

R Textbook Companion for
John E. Freund's Mathematical Statistics
With Applications
by John E. Freund¹

Created by
Arpit Dubey
M.Sc.
Mathematics
Indian institute of technology kharagpur
Cross-Checked by
R TBC Team

July 26, 2024

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

Book Description

Title: John E. Freund's Mathematical Statistics With Applications

Author: John E. Freund

Publisher: Pearson Education Limited ,india

Edition: 8

Year: 2013

ISBN: 978-9332519053

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

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

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

Contents

List of R Codes	4
1 Introduction	5
2 probability	15
3 Probability Distributions and Probability Densities	28
4 Mathematical Expectation	41
5 Special Probability Distributions	50
6 Special Probability Densities	57
7 Functions of Random Variables	61
8 Sampling Distributions	64
9 Decision Theory	66
10 Point Estimation	70
11 Interval Estimation	72
12 Hypothesis Testing	78
13 Tests of Hypothesis Involving Means Variances and Proportions	80
14 Regression and Correlation	88

List of R Codes

Exa 1.1	Counting travel options	5
Exa 1.2	Rolling two dice	5
Exa 1.3	Choosing inspection parts	6
Exa 1.4	Answering true false test	6
Exa 1.5	Permutations of abc	6
Exa 1.6	Introducing basketball players	7
Exa 1.7	Permutations of 2 letters	7
Exa 1.8	combination	7
Exa 1.9	Scheduling speakers for meetings	8
Exa 1.10	circular permutations	8
Exa 1.11	Permutations of the word book	8
Exa 1.12	Arranging novels on a shelf	9
Exa 1.13	Hanging paintings on a museum wall	9
Exa 1.14	Selecting households for market research	9
Exa 1.15	Coin toss outcomes	10
Exa 1.16	Forming committees of chemists and physicists	10
Exa 1.17	Partitioning a set into subsets	11
Exa 1.18	Assigning businessmen to hotel rooms	11
Exa 1.19	finding combination for n	12
Exa 1.20	finding combination of n	12
Exa 1.21	finding the combination	12
Exa 1.22	multinom	12
Exa 1.23	multinom	13
Exa 1.24	Selecting integrated circuit chips for assembly	13
Exa 1.25	Selecting units for inspection	14
Exa 2.1	ace in cards	15
Exa 2.2	pair of dice	15
Exa 2.3	divisible by 3	16

Exa 2.4	sum is 7	16
Exa 2.5	hit the target once and miss it twice	16
Exa 2.7	union	17
Exa 2.8	atleast one head	18
Exa 2.9	a number greater than 3	18
Exa 2.10	a probability measure	18
Exa 2.11	a full house	18
Exa 2.12	either or both kinds of sets	19
Exa 2.13	faulty brakes	19
Exa 2.14	dentist	20
Exa 2.15	services under warranty	20
Exa 2.16	good service under warranty	21
Exa 2.17	perfect square given that it is greater than 3	22
Exa 2.18	shipment	22
Exa 2.19	both be defective	22
Exa 2.20	replacement	23
Exa 2.21	3 fuses are defective	23
Exa 2.22	independent	24
Exa 2.23	independent	24
Exa 2.24	heads and die	25
Exa 2.25	construction job	25
Exa 2.26	oil change	26
Exa 2.27	rental agency 2	26
Exa 2.28	disease	26
Exa 2.29	reliability of the system	27
Exa 3.1	brown socks selected	28
Exa 3.2	number of heads	29
Exa 3.3	number of heads	30
Exa 3.4	probability distribution	30
Exa 3.5	number of heads	30
Exa 3.6	distribution function	31
Exa 3.7	probability distribution	31
Exa 3.9	Probability	32
Exa 3.10	distribution function	33
Exa 3.12	Two caplets are selected	33
Exa 3.13	a joint probability distribution	34
Exa 3.14	find F	34
Exa 3.15	Probability	35

Exa 3.17	Probability	35
Exa 3.18	Probability	36
Exa 3.19	trivariate probability density	36
Exa 3.23	conditional distribution	37
Exa 3.24	conditional density	37
Exa 3.27	r joint probability density	37
Exa 3.28	Construct a frequency distribution	38
Exa 3.29	frequency distribution	39
Exa 3.30	a histogram	39
Exa 4.1	expect	41
Exa 4.2	expected value	41
Exa 4.3	expected value	42
Exa 4.4	expected value	42
Exa 4.5	expected value	42
Exa 4.6	expected value	42
Exa 4.8	expected value	43
Exa 4.9	expected value	43
Exa 4.10	calculate the variance	44
Exa 4.11	standard deviation	44
Exa 4.12	Chebyshev theorem	45
Exa 4.14	moment generating function	45
Exa 4.15	covariance of X and Y	46
Exa 4.16	covariance	46
Exa 4.17	covariance	47
Exa 4.18	mean and the variance	48
Exa 4.19	covariance	48
Exa 4.20	conditional mean and the conditional variance	48
Exa 4.21	mean length and its standard deviation	49
Exa 5.1	binomial distribution	50
Exa 5.2	binomial distribution	50
Exa 5.3	negative binomial distribution	51
Exa 5.4	negative binomial distribution	51
Exa 5.5	geometric distribution	51
Exa 5.6	hypergeometric distribution	52
Exa 5.7	hyper and binomial	52
Exa 5.8	Poisson distribution	52
Exa 5.9	Poisson distribution	53
Exa 5.10	binomial probabilities	53

Exa 5.11	Poisson distribution	54
Exa 5.12	Poisson distribution	54
Exa 5.13	a Poisson distribution	54
Exa 5.14	MULTINOMIAL DISTRIBUTION	55
Exa 5.15	MULTINOMIAL DISTRIBUTION	55
Exa 5.16	producer and consumer risks	55
Exa 6.1	a Poisson distribution	57
Exa 6.2	standard normal distribution	57
Exa 6.4	cosmic radiation	58
Exa 6.5	chisquare distribution	58
Exa 6.6	binomial distribution	59
Exa 6.7	normal scores	59
Exa 6.8	normality	60
Exa 7.4	probability distribution	61
Exa 7.5	probability distribution	61
Exa 7.11	probability density	62
Exa 7.12	marginal density	62
Exa 7.19	probability	63
Exa 8.1	central limit theorem	64
Exa 8.2	Chisquare distribution	64
Exa 8.3	t distribution	65
Exa 9.1	deciding	66
Exa 9.2	decide	67
Exa 9.3	PAYOFF MATRIX	67
Exa 9.4	optimum strategies	68
Exa 9.5	minimax strategies	69
Exa 10.19	binomial trials	70
Exa 10.20	soft drink vending machines	70
Exa 10.21	voters	71
Exa 11.1	efficiency experts	72
Exa 11.2	confidence interval	72
Exa 11.3	confidence interval	73
Exa 11.4	paint manufacturer	73
Exa 11.5	confidence interval	74
Exa 11.6	two brands of cigarettes	74
Exa 11.7	a flu vaccine experienced	75
Exa 11.8	voters	75
Exa 11.9	running for governor of Illinois	75

Exa 11.10	gasoline consumption	76
Exa 11.11	confidence interval	76
Exa 11.12	car crossings	77
Exa 12.1	new medication	78
Exa 12.7	likelihood ratio technique	78
Exa 13.1	packages of cookies	80
Exa 13.2	high performance tires	81
Exa 13.3	ribbon call	81
Exa 13.4	nicotine	82
Exa 13.5	consumer testing service	82
Exa 13.6	a semiconductor	83
Exa 13.7	structural steel	83
Exa 13.8	patients suffering	84
Exa 13.9	oil company claims	84
Exa 13.10	shoppers favoring detergent	85
Exa 13.11	persons ability	86
Exa 13.12	a Poisson distribution	86
Exa 13.13	measurements of the heat producing capacity	87
Exa 14.4	equation of the least squares line	88
Exa 14.5	amount of time that 10 persons studied	88
Exa 14.6	confidence interval	89
Exa 14.7	complete a certain form	89
Exa 14.8	complete a certain form	90
Exa 14.9	onefamily houses sold	90
Exa 14.10	sales price of a threebedroom	91
Exa 14.11	least squares estimates	91
Exa 14.12	least squares estimates	92
Exa 14.13	least squares estimates using hypothesis	92

Chapter 1

Introduction

R code Exa 1.1 Counting travel options

```
1 # page no 2
2 transportation= 3
3 no_of_places= 5
4 total_ways=transportation*no_of_places
5 total_ways
6 print(total_ways)
```

R code Exa 1.2 Rolling two dice

```
1 # page no 3
2 selection_of_red_die=6
3 selection_of_green_die=6
4 total_ways=selection_of_green_die*selection_of_red_
  die
5 print(total_ways)
```

R code Exa 1.3 Choosing inspection parts

```
1 # page no 4
2 first_part=4
3 second_part=3
4 third_part=5
5 fourth_part=4
6 total_ways= first_part*second_part*third_part*fourth
  _part
7 print(total_ways)
```

R code Exa 1.4 Answering true false test

```
1 # page no 4
2 total_question=20
3 possible_ans=2
4 s=1
5 for (i in 1:20)
6 {
7   s=s*possible_ans
8
9 }
10 print(s)
```

R code Exa 1.5 Permutations of abc

```
1 # page number 4
2 total_letter= 3
3 letter_tobe_chosen=3
4 total_ways=choose(total_letter,letter_tobe_chosen)*
  factorial(letter_tobe_chosen)
5 print(total_ways)
```

R code Exa 1.6 Introducing basketball players

```
1 # page number 5
2 no_of_player=5
3 ways_wich_th_introdce=factorial(no_of_player)
4 print(ways_wich_th_introdce)
```

R code Exa 1.7 Permutations of 2 letters

```
1 # page number 5
2 num_of_lettr=4
3 selct_frm_lettr=2
4 s=1
5 for (i in 1:selct_frm_lettr)
6 {
7   s=s*num_of_lettr
8   num_of_lettr=num_of_lettr-1
9 }
10 print(s)
```

R code Exa 1.8 combination

```
1 # page number 6
2 total_member= 24
3 position_tobe_chosen=4
4 total_ways=choose(total_member,position_tobe_chosen)
5   *factorial(position_tobe_chosen)
6 print(total_ways)
```

R code Exa 1.9 Scheduling speakers for meetings

```
1 # page number= 6
2 total_dates= 5
3 num_of_spekr=3
4 total_ways= choose(total_dates,num_of_spekr)*
   factorial(num_of_spekr)
5 print(total_ways)
```

R code Exa 1.10 circular permutations

```
1 # page number = 7
2 no_of_prsn_play_circlr=4
3 ways=factorial(no_of_prsn_play_circlr-1)
4 print(ways)
```

R code Exa 1.11 Permutations of the word book

```
1 #page number= 7
2 letter="book"
3 n= nchar(letter)
4 letter_counts= table(strsplit(letter,"")[[1]])
5 s=1
6 for (i in 1:length(letter_counts))
7 {   s= s*(1/factorial(letter_counts[i][[1]]))
8     print(s)
9 }
10 total_ways=factorial(n)*s
11 print(total_ways)
```

R code Exa 1.12 Arranging novels on a shelf

```
1 #page number 7
2 copies_of_one_novel=3
3 novel_with_one_copy=4
4 total_novel=copies_of_one_novel+novel_with_one_copy
5 print(choose(total_novel,novel_with_one_copy)*
        factorial(novel_with_one_copy))
```

R code Exa 1.13 Hanging paintings on a museum wall

```
1 # page number = 8
2 library(iterpc)
3
4
5 painting_by_monet=2
6 painting_by_renoir=3
7 painting_by_degas=2
8 total_painting= painting_by_degas+painting_by_renoir
   +painting_by_monet
9
10
11 total_way= multichoose(c(painting_by_degas,painting_
   by_monet,painting_by_renoir))
12
13 print(total_ways)
```

R code Exa 1.14 Selecting households for market research

```

1 #page number= 8
2 total_houshld=20
3 select_houshld=3
4
5
6
7
8 total_ways =choose(total_houshld,select_houshld)*
   factorial(select_houshld)
9 print(total_ways)
10
11 order_not_matters=total_ways/factorial(select_
   houshld)
12 print(order_not_matters)

```

R code Exa 1.15 Coin toss outcomes

```

1 # page number 9
2
3
4 total_tossed_coins=6
5 no.of.heads=2
6 no.of.tails=4
7
8 total_ways= choose(total_tossed_coins,no.of.heads)
9 print(total_ways)

```

R code Exa 1.16 Forming committees of chemists and physicists

```

1 #page number= 9
2
3 total_chemist=4
4 number.of.chemist.selct=2

```

```

5 total_physicist=3
6 number.of.physicist.selct=1
7 way_chemist=choose(total_chemist,number.of.chemist.
    selct)
8 way_physicist=choose(total_physicist,number.of.
    physicist.selct)
9 total_ways=way_chemist*way_physicist
10 print(total_ways)

```

R code Exa 1.17 Partitioning a set into subsets

```

1 #page no 9
2 library(iterpc)
3 total_object=4
4 first_subset=1
5 second_subset=2
6 third_subset=1
7
8 total_subset= multichoose(c(first_subset,second_
    subset,third_subset))
9 print(total_subset)

```

R code Exa 1.18 Assigning businessmen to hotel rooms

```

1 # page number =11
2 total_hotel_rooms=7
3 assign_first_room=3
4 assign_second_room=2
5 assign_thired_room=2
6 total.ways= multichoose(c(assign_first_room,assign_
    second_room,assign_thired_room))
7 print(total.ways)

```

R code Exa 1.19 finding combination for n

```
1 # page number = 12
2 print(choose(4,3))
3 print(choose(4,4))
```

R code Exa 1.20 finding combination of n

```
1 # page number =12
2
3 print(choose(5,3))
4 print(choose(5,5))
```

R code Exa 1.21 finding the combination

```
1 # page number= 13
2 print(choose(20,12))
3 print(choose(17,10))
```

R code Exa 1.22 multinom

```
1 # page numebr= 15
2 m=2
3 n=3
4 k=4
5 sum=0
6 for(i in 0:k)
```

```

7 {
8   sum=sum+choose(m,i)*choose(n,k-i)
9 }
10
11 result= choose(m+n,k)
12
13 if(result==sum)
14 {
15   print(sum)
16 }

```

R code Exa 1.23 multinom

```

1 #page number= 15
2 library(iterpc)
3 n=6
4 r1=3
5 r2=1
6 r3=2
7 result= multichoose(c(r1,r2,r3))
8 print(result)

```

R code Exa 1.24 Selecting integrated circuit chips for assembly

```

1 # page number = 17
2 no.integrated.chips= 20
3 no_of_chips_must_solder=3
4 result= choose(no.integrated.chips,no_of_chips_must_
   solder)*factorial(no_of_chips_must_solder)
5 print(result)

```

R code Exa 1.25 Selecting units for inspection

```
1 # page number = 17
2 #- The answer provided in the textbook is wrong.
3 total_units=16
4 units_selected=4
5 result=choose(total_units,units_selected)
6 print(result)
```

Chapter 2

probability

R code Exa 2.1 ace in cards

```
1 # page number 21
2 library(MASS)
3 total_ace = 4
4 total_cards= 52
5 prob_of_ace= (total_ace)/total_cards
6 print(fractions(prob_of_ace))
```

R code Exa 2.2 pair of dice

```
1 # page number 23
2 red_dice <- 6
3 green_dice <- 6
4 dice_combinations <- expand.grid(1:red_dice, 1:green
  _dice)
5 dice_combinations
6 sums <- rowSums(dice_combinations)
7 cat(paste(sums, collapse = ","))
```

R code Exa 2.3 divisible by 3

```
1 # page numeber 24
2 red_dice <- 6
3 k <- 1:red_dice
4 divisible_by_3 <- k[k %% 3 == 0]
5 cat(paste(divisible_by_3, collapse = ","))
```

R code Exa 2.4 sum is 7

```
1 # page number=24
2
3 red_dice <- 6
4 green_dice <- 6
5 dice_combinations <- expand.grid(red_die = 1:red_
    dice, green_die = 1:green_dice)
6 valid_combinations <- subset(dice_combinations, red_
    die + green_die == 7)
7 x_coord <- valid_combinations$red_die
8 y_coord <- valid_combinations$green_die
9 cat(paste("(", x_coord, ", ", y_coord, ")", sep = " ",
    collapse = ", "))
10
11
12 plot(x_coord, y_coord, col = "blue", pch = 19, xlab
    = "Red die", ylab = "Green die")
```

R code Exa 2.5 hit the target once and miss it twice

```

1 # page number = 25
2 combinations <- expand.grid(m = 2:1, n = 2:1, l =
  2:1)
3 all_hit <- subset(combinations, m == 1 & n == 1 & l
  == 1)
4 one_miss <- subset(combinations, (m == 2 & n == 1 &
  l == 1) | (m == 1 & n == 2 & l == 1) | (m == 1 &
  n == 1 & l == 2))
5 cat(paste("(", all_hit$m-1, ", ", all_hit$n-1, ", ",
  all_hit$l-1, ")"), sep = " ", collapse = " ")
6 if (nrow(one_miss) > 0) {
7   cat(paste("(", one_miss$m-1, ", ", one_miss$n-1, ", ",
  one_miss$l-1, ")"), sep = " ", collapse = " ")
8 }

```

R code Exa 2.7 union

```

1 # page number 28
2 prob_A <- 0.12
3 prob_B <- 0.63
4 prob_C <- 0.45
5 prob_D <- -0.20
6 condition_one <- all(c(prob_A, prob_B, prob_C, prob_
  D) > 0)
7 condition_two <- sum(prob_A, prob_B, prob_C, prob_D)
  == 1
8 if(!condition_one){
9   print("violates Postulate 1")
10 }
11 if(condition_two){
12   print("violates Postulate 2.")
13 }

```

R code Exa 2.8 at least one head

```
1 # page number= 29
2 library(MASS)
3 prob_HH=1/4
4 prob_HT=1/4
5 prob_TH=1/4
6 probability=prob_HH+prob_HT+prob_TH
7 print(fractions(probability))
```

R code Exa 2.9 a number greater than 3

```
1 # page nummber= 30
2 library(MASS)
3 w= 1/9
4 s=c(1,2,1)
5 sum= w*sum(s)
6 print(fractions(sum))
```

R code Exa 2.10 a probability measure

```
1 # page number= 30
2 prob_0=1/2
3 sum=prob_0/(1-prob_0)
4 print(sum)
```

R code Exa 2.11 a full house

```
1 # page number 31
2 king_to_be_selected= 3
```

```

3 full_house_card=4
4 aces_to_be_selected=2
5 way_to_slct_king=choose(full_house_card,king_to_be_
    selected)
6 ways_to_selct_ace=choose(full_house_card,aces_to_be_
    selected)
7 total_cards_type1=13
8 n= total_cards_type1*(total_cards_type1-1)*way_to_
    slct_king*ways_to_selct_ace
9 total_ways_for5=choose(52,5)
10 prob= n/total_ways_for5
11 print(round(prob,4))

```

R code Exa 2.12 either or both kinds of sets

```

1 # page number =34
2 tv_set=0.86
3 hdtv_set=0.35
4 both_set=0.29
5 either_of_both=tv_set+hdtv_set-both_set
6 print(either_of_both)

```

R code Exa 2.13 faulty brakes

```

1 # page number = 34
2 faulty_breaks=0.23
3 badly_worn= 0.24
4 faulty_break_or_worn=0.38
5 both= faulty_breaks+badly_worn-faulty_break_or_worn
6 print(both)

```

R code Exa 2.14 dentist

```
1 # page number = 35
2 clean_teeth=0.44
3 cavity=0.24
4 extracted= 0.21
5 clean_filled=0.08
6 cleaned_extracted=0.11
7 filled_extracted=0.07
8 cleaned_filled_extracted=0.03
9 atleast_one_case= clean_teeth+cavity+extracted-clean
    _filled-cleaned_extracted-filled_extracted+
    cleaned_filled_extracted
10 print(atleast_one_case)
```

R code Exa 2.15 services under warranty

```
1 # page number= 37
2 good_sevice_under_warranty= c(16,10)
3 poor_service_under_warranty=c(4,20)
4 df=(data.frame(good_sevice_under_warranty,poor_
    service_under_warranty))
5 rownames(df)=c("In business 10 years or more","in
    business less than 10 years")
6
7 good_ser_provider= sum(df$good_sevice_under_warranty
    )
8 total_sum=0
9 for (i in colnames(df))
10 {
11     total_sum= total_sum+colSums(df[i])
12 }
13
14 prob_good_service= good_ser_provider/total_sum
15 print(prob_good_service[[1]])
```

```

16
17 provider_more_than_10= rowSums(df['In business 10
    years or more',])
18
19 value=df["In business 10 years or more","good_sevice
    _under_warranty"]
20 prob_good_service_with_10_years= value/provider_more
    _then_10
21 print(round(prob_good_service_with_10_years[[1]],2))

```

R code Exa 2.16 good service under warranty

```

1 # page number = 38
2 library(MASS)
3 good_sevice_under_warranty= c(16,10)
4 poor_service_under_warranty=c(4,20)
5 df=(data.frame(good_sevice_under_warranty,poor_
    service_under_warranty))
6 rownames(df)=c("In business 10 years or more","in
    business less than 10 years")
7 total_sum=0
8 total_sum = sum(sapply(df, sum))
9 less_than_10_good_ser=df["in business less than 10
    years",'good_sevice_under_warranty']
10 prob_less_than_10_good_ser=less_than_10_good_ser/
    total_sum
11 print(prob_less_than_10_good_ser[[1]])
12 less_than_10= rowSums(df["in business less than 10
    years",])
13 prob_less_than_10=less_than_10/total_sum
14 print(prob_less_than_10[[1]])
15
16 result= prob_less_than_10_good_ser/prob_less_than_10
17 print(fractions(result[[1]]))

```

R code Exa 2.17 perfect square given that it is greater than 3

```
1 # page number 39
2 library(MASS)
3 probb_odd= 2/9
4 probb_even= 1/9
5 probb_dice=c(2/9,1/9,2/9,1/9,2/9,1/9)
6 probb_B=probb_dice[1]+probb_dice[4]
7 probb_a_and_b= probb_dice[4]
8 probb_a=probb_dice[4]+probb_dice[5]+probb_dice[6]
9 probb_B_by_A=probb_a_and_b/probb_a
10 print(fractions(probb_B))
11 print(fractions(probb_B_by_A))
```

R code Exa 2.18 shipment

```
1 # page number 39
2 probb_R=0.8
3 probb_r_and_d=0.72
4 probb_d_by_r=probb_r_and_d/probb_R
5 print(probb_d_by_r)
```

R code Exa 2.19 both be defective

```
1 # page number= 40
2 library(MASS)
3 total_tv=240
4 no.defective.picked<- 2
5 total_defective=15
```

```

6 result= (total_defective/total_tv)*((total_defective
  -1)/(total_tv-1))
7 print(fractions(result))

```

R code Exa 2.20 replacement

```

1 # page number = 40
2 library(MASS)
3 total_cards= 52
4 no.of.drawn.cards=2
5 total_aces=4
6 result_without_replacement=1
7 result_without_replacement=(total_aces/total_cards)*
  ((total_aces-1)/(total_cards-1))
8 print(fractions(result_without_replacement))
9 result_with_replacement=(total_aces/total_cards)*(
  total_aces/total_cards)
10 print(fractions(result_with_replacement))

```

R code Exa 2.21 3 fuses are defective

```

1 # page number = 41
2 library(MASS)
3 fuses= 20
4 defective=5
5 selected= 3
6 prob_a=defective/fuses
7 prob_b_by_a=(defective-1)/(fuses-1)
8 prob_c_by_a_and_b=(defective-2)/(fuses-2)
9 result= prob_a*prob_b_by_a*prob_c_by_a_and_b
10 print(fractions(result))

```

R code Exa 2.22 independent

```
1 # page number= 42
2 prob_a=1/4
3 prob_b= 1/2
4 prob_c=3/8
5 prob_a_and_b=1/8
6 prob_b_and_c= 1/4
7 if(prob_a*prob_b==prob_a_and_b){
8   print("event A and B are independent")
9 }
10 if(!(prob_b*prob_c==prob_b_and_c)){
11   print("event B and C are not independent")
12 }
```

R code Exa 2.23 independent

```
1 # page number= 44
2 prob_a=1/2
3 prob_b= 1/2
4 prob_c=1/2
5 prob_a_and_b=1/4
6 prob_b_and_c= 1/4
7 prob_a_and_c=1/4
8 prob_a_b_c=1/4
9 if(prob_a*prob_b==prob_a_and_b)
10 {
11   print("P(A)*P(B)=1/4=P( A  B )")
12 }
13
14 if(prob_a*prob_c==prob_a_and_c){
15   print("P(A)*P(C)=1/4=P( A  C )")
16 }
```

```

16 }
17
18 if(prob_b*prob_c==prob_b_and_c){
19   print("P(B)*P(C)=1/4=P( B   C )")
20 }
21
22 if(!(prob_a*prob_b*prob_c==prob_a_b_c)){
23   print("P(A)*P(B)*P(C)=1/8   P ( A   B   C )")
24 }

```

R code Exa 2.24 heads and die

```

1 # page number = 45
2
3 library(MASS)
4 prob_of_head <- 1/2
5 tossed <- 3
6 dice_prob <- 1/6
7 result1 <- prob_of_head^tossed
8 result2 <- 5 * (dice_prob)^5
9 print(fractions(result1))
10 print(fractions(result2))

```

R code Exa 2.25 construction job

```

1 # page number = 45
2 prob_b=0.6
3 prob_a_given_b=0.35
4 prob_a_given_b_com=0.85
5 prob_a=prob_b*prob_a_given_b+(1-prob_b)*prob_a_given
  _b_com
6 print(prob_a)

```

R code Exa 2.26 oil change

```
1 # page number = 46
2 prob_b1=0.6
3 prob_b2=0.3
4 prob_b3=0.1
5 prob_a_given_b1=0.09
6 prob_a_given_b2=0.2
7 prob_a_given_b3= 0.06
8 prob_a= prob_b1*prob_a_given_b1+prob_b2*prob_a_given
    _b2+prob_b3*prob_a_given_b3
9 print(prob_a)
```

R code Exa 2.27 rental agency 2

```
1 # page number 47
2 prob_b1=0.6
3 prob_b2=0.3
4 prob_b3=0.1
5 prob_a_given_b1=0.09
6 prob_a_given_b2=0.2
7 prob_a_given_b3= 0.06
8 prob_b2_a= prob_b2*prob_a_given_b2/(prob_b1*prob_a_
    given_b1+prob_b2*prob_a_given_b2+prob_b3*prob_a_
    given_b3)
9 print(prob_b2_a)
```

R code Exa 2.28 disease

```

1 # page number = 48
2 prod_d=0.01
3 prod_p_d=0.98
4 prod_p_dcom=0.03
5 prod_dcom_p= (1-prod_d)*prod_p_dcom/(prod_d*prod_d
    +(1-prod_d)*prod_p_dcom)
6 print(round(prod_dcom_p,3))

```

R code Exa 2.29 reliability of the system

```

1 # page number = 51
2 reliability_CDE= 1-(1-0.70)^3
3 reliability_FG= 1-(1-0.75)^2
4 reliability_H= 0.90
5 reliability_a=0.95
6 reliability_b=0.99
7 result= reliability_a*reliability_b*reliability_CDE*
    reliability_FG*reliability_H
8 print(round(result,3))

```

Chapter 3

Probability Distributions and Probability Densities

R code Exa 3.1 brown socks selected

```
1 # page number 63
2 library(MASS)
3 total_brown <- 5
4 total_green <- 3
5
6 possible_outcomes <- expand.grid(sock1 = c("B", "G")
7                                ,
8                                sock2 = c("B", "G")
9                                )
10
11 possible_outcomes$brown <- rowSums(possible_outcomes
12 == "B")
13
14 probabilities <- table(possible_outcomes$brown) /
15   nrow(possible_outcomes)
16
17 prob <- function(n) {
18   (choose(total_brown, n)*factorial(n) * choose(
19     total_green, 2 - n)*factorial(2-n) )/ (choose(
```

```

        total_brown + total_green, 2)*factorial(2))
15 }
16
17 print(paste("P(BB)=",fractions(prob(2))))
18 print(paste("P(BG)=",fractions(prob(1))))
19 print(paste("P(GB)=",fractions(prob(1))))
20 print(paste("P(GG)=",fractions(prob(0))))

```

R code Exa 3.2 number of heads

```

1 #page number= 63
2 library(MASS)
3 coin_tossed <- 4
4 possible_outcomes <- expand.grid(
5   coin1 = c("H", "T"),
6   coin2 = c("H", "T"),
7   coin3 = c("H", "T"),
8   coin4 = c("H", "T")
9 )
10 possible_outcomes$combined <- paste(
11   possible_outcomes$coin1,
12   possible_outcomes$coin2,
13   possible_outcomes$coin3,
14   possible_outcomes$coin4,
15   sep = ""
16 )
17 possible_outcomes$sum_H <- rowSums(possible_outcomes
   == "H")
18 possible_outcomes$probability <- (1 / 16)
19 possible_outcomes$fraction_probability <- as.
   character(MASS::fractions(possible_outcomes$
   probability))
20 colnames(possible_outcomes)[5] <- "element of Sample
   Space"
21 colnames(possible_outcomes)[8] <- "Probability"

```

```
22 colnames(possible_outcomes)[6] <- "x"
23 print(possible_outcomes[, c(5, 8, 6)])
```

R code Exa 3.3 number of heads

```
1 # page number = 66
2 library(MASS)
3 coin_tossed <- 4
4 for( i in 0:coin_tossed)
5 {
6   print(paste("P(X=", i, ")=", fractions(choose(coin_
7     tossed, i)/2^coin_tossed)))
8   cat("\n")
9 }
```

R code Exa 3.4 probability distribution

```
1 # page number = 66
2
3 x=c(1,2,3,4,5)
4 fun= function(n)
5 {
6   return ((n+2)/25)
7 }
8 result=fun(x[1])+fun(x[2])+fun(x[3])+fun(x[4])+fun(x
9   [5])
10 print(result)
```

R code Exa 3.5 number of heads

```

1 # page number =69
2 library(MASS)
3 pdf=c(1/16,4/16,6/16,4/16,1/16)
4 sum=0
5 cdf=c()
6 for(i in pdf){
7   cdf=c(cdf,sum)
8   sum=sum+i
9
10 }
11 cdf=c(cdf,sum)
12 print(paste(fractions(cdf)))

```

R code Exa 3.6 distribution function

```

1 # page number 70
2 library(MASS)
3 f = c(3/28, 15/28, 5/14)
4 cdf = cumsum(f)
5 for (i in 1:3) {
6   print(paste("F(X=", i-1, ")=", fractions(cdf[i])))
7 }
8 plot(x = c(0, 1), y = c(cdf[1], cdf[1]), type = "l",
9      col = "black", xlim = c(0, 3), ylim = c(0, 1),
10      main = "Cumulative Distribution Function", xlab
11           = "X", ylab = "F(X)")
12 lines(x = c(1, 2), y = c(cdf[2], cdf[2]), type = "l",
13       , col = "black")
14 lines(x = c(2, 3), y = c(cdf[3], cdf[3]), type = "l",
15       , col = "black")

```

R code Exa 3.7 probability distribution

```

1 # page number = 71
2 library(MASS)
3 f=c(0,1/36,3/36,6/36,10/36,15/36,21/36,26/36,30/
      36,33/36,35/36,1)
4 f1=c(f[1])
5 for (i in 2:length(f))
6 {
7   sum=f[i]-f[i-1]
8   f1=c(f1,sum)
9 }
10
11 for( i in 2:(length(f1)))
12 {
13   print(paste("f(",i,")=",fractions(f1[i])))
14 }

```

R code Exa 3.9 Probability

```

1 # page number 76
2
3 fun <- function(x) {
4   return(exp(-3*x))
5 }
6
7 result <- integrate(fun, lower = 0, upper = Inf)
8
9 k = 1 / as.numeric(result$value)
10
11
12 print(k)
13
14 result<- integrate(fun,lower=0.5, upper= 1)
15 result=k*result$value
16 print(round(result,3))

```

R code Exa 3.10 distribution function

```
1 # page number = 78
2 f <- function(x) {
3   exp(-3*x)
4 }
5
6 result <- integrate(f, lower = 0, upper = Inf)
7
8 k = 1 / as.numeric(result$value)
9
10
11 print(k)
12
13 result1<- integrate(f,lower=0, upper= 1)
14 result2<- integrate(f,lower=0,upper=0.5)
15 result=k*(result1$value-result2$value)
16 print(round(result,3))
```

R code Exa 3.12 Two caplets are selected

```
1 # page number = 82
2
3 caplets <- c(rep("Aspirin", 3), rep("Sedative", 2),
4             rep("Laxative", 4))
5
6 combinations <- combn(caplets, 2)
7
8 count_table <- matrix(0, nrow = 3, ncol = 3,
9                       dimnames = list(0:2, 0:2))
```

```

10
11 for (i in 1:ncol(combinations)) {
12   combination <- combinations[,i]
13   X <- sum(combination=="Aspirin")
14   Y <- sum(combination=="Sedative")
15   count_table[X+1,Y+1]<-count_table[X+1,Y+1]+1
16 }
17
18 total_combinations <- ncol(combinations)
19 prob_table <- count_table / total_combinations
20 prob_table<- t(prob_table)
21
22 print(fractions(prob_table))

```

R code Exa 3.13 a joint probability distribution

```

1 # page number 83
2 library(MASS)
3 fun= function(x,y){
4   return (x*y)}
5 var <- expand.grid(1:3,1:3)
6 sum=sum(fun(var[1],var[2]))
7 k=fractions(1/sum)
8 print(k)

```

R code Exa 3.14 find F

```

1 # page number = 84
2 library(MASS)
3 f=c(1/6,2/9,1/3,1/6)
4 print(fractions(sum(f)))

```

R code Exa 3.15 Probability

```
1 # page number= 85
2 library(MASS)
3 fun= function(x,y){
4   return ((3/5)*x*(x+y))
5 }
6 result <-integrate(function(x) {sapply(x,function(x)
7   {
8     integrate(function(y) {
9       fun(x,y)
10      }, 0, 1-x)$value
11    })}, 0, 1)$value
12 print(round(result,3))
13 print(fractions(result))
```

R code Exa 3.17 Probability

```
1 # page number= 87
2
3 f <- function(x, y) {
4   return(exp(-x) * exp(-y))
5 }
6
7 x_lower <- 1
8 x_upper <- 3
9 y_lower <- 1
10 y_upper <- 2
11
12 P <- integrate(function(y) {
13   sapply(y, function(y_val) {
```



```

14     integrate(function(x) f(x, y_val), x_lower, x_
        upper)$value
15   })
16 }, y_lower, y_upper)$value
17 print(round(P,4))

```

R code Exa 3.18 Probability

```

1 # page nummber =89
2 f <- function(x, y, z) {
3   return((x + y) * z / 63)
4 }
5
6 P <- f(2, 1, 1) + f(2, 1, 2) + f(2, 2, 1)
7 print(fractions(P))

```

R code Exa 3.19 trivariate probability density

```

1 # page number 90
2 f <- function(x1, x2, x3) {
3   return((x1 + x2) * exp(-x3))
4 }
5 result <- integrate(function(x3) {sapply(x3,
        function(x3){
6     integrate(function(x2) {sapply(x2,function(x2){
7       integrate(function(x1) {
8         f(x1, x2, x3)
9       }, 0, 1/2)$value
10    })), 1/2, 1)$value
11  })), 0, 1)$value
12 cat(round(result,3))

```

R code Exa 3.23 conditional distribution

```
1 # page number = 95
2 f_0_1= 2/9
3 h_1=7/18
4 f_1_1=1/6
5 f_2_2=0
6
7 f0by1=f_0_1/h_1
8 f1by1=f_1_1/h_1
9 f2by2=f_2_2/h_1
10 print(fractions(f0by1))
11 print(fractions(f1by1))
12 print(fractions(f2by2))
```

R code Exa 3.24 conditional density

```
1 # page number = 96
2 fun_at_y= function(x){
3   return ((2*x+2)/3)
4 }
5
6 result= integrate(fun_at_y,0,1/2)$value
7 fractions(result)
```

R code Exa 3.27 r joint probability density

```
1 # page number = 99
2
```

```

3 fun = function(x1,x2,x3){
4   return (6*exp(-x1-2*x2-3*x3))
5 }
6 result <- integrate(function(x3) {sapply(x3,
      function(x3){
7       integrate(function(x2) {sapply(x2,function(x2){
8         integrate(function(x1) {
9           fun(x1, x2, x3)
10          }, 0, 1-x2)$value
11        })), 0, 1)$value
12      })), 1, Inf)$value
13 print(round(result,3))

```

R code Exa 3.28 Construct a frequency distribution

```

1 #page number = 103
2
3 data <- c(4890, 4830, 5490, 4820, 5230, 4860, 5040,
4          5060, 4500, 5260,
5          4610, 5100, 4730, 5250, 5540, 4910, 4430,
6          4850, 5040, 5000,
7          4600, 4630, 5330, 5160, 4950, 4480, 5310,
8          4730, 4700, 4390,
9          4710, 5160, 4970, 4710, 4430, 4260, 4890,
10         5110, 5030, 4850,
11         4820, 4550, 4970, 4740, 4840, 4910, 5200,
12         4880, 5150, 4890,
13         4900, 4990, 4570, 4790, 4480, 5060, 4340,
14         4830, 4670, 4750)
15
16 class_intervals <- seq(4200, 5600, by = 200)
17
18 binned_data <- cut(data, breaks = class_intervals,
19                    right = FALSE)
20

```

```

14 frequency_table <- table(t(binned_data))
15 for (i in 1:length(frequency_table))
16 print(frequency_table[[i]])
17 print(sum(frequency_table))

```

R code Exa 3.29 frequency distribution

```

1 #page number 104
2 # Given data
3 data <- c(5771, 5839, 5840, 5864, 5880, 5890, 5892,
           5902, 5908, 5912, 5914, 5918, 5924, 5926, 5928,
           5932, 5933, 5934, 5936, 5938, 5942, 5944, 5946,
           5948, 5950, 5952, 5954, 5956, 5958, 5960)
4
5 upper_limit= 4400
6 lower_limit= 4200
7 class_interval= upper_limit-lower_limit
8 print(class_interval)

```

R code Exa 3.30 a histogram

```

1 #page number= 104
2 compressive_strengths <- c(19.8, 13.9, 30.4, 16.4,
   11.6, 36.9, 14.8, 21.1, 13.5, 5.8,
3   10.0, 17.1, 14.1, 16.6,
   23.3, 12.1, 18.8,
   10.4, 9.4, 23.8,
4   14.2, 26.7, 7.8, 22.9,
   12.6, 6.8, 13.5, 10.7,
   12.2, 27.7,
5   9.0, 14.9, 24.0, 12.0,
   7.1, 12.8, 18.6, 26.0,
   37.4, 13.3)

```

```
6
7
8 hist(compressive_strengths,
9       xlab = "Histogram of solder-bond strengths",
10      ylab = "Frequency",
11      border = "black",
12      breaks = 10) # Adjust the number of breaks as
                    needed
```

Chapter 4

Mathematical Expectation

R code Exa 4.1 expect

```
1 #page number= 114
2 library(MASS)
3 N <- 12
4 W <- 2
5 k <- 3
6 probs <- dhyper(0:2, W, N - W, k)
7 expected_value <- sum(0:2 * probs)
8 print(fractions(expected_value))
```

R code Exa 4.2 expected value

```
1 # page number= 115
2 fun <- function(x) {
3   return (4*x / (pi * (1 + x^2)))
4 }
5 result <- integrate(fun, 0, 1)
6 print(round(result$value,4))
```

R code Exa 4.3 expected value

```
1 # page number=116
2 library(MASS)
3 fun <- function(x) {return ((2 * x * x + 1) / 6)}
4 sum <- sum(sapply(1:6, fun))
5 print(fractions(sum))
```

R code Exa 4.4 expected value

```
1 # page number = 116
2 fun= function(x){
3   return (exp(3*x/4)*exp(-x))
4 }
5 result= integrate(fun, lower= 0, upper=100)
6 fractions(result$value)
```

R code Exa 4.5 expected value

```
1 # page number= 118
2 library(MASS)
3 e_x_square=91/6
4 sum=2*e_x_square+1
5 print(fractions(sum))
```

R code Exa 4.6 expected value

```

1 # page number = 118
2 library(MASS)
3 fun <- function(r) {return (2 / ((r + 2) * (r + 1)))
  }
4 sum <- sum(4 * fun(c(2, 1))) + 1
5 print(fractions(sum))

```

R code Exa 4.8 expected value

```

1 # page numeber = 120
2 library(MASS)
3 fun <- function(x, y) {return (x + y)}
4 f <- c(1/6, 2/9, 1/36, 1/3, 1/6, 1/12)
5 xy <- matrix(c(0, 0, 0, 1, 0, 2, 1, 0, 1, 1, 2, 0),
  ncol = 2, byrow = TRUE)
6 sum <- sum(apply(xy, 1, function(v) fun(v[1], v[2]))
  * f)
7 print(fractions(sum))

```

R code Exa 4.9 expected value

```

1 # page number= 120
2 library(MASS)
3 fun <- function(x, y) {
4   return ((2*x/(7*y^3)) * (x + 2 * y))
5 }
6
7 result <- integrate(function(y) {sapply(y, function(
  y) {
8   integrate(function(x) {
9     fun(x, y)
10    }, 0, 1)$value
11  })), 1, 2)$value

```



```
12
13 print(fractions(result))
```

R code Exa 4.10 calculate the variance

```
1 #page number =124
2 library(MASS)
3 prob_each_dice=1/6
4 x=c(1,2,3,4,5,6)
5 expect_x=prob_each_dice*(sum(x))
6 expect_x_square=prob_each_dice*sum(x^2)
7 var=expect_x_square-(expect_x)^2
8 fractions(var)
```

R code Exa 4.11 standard deviation

```
1 # page number= 124
2
3 expect_x=0.4413
4 fun <- function(x) {
5   return ((4*x^2 / (pi * (1 + x^2))))
6 }
7 result <- integrate(fun, 0, 1)
8
9 result=round(result$value,4)
10 var=result-expect_x^2
11 var=round(var,4)
12 sd=sqrt(var)
13 print(round(sd,4))
```

R code Exa 4.12 Chebyshev theorem

```
1 # page number 127
2 library(stats)
3 library(MASS)
4
5 f <- function(x) {
6   ifelse(x > 0 & x < 1, 630 * x^4 * (1 - x)^4, 0)
7 }
8 mean_val <- integrate(function(x) x * f(x), 0, 1)$
  value
9 var_val <- integrate(function(x) (x - mean_val)^2 *
  f(x), 0, 1)$value
10 sd_val <- sqrt(var_val)
11 lower_bound <- mean_val - 2 * sd_val
12 upper_bound <- mean_val + 2 * sd_val
13 prob_within_interval <- integrate(f, lower_bound,
  upper_bound)$value
14 chebyshev_bound <- 1 - 1 / (2^2)
15 sprintf(paste(round(prob_within_interval,2)))
```

R code Exa 4.14 moment generating function

```
1 # page number 129
2 library(MASS)
3 fun= function(x){
4   return ((3*(1+exp(x))^2*exp(x))/8)
5 }
6 fun1= function(x){
7   return ((3*(1+exp(x))*exp(2*x))/4)
8 }
9 moment1=fractions(fun(0))
10 moment2=fractions(fun1(0))+fractions(fun(0))
11 print(moment1)
12 print(moment2)
```

R code Exa 4.15 covariance of X and Y

```
1 # page number= 133
2 library(MASS)
3 p=matrix(c(1/6,1/3,1/12,2/9,1/6,0,1/36,0,0),nrow=3,
           ncol=3,byrow=TRUE)
4 fractions(colSums(p))
5 mu1_1=0
6 x_values=0:2
7 y_values=0:2
8 for (x in x_values) {
9   for (y in y_values) {
10     mu1_1 <- mu1_1 + x * y * p[x+1, y+1]
11   }
12 }
13 print(fractions(mu1_1))
14 mu_x=fractions(sum(c(0:2)*colSums(p)))
15 mu_y=fractions(sum(c(0:2)*rowSums(p)))
16 print(mu_x)
17 print(mu_y)
18 sigma_xy=mu1_1-mu_x*mu_y
19 print(sigma_xy)
```

R code Exa 4.16 covariance

```
1 # page number = 133
2 library(cubature)
3 library(MASS)
4 nested_integrate <- function(outer_fun, inner_fun,
                               lower_limit, upper_limit) {
5   adaptIntegrate(function(x) {
```

```

6      sapply(x, function(x_val) {
7          adaptIntegrate(function(y) {
8              inner_fun(x_val, y)
9          }, lowerLimit = lower_limit(y), upperLimit =
            upper_limit(x_val))$integral
10     })
11 }, lowerLimit = lower_limit, upperLimit = upper_
    limit)$integral
12 }
13 fun1 <- function(x, y) { 2 * y }
14 fun2 <- function(x, y) { 2 * x }
15 fun3 <- function(x, y) { 2 * x * y }
16 lower_limit <- function(x) 0
17 upper_limit <- function(x) 1 - x
18 result1 <- nested_integrate(fun1, fun1, 0, 1)
19 result2 <- nested_integrate(fun2, fun2, 0, 1)
20 result3 <- nested_integrate(fun3, fun3, 0, 1)
21 print(fractions(result1))
22 print(fractions(result2))
23 print(fractions(result3))
24 final <- result3 - result1 * result2
25 print(fractions(final))

```

R code Exa 4.17 covariance

```

1 # page number = 135
2 joint_dist <- matrix(c(1/6, 1/3, 1/6, 0, 0, 0, 1/6,
3     0, 1/6), nrow = 3, ncol = 3, byrow = TRUE)
4 sum <- sum(sapply(c(-1, 0, 1), function(i) {sum(
5     sapply(c(-1, 0, 1), function(j) {(i * j) * joint_
6     dist[i + 2, j + 2]}})))))
7 print(sum)

```

R code Exa 4.18 mean and the variance

```
1 # page number= 137
2 mu <- c(2, -3, 4)
3 var <- c(1, 5, 2)
4 cov <- c(-2, -1, 1)
5 expect <- sum(c(3, -1, 2) * mu)
6 variance <- sum(c(9, 1, 4) * var) + sum(c(-6, 12,
      -4) * cov)
7 print(expect)
8 print(variance)
```

R code Exa 4.19 covariance

```
1 # page number = 138
2 mu <- c(3, 5, 2)
3 var <- c(8, 12, 18)
4 cov <- c(1, -3, 2)
5 covariance <- sum(c(3, -4, -2) * var) + sum(c(11, 5,
      -6) * cov)
6 print(covariance)
```

R code Exa 4.20 conditional mean and the conditional variance

```
1 # page number = 139
2 cond_x=function(x){
3   return ((2/3)*x*(x+1))
4 }
5
6 cond_x_sq=function(x){
7   return ((2/3)*x*x*(x+1))
8 }
9
```

```

10 expect_x=integrate(cond_x,0,1)$value
11 expect_x_sq=integrate(cond_x_sq,0,1)$value
12 var= expect_x_sq-expect_x^2
13 print(fractions(var))

```

R code Exa 4.21 mean length and its standard deviation

```

1 # page number = 142
2 obs= c
    (11.8,12.1,12.5,11.7,11.9,12.0,12.2,11.5,11.9,12.2)

3 s=sum(obs)
4 mean= mean(obs)
5 var=round(var(obs),3)
6 print(var)
7 print(round(var^(1/2),2))
8 print(mean)

```

Chapter 5

Special Probability Distributions

R code Exa 5.1 binomial distribution

```
1 # page number= 147
2 x=5
3 n=12
4 thita=1/2
5 result=dbinom(x, size=n, prob=thita)
6 print(round(result,2))
```

R code Exa 5.2 binomial distribution

```
1 # page number = 147
2 x=7
3 n=10
4 thita=0.8
5 result=dbinom(x, size=n, prob=thita)
6 print(round(result,2))
```

R code Exa 5.3 negative binomial distribution

```
1 # page number = 154
2 size <- 3
3 prob <- 0.40
4 failures <- 7
5 probability <- dnbinom(failures, size = size, prob =
  prob)
6 print(round(probability,4))
```

R code Exa 5.4 negative binomial distribution

```
1 # page number = 154
2 x <- 10
3 k <- 3
4 theta <- 0.40
5 binom_prob <- dbinom(k, size = x, prob = theta)
6 b_star <- (k / x) * binom_prob
7 print(round(b_star,4))
```

R code Exa 5.5 geometric distribution

```
1 #page number = 155
2 p <- 0.75
3 k <- 4
4 probability <- dgeom(k - 1, prob = p)
5 print(round(probability,4))
```

R code Exa 5.6 hypergeometric distribution

```
1 # page number = 156
2 x <- 0
3 m <- 4
4 n <- 20
5 k <- 6
6 probability <- dhyper(x, m, n, k)
7 print(round(probability,4))
```

R code Exa 5.7 hyper and binomial

```
1 # page number = 158
2 x_hyper <- 2
3 m_hyper <- 80
4 n_hyper <- 40
5 k_hyper <- 5
6 probability_hyper <- dhyper(x_hyper, m_hyper, n_
  hyper, k_hyper)
7 print(round(probability_hyper,3))
8 x_binom <- 2
9 n_binom <- 5
10 theta_binom <- 2/3
11 probability_binom <- dbinom(x_binom, size = n_binom,
  prob = theta_binom)
12 print(round(probability_binom,3))
```

R code Exa 5.8 Poisson distribution

```
1 # page number = 160
2 n <- 150
3 theta <- 0.05
4 lambda <- n * theta
```

```

5 x_values <- 5:15
6 poisson_probs <- dpois(x_values, lambda)
7 binomial_probs <- dbinom(x_values, size = n, prob =
  theta)
8 errors <- poisson_probs - binomial_probs
9 max_error_index <- which.max(abs(errors))
10 x_max_error <- x_values[max_error_index]
11 max_error <- errors[max_error_index]
12 cat(round(max_error,4))
13 cat(x_max_error)

```

R code Exa 5.9 Poisson distribution

```

1 # page number = 160
2
3 n <- 400
4 theta <- 0.02
5 lambda <- n * theta
6 x_values <- 5
7 poisson_probs <- dpois(x_values, lambda)
8 print(round(poisson_probs,3))
9 # The answer may slightly vary due to rounding off
  values.

```

R code Exa 5.10 binomial probabilities

```

1 # page number = 160
2 n <- 10000
3 theta <- 0.00005
4 lambda <- n * theta
5 x_a <- 2
6 prob_a <- dpois(x_a, lambda)
7 cat(round(prob_a,4))

```

```
8 x_b <- c(0:2)
9 prob_b <- dpois(x_b, lambda)
10 print(sum(round(prob_b,4)))
```

R code Exa 5.11 Poisson distribution

```
1 # page number = 162
2 cumulative_probs <- c
  (0.6065,0.3033,0.0758,0.0126,0.0016,0.0002)
3 print(cumulative_probs[3])
4 print(sum(cumulative_probs[c(1:3)]))
```

R code Exa 5.12 Poisson distribution

```
1 # page number = 163
2 lambda <- 12
3 prob_less_than_9 <- ppois(8, lambda)
4 print(round(prob_less_than_9,4))
```

R code Exa 5.13 a Poisson distribution

```
1 # page number = 164
2 lambda <- 7.5
3 x <- 5
4 prob_at_most_5 <- ppois(x, lambda)
5 prob_at_least_6 <- 1 - prob_at_most_5
6 cat(round(prob_at_least_6,4))
```

R code Exa 5.14 MULTINOMIAL DISTRIBUTION

```
1 # page number = 166
2 n <- 8
3 k <- c(5, 2, 1)
4 p <- c(0.50, 0.30, 0.20)
5 probb_multinomial <- factorial(n) / prod(factorial(k)
      ) * prod(p^k)
6 cat(probb_multinomial)
```

R code Exa 5.15 MULTINOMIAL DISTRIBUTION

```
1 # page number = 168
2 N1 <- c(6,3,7,4)
3 N <- sum(N1)
4 n <- 12
5 k <- c(4, 1, 5, 2)
6 result <- prod(choose(N1, k)) / choose(N, n)
7 print(round(result, 4))
```

R code Exa 5.16 producer and consumer risks

```
1 # page number = 171
2 p_AQL <- 0.05
3 p_LTPD <- 0.20
4 L_AQL <- 0.7358
5 L_LTPD <- 0.0692
6 producer_risk <- 1 - L_AQL
7 consumer_risk <- L_LTPD
8 cat(producer_risk)
9 cat(consumer_risk)
10
```

```
11 p <- c(0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35,
        0.40, 0.45)
12 L_p <- c(0.7358, 0.3917, 0.1756, 0.0692, 0.0243,
        0.0076, 0.0021, 0.0005, 0.0001)
13
14 plot(p, L_p, type = "o", col = "blue", xlab = "p",
        ylab = "L(p)",
15        main = "Oc curve for example 16")
```

Chapter 6

Special Probability Densities

R code Exa 6.1 a Poisson distribution

```
1 # page number = 181
2 lambda <- 8.4
3 exponential_density <- function(x) {
4   lambda * exp(-lambda * x)
5 }
6 upper_limit <- 5 / 30
7 result <- integrate(exponential_density, lower = 0,
8   upper = upper_limit)
9 probability <- result$value
10 print(round(probability,2))
```

R code Exa 6.2 standard normal distribution

```
1 #page number = 188
2 prob_a <- pnorm(1.72)
3 prob_b <- pnorm(-0.88)
4 prob_c <- pnorm(1.75) - pnorm(1.30)
5 prob_d <- pnorm(0.45) - pnorm(-0.25)
```

```

6 cat("(a) P(Z < 1.72) =", round(prob_a,4), "\n")
7 cat("(b) P(Z < -0.88) =", round(prob_b,4), "\n")
8 cat("(c) P(1.30 < Z < 1.75) =", round(prob_c,4), "\n")
9 cat("(d) P(-0.25 < Z < 0.45) =", round(prob_d,4), "\n")

```

R code Exa 6.4 cosmic radiation

```

1 # page number = 190
2 mean_radiation <- 4.35
3 sd_radiation <- 0.59
4 threshold <- 5.20
5 z_score <- (threshold - mean_radiation) / sd_radiation
6 prob_more_than_threshold <- 1 - pnorm(z_score)
7 cat(round(prob_more_than_threshold,4))
8 # The answer may slightly vary due to rounding off values.

```

R code Exa 6.5 chisquare distribution

```

1 # page number = 191
2 df <- 25
3 threshold <- 30
4 prob_chi_square <- 1 - pchisq(threshold, df)
5 cat(round(prob_chi_square,4))
6
7 mean_normal <- 18.7
8 sd_normal <- 9.1
9 lower <- 10.6
10 upper <- 24.8

```

```

11 prob_normal_lower <- pnorm(lower, mean_normal, sd_
    normal)
12 prob_normal_upper <- pnorm(upper, mean_normal, sd_
    normal)
13 prob_normal_between <- prob_normal_upper - prob_
    normal_lower
14 cat(round(prob_normal_between,4))

```

R code Exa 6.6 binomial distribution

```

1 # page number = 193
2 n <- 16
3 p <- 0.5
4 k <- 6
5 mean_binomial <- n * p
6 sd_binomial <- sqrt(n * p * (1 - p))
7 z_lower <- (k-(1/2) - mean_binomial) / sd_binomial
8 z_upper <- ((k + (1/2)) - mean_binomial) / sd_
    binomial
9 prob_normal_approx <- pnorm(z_upper) - pnorm(z_lower
    )
10 cat(round(prob_normal_approx,4))

```

R code Exa 6.7 normal scores

```

1 # page number = 199
2
3 observations <- c(3, 2, 7, 4, 3, 5)
4 ordered_observations <- sort(observations)
5 n <- length(ordered_observations)
6 normal_scores <- round(qnorm((1:n) / (n + 1)),2)
7 data <- data.frame(Ordered_Observations = ordered_
    observations, Normal_Scores = normal_scores)

```



```
8 print(data)
9 # The answer may slightly vary due to rounding off
  values.
```

R code Exa 6.8 normality

```
1 # page number = 200
2 original_data <- c(54.9, 8.3, 5.2, 32.4, 15.5)
3 print(original_data)
4 normal_scores <- c(-0.95, -0.44, 0, 0.44, 0.95)
5 ordered_original_data <- sort(original_data)
6 plot(normal_scores, ordered_original_data, xlab = "
   Normal Scores", ylab = "Ordered Observations",
   main = "Normal-Scores Plot for Original Data")
7 abline(lm(ordered_original_data ~ normal_scores),
   col = "red")
8 shapiro_test_original <- shapiro.test(original_data)
9 shapiro_test_original
10 transformed_data <- log10(original_data)
11 ordered_transformed_data <- sort(transformed_data)
12 plot(normal_scores, ordered_transformed_data, xlab =
   "Normal Scores", ylab = "Ordered Log-Transformed
   Observations", main = "Normal-Scores Plot for
   Transformed Data")
13 abline(lm(ordered_transformed_data ~ normal_scores),
   col = "red")
14 shapiro_test_transformed <- shapiro.test(transformed
   _data)
15 shapiro_test_transformed
```

Chapter 7

Functions of Random Variables

R code Exa 7.4 probability distribution

```
1 # page number = 211
2 library(MASS)
3 n <- 4
4 p <- 0.5
5 x_values <- 0:n
6 x_probabilities <- dbinom(x_values, size = n, prob =
  p)
7 x_probabilities=fractions(x_probabilities)
8 y_values <- 1 / (1 + x_values)
9 probability_distribution <- data.frame(
10   X = fractions(x_values),
11   P_X = fractions(x_probabilities),
12   Y = y_values
13 )
14 print((probability_distribution))
15 #- The answer may vary due to difference in
  representation
```

R code Exa 7.5 probability distribution

```

1 # page number = 212
2 n <- 4
3 p <- 0.5
4 x_values <- 0:n
5 x_probabilities <- dbinom(x_values, size = n, prob =
  p)
6 z_values <- (x_values - 2)^2
7 z_probabilities <- rep(0, max(z_values) + 1)
8 for (i in seq_along(z_values)) {
9   z_probabilities[z_values[i] + 1] <- z_
    probabilities[z_values[i] + 1] + x_
    probabilities[i]
10 }
11 probability_distribution_Z <- data.frame(
12   Z = seq_along(z_probabilities) - 1,
13   P_Z = z_probabilities
14 )
15 print(probability_distribution_Z)

```

R code Exa 7.11 probability density

```

1 # page number = 219
2 fun = function(u){
3   return ((u)*exp(-u))
4 }
5 result=integrate(fun,0,Inf )$value
6 print(result)

```

R code Exa 7.12 marginal density

```

1 # page number = 221
2 fun = function(y1){
3   return ((y1)*exp(-y1))

```

```

4 }
5
6 result=integrate(fun,0,Inf )$value
7 print(result)

```

R code Exa 7.19 probability

```

1 # page number = 230
2 lambda <- 5
3 n <- 2
4 gamma_cdf <- function(x, alpha, beta) {
5   pgamma(x, shape = alpha, rate = beta)
6 }
7 prob_T_leq_3 <- gamma_cdf(3, alpha = n, beta = 1/
   lambda)
8
9
10 cat("P(T <= 3) =", prob_T_leq_3, "\n")
11 #- The answer may vary due to difference in
   representation

```

Chapter 8

Sampling Distributions

R code Exa 8.1 central limit theorem

```
1 #page number = 238
2 mean_dispensed <- 200
3 sd_dispensed <- 15
4 sample_size <- 36
5 sample_mean_threshold <- 204
6 sd_sample_mean <- sd_dispensed / sqrt(sample_size)
7 z_score <- (sample_mean_threshold - mean_dispensed)
  / sd_sample_mean
8 cumulative_prob <- pnorm(z_score)
9 prob_at_least_threshold <- 1 - cumulative_prob
10 cat(round(prob_at_least_threshold,4))
```

R code Exa 8.2 Chisquare distribution

```
1 # page number = 245
2 n <- 20
3 sigma <- 0.60
4 s <- 0.84
```

```

5 alpha <- 0.01
6 test_statistic <- (n - 1) * (s^2) / (sigma^2)
7 cat(test_statistic)
8 critical_value <- qchisq(1 - alpha, df = n - 1)
9 cat(round(critical_value,3))
10 if (test_statistic > critical_value) {
11   cat("The process is declared out of control.\n")
12 } else {
13   cat("The process is in control.\n")
14 }

```

R code Exa 8.3 t distribution

```

1 # page number = 248
2 n <- 16
3 mu_hypothesis <- 12.0
4 x_bar <- 16.4
5 s <- 2.1
6 t_statistic <- (x_bar - mu_hypothesis) / (s / sqrt(n
  ))
7 df <- n - 1
8 p_value <- 1 - pt(t_statistic, df)
9 cat(round(t_statistic,2))
10 cat(df)
11 cat(p_value)
12 alpha <- 0.05
13 if (p_value < alpha) {
14   cat("Reject the null hypothesis\n")
15 } else {
16   cat("Fail to reject the null hypothesis: There is
      not enough evidence to conclude that the true
      average hourly gasoline consumption exceeds
      12.0 gallons.\n")
17 }

```

Chapter 9

Decision Theory

R code Exa 9.1 deciding

```
1 # page nummber =261
2 probb_recession <- 2 / 3
3 probb_good <- 1 / 3
4
5
6 profit_expand_good <- 164000
7 loss_expand_recession <- -40000
8 profit_wait_good <- 80000
9 profit_wait_recession <- 8000
10 expected_profit_expand <- (profit_expand_good * probb
    _good) + (loss_expand_recession * probb_recession)
11 expected_profit_wait <- (profit_wait_good * probb_
    good) + (profit_wait_recession * probb_recession)
12 cat(expected_profit_expand)
13 cat(expected_profit_wait)
14 if (expected_profit_expand > expected_profit_wait) {
15   cat("Decision: Expand now\n")
16 } else {
17   cat("Decision: Wait at least another year\n")
18 }
```

R code Exa 9.2 decide

```
1 # page number = 262
2 profit_expand_good <- 164000
3 loss_expand_recession <- -40000
4 profit_wait_good <- 80000
5 profit_wait_recession <- 8000
6
7 min_profit_expand <- loss_expand_recession
8 min_profit_wait <- profit_wait_recession
9
10 cat(min_profit_expand, "\n")
11 cat(min_profit_wait, "\n")
12
13 if (min_profit_expand > min_profit_wait) {
14   cat("Decision: Expand now\n")
15 } else {
16   cat("Decision: Wait another year\n")
17 }
```

R code Exa 9.3 PAYOFF MATRIX

```
1 # page number = 264
2 payoff_matrix <- matrix(c(7, -4, 8, 10), nrow = 2,
   byrow = TRUE)
3 rownames(payoff_matrix) <- c("A_I", "A_II")
4 colnames(payoff_matrix) <- c("B_1", "B_2")
5 print(payoff_matrix)
6
7 reduced_matrix <- payoff_matrix[, "B_2", drop =
   FALSE]
```



```

8 optimal_strategy_A <- rownames(reduced_matrix)[which
  .min(reduced_matrix)]
9 value_of_game <- min(reduced_matrix)
10
11 cat("Player A's optimal strategy:", optimal_strategy
  _A, "\n")
12 cat("Player B's optimal strategy: B_2\n")
13 cat("Value of the game:", value_of_game, "\n")

```

R code Exa 9.4 optimum strategies

```

1 # page number = 264
2 payoff_matrix <- matrix(c(-4, 1, 7, 4, 3, 5), nrow =
  2, byrow = TRUE)
3 rownames(payoff_matrix) <- c("A_I", "A_II")
4 colnames(payoff_matrix) <- c("B_1", "B_2", "B_3")
5 reduced_matrix <- payoff_matrix[, c("B_1", "B_2")]
6 print("Reduced matrix after removing dominated
  strategies for Player A:")
7 print(reduced_matrix)
8 final_matrix <- reduced_matrix[, "B_2", drop = FALSE
  ]
9 print("Final matrix after removing dominated
  strategies for Player B:")
10 print(final_matrix)
11 optimal_strategy_A <- rownames(final_matrix)[which.
  min(final_matrix)]
12 value_of_game <- min(final_matrix)
13 cat("Player A's optimal strategy:", optimal_strategy
  _A, "\n")
14 cat("Player B's optimal strategy: B_2\n")
15 cat("Value of the game:", value_of_game, "\n")

```

R code Exa 9.5 minimax strategies

```
1  payoff_matrix <- matrix(c(8, -5, 2, 6), nrow = 2,
   byrow = TRUE)
2  rownames(payoff_matrix) <- c("A_I", "A_II")
3  colnames(payoff_matrix) <- c("B_1", "B_2")
4
5  max_losses_A <- apply(payoff_matrix, 1, max)
6  minimax_strategy_A <- which.min(max_losses_A)
7
8  max_gains_B <- apply(payoff_matrix, 2, max)
9  minimax_strategy_B <- which.min(max_gains_B)
10
11 cat("Player A's minimax strategy:", rownames(payoff_
   matrix)[minimax_strategy_A], "\n")
12 cat("Player B's minimax strategy:", colnames(payoff_
   matrix)[minimax_strategy_B], "\n")
13
14 if (minimax_strategy_A == 2 && minimax_strategy_B ==
   2) {
15   cat("Player A switches to Strategy I, reducing
   loss to 2.\n")
16   cat("Player B switches to Strategy 1, increasing
   gain to 8.\n")
17 }
```

Chapter 10

Point Estimation

R code Exa 10.19 binomial trials

```
1 # page number = 308
2 x <- 42
3 n <- 120
4 alpha_prior <- 40
5 beta_prior <- 40
6 alpha_posterior <- alpha_prior + x
7 beta_posterior <- beta_prior + (n - x)
8 posterior_mean <- alpha_posterior / (alpha_posterior
  + beta_posterior)
9 cat(posterior_mean)
10 thita= x/n
11 print(thita)
```

R code Exa 10.20 soft drink vending machines

```
1 # page number = 309
2 mu1 <- 715
3 sigma1 <- 9.5
```

```

4 lower_bound <- 700
5 upper_bound <- 720
6 z_low <- (lower_bound - mu1) / sigma1
7 z_up <- (upper_bound - mu1) / sigma1
8 p_low <- pnorm(z_low)
9 p_up <- pnorm(z_up)
10 probability <- p_up - p_low
11 cat(mu1)
12 cat(sigma1)
13 cat(round(z_low,2))
14 cat(round(z_up,2))
15 cat(round(probability,3))

```

R code Exa 10.21 voters

```

1 # page number = 311
2 z_alpa_by_2=1.96
3 E=0.03
4
5 n= z_alpa_by_2^2/(4*E^2)
6 print(ceiling(n))

```

Chapter 11

Interval Estimation

R code Exa 11.1 efficiency experts

```
1 # page number= 318
2
3 n <- 150
4 sigma <- 6.2
5 z_value <- 2.575
6
7 max_error <- z_value * sigma / sqrt(n)
8
9 cat(round(max_error,2))
```

R code Exa 11.2 confidence interval

```
1 # page number= 319
2 n <- 20
3 x_bar <- 64.3
4 sigma <- 15
5 z_value <- 1.96
6 margin_of_error <- z_value * sigma / sqrt(n)
```

```

7
8 lower_bound <- x_bar - margin_of_error
9 upper_bound <- x_bar + margin_of_error
10
11 cat(round(lower_bound,1), "<      <",round(upper_bound
    ,1))

```

R code Exa 11.3 confidence interval

```

1 # page number = 320
2 data <- c(17, 13, 18, 19, 17, 21, 29, 22, 16, 28,
    21, 15,
3           26, 23, 24, 20, 8, 17, 17, 21, 32, 18, 25,
    22,
4           16, 10, 20, 22, 19, 14, 30, 22, 12, 24,
    28, 11)
5 conf_interval <- t.test(data)$conf.int
6 cat(round(conf_interval[1], 2))
7 cat(round(conf_interval[2], 2))
8
9 #— The answer may slightly vary due to rounding off
    values.

```

R code Exa 11.4 paint manufacturer

```

1 # page number = 321
2 x_bar <- 66.3
3 s <- 8.4
4 n <- 12
5 t_value <- 2.201
6 margin_of_error <- t_value * s / sqrt(n)
7 lower_bound <- x_bar - margin_of_error
8 upper_bound <- x_bar + margin_of_error

```

```
9 cat(round(lower_bound,1))
10 cat(round(upper_bound,1))
```

R code Exa 11.5 confidence interval

```
1 # page number = 322
2 x1_bar <- 418
3 x2_bar <- 402
4 n1 <- 40
5 n2 <- 50
6 sigma1 <- 26
7 sigma2 <- 22
8 z_value <- 1.88
9 se_diff <- sqrt((sigma1^2 / n1) + (sigma2^2 / n2))
10 margin_of_error <- z_value * se_diff
11 difference_means <- x1_bar - x2_bar
12 lower_bound <- difference_means - margin_of_error
13 upper_bound <- difference_means + margin_of_error
14 cat(round(lower_bound,1))
15 cat(round(upper_bound,1))
```

R code Exa 11.6 two brands of cigarettes

```
1 # page number= 324
2 n1 <- 10
3 n2 <- 8
4 x1_bar <- 3.1
5 x2_bar <- 2.7
6 s1 <- 0.5
7 s2 <- 0.7
8 t_value <- 2.120
9
10 se_diff <- sqrt(s1^2 / n1 + s2^2 / n2)
```

```

11
12 margin_of_error <- t_value * se_diff
13
14 cat(round(x1_bar - x2_bar - margin_of_error, 1), "\n
    ", round(x1_bar - x2_bar + margin_of_error, 1))

```

R code Exa 11.7 a flu vaccine experienced

```

1 # page number = 326
2 n <- 400
3 x <- 136
4 confidence_level <- 0.95
5
6 binom_result <- binom.test(x, n, conf.level =
    confidence_level)
7
8 cat( round(binom_result$conf.int[1], 2), "\n")
9 cat(round(binom_result$conf.int[2], 2))

```

R code Exa 11.8 voters

```

1 # page number = 327
2 n <- 400
3 x <- 140
4 confidence_level <- 0.99
5 binom_result <- binom.test(x, n, conf.level =
    confidence_level)
6 cat(round(binom_result$conf.int[2] - p_hat, 2))

```

R code Exa 11.9 running for governor of Illinois


```

1 # page number = 328
2
3 n1 <- 200
4 n2 <- 150
5 x1 <- 132
6 x2 <- 90
7 confidence_level <- 0.99
8 prop_result <- prop.test(c(x1, x2), c(n1, n2), conf.
   level = confidence_level)
9 cat("Confidence interval for (p1 - p2):", round(prop
   _result$conf.int[1], 3), "to", round(prop_result$
   conf.int[2], 3))
10 # The answer may slightly vary due to rounding off
   values.

```

R code Exa 11.10 gasoline consumption

```

1 # page number = 330
2 n <- 16
3 s <- 2.2
4 chi_sq_low <- 4.601
5 chi_sq_high <- 32.801
6 lower_bound <- (n - 1) * (s^2) / chi_sq_high
7 upper_bound <- (n - 1) * (s^2) / chi_sq_low
8 cat(round(lower_bound, 2), "< 2 < ", round(upper_
   bound, 2))

```

R code Exa 11.11 confidence interval

```

1 # page number = 331
2 n1 <- 10
3 n2 <- 8
4 s1 <- 0.5

```

```

5 s2 <- 0.7
6 f0.01_9_7 <- 6.72
7 f0.01_7_9 <- 5.61
8 lower_bound <- (s1^2 / s2^2) / f0.01_9_7
9 upper_bound <- (s1^2 / s2^2) * f0.01_7_9
10 cat(round(lower_bound,3), " < 2 ^2/ 2 ^2< ", round(
    upper_bound,3))

```

R code Exa 11.12 car crossings

```

1 # page number = 332
2 car_crossings <- c(142600, 167800, 136500, 108300,
    126400, 133700, 162000, 149400)
3 sample_mean <- mean(car_crossings)
4 sample_sd <- sd(car_crossings)
5 n <- length(car_crossings)
6 confidence_level <- 0.95
7 t_critical <- qt((1 + confidence_level) / 2, df = n
    - 1)
8 standard_error <- sample_sd / sqrt(n)
9 margin_of_error <- t_critical * standard_error
10 lower_bound <- sample_mean - margin_of_error
11 upper_bound <- sample_mean + margin_of_error
12 cat((lower_bound), ", ", upper_bound)
13 #— The answer provided in the textbook is wrong.

```

Chapter 12

Hypothesis Testing

R code Exa 12.1 new medication

```
1 # page number = 339
2 n <- 20
3 theta_0 <- 0.90
4 theta_1 <- 0.60
5 critical_value <- 14
6 alpha <- pbinom(critical_value, n, theta_0, lower.
  tail = TRUE)
7 beta <- pbinom(critical_value, n, theta_1, lower.
  tail = FALSE)
8 cat(round(alpha,4))
9 cat(round(beta,4))
10 #- The answer may slightly vary due to rounding off
  values.
```

R code Exa 12.7 likelihood ratio technique

```
1 # page number= 353
2 library(MASS)
```

```

3 f_x <- c(1/12, 1/12, 1/12, 1/4, 1/6, 1/6, 1/6)
4 g_x <- c(1/3, 1/3, 1/3, 2/3, 0, 0, 0)
5 x_values <- 1:7
6 lambda <- ifelse(g_x != 0, f_x / g_x, 1)
7 cat("Lambda values for each x:\n")
8 data.frame(x = x_values, lambda = lambda)
9 critical_region_lrt <- x_values[lambda == 1/4]
10 alpha_lrt <- sum(f_x[critical_region_lrt])
11 beta_lrt <- sum(g_x[c(4,5,6)])
12 print(fractions(beta_lrt))
13 critical_region_alt <- c(4)
14 cat(critical_region_alt)
15 alpha_alt <- sum(f_x[c(4)])
16 print(fractions(alpha_alt))
17 beta_alt <- sum(g_x[c(1,2,3)])
18 if(beta_lrt > alpha_lrt){
19   cat("null hypothesis is rejected")
20 }

```

Chapter 13

Tests of Hypothesis Involving Means Variances and Proportions

R code Exa 13.1 packages of cookies

```
1 # page number = 364
2 x_bar <- 8.091
3 mu_0 <- 8
4 sigma <- 0.16
5 n <- 25
6 alpha <- 0.01
7 z <- (x_bar - mu_0) / (sigma / sqrt(n))
8 z_critical_lower <- qnorm(alpha / 2)
9 z_critical_upper <- qnorm(1 - alpha / 2)
10 cat(round(z,2))
11 cat(round(z_critical_lower,3))
12 cat(round(z_critical_upper,3))
13 if (z < z_critical_lower || z > z_critical_upper) {
14   cat(" the null hypothesis must be rejected \n")
15 } else {
16   cat(" Fail to reject the null hypothesis\n")
17 }
```

R code Exa 13.2 high performance tires

```
1 # page number= 365
2 x_bar <- 21819
3 mu_0 <- 22000
4 sigma <- 1295
5 n <- 100
6 alpha <- 0.05
7 z <- (x_bar - mu_0) / (sigma / sqrt(n))
8 z_critical <- qnorm(alpha)
9 cat("Test Statistic (z):", round(z,1), "\n")
10 cat("Critical Value:", round(z_critical,3), "\n")
11 if (z < z_critical) {
12   cat("Reject the null hypothesis.")
13 } else {
14   cat("Fail to reject the null hypothesis.")
15 }
```

R code Exa 13.3 ribbon call

```
1 # page number= 365
2 sample_data <- c(171.6, 191.8, 178.3, 184.9, 189.1)
   # sample breaking strengths
3 mu_0 <- 185
4 alpha <- 0.05
5 sample_mean <- mean(sample_data)
6 sample_sd <- sd(sample_data)
7 n <- length(sample_data)
8 t_statistic <- (sample_mean - mu_0) / (sample_sd /
   sqrt(n))
9 critical_value <- qt(alpha, df = n - 1)
```

```

10 p_value <- pt(t_statistic, df = n - 1)
11 cat(round(t_statistic,2))
12 if (t_statistic < critical_value) {
13   cat("Reject the null hypothesis\n")
14 } else {
15   cat("the null hypothesis cannot be rejected\n")
16 }

```

R code Exa 13.4 nicotine

```

1 # page number = 367
2 x <- c(2.61,2.38)
3 s <- c(0.12,0.14)
4 n <- c(50,40)
5 delta <- 0.20
6 z <- (x[1] - x[2] - delta) / sqrt((s[1]^2 / n[1]) +
   (s[2]^2 / n[2]))
7 p_value <- 1 - pnorm(z)
8 cat(round(z,2))
9 alpha <- 0.05
10 if (p_value < alpha) {
11   cat("Reject the null hypothesis.\n")
12 } else {
13   cat("Fail to reject the null hypothesis.\n")
14 }

```

R code Exa 13.5 consumer testing service

```

1 # page number = 368
2 x <- c(546,492)
3 s <- c(31,26)
4 n <- c(4,4)
5

```

```

6 sp <- sqrt(((n[1]- 1) * s[1]^2 + (n[2] - 1) * s
  [2]^2) / (n[1] + n[2] - 2))
7 t <- (x[1] - x[2]) / (sp * sqrt(1/n[1] + 1/n[2]))
8 critical_value <- qt(0.95, df = n[1] + n[2] - 2)
9 cat(round(sp,3))
10 cat(round(t,2))
11 if (t > critical_value) {
12   "reject null hypothesis."
13 } else {
14   "Fail to reject null hypothesis"
15 }

```

R code Exa 13.6 a semiconductor

```

1 # page number = 370
2 s2 <- 0.68
3 sigma0 <- 0.36
4 n <- 18
5 chi_square <- (n - 1) * s2 / sigma0
6 print(round(chi_square,2))
7 critical_value <- qchisq(0.05, df = n - 1)
8 if (chi_square > critical_value) {
9   "Reject null hypothesis"
10 } else {
11   "Fail to reject null hypothesis"
12 }

```

R code Exa 13.7 structural steel

```

1 # page numebr= 371
2 s_squared <- c(19.2,3.5)
3 alpha <- 0.02
4 F_statistic <- s_squared[1] / s_squared[2]

```



```

5 critical_value <- qf(1 - alpha, df1 = 12, df2 = 15)
6 cat(round(F_statistic, 2))
7 if (F_statistic > critical_value) {
8   "Reject null hypothesis"
9 } else {
10  "Fail to Reject null hypothesis"
11 }

```

R code Exa 13.8 patients suffering

```

1 # page number = 373
2 x <- 4
3 n <- 20
4 alpha <- 0.05
5 theta_hat <- x / n
6 p_value <- 2*pbinom(x, size = n, prob = 0.5, lower.
   tail = TRUE)
7 cat("P-value:", round(p_value,4), "\n")
8 if (p_value < alpha) {
9   cat("Reject the null hypothesis")
10 } else {
11   cat("Fail to reject the null hypothesis")
12 }

```

R code Exa 13.9 oil company claims

```

1 # page number = 374
2 x <- 22
3 n <- 200
4 theta0 <- 0.20
5 alpha <- 0.01
6 theta_hat <- x / n

```

```

7 z <- (x - n * theta0) / sqrt(n * theta0 * (1 -
  theta0))
8 cat(round(z,2))
9 p_value <- pnorm(z, lower.tail = TRUE)
10 if (p_value < alpha) {
11   cat("Reject the null hypothesis.\n")
12 } else {
13   cat("Fail to reject the null hypothesis.\n")
14 }

```

R code Exa 13.10 shoppers favoring detergent

```

1 # page number= 376
2 observed <- matrix(c(232, 168, 260, 240, 197, 203),
  nrow = 3, byrow = TRUE,
3
4
5   dimnames = list(c("Los Angeles",
6                     "San Diego", "Fresno"),
7                     c("Detergent A",
8                       "Detergent B")
9   ))
10 chi_square_test <- chisq.test(observed)
11 cat(round(chi_square_test$statistic, 2), "\n")
12 cat(round(qchisq(0.95, chi_square_test$parameter),
13   3), "\n")
14
15 if (chi_square_test$p.value < 0.05) {
16   cat("Reject the null hypothesis\n")
17 } else {
18   cat("Fail to reject the null hypothesis\n")
19 }
20 # The answer may slightly vary due to rounding off
21 values.

```

R code Exa 13.11 persons ability

```
1 # page numeber= 380
2 observed <- matrix(c(63, 42, 15, 58, 61, 31, 14, 47,
3                     29), nrow = 3, byrow = TRUE,
4                     dimnames = list(c("Low", "Average",
5                                       "High"),
6                                       c("Low", "Average",
7                                       "High"))))
5 chi_square <- chisq.test(observed)$statistic
6 df <- chisq.test(observed)$parameter
7 critical_value <- qchisq(0.99, df)
8 cat(round(chi_square,2))
9 if (chi_square > critical_value) {
10   cat("Reject the null hypothesis")
11 } else {
12   cat("Fail to reject the null hypothesis")
13 }
```

R code Exa 13.12 a Poisson distribution

```
1 # page number = 382
2 observed <- c(18, 53, 103, 107, 82, 46, 18, 10, 3)
3 expected <- c(21.9, 65.7, 98.6, 98.6, 73.9, 44.4,
4              22.2, 9.5, 5.3)
5 chi_square <- sum((observed - expected)^2 / expected)
6 df <- length(observed) - 2
7 critical_value <- qchisq(0.95, df)
8 cat(round(chi_square,2))
9 if (chi_square > critical_value) {
10   cat("Reject the null hypothesis. \n")
11 } else {
12   cat(" the null hypothesis cannot be rejected.\n")
13 }
```

R code Exa 13.13 measurements of the heat producing capacity

```
1 # page number = 382
2 mine1 <- c(8400, 8230, 8380, 7860, 7930)
3 mine2 <- c(7510, 7690, 7720, 8070, 7660)
4 t_test_result <- t.test(mine1, mine2, var.equal =
  TRUE)
5 print(t_test_result)
```

Chapter 14

Regression and Correlation

R code Exa 14.4 equation of the least squares line

```
1 # page number = 400
2 x <- c(4, 9, 10, 14, 4, 7, 12, 22, 1, 17)
3 y <- c(31, 58, 65, 73, 37, 44, 60, 91, 21, 84)
4 model <- lm(y ~ x)
5 alpha_hat <- coef(model)[1]
6 beta_hat <- coef(model)[2]
7 cat("y =", round(alpha_hat, 2), "+", round(beta_hat,
      3), "x\n")
8 x_new <- 14
9 y_hat <- predict(model, newdata = data.frame(x = x_
      new))
10 cat(floor(y_hat))
```

R code Exa 14.5 amount of time that 10 persons studied

```
1 # page number = 404
2
3 x <- c(4, 9, 10, 14, 4, 7, 12, 22, 1, 17)
```

```

4 y <- c(31, 58, 65, 73, 37, 44, 60, 91, 21, 84)
5 model <- lm(y ~ x)
6 summary_model <- summary(model)
7 beta_hat <- coef(model)[2]
8 se_beta_hat <- summary_model$coefficients[2, "Std.
  Error"]
9 beta_null <- 3
10 t_stat <- (beta_hat - beta_null) / se_beta_hat
11 alpha <- 0.01
12 df <- summary_model$df[2]
13 t_critical <- qt(1 - alpha, df)
14 cat(round(t_stat, 2), "\n")
15 if (t_stat > t_critical) {
16   cat("Reject the null hypothesis")
17 } else {
18   cat("Fail to reject the null hypothesis")
19 }

```

R code Exa 14.6 confidence interval

```

1 # page number = 405
2 x <- c(4, 9, 10, 14, 4, 7, 12, 22, 1, 17)
3 y <- c(31, 58, 65, 73, 37, 44, 60, 91, 21, 84)
4 model <- lm(y ~ x)
5 conf_interval <- confint(model, level = 0.95)
6 lhs <- conf_interval["x", "2.5 %"]
7 rhs <- conf_interval["x", "97.5 %"]
8 cat(round(lhs, 2), "<   <", round(rhs, 2))

```

R code Exa 14.7 complete a certain form

```

1 # page number = 410
2

```

```

3 x <- c(8.2, 9.6, 7.0, 9.4, 10.9, 7.1, 9.0, 6.6, 8.4,
        10.5)
4 y <- c(8.7, 9.6, 6.9, 8.5, 11.3, 7.6, 9.2, 6.3, 8.4,
        12.3)
5 r <- cor(x, y)
6 cat(round(r, 3))
7 plot(x, y, xlim = c(0, 14), ylim = c(0, 14), xlab =
    "Morning", ylab = "Afternoon")
8 abline(a = 0, b = 1) # Adds a line y = x to the
    plot

```

R code Exa 14.8 complete a certain form

```

1 # page number = 411
2 n=10
3 r=0.936
4 critical= 2.575
5 z=sqrt((n-3))/2*log((r+1)/(1-r))
6 cat(round(z,1))
7 if(z>critical)
8 {
9     cat("null hypothesis rejected")
10 }

```

R code Exa 14.9 onefamily houses sold

```

1 # page number = 414
2 bedrooms <- c(3, 2, 4, 2, 3, 2, 5, 4)
3 baths <- c(2, 1, 3, 1, 2, 2, 3, 2)
4 price <- c(292000, 264600, 317500, 265500, 302000,
            275500, 333000, 307500)
5 housing_data <- data.frame(bedrooms, baths, price)

```

```

6 model <- lm(price ~ bedrooms + baths, data = housing
  _data)
7 coefficients <- coef(model)
8 intercept <- coefficients[1]
9 bedrooms_coef <- coefficients[2]
10 baths_coef <- coefficients[3]
11 cat("y^cap =", round(intercept,0), "+", round(
  bedrooms_coef,0), " x1 +", round(baths_coef,0), "
  x2\n")

```

R code Exa 14.10 sales price of a threebedroom

```

1 # page number= 414
2 intercept= 224929
3 bedroom_coef=15314
4 bathroom_coef= 10957
5 x1=3
6 x2=2
7 result= intercept+bedrooms_coef*x1+bathroom_coef*x2
8 cat(floor(result))

```

R code Exa 14.11 least squares estimates

```

1 # page number = 417
2 bedrooms <- c(3, 2, 4, 2, 3, 2, 5, 4)
3 baths <- c(2, 1, 3, 1, 2, 2, 3, 2)
4 price <- c(292000, 264600, 317500, 265500, 302000,
  275500, 333000, 307500)
5 housing_data <- data.frame(bedrooms, baths, price)
6 model <- lm(price ~ bedrooms + baths, data = housing
  _data)
7 coefficients <- coef(model)
8 intercept <- coefficients[1]

```



```

9 bedrooms_coef <- coefficients[2]
10 baths_coef <- coefficients[3]
11 cat("y^cap =", round(intercept,0), "+", round(
    bedrooms_coef,0), " x1 +", round(baths_coef,0), "
    x2\n")

```

R code Exa 14.12 least squares estimates

```

1 # page number = 414
2 y <- c(292000, 264600, 317500, 265500, 302000,
    275500, 333000, 307500)
3 Y= sum(y^2)
4 x=matrix(c(224929,15314,10957),nrow=1)
5 y=matrix(c(637000,7558200,4835600),ncol=1)
6 dim(y)
7 result= x%*%y
8 sigma=sqrt((Y-result)/8)
9 cat(round(sigma,0))
10 # The answer provided in the textbook is wrong.

```

R code Exa 14.13 least squares estimates using hypothesis

```

1 # page number =419
2 critical= 2.015
3 bita_cap= 15314
4 bita=9500
5 c11=32/84
6 n=8
7 sigma= 3546
8 t=(bita_cap-bita)/(sigma*sqrt(n*c11/5))
9 cat(round(t,3))
10 if(critical<t)
11 {

```

```
12  cat("null hypothesis must be rejected")
13  }
14  # The answer provided in the textbook is wrong.
```
