1.5
xbar<-131.08

z<-(xbar-mu0)/(sd/sqrt(n))        #Test statistic
    value
print(paste("Test statistic value:",z))

#To determine rejection region
alpha<-0.01
a<-qnorm(1-alpha/2)
l<-(-a)
r<-a
print(paste("Rejection region: z<=",l,",z>=",r))
if(z<=l || z>=r)print(paste(z,"falls in the
    rejection region and H0 is rejected"))else print(
    paste(z,"does not fall in the rejection region so
     H0 cannot be rejected at significance level",
    alpha))
```

R code Exa 8.7 Tests about a population mean

```
#Ex8.7, Page 314
```

```
#Answers may vary slightly due to rounding off of
   values

n<-16
sd<-1500
alpha1<-0.01
xbar<-30000
mu0<-31000

zalpha1<-qnorm(1-alpha1)

z<-(xbar-mu0)/(sd/sqrt(n))

#To find beta(31000) when alpha=0.01
beta1<-pnorm(zalpha1+z)
print(paste("Beta(31000):",beta1,"when alpha=",
   alpha1))

alpha2<-0.1
zalpha2<-qnorm(1-alpha2)

#To find beta(31000) when alpha=0.1
beta2<-pnorm(zalpha2+z)
print(paste("Beta(31000):",round(beta2,digits=1),"
   when alpha=",alpha2))

#To find n
n<-(sd*(zalpha1+zalpha2)/(xbar-mu0))^2
print(paste("n value:",n))
print(paste("Sample size=",ceiling(n)))
```

R code Exa 8.8 Tests about a population mean

```
#Ex8.8, Page 315
#Answers may vary slightly due to rounding off of
```

```
values

n<-52
mu0<-30
data<-c
    (14.1,14.5,15.5,16.0,16.0,16.7,16.9,17.1,17.5,17.8,17.8,18.1,18.2
        19.0,19.2,19.4,20.0,20.0,20.8,20.8,21.0,21.5,23.5,27.5,27.5,
        30.0,31.6,31.7,31.7,32.5,33.5,33.9,35.0,35.0,35.0,36.7,40.0,
        47.5,50.0,51.0,51.8,54.4,55.0,57.0)
xbar<-mean(data)
sd<-sd(data)
print(paste("Mean:",xbar))
print(paste("Standard deviation:",sd))

alpha<-0.05
#To find zalpha
z1<-qnorm(alpha)

#To find z
z<-(xbar-mu0)/(sd/sqrt(n))
print(paste("z value:",z))

if(z>z1) print(paste("H0 cannot be rejected since",z
    ,">",z1)) else print(paste("H0 is rejected since"
    ,z,"<",z1))
```

R code Exa 8.9 Tests about a population mean

```
#Ex8.9, Page 317
#Answers may vary slightly due to rounding off of
    values

```

```
data<-c(2.67,4.62,4.14,3.81,3.83)
n<-length(data)

alpha<-0.05
mu0<-4

#To determine rejection region
t1<-qt(1-alpha/2,df=n-1)
l<-(-t1)
r<-t1
print(paste("Rejection region: t<=",l,",t>=",r))

#To find mean
print(paste("Sum of sample:",sum(data)))
print(paste("Mean:",mean(data)))

#To find standard deviation
print(paste("Standard deviation:",sd(data)))
print(paste("Estimated standard error of the mean:",
    sd(data)/sqrt(n)))

#Test statistic value
t1<-t.test(data,alternative="two.sided",mu=mu0)
print(t1)
```

R code Exa 8.11 Tests concerning a population proportion

```
#Ex8.11, Page 324
#Answers may vary slightly due to rounding off of
    values

#H0:p=0.15
#Ha:p>0.15

p0<-0.15
```

```
q0<-1-p0
n<-91
n1<-16

#Check validity of test procedures
if(n*p0>10 && n*q0>10)  print(paste("Large sample z
    test can be used")) else  print(paste("z test
    cannot be used"))

alpha<-0.1
zalpha<-qnorm(1-alpha)
print(paste("Rejection region: z>=",zalpha))

p<-n1/n
print(paste("p-cap:",p))

#To find z
z<-(p-p0)/(sqrt(p0*q0/n))
print(paste("Test statistic value:",z))

if(z<zalpha)print(paste("z=",z,"is not in the
    rejection region and hence, H0 cannot be rejected
    ")) else print(paste("z is in the rejection
    region"))
```

R code Exa 8.12 Tests concerning a population proportion

```
#Ex8.12, Page 326
#Answers may vary slightly due to rounding off of
    values

#H0: p=0.9
#Ha: p<0.9

n<-225
```

```
p0<-0.9
pdash<-0.8
alpha<-0.01

zalpha<-qnorm(1-alpha)

#To find beta(0.8)
num<-p0-pdash-(zalpha*sqrt((p0*(1-p0))/n))
denom<-sqrt(pdash*(1-pdash)/n)
beta<-1-pnorm(num/denom)
print(paste("Beta(0.8):",beta))

#P(H0 rejected)
prob<-1-beta
print(paste("P(H0 rejected)=",prob,"and thus,",
    ceiling(prob*100),"% of samples will result in
    correct H0 rejection"))

#To find n
n<-((zalpha*sqrt(p0*(1-p0))+zalpha*sqrt(pdash*(1-
    pdash)))/(pdash-p0))^2
print(paste("Sample size:",ceiling(n)))
```

R code Exa 8.13 Tests concerning a population proportion

```
#Ex8.13, Page 327

#H0: p=0.9
#Ha: p<0.9
#Rejection region: x<=15

n<-20
pdash<-0.8
alpha<-0.05

```

```
#To find beta=P(H0 not rejected when X follows
    binomial dist)
beta<-1-pbinom(15,n,pdash)
print(paste("Beta(",pdash,")=",round(beta,digits=2))
    )

print(paste(ceiling(beta*100),"% of all samples
    causes H0 to be rejected incorrectly when p=",
    pdash))
```

R code Exa 8.14 P values

```
#Ex8.14, Page 329
#Answers may vary slightly due to rounding off of
    values

#H0: mu=2.0
#Ha: mu>2.0

mu0<-2
xbar<-2.06
sd<-0.141
n<-51

#To find z value
z<-(xbar-mu0)/(sd/sqrt(n))
print(paste("Test statistic value:",z))

#To find P-value=P(Z>=z when mu=2.0)
p<-1-pnorm(z)
print(paste("P-value:",p))

alpha<-c(0.01,0.001)
for(i in alpha){
      if(p<=i)
```

```
      print(paste("H0 rejected when significance level is at",i))
    else
      print(paste("H0 not rejected when significance level is at",i))
}
```

R code Exa 8.15 P values

```
#Ex8.15, Page 330
#Answers may vary slightly due to rounding off of values

#H0: mu=15
#Ha: mu>1.5

z<-2.10

#To determine p-value
p<-1-pnorm(z)
print(paste("P-value:",p))

alpha<-c(0.1,0.05,0.01)
for(i in alpha){
  z1<-qnorm(1-i)
  if(z>=z1)
    print(paste("Reject H0 at significance level",i))
  else
    print(paste("Do not reject H0 at significance level",i))
}

#Using p-value approach
print(paste("Using p-value approach:"))
```

```
for(j in alpha){
  if(j>=p)
    print(paste("Reject H0 at significance level",j,
        "since",j,">=",p))
  else
    print(paste("Do not reject H0 at significance
        level",j,"since",j,"<",p))
}
```

R code Exa 8.17 P values for z tests

```
#Ex8.17, Page 333
#Answers may vary slightly due to rounding off of
    values

#H0: mu=245
#Ha: mu!=245

xbar<-246.18
mu0<-245
n<-50
sd<-3.6
alpha<-0.01

#To find z value
z<-(xbar-mu0)/(sd/sqrt(n))
print(paste("Test statistic value:",z))

#To find p value in two tailed experiment
p<-2*(1-pnorm(z))
print(paste("P-value:",p))

if(p>alpha) print(paste("H0 not rejected at
    significance level",alpha)) else print(paste("H0
    rejected at significance level",alpha))
```

R code Exa 8.18 P values for t tests

```
#Ex8.18, Page 335

#H0: mu=4
#Ha: mu!=4

n<-5
t<-round(-0.594,digits=1)
df<-n-1

alpha<-0.05

#To find critical values
a<-qt(1-alpha/2,df=n-1)

#Since test is two tailed
p<-2*pt(t,df=n-1)
print(paste("P-value:",round(p,digits=2)))

b<-c(0.01,0.05,0.1)
for(i in b){
  if(p>i)
    print(paste("H0 is not rejected at significance level",i))
  else
    print(paste("H0 rejected at significance level",i))
}
```

R code Exa 8.19 Interpreting P values

```
#Ex8.19, Page 335
#Answers may vary slightly due to rounding off of
   values
#Tests for mean=20 have only been done....

#H0: mu=20
#Ha: mu>20

n<-4
df<-n-1
mu0<-20
sd<-2
x<-c(20.83,22.232,20.276,17.718)

xbar<-mean(x)
print(paste("Mean:",xbar))
s<-sd(x)
print(paste("Standard deviation:",s))

#To find t
t<-(xbar-mu0)/(s/sqrt(n))
print(paste("Test statistic value:",t))

#To find p value
p<-pt(t,df=n-1,lower.tail = FALSE)

alpha<-0.05
if(p>alpha) print(paste("H0 is not rejected at
   significance level",alpha)) else print(paste("H0
    is rejected at significance level",alpha))
```

Chapter 9

Inferences based on two samples

R code Exa 9.1 Test procedures for normal population with known variances

```
#Ex9.1, Page 348
#Answers may slightly vary due to rounding off of
   values

#H0: mu1-mu2=0
#Ha: mu1-mu2!=0

xbar<-29.8
ybar<-34.7
s1<-4
s2<-5
m<-20
n<-25

#To find percentile
alpha<-0.01
H0<-qnorm(1-alpha/2)

```

```
#To find test statistic value
z<-(xbar-ybar)/sqrt((s1^2)/m+(s2^2)/n)
print(paste("Test statistic value:",z))

if(z>=H0||z<=H0)print(paste("H0 is rejected at level",alpha)) else print(paste("H0 is accepted at level",alpha))
```

R code Exa 9.2 Using a comparison to identify causality

```
#Ex9.2, Page 349
#Answers may slightly vary due to rounding off of values

#H0: mu1-mu2=0
#Ha: mu1-mu2>0

xbar<-48.9
ybar<-43.2
s1<-14.6
s2<-14.4
m<-125
n<-90
alpha<-0.01

#To find test statistic value
z<-(xbar-ybar)/sqrt((s1^2)/m+(s2^2)/n)
print(paste("Test statistic value:",z))

#To find p-value
p<-pnorm(-abs(z))
print(paste("p-value:",p))

if(alpha>p) print(paste("H0 is rejected at level",alpha)) else  print(paste("H0 is accepted at
```

```
level",alpha))
```

R code Exa 9.3 Beta and the choice of sample size

```
#Ex9.3, Page 350
#Answers may slightly vary due to rounding off of
    values

t<-5
t1<-0
s<-1.34
n<-qnorm(1-0.01/2)
beta<-pnorm(n-((t-t1)/s))-pnorm(-n-((t-t1)/s))
print(paste("Beta(5):",beta))
```

R code Exa 9.4 Large sample tests

```
#Ex9.4, Page 352
#Answers may vary slightly due to rounding off of
    values

mu1<-2258
mu2<-2637
s1<-1519
s2<-1138
m<-663
n<-413
l<-0.05
H0<-(-200)

#To find z0.05
z1<-qnorm(1-l)

```

```
#To find test statistic value
z<-(mu1-mu2-H0)/sqrt(((s1^2)/m+(s2^2)/n))
print(paste("Test statistic value:",z))

if(z<=-z1) print(paste("H0 is rejected since",z,"<=",-z1))

#To find p-value
p<-pnorm(-abs(z))
print(paste("P-value:",p))
if(p<=l) print(paste("H0 is rejected since",p,"<=",l))
```

R code Exa 9.5 Confidence intervals for difference of means

```
#Ex9.5, Page 354
#Answers may vary slightly due to rounding off values

c1<-c('Variable_diam', 'N', 'Mean', 'Median', 'TrMean', 'StDev', 'SEMean', 'Min', 'Max', 'Q1','Q3')
c2<-c('3/8',78,4.250,4.230,4.238,1.300,0.147,1.634,7.327,3.389,5.075)
c3<-c('1/2',88,7.140,7.113,7.150,1.680,0.179,2.450,11.343,5.965,8.447)
df<-data.frame(c1,c2,c3)
print(df)

mu1<-4.25
mu2<-7.14
z<-qnorm(1-(0.05/2))
s1<-1.3
```

```
s2<-1.68
m<-78
n<-88

#To find confidence intervals
l<-(mu1-mu2)-z*sqrt((s1^2)/m+(s2^2)/n)
r<-(mu1-mu2)+z*sqrt((s1^2)/m+(s2^2)/n)
print(paste("Left endpoint of interval:",l))
print(paste("Right endpoint of interval:",r))
```

R code Exa 9.6 Two sample t test and confidence interval

```
#Ex9.6, Page 358
#Answers may vary slightly due to rounding off of
    values

Fabric_type<-c('Cotton','Triacetate')
Sample_size<-c(10,10)
Sample_mean<-c(51.71,136.14)
Sample_standard_deviation<-c(.79,3.59)
data<-data.frame(Sample_size,Sample_mean,Sample_
    standard_deviation,row.names=Fabric_type)
print(data)

#Assigning values to variables
s1<-Sample_standard_deviation[1]
s2<-Sample_standard_deviation[2]

xbar<-Sample_mean[1]
ybar<-Sample_mean[2]

m<-Sample_size[1]
n<-Sample_size[2]

#To find df
```

```r
num<-((s1^2/m)+(s2^2/n))^2
denom<-((s1^2/m)^2)/(m-1)+((s2^2/n)^2)/(n-1)
df<-num/denom
print(paste("df:",df))

#To find t value
alpha<-0.05
t<-qt(1-alpha/2,df=m-1)
print(paste("t:",t))

#To find intervals
l<-xbar-ybar-t*sqrt((s1^2/m)+(s2^2/n))
r<-xbar-ybar+t*sqrt((s1^2/m)+(s2^2/n))
print(paste("Left endpoint of interval:",l))
print(paste("Right endpoint of interval:",r))

print(paste("The true average porosity of the
    triacetate fabric is greater than that of the
    cotton fabric by between",round(-r,digits=2),"and
    ",round(-l,digits=2),"cm^3/cm^2/sec"))
```

R code Exa 9.7 Two sample t test and confidence interval

```r
#Ex9.7, Page 359
#Answers may vary slightly due to rounding off of
    values

No_fusion<-c
    (2748,2700,2655,2822,2511,3149,3257,3213,3220,2753)

Fused<-c(3027,3356,3359,3297,3125,2910,2889,2902)

#H0: mu1-mu2=0
#Ha: mu1-mu2<0

```

```
#To find test staistic value
t<-t.test(No_fusion,Fused)
print(t)
print(paste("t statistic value:",t$statistic))
print(paste("df:",t$parameter))

#To plot normal probability plots
par(mfrow=c(2,1))
qqnorm(No_fusion,datax=TRUE,ylab="Not fused",xlab="Probability",main="Normal probability plot for not fused data")
qqline(No_fusion,datax=TRUE)
qqnorm(Fused,datax=TRUE,ylab="Fused",xlab="Probability",main="Normal probability plot for fused data")
qqline(Fused,datax=TRUE)
```

R code Exa 9.9 Paired t test

```
#Ex9.9, Page 367
#Answers may vary slightly due to rounding off of values

Subject<-c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
Before<-c(81,87,86,82,90,86,96,73,74,75,72,80,66,72,56,82)
After<-c(78,91,78,78,84,67,92,70,58,62,70,58,66,60,65,73)
Difference<-Before-After
data<-data.frame(Before,After,Difference,row.names=Subject)
print(data)

#Normal probability plot
par(mfrow=c(1,2))
```

```
qqnorm(data$Difference,main="Normal probability plot
    of the differences",xlab="diff",ylab="
   Probability")
qqline(data$Difference)
boxplot(Difference,horizontal=TRUE,xlab="Difference"
   )

#H0: mu=0
#Ha: mu!=0
t1<-t.test(Difference,mu=0)
print(paste("t statistic value:",t1$statistic))
print(paste("p value:",t1$p.value))

alpha<-0.01
if(t1$p.value<alpha) print(paste("H0 rejected"))
   else print(paste("H0 not rejected"))
```

R code Exa 9.10 Paired t confidence interval

```
#Ex9.10, Page 369
#Answers may vary slightly due to rounding off of
   values

Subject<-c(1,2,3,4,5,6,7,8,9,10,11,12,13)
Slide<-c(30,35,40,25,20,30,35,62,40,51,25,42,33)
Digital<-c(25,16,15,15,10,20,7,16,15,13,11,19,19)
Difference<-Slide-Digital
data<-data.frame(Slide,Digital,Difference,row.names=
   Subject)
print(data)

#Normal probability plot
qqnorm(data$Difference,main="Normal probability plot
    of the differences",ylab="diff",xlab="
   Probability")
```

```
qqline(data$Difference)

#To find t critical value
alpha<-0.05
t<-qt(1-alpha/2,df=12)
print(paste("t statistic value:",t))

#To find confidence intervals
t1<-t.test(Slide,Digital,paired=TRUE)
print(t1)
```

R code Exa 9.11 Inferences concerning difference between population proportions

```
#Ex9.11, Page 376
#Answers may slightly vary due to rounding off of values

alpha<-0.05

#H0: p1-p2=0
#Ha: p1-p2<0

a<-81
tot1<-549
p1cap<-a/tot1
q1cap<-1-p1cap

b<-141
tot2<-730
p2cap<-b/tot2
q2cap<-1-p2cap

pcap<-(a+b)/(tot1+tot2)
qcap<-1-pcap
```

```
#To find test statistic value
if((tot1*p1cap)>=10 && (tot1*q1cap)>=10 && (tot2*p2cap)>=10 && (tot2*q2cap)>=10)
  z<-(p1cap-p2cap)/sqrt(pcap*qcap*(1/tot1+1/tot2))
print(paste("Test statistic value:",z))

#To find P-value
p<-pnorm(z)
print(paste("P-value:",p))

if(p<=alpha)  print(paste("H0 can be rejected at significance level",alpha)) else  print(paste("H0 cannot be rejected"))
```

R code Exa 9.12 Type II error probabilities and sample sizes

```
#Ex9.12, Page 378
#Answers may vary slightly due to rounding off of values

#H0: p1-p2=0
#Ha: p1-p2>0

a<-30
tot1<-100000
p1<-a/tot1
print(paste("p1:",p1))
q1<-1-p1

b<-a/2
p2<-b/tot1
print(paste("p2:",p2))
q2<-1-p2

```

```
alpha<-0.05
beta<-0.1

zalpha<-qnorm(1-alpha)
zbeta<-qnorm(1-beta)
d<-p1-p2

#To find n value
n<-(zalpha*sqrt((p1+p2)*(q1+q2)/2)+zbeta*sqrt(p1*q1+
    p2*q2))^2/d^2
print(paste("Value of n:",n))
```

R code Exa 9.13 A large sample confidence interval

```
#Ex9.13, Page 380
#Answers may vary slightly due to rounding off of
    values

alpha<-0.01
a<-76         #number of people who survived through
    chemotherapy treatment
m<-154    #number of people who received
    chemotherapy treatment
p1cap<-a/m
q1cap<-1-p1cap
print(paste("p1cap:",p1cap))

b<-98         #number of people who survived through
    hybrid treatment
n<-164    #number of people who received hybrid
    treatment
p2cap<-b/n
q2cap<-1-p2cap
print(paste("p2cap:",p2cap))

```

```
#To find z
zalpha<-qnorm(1-alpha/2)

if(m*p1cap>=10 && m*q1cap>=10 && n*p2cap>=10 && n*q2cap>=10)
  l<-p1cap-p2cap-zalpha*sqrt((p1cap*q1cap/m)+(p2cap*q2cap/n))
  r<-p1cap-p2cap+zalpha*sqrt((p1cap*q1cap/m)+(p2cap*q2cap/n))

print(paste("Left endpoint of confidence interval:",l))
print(paste("Right endpoint of confidence interval:",r))
print(paste("Hence",round(l,digits=3),"<p1-p2<",round(r,digits=3)))
```

R code Exa 9.14 F test for equality of variances

```
#Ex9.14, Page 385
#Answers may vary slightly due to rounding off of values

s1<-52.6
s2<-84.2
m<-28
n<-26

#H0: s1^2=s2^2
#Ha: s1^2<s2^2

alpha<-0.01
#To find F value
Falpha<-qf(alpha,df1=m-1,df2=n-1)
print(paste("F value:",Falpha))
```

```r

f<-(s1^2)/(s2^2)
print(paste("f value:",f))

if(f<=Falpha) print(paste("H0 is rejected at significance level",alpha)) else print(paste("H0 is not rejected"))
```

Chapter 10

The analysis of variance

R code Exa 10.1 Single factor ANOVA

```
#Ex10.1, Page 392
#Answers may vary slightly due to rounding off of
   values

Type_of_box<-c(1,2,3,4)
CS1<-c(655.5,789.2,737.1,535.1)
CS2<-c(788.3,772.5,639.0,628.7)
CS3<-c(734.3,786.9,696.3,542.4)
CS4<-c(721.4,686.1,671.7,559.0)
CS5<-c(679.1,732.1,717.2,586.9)
CS6<-c(699.4,774.8,727.1,520.0)
Mean<-c(NULL)
SD<-c(NULL)

#To find mean of observations
for(i in 1:4){
  sum<-CS1[i]+CS2[i]+CS3[i]+CS4[i]+CS5[i]+CS6[i]
  mean<-sum/6
  Mean[i]<-mean
}
#To find standard deviation of values
```

```
for(i in 1:4){
  s<-(CS1[i]-Mean[i])^2+(CS2[i]-Mean[i])^2+(CS3[i]-Mean[i])^2+(CS4[i]-Mean[i])^2+(CS5[i]-Mean[i])^2+(CS6[i]-Mean[i])^2
  sd<-sqrt(s/5)
  SD[i]<-sd
}

data<-data.frame(Type_of_box,CS1,CS2,CS3,CS4,CS5,CS6,Mean,SD)
print(data)

#To find grand mean of mean values
tot_mean<-0
for(i in 1:4){
  tot_mean<-tot_mean+Mean[i]
}
tot_mean<-tot_mean/4
print(paste("Grand mean:",tot_mean))

par(mfrow=c(1,2))
#To create boxplots of the data
#First element of each CS column gives us the compression strength of one box
a<-c(CS1[1],CS2[1],CS3[1],CS4[1],CS5[1],CS6[1])
b<-c(CS1[2],CS2[2],CS3[2],CS4[2],CS5[2],CS6[2])
c<-c(CS1[3],CS2[3],CS3[3],CS4[3],CS5[3],CS6[3])
d<-c(CS1[4],CS2[4],CS3[4],CS4[4],CS5[4],CS6[4])
boxplot(a,b,c,d)
title(main="Boxplots for original data")

#Altering data in sample 2 by adding 120 to every value in Sample 4
e<-c(NULL)
for(i in 1:6){
  e[i]<-d[i]+120
}
boxplot(a,b,c,e)
```

```
title(main="Boxplots for altered data")
```

R code Exa 10.2 F distribution and the F test

```
#Ex10.2, Page 397
#Answers may vary slightly due to rounding off of
   values

#From Ex10.1
Mean<-c(713.00,756.93,698.07,562.02)
SD<-c(46.55,40.34,37.20,39.87)
tot_mean<-682.50

I<-4
J<-6

#Numerator
v1<-I-1

#Denominator
v2<-I*(J-1)

alpha<-0.05

#H0:mu1=mu2=mu3=mu4
#Ha:at least two mean values will be different

#To find F critical value
F<-qf(p=1-alpha,df1=v1,df2=v2)
print(paste("Critical value F:",F))

#To find mean square for treatment
MSTr<-(J/(I-1))*sum((Mean-tot_mean)^2)
print(paste("MSTr:",MSTr))

```

```
#To find mean square for error
MSE<-(1/I)*sum(SD^2)
print(paste("MSE:",MSE))

#To find f value
f<-MSTr/MSE
print(paste("f:",f))

if(f>=F)  print(paste("H0 is rejected at
    significance level",alpha,"since f>=F"))  else
    print(paste("H0 not rejected"))
```

R code Exa 10.3 F distribution and the F test

```
#Ex10.3, Page 397
#Answers may vary slightly due to rounding off of
    values

Treatment<-c(1,2,3,4,5)
Sample_mean<-c(10.5,14.8,15.7,16.0,21.6)
Sample_sd<-c(4.5,6.8,6.5,6.7,6.0)

#To find total mean of mean values
tot_mean<-mean(Sample_mean)
print(paste("Grand mean:",tot_mean))

#H0: mu1=mu2=mu3=mu4=mu5
#Ha: at least two of the mean values are different

I<-5
J<-10
alpha<-0.01

#Numerator
v1<-I-1
```

```
#Denominator
v2<-I*(J-1)

#To find F value
F<-qf(1-alpha,v1,v2)
print(paste("Critical value F:",F))

#To find mean square for treatment
s<-Sample_mean-tot_mean
MSTr<-(J/(I-1))*sum(s^2)
print(paste("MSTr:",MSTr))

#To find mean square for error
MSE<-mean(Sample_sd^2)
print(paste("MSE:",MSE))

#To find f value
f<-MSTr/MSE
print(paste("f:",f))

if(f>=F)  print(paste("H0 is rejected at
   significance level",alpha,"since f>=F"))  else
   print(paste("H0 not rejected"))
```

R code Exa 10.4 Sums of squares

```
#Ex10.4, Page 400
#Answers may vary slightly due to rounding off of
   values

Mixture1<-c(0.56,1.12,0.90,1.07,0.94)
Mixture2<-c(0.72,0.69,0.87,0.78,0.91)
Mixture3<-c(0.62,1.08,1.07,0.99,0.93)
```

```
data1<-data.frame(Mixture1,Mixture2,Mixture3)

I<-3
J<-5

#Numerator
v1<-I-1

#Denominator
v2<-I*(J-1)

alpha<-0.01
#To find F value
F<-qf(1-alpha,v1,v2)
print(paste("Critical value, F:",F))


#To create ANOVA table
mix<-stack(data1)
names(mix)<-c("Degree","Treatment")

a<-aov(Degree~Treatment,data=mix)
print(paste("ANOVA table:"))
a1<-summary(a)
print(a1)
print(paste("f value:",a1[[1]]$F[1]))
```

R code Exa 10.6 Multiple comparisons in ANOVA

```
#Ex10.6, Page 404

Treatment1<-c(88.6,73.2,91.4,68.0,75.2)
Treatment2<-c(63.0,53.9,69.2,50.1,71.5)
Treatment3<-c(44.9,59.5,40.2,56.3,38.7)
Treatment4<-c(31.0,39.6,45.3,25.2,22.7)
```

```
data1<-data.frame(Treatment1,Treatment2,Treatment3,
   Treatment4)

Treatment<-stack(data1)
names(Treatment)<-c("Time","Model")

#To display ANOVA table
a<-aov(Time~Model,data=Treatment)
print(summary(a))
```

R code Exa 10.9 Unequal sample sizes

```
#Ex10.9, Page 412

Permanent_molding<-c
   (45.5,45.3,45.4,44.4,44.6,43.9,44.6,44.0)
Die_casting<-c
   (44.2,43.9,44.7,44.2,44.0,43.8,44.6,43.1)
Plaster_molding<-c(46.0,45.9,44.8,46.2,45.1,45.5)

#To display ANOVA table
data1<-data.frame(Mold=c(Permanent_molding,Die_
   casting,Plaster_molding),Treatment=factor(rep(c("
   Mold1","Mold2","Mold3"),times=c(length(Permanent_
   molding),length(Die_casting),length(Plaster_
   molding)))))

cat("ANOVA table\n")
a<-aov(Mold~Treatment,data=data1)
print(summary(a))
```

R code Exa 10.10 Unequal sample sizes

```
#Ex10.10, Page 413

alpha<-0.95
n<-22
J1<-8
J2<-8
J3<-6
I<-3
J<-n-I
MSE<-0.316
Q<-qtukey(alpha,I,J)
xibar<-c(44.71,44.06,45.58)

#To find w12
w12<-Q*sqrt((MSE/2)*(1/J1+1/J2))
print(paste("w12:",w12))
w23<-Q*sqrt((MSE/2)*(1/J2+1/J3))
print(paste("w23:",w23))
w31<-Q*sqrt((MSE/2)*(1/J3+1/J1))
print(paste("w31:",w31))

if(xibar[1]-xibar[2]< w12){
  print(paste("mu1 and mu2 are not significantly
      different"))
}
if(xibar[2]-xibar[3]< w23){
  print(paste("mu2 and mu3 are not significantly
      different"))
}
if(xibar[3]-xibar[1]< w31){
  print(paste("mu3 and mu1 are not significantly
      different"))
}
```

R code Exa 10.11 A random effects model

```
#Ex.10.11, Page 415

#H0: sigmaA^2=0

C1<-c(55,53,54)
C2<-c(26,37,32)
C3<-c(78,91,85)
C4<-c(92,100,96)
C5<-c(49,51,50)
C6<-c(80,85,83)
data1<-data.frame(C1,C2,C3,C4,C5,C6)
print(paste("x..=",sum(data1)))


#To create ANOVA table
col<-stack(data1)
names(col)<-c("Type","Treatments")
a<-aov(Type~Treatments,data=col)
print(paste("ANOVA table:"))
print(summary(a))
```

Chapter 11

Multifactor analysis of variance

R code Exa 11.1 Two factor ANOVA

```
#Ex11.1, Page 420

dat<-c(0.97,0.48,0.48,0.46,0.77,0.14,0.22,0.25,0.67,0.39,0.57,0.19)
mat1<-matrix(dat,nrow=3,ncol=4,byrow=TRUE)
rownames(mat1)<-c("Brand1","Brand2","Brand3")
colnames(mat1)<-c("WT1","WT2","WT3","WT4")
print(mat1)

print(paste("Number of levels of factor A(Brand of pen), I:",nrow(mat1)))
print(paste("Number of levels of factor B(Washing treatment), J:",ncol(mat1)))
```

R code Exa 11.3 Fixed effects model

```
#Ex11.3, Page 424
```

```
#Answers may vary slightly due to rounding off of
   values

#To create data frame
brand<-rep(c("I","II","III"),4)
WT<-rep(c("1","2","3","4"),each=3)  #WT<-Washing
   treatment
col_change<-c
   (0.97,0.77,0.67,0.48,0.14,0.39,0.48,0.22,0.57,0.46,0.25,0.19)

df1<-data.frame(brand,WT,col_change)
print(df1)

#To display ANOVA table
a<-aov(col_change~brand+WT,data=df1)
print(a)
cat("\nSummarized table:\n")
print(summary(a))

#Diagnostic plots
model<-lm(col_change~brand+WT,data=df1)
par(mfrow=c(1,2))
qqnorm(model$residuals,datax=TRUE,main="Normal
   probability plot",ylab="Residual",xlab="Percent")
qqline(model$residuals,datax=TRUE)

plot(df1$col_change,resid(model),main="Residual plot
   ",xlab="Fitted value",ylab="Residual")
abline(0,0)
```

R code Exa 11.5 Randomized block experiments

```
#Ex11.5, Page 427
#Answers may vary slightly due to rounding off of
```

```
    values

#To create data frame
Treatment<-rep(c("I","II","III","IV","V"),4)
Block<-rep(c("1","2","3","4"),each=5)
Power<-c
    (685,722,733,811,828,792,806,802,888,920,838,893,880,952,978,875,

df1<-data.frame(Treatment,Block,Power)
print(df1)

#To display ANOVA table
a<-aov(Power~Treatment+Block,data=df1)
print(a)
cat("\nSummarized table:\n")
print(summary(a))
```

R code Exa 11.6 Randomized block experiments

```
#Ex11.6, Page 428
#Answers may vary slightly due to rounding off of
    values

#To create data frame
Tension<-rep(c("210","235","260","285"),6)
Player<-rep(c("1","2","3","4","5","6"),each=4)
Speed<-c
    (105.7,113.3,117.2,110.0,116.6,119.9,124.4,106.8,106.6,120.5,122.

df1<-data.frame(Tension,Player,Speed)
print(df1)

#To find mean for corresponding tension data
```

```
cat("\nTension mean values:\n")
m1<-tapply(df1$Speed,list(Tension),mean)
print(m1)

#To find mean of corresponding player data
cat("\nPlayer mean values:\n")
m2<-tapply(df1$Speed,list(Player),mean)
print(m2)

#To display ANOVA table
a<-aov(Speed~Tension+Player,data=df1)
print(a)
print(summary(a))
```

R code Exa 11.7 Two factor ANOVA with Kij greater than 1

```
#Ex11.7, Page 435

AggCont<-rep(c("38","41","44"),each=6)
AsphGr<-rep(c("PG58","PG64","PG70"),6)
data<-c
    (0.835,0.855,0.815,0.845,0.865,0.825,0.822,0.832,0.800,0.826,0.83

df1<-data.frame(AggCont,AsphGr,data)
print(df1)

#To find mean for corresponding content data
cat("\nCoarse aggregate content(%) mean values:\n")
m1<-tapply(df1$data,list(AggCont),mean)
print(m1)

#To find mean of corresponding Asphalt binder grade
    data
cat("\nAsphalt binder grade mean values:\n")
m2<-tapply(df1$data,list(AsphGr),mean)
```

```
print(m2)


#To display ANOVA table
model<-lm(data~AsphGr+AggCont+AsphGr:AggCont)
a<-aov(data~AsphGr+AggCont+AsphGr:AggCont)
print(a)
print(summary(a))

#To display interaction plots(Thermal diffusivity
    data is not given)
interaction.plot(AggCont,AsphGr,response=data)

#Plots for checking normality and constant variance
    assumptions
par(mfrow=c(1,2))
qqnorm(model$residuals,datax=TRUE,main="Normal
    probability plot",ylab="Residual",xlab="Percent")
qqline(model$residuals,datax=TRUE)

plot(df1$data,resid(model),main="Residual plot",xlab
    ="Fitted value",ylab="Residual")
abline(0,0)
```

R code Exa 11.9 Models with mixed and random effects

```
#Ex11.9, Page 439
#Answers may vary slightly from textbook values

#To create data frame
Casmater<-rep(c("Steel","Aluminium","Plastic"),10)
Source<-rep(c("1","2","3","4","5"),each=6)
Power<-c
    (13.1,15.0,14.0,13.2,14.8,14.3,16.3,15.7,17.2,15.8,16.4,16.7,13.7
```

```
df1<-data.frame(Casmater,Source,Power)
print(df1)

#To create ANOVA table
a<-aov(Power~Casmater+Source+Casmater:Source,data=
    df1)
print(a)
cat("\nSummarized table:\n")
print(summary(a))
```

R code Exa 11.10 Three factor ANOVA

```
#Ex11.10, Page 444
#Answers may vary slightly due to rounding off of
    values

#To create data frame
A<-rep(c(rep(c("A1","A2","A3","A4"),each=3)),8)
B<-rep(c("B1","B2"),each=48)
C<-rep(c("C1","C2","C3","C4","C1","C2","C3","C4"),
    each=12)
data<-c
    (3.6,3.8,3.9,3.8,3.6,4.0,3.7,3.9,4.2,3.6,3.5,3.8,3.4,3.7,3.9,3.8,

df1<-data.frame(A,B,C,data)
print(df1)

#To find cell totals for all combinations of the
    three factors
cat("\nCell totals:\n")
m1<-tapply(df1$data,list(A,B,C),sum)
print(m1)

```

```
#To create ANOVA table
a<-aov(data~A+B+C+A:B+A:C+B:C+A:B:C,data=df1)
print(a)
cat("\nSummarized table:\n")
print(summary(a))
```

R code Exa 11.11 Latin square designs

```
#Ex11.11, Page 447
#Answers may vary from textbook values

library(nlme)
library(multcomp)

#Dependent variable
data<-c
   (7.38,7.15,6.75,8.05,5.65,6.00,5.39,8.16,5.64,6.45,5.44,6.55,5.03

#Repeated measures
##B(Columns)
B<-factor(rep(c("1","2","3","4","5","6"),each=6))

##A(Rows)
A<-factor(rep(c("I","II","III","IV","V","VI"),6))

df1<-data.frame(A,B,data)
print(df1)

#To create ANOVA table
#using aov() function
cat("\nUsing aov():\n")
a1<-aov(data~A+Error(B/A),data=df1)
print(summary(a1))

```

```
#using lme()
cat("\nUsing lme():\n")
model<-lme(data~A,random=~1|B/A,data=df1)
a2<-anova(model)
print(a2)

#Tukey test
print(summary(glht(model,linfct=mcp(A="Tukey"))))
```

R code Exa 11.12 Factorial experiments with p factors at two levels

```
#Ex11.12, Page 452

Age<-rep(c("1","2"),each=2,4)
Temp<-rep(c("1","2"),8)
Soil<-rep(c("1","2"),each=8)
Comp_Stre<-c
   (471,485,712,712,413,552,637,789,385,530,770,741,434,593,705,806)

df1<-data.frame(Age,Temp,Soil,Comp_Stre)

#To compute cell totals
s1<-s2<-s3<-s4<-s5<-s6<-s7<-s8<-c(NULL)
for(i in 1:length(Comp_Stre)){
  if(df1$Age[i]=="1" && df1$Temp[i]=="1" && df1$Soil
     [i]=="1"){
    s1[i]<-df1$Comp_Stre[i]
    x111<-sum(s1,na.rm=TRUE)
  }
  else if(df1$Age[i]=="1" && df1$Temp[i]=="2" && df1
     $Soil[i]=="1"){
    s2[i]<-df1$Comp_Stre[i]
    x121<-sum(s2,na.rm=TRUE)
  }
  else if(df1$Age[i]=="1" && df1$Temp[i]=="1" && df1
```

```
      $Soil[i]=="2"){
      s3[i]<-df1$Comp_Stre[i]
      x112<-sum(s3,na.rm=TRUE)
    }
    else if(df1$Age[i]=="2" && df1$Temp[i]=="1" && df1$Soil[i]=="1"){
      s4[i]<-df1$Comp_Stre[i]
      x211<-sum(s4,na.rm=TRUE)
    }
    else if(df1$Age[i]=="2" && df1$Temp[i]=="1" && df1$Soil[i]=="2"){
      s5[i]<-df1$Comp_Stre[i]
      x212<-sum(s5,na.rm=TRUE)
    }
    else if(df1$Age[i]=="1" && df1$Temp[i]=="2" && df1$Soil[i]=="2"){
      s6[i]<-df1$Comp_Stre[i]
      x122<-sum(s6,na.rm=TRUE)
    }
    else if(df1$Age[i]=="2" && df1$Temp[i]=="2" && df1$Soil[i]=="1"){
      s7[i]<-df1$Comp_Stre[i]
      x221<-sum(s7,na.rm=TRUE)
    }
    else if(df1$Age[i]=="2" && df1$Temp[i]=="2" && df1$Soil[i]=="2"){
      s8[i]<-df1$Comp_Stre[i]
      x222<-sum(s8,na.rm=TRUE)
    }

}
#Parameters of the model
n<-2
alpha1_cap<-(x111+x121+x112+x122-x211-x212-x221-x222)/(8*n)
alpha2_cap<-(-1)*alpha1_cap

gamma11_cap<-(x111-x121-x211+x221+x112-x122-x212+
```

```
    x222)/(8*n)
gamma12_cap<-(-1)*gamma11_cap
gamma21_cap<-(-1)*gamma11_cap
gamma22_cap<-gamma11_cap

print(paste("alpha1_cap:",alpha1_cap))
print(paste("alpha2_cap:",alpha2_cap))
print(paste("gamma11_cap:",gamma11_cap))
print(paste("gamma12_cap:",gamma12_cap))
print(paste("gamma21_cap:",gamma21_cap))
print(paste("gamma22_cap:",gamma22_cap))
```

R code Exa 11.13 Factorial experiments with p factors at two levels

```
#Ex11.13, Page 453
#Answers may vary slightly due to rounding off of
    values

Age<-factor(rep(c("1","2"),each=2,4))
Temp<-factor(rep(c("1","2"),8))
Soil<-factor(rep(c("1","2"),each=8))
Comp_Stre<-c
    (471,485,712,712,413,552,637,789,385,530,770,741,434,593,705,806)

df1<-data.frame(Age,Temp,Soil,Comp_Stre)

#To compute cell totals
s1<-s2<-s3<-s4<-s5<-s6<-s7<-s8<-c(NULL)
for(i in 1:length(Comp_Stre)){
  if(df1$Age[i]=="1" && df1$Temp[i]=="1" && df1$Soil
     [i]=="1"){
    s1[i]<-df1$Comp_Stre[i]
    x111<-sum(s1,na.rm=TRUE)
  }
  else if(df1$Age[i]=="1" && df1$Temp[i]=="2" && df1
```

```
$Soil[i]=="1"){
    s2[i]<-df1$Comp_Stre[i]
    x121<-sum(s2,na.rm=TRUE)
  }
  else if(df1$Age[i]=="1" && df1$Temp[i]=="1" && df1$Soil[i]=="2"){
    s3[i]<-df1$Comp_Stre[i]
    x112<-sum(s3,na.rm=TRUE)
  }
  else if(df1$Age[i]=="2" && df1$Temp[i]=="1" && df1$Soil[i]=="1"){
    s4[i]<-df1$Comp_Stre[i]
    x211<-sum(s4,na.rm=TRUE)
  }
  else if(df1$Age[i]=="2" && df1$Temp[i]=="1" && df1$Soil[i]=="2"){
    s5[i]<-df1$Comp_Stre[i]
    x212<-sum(s5,na.rm=TRUE)
  }
  else if(df1$Age[i]=="1" && df1$Temp[i]=="2" && df1$Soil[i]=="2"){
    s6[i]<-df1$Comp_Stre[i]
    x122<-sum(s6,na.rm=TRUE)
  }
  else if(df1$Age[i]=="2" && df1$Temp[i]=="2" && df1$Soil[i]=="1"){
    s7[i]<-df1$Comp_Stre[i]
    x221<-sum(s7,na.rm=TRUE)
  }
  else if(df1$Age[i]=="2" && df1$Temp[i]=="2" && df1$Soil[i]=="2"){
    s8[i]<-df1$Comp_Stre[i]
    x222<-sum(s8,na.rm=TRUE)
  }

}

Effect_contrasts<-c(NULL)
```

```
#To compute effect contrasts
for(j in 1:7){
  if(j==1){
    Effect_contrasts[j]=-x111+x211-x121+x221-x112+
       x212-x122+x222
  }
  else if(j==2){
    Effect_contrasts[j]=-x111-x211+x121+x221-x112-
       x212+x122+x222
  }
  else if(j==3){
    Effect_contrasts[j]=x111-x211-x121+x221+x112-
       x212-x122+x222
  }
  else if(j==4){
    Effect_contrasts[j]=-x111-x211-x121-x221+x112+
       x212+x122+x222
  }
  else if(j==5){
    Effect_contrasts[j]=x111-x211+x121-x221-x112+
       x212-x122+x222
  }
  else if(j==6){
    Effect_contrasts[j]=x111+x211-x121-x221-x112-
       x212+x122+x222
  }
  else if(j==7){
    Effect_contrasts[j]=-x111+x211+x121-x221+x112-
       x212-x122+x222
  }
}

SS<-(Effect_contrasts^2)/(8*n)
df2<-data.frame(Effect_contrasts,SS)
cat("\nEffect contrast and SS values:\n")
print(df2)
SSsum<-sum(SS)

```

```r
#To find sum of squares value
x2_ijkl<-0
X2<-0
for(k in 1:length(Comp_Stre)){
  x2_ijkl<-x2_ijkl+(Comp_Stre[k]^2)
}
x2<-(x111+x112+x211+x121+x212+x221+x122+x222)^2/16
print(paste("Sum of square of data:",x2_ijkl))
print(paste("x...^2/16:",x2))

#To find SST value
SST<-x2_ijkl-x2
print(paste("SST:",SST))

#To find SSE value
SSE<-SST-SSsum
print(paste("SSE:",SSE))

#To display corresponding ANOVA tables
model1<-lm(Comp_Stre~Age:Temp:Soil)
model2<-lm(Comp_Stre~Age*Temp*Soil)
a<-aov(model1,data=df1)
b<-aov(model2,data=df1)
print(summary(a))
print(summary(b))
```

Chapter 12

Simple linear regression and correlation

R code Exa 12.1 The simple linear regression model

```
#Ex12.1, Page 470

i<-c
   (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,2
xi<-c
   (0.40,0.42,0.48,0.51,0.57,0.60,0.70,0.75,0.75,0.78,0.84,0.95,0.99
yi<-c
   (1.02,1.21,0.88,0.98,1.52,1.83,1.50,1.80,1.74,1.63,2.00,2.80,2.48
data<-data.frame(xi,yi,row.names=i)
print(data)

par(mfrow=c(1,3))
plot(xi,yi,main="Scatterplot",xlab="Width of
   palprebal fissure",ylab="OSA")
dotchart(xi,labels=row.names(xi),main="Dot plot of x
    values")
```

```
dotchart(yi,labels=row.names(yi),main="Dot plot of y values")
```

R code Exa 12.2 The simple linear regression model

```
#Ex12.2, Page 471

x<-c(7.01,7.11,7.12,7.24,7.94,7.94,8.04,8.05,8.07,8.90,8.94,8.95,8.97
y<-c(60,67,66,52,50,45,52,48,40,23,20,40,31,26,9,22,13,7)
data1<-data.frame(x,y)
print(data)

plot(x,y,main="Scatterplot",xlab="pH",ylab="Arsenic removed(%)")
```

R code Exa 12.3 A linear probabilistic model

```
#Ex12.3, Page 474
#Answers may slightly vary due to rounding off of probabilities

y<- function(x) (65-1.2*x)
sd<-8

#P(time-to-failure exceeds 50 when applied stress is 20)
x1<-20
mu1<-y(x1)
q<-50
```

```
P1<-1-pnorm(q,mu1,sd)
print(paste("Probability that the time-to-failure>50 when applied stress=20):",P1))

#P(time-to-failure exceeds 50 when applied stress is 25)
x2<-25
mu2<-y(x2)
q<-50
P2<-1-pnorm(q,mu2,sd)
print(paste("Probability that the time-to-failure>50 when applied stress=25):",P2))

#Let Y1<-observation made with x=25 and Y2<-observation made with x=24
#Mean value
X1<-25
X2<-24
E<-y(X1)-y(X2)
#Variance
V<-sd^2+sd^2
SD<-sqrt(V)
#Probability that Y1 exceeds Y2
P3<-1-pnorm(0,E,V)
print(paste("P(Y1>Y2):",P3))
```

R code Exa 12.4 Estimating model parameters

```
#Ex12.4, Page 479
#Answers may slightly vary due to rounding off of values

x<-c(132.0,129.0,120.0,113.2,105.0,92.0,84.0,83.2,88.4,59.0,80.0,81.5
```

```
y<-c
   (46.0,48.0,51.0,52.1,54.0,52.0,59.0,58.7,61.6,64.0,61.4,54.6,58.8
data1<-data.frame(x,y)

#To display regression model
model<-lm(y~x,data=data1)
cat("Regression model of data:\n")
a<-summary(model)
print(a)

#To display the scatterplot
plot(x,y,main="Scatterplot with least square line
   superimposed",xlab="iod val",ylab="cet num")
abline(a$coefficients[1],a$coefficients[2])
```

R code Exa 12.5 Estimating model parameters

```
#Ex12.5, Page 480

y<- function(x) {75.212-0.2094*x}
#If iodine value 100 is selected...
x<-100
print(paste("Point prediction for the resulting
   cetane number:",round(y(x),digits=2)))
```

R code Exa 12.6 Estimating model parameters

```
#Ex12.6, Page 482
#Answers may slightly vary due to rounding off of
   values

```

```
x<-c(125.3,98.2,201.4,147.3,145.9,124.7,112.2,120.2,161.2,178.9,159.5

y<-c(77.9,76.8,81.5,79.8,78.2,78.3,77.5,77.0,80.1,80.2,79.9,79.0,76.7

data1<-data.frame(x,y)

model<-lm(y~x)

#To display final table of predicted values and residuals
yb<-predict(model,data1)
Residual<-resid(model)
data1<-data.frame(x,y,yb,Residual)
colnames(data1)[colnames(data1)=="x"]<-"Filtrate"
colnames(data1)[colnames(data1)=="y"]<-"Moistcon"
colnames(data1)[colnames(data1)=="yb"]<-"Fit"
print(data1)
```

R code Exa 12.7 Estimating model parameters

```
#Ex12.7, Page 483
#Answers may vary slightly due to rounding off of values

#From Ex12.6
x<-c(125.3,98.2,201.4,147.3,145.9,124.7,112.2,120.2,161.2,178.9,159.5

y<-c(77.9,76.8,81.5,79.8,78.2,78.3,77.5,77.0,80.1,80.2,79.9,79.0,76.7

data1<-data.frame(x,y)

```

```
model<-lm(y~x)
Residual<-resid(model)
n<-length(Residual)

SSE<-sum(Residual^2)
print(paste("SSE:",SSE))

#To find variance(sigma^2)
s2<-SSE/(n-2)
print(paste("Estimate of variance:",s2))

#To find standard deviation(sigma)
sd<-sqrt(s2)
print(paste("Estimate of standard deviation:",sd))

#To display ANOVA table
a<-aov(model,data=data1)
cat("ANOVA table:\n")
print(summary(a))
```

R code Exa 12.8 Estimating model parameters

```
#Ex12.8, Page 484
#Answers may vary slightly due to rounding off of
    values

x<-c(12,30,36,40,45,57,62,67,71,78,93,94,100,105)
y<-c
    (3.3,3.2,3.4,3.0,2.8,2.9,2.7,2.6,2.5,2.6,2.2,2.0,2.3,2.1)

data1<-data.frame(x,y)

#To display ANOVA table
model<-lm(y~x)
print(summary(model))
```

```
a<-aov(model,data=data1)
cat("ANOVA table:\n")
print(summary(a))
```

R code Exa 12.9 Coefficient of determination

```
#Ex12.9, Page 486
#Answers may vary slightly due to rounding off of
   values

#From Ex12.4
x<-c
   (132.0,129.0,120.0,113.2,105.0,92.0,84.0,83.2,88.4,59.0,80.0,81.5

y<-c
   (46.0,48.0,51.0,52.1,54.0,52.0,59.0,58.7,61.6,64.0,61.4,54.6,58.8

data1<-data.frame(x,y)

n<-14
beta0<-75.212432
beta1<-(-0.20938742)
yi<-779.2
xiyi<-71347.30
yi2<-43745.22

#To find SST and SSE
SST<-yi2-(yi^2)/n
print(paste("SST:",SST))
SSE<-yi2-beta0*yi-beta1*xiyi
print(paste("SSE:",SSE))

#Coefficient of determination
r2<-1-(SSE/SST)
print(paste("Coefficient of determination:",r2))
```

```

#To create regression model using ANOVA table
a<-lm(y~x)
print(summary(a))
a1<-aov(a,data=data1)
print(summary(a1))
```

R code Exa 12.11 Confidence interval for beta1

```
#Ex12.11, Page 494
#Answers may vary slightly due to rounding off of
    values

x<-c
    (5.7,6.8,9.6,10.0,10.7,12.6,14.4,15.0,15.3,16.2,17.8,18.7,19.7,20

y<-c
    (119.0,121.3,118.2,124.0,112.3,114.1,112.2,115.1,111.3,107.2,108.

plot(x,y,main="Scatterplot",xlab="Air content",ylab=
    "Density")

n<-length(x)
xi<-sum(x)
yi<-sum(y)
#To find sum of xi^2, yi^2 and xi*yi
xi2<-sum(x^2)
yi2<-sum(y^2)
xiyi<-sum(x*y)

#To find Sxx and Sxy
Sxx<-xi2-(xi^2)/length(x)
Sxy<-xiyi-(yi*xi)/length(y)
print(paste("Sxx:",Sxx))
```

```
print(paste("Sxy:",Sxy))

#To find beta1 and beta0
xbar<-mean(x)
ybar<-mean(y)
beta1<-Sxy/Sxx
beta0<-ybar-beta1*xbar
print(paste("Beta1:",beta1))
print(paste("Beta0:",beta0))

#To find SSE
SSE<-yi2-beta0*yi-beta1*xiyi
print(paste("SSE:",SSE))

#To find SST
SST<-yi2-yi^2/n
print(paste("SST:",SST))

#To find r^2
r2<-1-(SSE/SST)
print(paste("r^2:",r2))

I<-15
J<-2
df<-I-J

#s^2
s2<-SSE/df
s<-sqrt(s2)

#Estimated standard deviation of beta1
sb<-s/sqrt(Sxx)
print(paste("Estimated standard deviation of beta1:"
   ,sb))

alpha<-0.05
t<-qt(1-alpha/2,df)

```

```
#To find confidence intervals
l<-beta1-t*sb
r<-beta1+t*sb
print(paste("Left endpoint of confidence interval:",l))
print(paste("Right endpoint of confidence interval:",r))

#To display analysis of variance and parameter estimates
model<-lm(y~x)
cat("\nParameter estimates\n")
print(summary(model))
cat("\nANOVA table:\n")
print(summary(aov(model)))
a<-predict(model)
print(paste("Predict values:"))
print(a)
print(paste("Residuals:"))
print(resid(model))
```

R code Exa 12.12 Hypothesis testing procedures

```
#Ex12.12, Page 496
#Answers may vary slightly due to rounding off of values

x<-c(42.2,42.6,43.3,43.5,43.7,44.1,44.9,45.3,45.7,45.7,45.9,46.0,46.2
y<-c(44,44,44,45,45,46,46,46,47,48,48,48,47,48,48,49,49,49)
data1<-data.frame(x,y)

```

```
model<-lm(y~x,data=data1)
cat("Regression model:\n")
print(summary(model))

#To display ANOVA table
a<-aov(model)
cat("ANOVA table:\n")
print(summary(a))
```

R code Exa 12.13 Inferences concerning mean

```
#Ex12.13, Page 501
#Answers may vary slightly due to rounding off of
   values

library(ggplot2)

x<-c
   (8.0,15.0,16.5,20.0,20.0,27.5,30.0,30.0,35.0,38.0,40.0,45.0,50.0,

y<-c
   (22.8,27.2,23.7,17.1,21.5,18.6,16.1,23.4,13.4,19.5,12.4,13.2,11.4

data1<-data.frame(x,y)

#To plot data using ggplot
ggplot(data=data1,mapping=aes(x,y))+geom_point(size
   =2)+geom_smooth(method=lm)

#To display linear regression model
model<-lm(y~x)
print(summary(model))

#To display analysis of variance
cat("ANOVA table:\n")
```

```r
a<-aov(model,data=data1)
print(summary(a))

#To display confidence and prediction intervals
d<-data.frame(x=c(35,45))
cat("Confidence intervals:\n")
print(predict(model,newdata=d,interval="confidence",
   level=0.95))
cat("Prediction intervals:\n")
print(predict(model,newdata=d,interval="predict",
   level=0.95))
```

R code Exa 12.14 Prediction interval for a future value of Y

```r
#Ex12.14, Page 505
#Answers may vary slightly due to rounding off of
   values

ycap<-13.79
sy<-0.7582
s<-2.8640
n<-18
df<-n-2
t<-2.120

#To find confidence intervals
l<-ycap-t*sqrt(s^2+sy^2)
r<-ycap+t*sqrt(s^2+sy^2)
print(paste("Left endpoint of confidence interval:",
   l))
print(paste("Right endpoint of confidence interval:"
   ,r))
```

R code Exa 12.15 Correlation

```
#Ex12.15, Page 509
#Answers may vary slightly due to rounding off of
   values

x<-c(2.4,3.4,4.6,3.7,2.2,3.3,4.0,2.1)
y<-c(1.33,2.12,1.80,1.65,2.00,1.76,2.11,1.63)

#To find sample correlation coefficient
r<-cor(x,y)
print(paste("Sample correlation coefficient,r:",r))
```

R code Exa 12.16 Inferences about the population correlation coefficient

```
#Ex12.16, Page 511
#Answers may vary slightly due to rounding off of
   values

x<-c
   (0.066,0.088,0.120,0.050,0.162,0.186,0.057,0.100,0.112,0.055,0.15

y<-c
   (4.6,11.6,9.5,6.3,13.8,15.4,2.5,11.8,8.0,7.0,20.6,16.6,9.2,17.9,2


#To find r(point estimate of population correlation
   coefficient)
r<-cor(x,y)
print(paste("r:",r))
```

R code Exa 12.17 Inferences about the population correlation coefficient

```r
#Ex12.17, Page 513
#Answers may vary slightly due to rounding off of values

r<-0.29
n<-45
#To find the test statistic
t<-r*sqrt(n-2)/sqrt(1-r^2)
print(paste("t:",round(t,digits=0)))

#P value for two tailed test
p<-2*pt(-abs(t),df=n-1)
print(p)
```

R code Exa 12.18 Other inferences concerning rho

```r
#Ex12.18, Page 515
#Answers may vary slightly due to rounding off of values

x<-c
   (55.10,44.83,46.32,51.10,49.89,45.20,48.18,46.70,54.31,41.50,47.5
y<-c
   (49.10,31.20,32.80,42.60,42.50,32.70,36.21,40.40,37.42,30.80,35.3

#To find r
r<-cor(x,y)
print(paste("Correlation coefficient, r:",r))

#H0: rho=0.5
#Ha: rho>0.5
rho<-0.5

```

```
#To find v
v<-0.5*log((1+r)/(1-r))
print(paste("v:",v))

muv<-0.5*log((1+rho)/(1-rho))
print(paste("muv:",muv))

z<-(v-muv)*sqrt(length(x)-3)
print(paste("z:",z))

#To find P value
p<-pnorm(z,lower.tail=FALSE)
print(paste("p value:",p))

print(paste("H0 is rejected..."))
```

R code Exa 12.19 Other inferences concerning rho

```
#Ex12.19, Page 516
#Answers may vary slightly due to rounding off of
    values

xi<-285.90
xi2<-4409.55
yi<-690.30
yi2<-29040.29
xiyi<-10818.56
r<-0.733
v<-0.935
n<-20
alpha<-0.05

#To find z
zalpha2<-qnorm(1-alpha/2)

```

```
#95% interval for muv
c1<-v-zalpha2/sqrt(n-3)
c2<-v+zalpha2/sqrt(n-3)
print(paste("Left endpoint of CI for muv:",c1))
print(paste("Right endpoint of CI for muv:",c2))

#95% interval for rho
l<-(exp(1)^(2*c1)-1)/(exp(1)^(2*c1)+1)
r<-(exp(1)^(2*c2)-1)/(exp(1)^(2*c2)+1)
print(paste("Left endpoint of CI for rho:",l))
print(paste("Right enpoint of CI for rho:",r))
```

Chapter 13

Nonlinear and multiple regression

R code Exa 13.2 Diagnostic plots

```
#Ex13.2, Page 526

x<-c
   (100,125,125,150,150,200,200,250,250,300,300,350,400,400)
y<-c
   (150,140,180,210,190,320,280,400,430,440,390,600,610,670)
y_cap<-(-45.55)+(1.71*x)
m1<-lm(y~x)
ei<-round(resid(m1),digits=1)
ei_std<-round(rstandard(m1),digits=2)
df<-data.frame(x,y,y_cap,ei,ei_std)
print(df)

par(mfrow=c(3,2))
plot(x,y,main="y vs. x")
abline(m1)
plot(y_cap,ei_std,main="Standardized residuals vs. y
```

```
    _cap")
abline(0,0)
plot(y,y_cap,main="y_cap vs. y")
abline(lm(y_cap~y))
plot(x,ei_std,main="Standardized residuals vs. x")
abline(0,0)
qqnorm(ei_std,xlab="z percentile",ylab="e*",main="
    Normal probability plot")
```

R code Exa 13.4 Regression with transformed variables

```
#Ex13.4, Page 534
#Answers may vary slightly due to rounding off of
    values

x<-c(2,10,20,30,40,50,60,70,80,90,100)
y<-c(408,274,196,137,90,78,51,40,30,22,15)
y_dash<-log(y,base=exp(1))
m1<-lm(y~x)
m2<-lm(y_dash~x)
ei_std1<-rstandard(m1)
ei_std2<-rstandard(m2)

par(mfrow=c(2,2))
plot(x,y,main="Scatterplot")
abline(m1)
plot(x,ei_std1,xlab="x",ylab="e*",main="Residual
    plot from linear regression for the data")
abline(0,0)

#To find the coefficients
beta0<-m2$coefficients[1]
beta1<-round(m2$coefficients[2],digits=4)

y_cap<-(exp(1)^(beta0))*(exp(1)^(beta1*x))
```

```
df<-data.frame(x,y,y_dash,y_cap)
print(df)

plot(x,ei_std2,ylab="e*",main="Standardized
   residuals(after transforming) vs. x")
abline(0,0)
plot(y,y_cap,main="ycap vs. y")
abline(lm(y_cap~y))
```

R code Exa 13.7 Polynomial regression

```
#Ex13.7, Page 544

Thickness<-c
   (220,220,220,220,370,370,370,370,440,440,440,440,680,680,680,680,

Strength<-c
   (24.0,22.0,19.1,15.5,26.3,24.6,23.1,21.2,25.2,24.0,21.7,19.2,17.0


df<-data.frame(Thickness,Strength)
cat("Regression model:\n")
relation<-lm(Strength~Thickness+I(Thickness^2))
print(summary(relation))

cat("Analysis of variance:\n")
model<-lm(Strength~poly(Thickness,2))
a<-aov(model,data=df)
print(summary(a))

#To find confidence and prediction intervals
cat("\nWhen thickness=500\n")
cat("95% CI:\n")
print(predict(relation,newdata=data.frame(Thickness
```

```
   =500),interval="confidence"))
cat("95% PI:\n")
print(predict(relation,newdata=data.frame(Thickness
   =500),interval="prediction"))

cat("\nWhen thickness=800\n")
cat("95% CI:\n")
print(predict(relation,newdata=data.frame(Thickness
   =800),interval="confidence"))
cat("95% PI:\n")
print(predict(relation,newdata=data.frame(Thickness
   =800),interval="prediction"))
```

R code Exa 13.8 Estimating parameters

```
#Ex13.8, Page 546
#Answers may vary slightly due to rounding off of
   values

#From Ex13.7
Thickness<-c
   (220,220,220,220,370,370,370,370,440,440,440,440,680,680,680,680,

Strength<-c
   (24.0,22.0,19.1,15.5,26.3,24.6,23.1,21.2,25.2,24.0,21.7,19.2,17.0

df<-data.frame(Thickness,Strength)
relation<-lm(df$Strength~poly(df$Thickness,2),data=
   df)
a<-summary(aov(relation,data=df))
print(a)

n<-20
k<-2
SSE<-a[[1]][, 'Sum Sq'][2]
```

```
SST<-a[[1]][, 'Sum Sq'][1]+a[[1]][, 'Sum Sq'][2]
R2<-1-(SSE/SST)
print(paste("R-squared value:",R2))

#To find variance
sigma2<-SSE/(n-(k+1))
print(paste("sigma^2:",sigma2))
print(paste("sigma:",sqrt(sigma2)))
```

R code Exa 13.9 Statistical intervals and test procedures

```
#Ex13.9, Page 548
#Answers may vary slightly due to rounding off of values

Thickness<-c(220,220,220,220,370,370,370,370,440,440,440,440,680,680,680,680,
Strength<-c(24.0,22.0,19.1,15.5,26.3,24.6,23.1,21.2,25.2,24.0,21.7,19.2,17.0

model<-lm(Strength~Thickness+I(Thickness^2))

new<-data.frame(Thickness=500)
y_cap<-predict(model,new)
print(paste("ycap when x=500:",y_cap))

#To find confidence and prediction intervals
cat("95% CI:\n")
print(predict(model,newdata=data.frame(Thickness=500),interval="confidence"))
cat("95% PI:\n")
print(predict(model,newdata=data.frame(Thickness=500),interval="prediction"))
```

R code Exa 13.10 Centering x values

```
#Ex13.10, Page 549
#Answers may vary slightly due to rounding off of values

x<-c(280,284,292,295,298,305,308,315)
x_dash<-c(-17.13,-13.13,-5.13,-2.13,0.87,7.87,10.87,17.87)
y<-c(770,800,840,810,735,640,590,560)
df<-data.frame(x,x_dash,y)

#To find estimated coefficients and standard deviations using original model
m1<-lm(y~x+I(x^2))
print(summary(m1))
Estimate1<-c(NULL)
Estimated_SD1<-c(NULL)
for(i in 1:3){
  Estimate1[i]<-m1$coefficients[i]
  Estimated_SD1[i]<-coef(summary(m1))[,"Std. Error"][i]
}
cat("Estimated coefficients and standard deviations using original model\n")
Parameter1<-c("beta0","beta1","beta2")
df1<-data.frame(Parameter1,Estimate1,Estimated_SD1)
print(df1)

#To find estimated coefficients and standard deviations using centered model
Estimate2<-c(NULL)
Estimated_SD2<-c(NULL)
m2<-lm(y~x_dash+I(x_dash^2))
```

```
print(summary(m2))
for(i in 1:3){
  Estimate2[i]<-m2$coefficients[i]
  Estimated_SD2[i]<-coef(summary(m2))[,"Std. Error"][i]
}
cat("Estimated coefficients and standard deviations using centered model\n")
Parameter2<-c("beta0*","beta1*","beta2*")
df2<-data.frame(Parameter2,Estimate2,Estimated_SD2)
print(df2)
```

R code Exa 13.12 Estimating parameters

```
#Ex13.12, Page 558
#Answers may vary slightly due to rounding off of values

Observation<-1:30
Force<-c(30,40,30,40,30,40,30,40,30,40,30,40,30,40,30,40,25,45,35,35,35,3

Power<-c(60,60,90,90,60,60,90,90,60,60,90,90,60,60,90,90,75,75,45,105,75,

Temperature<-c(175,175,175,175,225,225,225,225,175,175,175,175,225,225,225,225,

Time<-c(15,15,15,15,15,15,15,15,25,25,25,25,25,25,25,25,20,20,20,20,20,2

Strength<-c(26.2,26.3,39.8,39.7,38.6,35.5,48.8,37.8,26.6,23.4,38.6,52.1,39.5

```

```
m1<-lm(Strength~(Force+Power+Temperature+Time))

#To form estimated regression equation
Force<-35  #Force
Power<-75  #Power
Temperature<-200 #Temperature
Time<-20  #Time

new<-data.frame(Force,Power,Temperature,Time)
y_cap<-predict(m1,new)
print(paste("Point prediction of strength at
   (35,75,200,20):",y_cap,"gm"))
```

R code Exa 13.13 Estimating parameters

```
#Ex13.13, Page 560
#Answers may vary slightly due to rounding off of
   values

temp1<-x1<-c(21,21,7,7,28,0,14,14,14)
temp2<-x2<-c
   (0.65,0.55,0.65,0.55,0.60,0.60,0.70,0.50,0.60)
x1x2<-x1*x2
Comp_str<-c
   (33.55,47.55,35.00,35.90,40.90,39.10,31.55,48.00,42.30)

Adsorbability<-c
   (8.42,6.26,6.74,6.59,7.28,6.90,10.80,5.63,7.43)
df1<-data.frame(x1,x2,x1x2,Comp_str,Adsorbability)

#To find mean and SST values
ybar_compstr<-mean(Comp_str)
ybar_Adsorb<-mean(Adsorbability)
SST_CompStr<-sum((Comp_str-ybar_compstr)^2)
SST_Adsorb<-sum((Adsorbability-ybar_Adsorb)^2)
```

```r
cat("Mean of compression strength values:",ybar_compstr,"\n")
cat("Mean of adsorbability values:",ybar_Adsorb,"\n")
cat("SST of compression strength:",SST_CompStr,"\n")
cat("SST of adsorbability:",SST_Adsorb,"\n\n")

#First-order regression model
m1<-lm(Comp_str~(x1+x2))
SSE1<-sum(resid(m1)^2)
cat("SSE for first-order model:",SSE1,"\n")
cat("R squared value:",summary(m1)$r.squared,"\n\n")

#Model including interaction predictor
m2<-lm(Comp_str~(x1*x2))
SSE2<-sum(resid(m2)^2)
cat("SSE for first-order model including interaction predictor:",SSE2,"\n")
cat("R squared value:",summary(m2)$r.squared,"\n\n")

#To create estimated regression function
x1<-14 #% limestone
x2<-0.6 #Water-cement ratio
new<-data.frame(x1,x2)
y_cap<-predict(m2,new)
cat("Prediction of compression strength at (14,0.6):",y_cap,"\n")

#When adsorbability is taken as the dependent variable
#First-order regression model
m3<-lm(Adsorbability~(temp1+temp2))
cat("R squared value for first-order model:",summary(m3)$r.squared,"\n")

#Model including interaction predictor
m4<-lm(Adsorbability~(temp1*temp2))
cat("R squared value when using interaction
```

```
    predictor:",summary(m4)$r.squared,"\n\n")
```

R code Exa 13.14 Model utility test

```
#Ex13.14, Page 562

#H0: beta1=beta2=beta3=beta4=0
#Ha: at least one of these four betas!=0

Observation<-1:30
Force<-c
    (30,40,30,40,30,40,30,40,30,40,30,40,30,40,30,40,25,45,35,35,35,3

Power<-c
    (60,60,90,90,60,60,90,90,60,60,90,90,60,60,90,90,75,75,45,105,75,

Temperature<-c
    (175,175,175,175,225,225,225,225,175,175,175,175,225,225,225,225,

Time<-c
    (15,15,15,15,15,15,15,15,25,25,25,25,25,25,25,25,20,20,20,20,20,2

Strength<-c
    (26.2,26.3,39.8,39.7,38.6,35.5,48.8,37.8,26.6,23.4,38.6,52.1,39.5


Model<-cbind(Force,Power,Temperature,Time)

m1<-lm(Strength~Model)
print(summary(m1))

#To display analysis of variance table
a<-summary(aov(Strength~Model))
print(a)

```

```
print(paste("F value:",a[[1]]$F[1]))
```

R code Exa 13.15 Inferences in multiple regression

```
#Ex13.15, Page 564
#Answers may vary slightly due to rounding off of values

x1<-c(61,175,111,124,130,173,169,169,160,244,257,333,199)
x2<-c(13,21,24,23,64,38,33,61,39,71,112,88,54)
y<-c(4,18,14,18,26,26,21,30,28,36,65,62,40)
data1<-data.frame(x1,x2,y)

model<-lm(y~(x1+x2))
print(summary(model))

new<-data.frame(x1=160,x2=39)
y_cap<-predict(model,new)
print(paste("y_cap:",y_cap))

cat("\n99% CI:\n")
print(confint(model,level=0.99))

#To find confidence and prediction intervals
cat("95% CI:\n")
print(predict(model,new,interval="confidence"))
cat("95% PI:\n")
print(predict(model,new,interval="prediction"))
```

R code Exa 13.16 Inferences in multiple regression

```
#Ex13.16, Page 566
#Answers may slightly vary due to rounding off of values

Observation<-1:30
x1<-c(8,2,7,10,7,7,7,5,4,5,8,2,4,6,10,4,4,10,5,8,10,2,6,7,5,8,4,6,4,7)
x2<-c(4,4,4,7,4,7,13,4,7,1,10,4,10,7,13,10,13,10,4,13,1,13,13,1,13,1,1
x3<-c(100,180,180,120,180,180,140,160,140,100,140,100,180,120,180,160,
x4<-c(1,7,1,5,5,1,1,7,3,7,3,3,3,7,3,5,7,7,1,1,1,1,7,7,1,7,7,1,1,7)
y<-c(1.4,2.2,4.6,4.9,4.6,4.7,4.6,4.5,4.8,1.4,4.7,1.6,4.5,4.7,4.8,4.6,
x12<-x1^2
x22<-x2^2
x32<-x3^2
x42<-x4^2
x1x2<-x1*x2
x1x3<-x1*x3
x1x4<-x1*x4
x2x3<-x2*x3
x2x4<-x2*x4
x3x4<-x3*x4
df1<-data.frame(Observation,x1,x2,x3,x4,y)
print(df1)

#To find estimate for reduced model
cat("\nReduced model:\n")
m1<-lm(y~x1+x2+x3+x4)
print(summary(m1))

```

```
#To find estimate for full model
cat("Full model:\n")
m2<-lm(y~(x1+x2+x3+x4+x12+x22+x32+x42+x1x2+x1x3+x1x4+x2x3+x2x4+x3x4))
print(summary(m2))

#H0: beta5=beta6=.....=beta14=0
#Ha: at least one among beta5,....beta14 is not 0
n<-30
k<-14
l<-4
alpha<-0.01
#To find F value
Fval<-qf(1-alpha,df1=k-l,df2=n-(k+1))
cat("F value:",Fval,"\n")

#Unexplained variation for the full model
SSEk<-sum(resid(m2)^2)

#Unexplained variation for the reduced model
SSEl<-sum(resid(m1)^2)

#f value
f<-((SSEl-SSEk)/(k-l))/(SSEk/(n-(k+1)))
cat("f value:",f,"\n")

if(f>=Fval) cat("H0 is rejected since",f,">=",Fval)
```

R code Exa 13.18 Transformations

```
#Ex13.18, Page 575

s<-c
  (20,20,20,20,20,20,20,20,20,60,60,60,60,60,60,60,60,60,100,100,10
```

```
l_1000s<-c(3,3,3,6,6,6,10,10,10,3,3,3,6,6,6,10,10,10,3,3,3,6,6,6,10,10,10)
w<-c(300.2,310.8,333.0,99.6,136.2,142.4,20.2,28.2,102.7,67.3,77.9,93.

si<-log(s,base=exp(1))
li<-log(l_1000s,base=exp(1))
wi<-log(w,base=exp(1))

#To display estimated coefficients and t ratios
df1<-data.frame(si,li,wi)
model<-lm(wi~(si+li),data=df1)
print(summary(model))
```

Chapter 14

Goodness of fit tests and categorical data analysis

R code Exa 14.1 Goodness of fit tests when category probabilities are completely specified

```
#Ex14.1 , Page 597
#Answers may vary slightly due to rounding off of
   value

p<-c(9/16,3/16,3/16,1/16)
ni<-c(926,288,293,104)
n<-1611
k<-4
df<-k-1

npi<-round(n*p,digits=1)
df1<-data.frame(ni,npi)

#To find contribution to chi square from each cell
chi<-c(NULL)
for(i in 1:length(ni)){
  s<-((ni[i]-npi[i])^2)/npi[i]
  chi[i]<-s
```

```
}
print(paste("Chi-square values:"))
print(chi)

sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.1
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))

if(sum < chival) print(paste("H0 cannot be rejected"
  )) else  print(paste("H0 can be rejected"))
```

R code Exa 14.2 P values for chi squared tests

```
#Ex14.2, Page 599
#Answers may vary slightly due to rounding off of
  values

p<-9/16
n<-4
k<-5
df<-k-1
b<-c(NULL)
prob<-dbinom(0:4,n,p)
print(prob)

npi<-269*prob
print(npi)

Observed<-c(16,45,100,82,26)
Expected<-npi

```

```
df1<-data.frame(Observed,Expected)

#To find contribution to chi square from each cell
chi<-c(NULL)
for(i in 1:length(prob)){
  s<-((Observed[i]-Expected[i])^2)/Expected[i]
  chi[i]<-round(s,digits=3)
}
print(paste("Chi-square values:"))
print(chi)

sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.01
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))

if(sum < chival) print(paste("H0 cannot be rejected"
    )) else  print(paste("H0 can be rejected"))
```

R code Exa 14.3 Chi square when underlying distribution is continuous

```
#Ex14.3, Page 600
#Answers may vary slightly due to rounding off of
    values

k<-24
df<-k-1
p<-1/24

#Cell counts
Observed<-c
    (52,73,89,88,68,47,58,47,48,53,47,34,21,31,40,24,37,31,47,34,36,4
```

```
npi<-1186*p
Expected<-c(NULL)
for(i in 1:length(Observed)){
  Expected[i]<-npi
}
chi<-c(NULL)
for(i in 1:length(Observed)){
  s<-((Observed[i]-Expected[i])^2)/Expected[i]
  chi[i]<-round(s,digits=3)
}
sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.01
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))

if(sum < chival) print(paste("H0 cannot be rejected"
   )) else  print(paste("H0 can be rejected"))
```

R code Exa 14.4 Chi square when underlying distribution is continuous

```
#Ex14.4, Page 600
#Answers may vary slightly due to rounding off fo
   values

#Cell counts
Observed<-c(21,17,12,16,10,15,19,10)
p<-1/8
n<-120
k<-8
df<-k-1
npi<-n*p
```

```
Expected<-c(NULL)
for(i in 1:length(Observed)){
  Expected[i]<-npi
}
chi<-c(NULL)
for(i in 1:length(Observed)){
  s<-((Observed[i]-Expected[i])^2)/Expected[i]
  chi[i]<-round(s,digits=3)
}
sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.1
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))

if(sum < chival) print(paste("H0 cannot be rejected"
  )) else  print(paste("H0 can be rejected"))
```

R code Exa 14.5 Chi square when parameters are estimated

```
#Ex14.5, Page 603

Type<-c("Observed")
M<-c(125)
MN<-c(225)
N<-c(150)
print(paste("Observed counts:"))
df<-data.frame(Type,M,MN,N)
print(df)

n<-500

```

```
n1<-M
n2<-MN
theta<-(2*n1+n2)/(2*n)
print(paste("Resulting estimator:",theta))
```

R code Exa 14.6 Chi square when parameters are estimated

```
#Ex14.6, Page 604
#Answers may slightly due to rounding off of values

k<-3
m<-1
theta<-0.475
n<-500
npi1<-500*theta^2
npi2<-n*2*theta*(1-theta)
npi3<-n-npi1-npi2

Observed<-c(125,225,150)
Estimated_expected<-c(npi1,npi2,npi3)

chi<-c(NULL)
for(i in 1:length(Observed)){
  s<-((Observed[i]-Estimated_expected[i])^2)/
     Estimated_expected[i]
  chi[i]<-round(s,digits=3)
}
sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.05
df<-k-m-1
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))
```

```
if(sum < chival) print(paste("H0 is rejected")) else
    print(paste("H0 can be rejected"))
```

R code Exa 14.9 Goodness of fit for discrete distributions

```
#Ex14.9, Page 608
#Answers may vary slightly due to rounding off of
   values

k<-5
m<-1
mu<-2.10
n<-48
df<-k-m-1

Frequency<-c(9,9,10,14,6)

npi<-c(NULL)
for(i in 1:(k-1)){
  num<-(exp(1)^((-1)*mu))*(mu^(i-1))
  npi[i]<-n*(num/factorial(i-1))

}
npi[5]<-n-npi[1]-npi[2]-npi[3]-npi[4]
print(npi)

chi<-c(NULL)
for(i in 1:length(Frequency)){
  s<-((Frequency[i]-npi[i])^2)/npi[i]
  chi[i]<-round(s,digits=3)
}
sum<-sum(chi)
print(paste("Chi square value:",sum))

```

```
alpha<-0.05
#To find chi square value
chival<-qchisq(1-alpha,df)
print(paste("Critical value:",chival))

if(sum < chival) print(paste("H0 is not rejected
   since",sum,"<",chival)) else  print(paste("H0 can
    be rejected"))
```

R code Exa 14.10 Goodness of fit for continuous distributions

```
#Ex14.10, Page 608
#Answers may vary slightly from textbook values

data<-c
   (204,108,140,152,158,129,175,146,157,174,192,194,144,152,135,223,

n<-length(data)
mu_cap<-round(mean(data),digits=2)
sigma_cap<-round(sqrt((n-1)*(sd(data)^2)/n),digits
   =2)
print(paste("mu_cap:",mu_cap))
print(paste("sigma_cap:",sigma_cap))

#To find estimated expected counts
interval<-c(-Inf
   ,117.9,132.9,144.6,155.4,167.1,182.1,Inf)
p<-c(NULL)
npi<-c(NULL)
for(i in 1:length(interval)-1){
  p[i]<-pnorm(interval[i+1],mean=mu_cap,sd=sigma_cap
     )-pnorm(interval[i],mean=mu_cap,sd=sigma_cap)
  npi[i]<-n*p[i]
}
```

```
Cell<-c("(-Inf,117.9)","(117.9,132.9)","(132.9,144.6)","(144.6,155.4)","(155.4,167.1)","(167.1,182.1)","(182.1,Inf)")
Observed<-c(5,5,11,6,6,7,9)
Estimated_expected<-npi
data1<-data.frame(Cell,Observed,Estimated_expected)
print(data1)

#To find chi-squared values
alpha<-0.95
df1<-length(npi)-1
m<-2
q1<-qchisq(alpha,df=df1)
print(paste("X^2(0.05,k-1):",q1))

df2<-df1-m
q2<-qchisq(alpha,df=df2)
print(paste("X^2(0.05,k-1-m):",q2))
```

R code Exa 14.12 A special test for normality

```
#Ex14.12, Page 611

#Package to be installed: DescTools
library(DescTools)

yi<-c(-1.871,-1.404,-1.127,-0.917,-.742,-0.587,-0.446,-0.313,-0.186,-0
xi<-c(24.46,25.61,26.25,26.42,26.66,27.15,27.31,27.54,27.74,27.94,27.9

z<-c(NULL)
```

```
#To find z percentiles
for(i in 1:length(xi)){
  z[i]<-round(qnorm(1-((i-0.5)/length(xi)),lower.
     tail=FALSE),digits=2)
}

#Shapiro test is similar to Ryan-Joiner test which
   is not available in R
t<-shapiro.test(xi)
print(t)

#To create normal probability plot
qqnorm(xi,datax=TRUE,ylab="Voltage",xlab="
   Probability",main="Normal probability plot")
qqline(xi,datax=TRUE)
```

R code Exa 14.13 Testing for homogeneity

```
#Ex14.13, Page 615

print(paste("Reason for nonconformity:"))
Blemish<-c(34,23,32)
Crack<-c(65,52,28)
Location<-c(17,25,16)
Missing<-c(21,19,14)
Other<-c(13,6,10)
Sample_size<-c(150,125,100)
df1<-data.frame(Blemish,Crack,Location,Missing,Other
   ,Sample_size)
print(df1)


#Total values
Blemish_tot<-sum(Blemish)
Crack_tot<-sum(Crack)
```

```
Location_tot<-sum(Location)
Missing_tot<-sum(Missing)
Other_tot<-sum(Other)
Sample_tot<-sum(Sample_size)

#H0: production lines are homogeneous
#Ha: production lines are not not homogeneous

tbl<-cbind.data.frame(df1$Blemish,df1$Crack,df1$Location,df1$Missing,df1$Other)
c<-chisq.test(tbl,df1$Sample_size)
print(c)

print(paste("P value:",c$p.value))

print(paste("H0 should not be rejected at levels 0.05 or 0.01"))
```

R code Exa 14.14 Testing for independence

```
#Ex14.14, Page 617
#Answers may vary slightly due to rounding off of values

O<-c(24,15,17,52,73,80,58,86,36)
E<-c(17.02,22.10,16.89,62.29,80.88,61.83,54.69,71.02,54.29)

print(paste("Observed pricing policy"))
Observed<-matrix(data=O,nrow=3,ncol=3,byrow=TRUE)
rownames(Observed)<-c("Substandard","Standard","Modern")
colnames(Observed)<-c("Aggressive","Neutral","Nonaggressive")
```

```
print(Observed)

print(paste("Expected pricing policy"))
Expected<-matrix(data=E,nrow=3,ncol=3,byrow=TRUE)
rownames(Expected)<-c("Substandard","Standard","
    Modern")
colnames(Expected)<-c("Aggressive","Neutral","
    Nonaggressive")
print(Expected)

chi<-c(NULL)
for(i in 1:length(Observed)){
    chi[i]<-(Observed[i]-Expected[i])^2/(Expected[i
        ])
}
sum<-sum(chi)
print(paste("Chi square value:",sum))

alpha<-0.01
#To find chi square value
chival<-qchisq(1-alpha,df=4)
print(paste("Critical value (Chi square):",chival))
```

Chapter 15

Distribution free procedures

R code Exa 15.1 The Wilcoxon signed rank test

```
#Ex15.1, Page 629

library(stats)

x<-c(494.6,510.8,487.5,493.2,502.6,485.0,495.9,498.2,501.6,497.3,492.
y<-x-500

#Performing Wilcoxon signed rank test using wilcox.test function
print(wilcox.test(y))
```

R code Exa 15.2 Paired observations

```
#Ex15.2, Page 630

library(stats)
```

```r

IF_REE<-c(1753.7,1604.4,1576.5,1279.7,1754.2,1695.5,1700.1,1717.0)
Std_REE<-c(1755.0,1691.1,1697.1,1477.7,1785.2,1669.7,1901.3,1735.3)
Difference<-IF_REE-Std_REE

#Performing wilcoxon signed rank test
w1<-wilcox.test(Difference)     #or wilcox.test(IF_REE,Std_REE,paired=TRUE)
print(w1)

#Along with continuity correction
w2<-wilcox.test(Difference,exact=FALSE)    #or wilcox.test(IF_REE,Std_REE,paired=TRUE,exact=FALSE)
print(w2)
```

R code Exa 15.3 Large sample approximation

```r
#Ex15.3, Page 631
#Here p value is used to make an inference regarding the rejection of hypotheses
#wilcoxsign_test() from coin package can also be used to find z value if needed

#H0: mu=50000
#Ha: mu<50000

Signed_rank<-c(-1,-2,+3,-4,+5,-6,-7,+8,-9,-10,+11,-12,-13,+14,-15,-16,-17,+18,-
xi_50000<-c(-10,-27,36,-55,73,-77,-81,90,-95,-99,113,-127,-129,136,-150,-155
```

```
alpha_level<-0.01

w1<-wilcox.test(xi_50000,Signed_rank,paired=T,exact=F,correct=F,alt="two.sided")

#Using p value:
if(w1$p.value<alpha_level) print(paste("H0 is rejected since p value,",w1$p.value,"is smaller than level,",alpha_level)) else print(paste("H0 is not rejected"))
```

R code Exa 15.4 The Wilcoxon rank sum test

```
#Ex15.4, Page 637

Polluted<-c(21.3,18.7,23.0,17.1,16.8,20.9,19.7)
Unpolluted<-c(14.2,18.3,17.2,18.4,20.0)

#Performing Wilcoxon rank sum test....
print(wilcox.test(Polluted,Unpolluted,conf.level=0.99))
```

R code Exa 15.5 A normal approximation for W

```
#Ex15.5, Page 638
#Answers may vary slightly due to rounding off of values

Allergics<-c(67.6,39.6,1651.0,100.0,65.9,1112.0,31.0,102.4,64.7)
```

```
Nonallergics<-c
  (34.3,27.3,35.4,48.1,5.2,29.1,4.7,41.7,48.0,6.6,18.9,32.4,45.5)

m<-length(Allergics)
n<-length(Nonallergics)

#H0: mu1-mu2=0

new<-c(Allergics,Nonallergics)

#To find rank of combined vectors
r<-rank(new)
s<-0
for(i in 1:length(Allergics)){
  s<-s+r[i]
}
print(paste("Rank sum of allergics:",s))

#To find mean and variance of W
mu_w<-(m+n+1)*m/2
print(paste("Mean of W:",mu_w))
var_w<-m*n*(m+n+1)/12
print(paste("Variance of W:",var_w))

#To find test statistic value
z<-(s-mu_w)/sqrt(var_w)
print(paste("Test statistic value:",z))

alpha<-0.01
z1<-qnorm(alpha/2)

if(z>=z1 || z<=z1)  print(paste("H0 is rejected"))
  else print(paste("H0 is not rejected"))
```

R code Exa 15.6 The Wilcoxon signed rank interval

```
#Ex15.6, Page 642
#Interval values may vary slightly.....

#Given data
averages<-c(4.51,4.55,4.59,4.705,4.72,4.745,4.76,4.795,4.835,4.90,4.915,4.93

print(summary(averages))

#To display confidence intervals of the data
print(wilcox.test(averages,conf.int=TRUE,conf.level=0.95))
```

R code Exa 15.8 The Wilcoxon rank sum interval

```
#Ex15.8, Page 643
#Answers vary from that of text book(CI in text =>(4830,8220))

Epoxy<-c(10860,11120,11340,12130,14380,13070)
Other<-c(4590,4850,6510,5640,6390,0)

dat<-data.frame(Crush_Strength=c(Epoxy,Other),Bark_board=rep(c("Epoxy","Other"),each=6))
print(dat)

#To display differences between crushing strengths of both bark boards
o<-outer(Epoxy,Other,"-")
print(o)

#To find corresponding CI intervals
```

```
print(wilcox.test(Crush_Strength~Bark_board,data=dat,conf.int=TRUE,conf.level=0.95))
```

R code Exa 15.9 The Kruskal Wallis test

```
#Ex15.9, Page 646
#Answers may vary slightly due to rounding off of values

data<-c(309.2,309.7,311.0,316.8,326.5,349.8,409.5,331.0,347.2,348.9,361.
m1<-matrix(data,nrow=5,dimnames=list(c("4''","6''","8''","10''","12''")),byrow=TRUE)
print(m1)

N<-length(data)

#To find ranks of respective data
r1<-matrix(rank(m1),nrow=5)
cat("Ranks of data\n")
print(r1)

I<-nrow(r1)
J<-ncol(r1)

#To find rank sum and mean with respect to each plate length type
ri<-c(NULL)
ribar<-c(NULL)
for(i in 1:nrow(r1)){
  ri[i]<-sum(r1[i,])
  ribar[i]<-round(mean(r1[i,]),digits=2)
}
df1<-data.frame(ri,ribar)
```

```
cat("Rank sum and mean\n")
print(df1)

#To find k value
sum<-0
for(j in 1:nrow(r1)){
  sum<-sum+(ri[j])^2/J
}
k<-(12/(N*(N+1)))*sum-3*(N+1)
print(paste("k value:",k))

#To find chi-square value
alpha<-0.99
chival<-qchisq(alpha,df=I-1)
print(paste("Chi-squared value:",chival))

if(k>=chival) print(paste("H0 is not rejected since"
   ,k,">=",chival))  else  print(paste("H0 is
   rejected"))
```

R code Exa 15.10 Friedman test for a randomized block experiment

```
#Ex15.10, Page 647
#Answers may vary slightly due to rounding off of
   values

data<-c
   (23.1,57.6,10.5,23.6,11.9,54.6,21.0,20.3,22.7,53.2,9.7,19.6,13.8,
m1<-matrix(data,nrow=4,dimnames=list(c("Fear","
   Happiness","Depression","Calmness"),c("1","2","3"
   ,"4","5","6","7","8")),byrow=TRUE)
print(m1)

#To find ranks of respective data
```

```r
r1<-rank(m1[,1])
r2<-rank(m1[,2])
r3<-rank(m1[,3])
r4<-rank(m1[,4])
r5<-rank(m1[,5])
r6<-rank(m1[,6])
r7<-rank(m1[,7])
r8<-rank(m1[,8])
data1<-data.frame(r1,r2,r3,r4,r5,r6,r7,r8)
cat("\nRanks of data:\n")
print(data1)

I<-nrow(data1)
J<-ncol(data1)

#To find rank sum and mean with respect to each plate length type
ri<-c(NULL)
ri2<-c(NULL)
for(i in 1:nrow(data1)){
  ri[i]<-sum(data1[i,])
  ri2[i]<-(ri[i])^2
}
df1<-data.frame(ri,ri2)
print(df1)

#To find test statistic
Fr<-(12/(I*J*(I+1)))*sum(ri2)-3*J*(I+1)
print(paste("Test statistic value:",Fr))

#To find chi-square value
alpha<-0.95
chival<-qchisq(alpha,df=I-1)
print(paste("Chi-squared value:",chival))

if(Fr<chival) print(paste("H0 is not rejected since",Fr,"<",chival)) else print(paste("H0 is rejected"))
```

Chapter 16

Quality control methods

R code Exa 16.1 X bar chart based on known parameter values

```
#Ex16.1, Page 655
#Answers may vary slightly due to rounding off of values

#Package to be installed: qcc(Quality control charts)
library(qcc)

Vis_obs<-c(10.37,10.48,10.77,10.47,10.84,10.48,10.41,10.40,10.33,10.73,10.4
s<-c(rep(1:25,3))

q<-qcc.groups(Vis_obs,s)

#To plot X-bar chart
q1<-qcc(q,type="xbar",center=10.5,std.dev=0.18)
plot(q1)
print(summary(q1))
```

R code Exa 16.2 X bar charts based on estimated parameters

```
#Ex16.2, Page 657
#Answers may vary slightly due to rounding off of
   values

n<-3
k<-25
xdoublebar<-261.896/k
sbar<-3.834/k
a3<-0.886

#To find control limits
LCL<-xdoublebar-n*sbar/(a3*sqrt(n))
UCL<-xdoublebar+n*sbar/(a3*sqrt(n))
print(paste("LCL:",LCL))
print(paste("UCL:",UCL))

#Old limits:
#mu=10.5, sigma=0.18

#New limits:
print(paste("mu_cap:",xdoublebar))
print(paste("sigma_cap:",sbar/a3))
```

R code Exa 16.3 X bar charts based on estimated parameters

```
#Ex16.3, Page 658

n<-3
k<-25
xdoublebar<-261.896/k
```

```
rbar<-0.292
b3<-1.693
sigma_cap<-0.292/b3

#To find control limits
LCL<-xdoublebar-3*rbar/(b3*sqrt(n))
UCL<-xdoublebar+3*rbar/(b3*sqrt(n))
print(paste("LCL:",round(LCL,digits=3)))
print(paste("UCL:",round(UCL,digits=3)))
```

R code Exa 16.4 The S chart

```
#Ex16.4, Page 663
#Answers may vary slightly due to rounding off of values

#Package to be installed: qcc(Quality control charts)
library(qcc)

n<-4
k<-22
obs1<-c(29.7,32.2,35.9,28.8,30.9,30.6,32.3,32.0,24.2,33.7,35.3,28.1,28.7
obs2<-c(29.0,29.3,29.1,27.2,32.6,34.3,27.7,27.9,27.5,24.4,33.2,34.0,28.9
obs3<-c(28.8,32.2,32.1,28.5,28.3,34.8,30.9,31.0,28.5,34.3,31.4,31.0,25.8
obs4<-c(30.2,32.9,31.3,35.7,28.3,26.3,27.8,30.8,31.1,31.0,28.0,30.8,29.7
SD<-c
```

```
    (0.64,1.60,2.83,3.83,2.11,3.94,2.30,1.76,2.85,4.53,3.09,2.41,1.71
Range<-c
    (1.4,3.6,6.8,8.5,4.3,8.5,4.6,4.1,6.9,9.9,7.3,5.9,3.9,4.0,4.4,5.2,
data1<-data.frame(obs1,obs2,obs3,obs4)

#Sum of SD values
si<-sum(SD)
print(paste("Sum of SD values:",si))
sbar<-mean(SD)
print(paste("Mean of SD values:",sbar))

a4<-0.921
LCL<-0
UCL<-sbar+3*sbar*sqrt(1-a4^2)/a4
print(paste("LCL:",LCL))
print(paste("UCL:",UCL))

#To plot S chart
q4<-qcc(data1,type="S")
print(summary(q4))
```

R code Exa 16.5 R chart

```
#Ex16.5, Page 665
#Answers may vary slightly due to rounding off of
    values

#Package to be installed: qcc(Quality control charts
    )
library(qcc)

des_dim<-c
    (200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,
```

```
obs1<-c(12,6,5,19,9,9,8,4,11,13,10,8,14,7,14,10,7)
obs2<-c(17,9,9,6,14,15,11,14,14,9,14,9,7,9,5,12,11)
obs3<-c(6,17,15,11,9,8,12,11,7,9,8,4,9,12,8,10,15)
mean<-c(11.7,10.7,9.7,12.0,10.7,10.7,10.3,9.7,10.7,10.3,10.7,7.0,10.0,9.
range<-c(11,11,10,13,5,7,4,10,7,4,6,5,7,5,9,2,8)
st_dev<-c(5.51,5.69,5.03,6.56,2.84,3.79,2.08,5.13,3.51,2.31,3.06,2.65,3.61
data1<-data.frame(obs1,obs2,obs3)

n<-3
b3<-1.693
c3<-0.888

#To find sum of range values
ri<-sum(range)
rbar<-mean(range)

#To find control limits
UCL<-rbar+n*c3*rbar/b3
print(paste("LCL:",LCL))
print(paste("UCL:",UCL))

#To display X-bar chart (can only be displayed separately by using run method)
q1<-qcc(data1,type="xbar")
print(summary(q1))

#To display R chart (can only be displayed separately by using run method)
q2<-qcc(data1,type="R")
print(summary(q2))
```

R code Exa 16.6 The p chart

```
#Ex16.6, Page 669
#Answers may vary slightly due to rounding off of values

#Package to be installed: qcc(Quality control charts)
library(qcc)

Dayi<-c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,2
xi<-c(7,4,3,6,4,9,6,7,5,3,7,8,4,6,2,9,7,6,7,11,6,7,4,8,6)
pi<-c(0.07,0.04,0.03,0.06,0.04,0.09,0.06,0.07,0.05,0.03,0.07,0.08,0.04
data1<-data.frame(xi,pi)

#Sum of pi values
p<-sum(pi)
print(paste("Sum of pi values:",p))
pbar<-mean(pi)
print(paste("Mean of p values:",pbar))

#To find control limits
LCL<-pbar-3*sqrt(pbar*(1-pbar)/100)
UCL<-pbar+3*sqrt(pbar*(1-pbar)/100)
print(paste("LCL:",LCL))
print(paste("UCL:",UCL))

#To display p chart
```

```
q1<-qcc(data1,sizes=length(Dayi),type="p")
print(summary(q1))
```

R code Exa 16.7 c chart for number of defectives

```
#Ex16.7, Page 670
#Answers may slightly vary due to rounding off of values

#Package to be installed: qcc(Quality control charts)
library(qcc)

x<-c(7,10,9,12,13,6,13,7,5,11,8,10,13,9,21,10,6,8,3,12,7,11,14,10)

#To find the sum of x values
xi<-sum(x)
xbar<-mean(x)
print(paste("Sum of x values:",xi))
print(paste("Mean of x values:",xbar))

#To find control limits
LCL<-xbar-3*sqrt(xbar)
UCL<-xbar+3*sqrt(xbar)
print(paste("LCL:",LCL))
print(paste("UCL:",UCL))

#To display c chart
q1<-qcc(x,sizes=length(x),type="c")
print(summary(q1))

#From the chart, the 15th value lies above the UCL
#Eliminating that observation......
```

```
xbar1<-(sum(x)-x[15])/(length(x)-1)
print(paste("Mean value when the 15th value is
    eliminated:",xbar1))
LCL1<-xbar1-3*sqrt(xbar1)
UCL1<-xbar1+3*sqrt(xbar1)
print(paste("Control limits when 15th value is
    eliminated:"))
print(paste("New LCL value:",LCL1))
print(paste("New UCL value:",UCL1))

#Remaining 23 observation lie within control lines (
    in-control)
```

R code Exa 16.8 CUSUM procedures

```
#Ex16.8, Page 674

#Package to be installed: qcc(Quality control charts
    )
library(qcc)

n<-4
mu=mu0=40
sigma<-0.5
obs<-c
    (40.77,39.95,40.86,39.21,38.94,39.70,40.37,39.88,40.43,40.27,40.9

sa<-c(rep(1,4),rep(2,4),rep(3,4),rep(4,4),rep(5,4),
    rep(6,4),rep(7,4),rep(8,4),rep(9,4),rep(10,4),rep
    (11,4),rep(12,4),rep(13,4),rep(14,4),rep(15,4),
    rep(16,4))

#To display X-bar chart
q<-qcc.groups(data=obs,sample=sa)
q4<-qcc(q,center=mu0,std.dev=sigma,type="xbar")
```

```r
print(summary(q4))
```

R code Exa 16.9 Computational version

```r
#Ex16.9, Page 676

#Package to be installed: qcc(Quality control charts)
library(qcc)

mu0<-40
delta<-0.3
k<-delta/2
l<-mu0-k
r<-mu0+k
obs<-c(40.77,39.95,40.86,39.21,38.94,39.70,40.37,39.88,40.43,40.27,40.9

sa<-c(rep(1,4),rep(2,4),rep(3,4),rep(4,4),rep(5,4),rep(6,4),rep(7,4),rep(8,4),rep(9,4),rep(10,4),rep(11,4),rep(12,4),rep(13,4),rep(14,4),rep(15,4),rep(16,4))
xbar<-c(40.20,39.72,40.42,39.98,40.06,39.76,39.65,40.41,40.32,39.84,40.4


xbar1<-xbar-r
xbar2<-xbar-l

#To find d and e values
d0<-0
d1<-max(0,d0+(xbar[1]-r))
e0<-0
e1<-max(0,e0-(xbar[1]-l))

```

```
d<-c(d1)
e<-c(e1)
for(i in 2:length(xbar)){
  d[i]<-max(0,d[i-1]+(xbar[i]-r))
  e[i]<-max(0,e[i-1]-(xbar[i]-l))
}

df<-data.frame(xbar,xbar1,d,xbar2,e)
print(df)

q<-qcc.groups(data=obs,sample=sa)
c<-cusum(q,sizes=4,se.shift=delta,center=mu0)
print(summary(c))
```

R code Exa 16.10 Designing a CUSUM procedure

```
#Ex16.10, Page 680

sigma<-0.004
kbar<-0.74
delta<-0.003

n<-round((2*kbar*sigma/delta)^2,digits=0)
print(paste("Sample size:",n))

hbar<-3.2
h<-(sigma/sqrt(n))*hbar
print(paste("h value:",h))

print(paste("Out of control signal if d>",h,"or e>",
   h))
```

R code Exa 16.11 Acceptance sampling

```
#Ex16.11, Page 681

P<- function(p) {(1-p)^50+50*p*(1-p)^49+1225*p^2*(1-p)^48}

p<-c(0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.10,0.12,0.15)

#Finding P(A) values
PA<-c(NULL)
for(i in 1:length(p)){
  PA[i]<-round(P(p[i]),digits=3)
}

df<-data.frame(p,PA)
print(df)

#To plot data
plot(p,PA,ylab="P(A)",main="Curve plotting P(A) and p values")
```

R code Exa 16.12 Designing a single sample plan

```
#Ex16.12, Page 683
#Answers may vary slightly due to rounding off of values

#Package to be installed: AcceptanceSampling
library(AcceptanceSampling)

c<-0:15
np1<-c(0.051,0.355,0.818,1.366,1.970,2.613,3.285,3.981,4.695,5.425,6.16
```

```
np2<-c
   (2.30,3.89,5.32,6.68,7.99,9.28,10.53,11.77,12.99,14.21,15.41,16.6
p2_p1<-np2/np1
df1<-data.frame(c,np1,np2,p2_p1)
print(df1)
cat("\n")

AQL<-p1<-0.01
LTPD<-p2<-0.045

#Ratio of p2 to p1
r<-LTPD/AQL
print(paste("Ratio of p2 to p1:",r))

for(i in 1:length(c)){
  if(r>p2_p1[i+1] && r<p2_p1[i]){
    j<-i
    np_1<-np1[i]
    c1<-c[i]
    c2<-c[i+1]
    np_2<-np2[i+1]

  }
}
cat("Ratio lies between",p2_p1[j],"and",p2_p1[j+1],"
   values corresponding to c values",c[j],"and",c[j
   +1],"\n")

#Single sample plan: 1st c value
cat("When c value is",c1,"\n")

##To find sample size
n<-round(np_1/p1,digits=0)
print(paste("Sample size:",n))

##To find alpha and beta values
alpha1<-1-pbinom(c1,n,p1)
```

```
print(paste("Alpha value:",alpha1))

beta1<-pbinom(c1,n,p2)
print(paste("Beta value:",beta1))

#Single sample plan: 2nd c value
cat("When c value is",c2,"\n")

#To find sample size
n<-round(np_2/p2,digits=0)
print(paste("Sample size:",n))

##To find alpha and beta values
alpha2<-1-pbinom(c2,n,p1)
print(paste("Alpha value:",alpha2))

beta2<-pbinom(c2,n,p2)
print(paste("Beta value:",beta2))
```