

R Textbook Companion for
A First Course in Probability
by Sheldon Ross¹

Created by
Anuj Singh
B.Tech.

Computer Science and Engineering
Indian Institute of Technology, Bhilai

Cross-Checked by
R TBC Team

May 26, 2020

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

Book Description

Title: A First Course in Probability

Author: Sheldon Ross

Publisher: Pearson, USA

Edition: 8

Year: 2008

ISBN: 013603313X

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 Combinatorial Analysis	5
2 Axioms of Probability	14
3 Conditional Probability and Independence	22
4 Random Variables	33
5 Continuous Random Variables	44
6 Jointly Distributed Random Variables	55
7 Properties of Expectation	63
8 Limit Theorems	68

List of R Codes

Exa 2.a	Mother and child of the year	5
Exa 2.b	Committee selection	5
Exa 2.c	Licence plates	6
Exa 2.e	Licence plates without repetitions	6
Exa 3.a	Batting orders	7
Exa 3.b.a	Universal rankings	7
Exa 3.b.b	Gender rankings	7
Exa 3.c	Book arrangements	8
Exa 3.d	Pepper permuatations	8
Exa 3.e	Chess tournament outcomes	9
Exa 3.f	Flag signals	9
Exa 4.a	Possible committees	10
Exa 4.b	Committee selection with genders and feuds	10
Exa 5.a	Possible divisions	11
Exa 5.b	Division into team A and team B	11
Exa 5.c	Basketball divisions	11
Exa 5.d	Tournaments	12
Exa 6.a	Distinct non-negative integral solutions	12
Exa 6.b	Investments	13
Exa 3.b	Even number on a dice roll	14
Exa 4.a	Books and vacations	14
Exa 5.a	Sum of dice rolls	15
Exa 5.b	Drawing balls from a bowl	15
Exa 5.c	Committee selection	16
Exa 5.f	Probability of a straight	16
Exa 5.g	Probability of full house	17
Exa 5.h.a	Bridge player gets all spades	17
Exa 5.h.b	Bridge each player gets 1 spade	17

Exa 5.i	Birthday problem	18
Exa 5.k	Football team	18
Exa 5.l	Club sports	19
Exa 5.n	Married couples	20
Exa 7.a	Horse race	20
Exa 2.a	Student exams	22
Exa 2.b.a	First coin flip heads	22
Exa 2.b.b	At least one flip heads	23
Exa 2.c	Bridge	23
Exa 2.e	Course selection	24
Exa 2.f.a	Drawing red balls from urn	24
Exa 2.h	Each pile contains 1 ace	25
Exa 3.d	Laboratory blood test	25
Exa 3.e	Surgery dilemma	25
Exa 3.f	Criminal investigation	26
Exa 3.i	Urn and coins	26
Exa 3.l	Coloured cards	27
Exa 3.n.a	Flashlight gives more than 100 hours of use	27
Exa 3.n.b	Type of flashlight	28
Exa 3.o	Crime probability	28
Exa 4.h	Independent die trials	30
Exa 5.a	Conditional probability of accident prone policy holder	31
Exa 3.a.1	Accident within a year of purchasing policy	31
Exa 3.a.2	Policy holder is accident prone	31
Exa 1.b	Urn with replacement	33
Exa 1.d	Winnings from urn experiment	33
Exa 10.a	Properties of random variables	34
Exa 3.a	Expectation of a die roll	35
Exa 3.d	Expectation of number of students	36
Exa 4.a	Expectation of a random variable squared	36
Exa 5.a	Variance of a die roll	37
Exa 6.a	Binomial random variable coin experiment	37
Exa 6.b	Package replacement	38
Exa 6.c	Wheel of fortune	38
Exa 6.d	Genetic traits	39
Exa 6.h	Computing binomial distribution function	39
Exa 6.i	Binomial distribution generation	40
Exa 7.a	Typos in a page	40

Exa 7.b	Defective item	40
Exa 7.c	Radioactive particles	41
Exa 7.e.a	Earthquake occourence	41
Exa 7.f	Generating poisson distribution	42
Exa 8.g	Throws of die required	42
Exa 8.i	Electrical components	43
Exa 1.a.a	Finding the constant in "fx"	44
Exa 1.a.b	Probability of a continuous random variable	44
Exa 1.b.a	Computer functioning "a"	45
Exa 1.b.b	Computer functioning "b"	45
Exa 1.c	Lifetime of radio tube	46
Exa 2.a	Expectation of a continuous random variable	46
Exa 2.e	Variance of a continuous random variable	47
Exa 3.b	Uniform distribution	47
Exa 3.c	Waiting time for bus	48
Exa 4.b	Normal random variable	49
Exa 4.c	Grading on the curve	49
Exa 4.d	Paternity suit	50
Exa 4.e	Signals	50
Exa 4.f	Normal approximation	51
Exa 4.g	Probability of attendance in a college	51
Exa 4.h	Effectiveness of diet	52
Exa 4.i	Outlawing cigarettis	52
Exa 5.b	Exponential random variable	53
Exa 5.e	Laplace distribution	53
Exa 1.a	Joint PMF of drawing balls from urn	55
Exa 1.b	Joint PMF of number of boys and girls in a family	56
Exa 1.c.a	Joint density function "a"	57
Exa 1.c.b	Joint density function "b"	57
Exa 2.c	Waiting time	58
Exa 3.c.a	Basketball wins "a"	59
Exa 3.c.b	Basketball wins "b"	59
Exa 3.d.a	Price of security "a"	60
Exa 3.d.b	Price of security "b"	60
Exa 4.a	Conditional probability on joint PMF	61
Exa 6.b	Probability of sample median	61
Exa 2.r	Chipmunks and groves	63
Exa 5.e.a	Game of carps "a"	63

Exa 5.e.b	Game of carps "b"	64
Exa 5.e.c	Game of carps "c"	65
Exa 2.b	Chebyshev inequality	68
Exa 3.a	Astronomical distances	68
Exa 3.b	Probability of a professor teaching two sections	69
Exa 3.c	Sum of dice rolls	69
Exa 3.d	Approximation of uniform random variables	70
Exa 3.e	Time for grading exams	71
Exa 5.a	Factory production	71
Exa 5.b	Pairs	71

Chapter 1

Combinatorial Analysis

R code Exa 2.a Mother and child of the year

```
1 # Page No. 2
2
3 no_of_women = 10
4 children_per_women = 3
5
6 no_of_choices = no_of_women * children_per_women
7
8 print(no_of_choices)
```

R code Exa 2.b Committee selection

```
1 # Page No. 2
2
3 no_of_freshmen = 3
4 no_of_sophmores = 4
5 no_of_juniors = 5
6 no_of_seniors = 2
7
```

```
8 ans = no_of_seniors * no_of_juniors * no_of_
    sophmores * no_of_freshmen
9
10 print(ans)
```

R code Exa 2.c Licence plates

```
1 # Page No. 3
2
3 no_of_letters = 26
4 no_of_digits = 10
5
6 ans = (no_of_digits^4) * (no_of_letters^3)
7
8 print(ans)
```

R code Exa 2.e Licence plates without repetitions

```
1 # Page No. 3
2
3 no_of_letters = 26
4 no_of_digits = 10
5 ans = 1
6
7 for(i in 0:3)
8 {
9     ans = ans * (no_of_digits - i);
10 }
11 for(i in 0:2)
12 {
13     ans = ans * (no_of_letters - i);
14 }
15
```

```
16 print(ans)
```

R code Exa 3.a Batting orders

```
1 # Page No. 3
2
3 no_of_players = 9
4
5 ans = factorial(no_of_players)
6
7 print(ans)
```

R code Exa 3.b.a Universal rankings

```
1 # Page No. 4
2
3 no_of_women = 4
4 no_of_men = 6
5
6 ans = factorial(no_of_men + no_of_women)
7
8 print(ans)
```

R code Exa 3.b.b Gender rankings

```
1 # Page No. 4
2
3 no_of_women = 4
4 no_of_men = 6
5
```

```
6 ans = factorial(no_of_men) * factorial(no_of_women)
7
8 print(ans)
```

R code Exa 3.c Book arrangements

```
1 # Page No. 4
2
3 no_of_math_books = 4
4 no_of_chem_books = 3
5 no_of_history_books = 2
6 no_of_lang_book = 1
7 no_of_types_of_books = 4
8
9 no_of_orderings = factorial(no_of_types_of_books)
10 ans = no_of_orderings * factorial(no_of_math_books)
    * factorial(no_of_chem_books) * factorial(no_of_
    history_books) * factorial(no_of_lang_book)
11
12 print(ans)
```

R code Exa 3.d Pepper permutations

```
1 # Page No. 4
2
3 no_of_p = 3
4 no_of_r = 1
5 no_of_e = 2
6
7 total_chars = no_of_p + no_of_r + no_of_e
8 ans = factorial(total_chars) / (factorial(no_of_p) *
    factorial(no_of_r) * factorial(no_of_e))
9
```

```
10 print(ans)
```

R code Exa 3.e Chess tournament outcomes

```
1 # Page No. 5
2
3 no_of_russians = 4
4 no_of_americans = 3
5 no_of_brits = 2
6 no_of_brazillians = 1
7
8 ans = factorial(no_of_brazillians + no_of_americans
  + no_of_brits + no_of_russians) / (factorial(no_
  of_russians) * factorial(no_of_brits) * factorial
  (no_of_americans) * factorial(no_of_brazillians))
9
10 print(ans)
```

R code Exa 3.f Flag signals

```
1 # Page No. 5
2
3 no_of_white_flags = 4
4 no_of_red_flags = 3
5 no_of_blue_flags = 2
6
7 ans = factorial(no_of_white_flags + no_of_red_flags
  + no_of_blue_flags) / (factorial(no_of_white_
  flags) * factorial(no_of_red_flags) * factorial(
  no_of_blue_flags))
8
9 print(ans)
```

R code Exa 4.a Possible committees

```
1 # Page No. 6
2
3 total_people = 20
4 committee_size = 3
5
6 ans = choose(total_people, committee_size)
7
8 print(ans)
```

R code Exa 4.b Committee selection with genders and feuds

```
1 # Page No. 6
2
3 total_men = 7
4 req_men = 3
5 total_women = 5
6 req_women = 2
7
8 ans1 = choose(total_men, req_men) * choose(total_
  women, req_women)
9
10 print(ans1)
11
12 no_of_feuding_men = 2
13
14 feuding_groups = choose(no_of_feuding_men, no_of_
  feuding_men) * choose(total_men - no_of_feuding_
  men, req_men - no_of_feuding_men)
15 ans2 = (choose(total_men, req_men) - feuding_groups)
  * choose(total_women, req_women)
```

```
16
17 print(ans2)
```

R code Exa 5.a Possible divisions

```
1 # Page No. 10
2
3 patrollers = 5
4 station_officers = 2
5 reserve = 3
6
7 ans = factorial(patrollers + station_officers +
8               reserve) / (factorial(patrollers) * factorial(
9               station_officers) * factorial(reserve))
10
11 print(ans)
```

R code Exa 5.b Division into team A and team B

```
1 # Page No. 10
2
3 teamA_size = 5
4 teamB_size = 5
5
6 ans = factorial(teamA_size + teamB_size) / (
7       factorial(teamA_size) * factorial(teamB_size))
8
9 print(ans)
```

R code Exa 5.c Basketball divisions

```

1 # Page No. 10
2
3 team_size = 5
4 no_of_teams = 2
5
6 ans = factorial(team_size * no_of_teams) / ((
    factorial(team_size)^2) * factorial(no_of_teams))
7
8 print(ans)

```

R code Exa 5.d Tournaments

```

1 # Page No. 11
2
3 no_of_players = 8
4 no_of_winners = 4
5
6 ans1 = factorial(no_of_winners) * choose(no_of_
    players, no_of_winners)
7
8 cat("Ans to a)", ans1, "\n")
9
10 ans2 = factorial(no_of_players)
11
12 cat("Ans to b)", ans2, "\n")
13
14 # The answer may vary due to difference in
    representation.

```

R code Exa 6.a Distinct non-negative integral solutions

```

1 # Page No. 13
2

```



```
3 const = 3
4 no_of_vars = 2
5
6 ans = choose(const + no_of_vars - 1, no_of_vars - 1)
7
8 print(ans)
```

R code Exa 6.b Investments

```
1 # Page No. 13
2
3 no_of_vars = 4
4 const = 20
5
6 ans1 = choose(const + no_of_vars - 1, no_of_vars -
7             1)
8 print(ans1)
9
10 no_of_vars_updated = 5
11
12 ans2 = choose(const + no_of_vars_updated - 1, no_of_
13             vars_updated - 1)
14 print(ans2)
```

Chapter 2

Axioms of Probability

R code Exa 3.b Even number on a dice roll

```
1 # Page No. 28
2
3 p_of_a_side = 1/6
4 no_of_even_sides = 3
5
6 ans = p_of_a_side * no_of_even_sides
7
8 print(ans)
9
10 # The answer may vary due to difference in
    representation.
```

R code Exa 4.a Books and vacations

```
1 # Page No. 30
2
3 p_book1 = 0.5
4 p_book2 = 0.4
```

```
5 p_both_books = 0.3
6
7 ans = 1 - (p_book1 + p_book2 - p_both_books)
8
9 print(ans)
```

R code Exa 5.a Sum of dice rolls

```
1 # Page No. 34
2
3 library(MASS)
4
5 favourable_outcomes = 6
6 total_outcomes = 36
7
8 ans = favourable_outcomes / total_outcomes
9
10 print(fractions(ans))
```

R code Exa 5.b Drawing balls from a bowl

```
1 # Page No. 34
2
3 library(MASS)
4
5 total_white_balls = 6
6 total_black_balls = 5
7 fav_no_of_white_balls = 1
8 fav_no_of_black_balls = 2
9 drawn_balls = 3
10
11 ans = (choose(total_white_balls, fav_no_of_white_
               balls) * choose(total_black_balls, fav_no_of_
```

```

        black_balls)) / choose(total_black_balls + total_
        white_balls, drawn_balls)
12
13 print(fractions(ans))

```

R code Exa 5.c Committee selection

```

1 # Page No. 35
2
3 library(MASS)
4
5 total_men = 6
6 total_women = 9
7 fav_no_of_men = 3
8 fav_no_of_women = 2
9 committee_size = 5
10
11 ans = (choose(total_women, fav_no_of_women) * choose
        (total_men, fav_no_of_men)) / choose(total_men +
        total_women, committee_size )
12
13 print(fractions(ans))

```

R code Exa 5.f Probability of a straight

```

1 # Page No. 36
2
3 hand_size = 5
4 total_cards = 52
5
6 no_of straights = 10 * (4^5 - 4)
7 ans = no_of_straights / choose(total_cards, hand_
        size)

```

```
8
9 print(ans)
```

R code Exa 5.g Probability of full house

```
1 # Page No. 37
2
3 total_hands = 5
4 total_cards = 52
5
6 total_full_houses = 13 * 12 * choose(4, 2) * choose
  (4, 3)
7 ans = total_full_houses / choose(total_cards, total_
  hands)
8
9 print(ans)
```

R code Exa 5.h.a Bridge player gets all spades

```
1 # Page No. 37
2
3 total_cards = 52
4 suite_size = 13
5
6 p_13spades = 1 / choose(total_cards, suite_size)
7 ans = 4 * p_13spades
8
9 print(ans)
```

R code Exa 5.h.b Bridge each player gets 1 spade

```

1 # Page No. 37
2
3 total_cards = 52
4 suite_size = 13
5 no_of_aces = 4
6 ace_per_suite = 1
7
8 ans = (factorial(no_of_aces) * (factorial(total_
      cards - no_of_aces) / factorial(suite_size - ace_
      per_suite)^4)) / (factorial(total_cards) /
      factorial(suite_size)^4)
9
10 print(ans)

```

R code Exa 5.i Birthday problem

```

1 # Page No. 38
2
3 days_in_a_year = 365
4 n = 0
5 p = 1
6 i = 0
7
8 while(p >= 0.5)
9 {
10   p = p * (days_in_a_year - i) / days_in_a_year
11   n = n + 1
12   i = i + 1
13 }
14
15 print(n)

```

R code Exa 5.k Football team

```

1 # Page No. 39
2
3 P_2i <- function(i, o = 20, d = 20)
4 {
5   k = (factorial(o - 2 * i) / (2^(o / 2 - i) *
6     factorial(o / 2 - i)))^2
7   number = ((dim(combn(o, 2 * i))[2])^2) * factorial
8     (2 * i) * k
9   denom = (factorial(o + d) / (2^20 * factorial(o)))
10  return(number / denom)
11 }
12
13 for(i in 0:10)
14 {
15   cat("P(", i, ") =", P_2i(i), "\n")
16 }
17
18 # The answer may vary due to difference in
19 representation.

```

R code Exa 5.1 Club sports

```

1 # Page No. 40
2
3 no_tennis = 36
4 no_squash = 28
5 no_badminton = 18
6 no_tennis_badminton = 12
7 no_tennis_squash = 22
8 no_badminton_squash = 9
9 no_all = 4
10
11 ans = no_tennis + no_squash + no_badminton - no_
12   badminton_squash - no_tennis_badminton - no_
13   tennis_squash + no_all

```

```
12
13 print(ans)
```

R code Exa 5.n Married couples

```
1 # Page No. 42
2
3 no_of_couples = 10
4 ans = 0
5
6 for(i in 1:10)
7 {
8   ans = ans + ((-1)^(i + 1)) * choose(10, i) * (2^i)
9     * factorial(19 - i) / factorial(19)
10 }
11 ans = 1 - ans
12 print(ans)
```

R code Exa 7.a Horse race

```
1 # Page No. 48
2
3 P = c(0.2, 0.2, 0.15, 0.15, 0.1, 0.1, 0.1)
4
5 v1 = P[1] + P[2] + P[3]
6
7 print(v1)
8
9 v2 = P[1] + P[5] + P[6] + P[7]
10
11 if(v1 > v2)
12 {
```



```
13     print("First  wager")
14 } else
15 {
16     print("Second  wager")
17 }
```

Chapter 3

Conditional Probability and Independence

R code Exa 2.a Student exams

```
1 # Page No. 59
2
3 P_Lx <- function(x)
4 {
5   return(x/2)
6 }
7 P_F = 1 - P_Lx(1)
8 ans = P_F / (1 - P_Lx(0.75))
9
10 print(ans)
```

R code Exa 2.b.a First coin flip heads

```
1 # Page No. 59
2
3 library(MASS)
```

```
4
5 P_B = 1/4
6 P_F = 1/2
7
8 ans = P_B / P_F
9
10 print(fractions(ans))
```

R code Exa 2.b.b At least one flip heads

```
1 # Page No. 59
2
3 library(MASS)
4
5 P_B = 1/4
6 P_A = 3/4
7
8 ans = P_B / P_A
9
10 print(fractions(ans))
```

R code Exa 2.c Bridge

```
1 # Page No. 60
2
3 total_north_south = 26
4 suite_size = 13
5
6 ans = choose(5,3) * choose(total_north_south - 5,
7                             suite_size - 3) / choose(total_north_south, suite
8                                                         _size)
9
10 print(ans)
```

R code Exa 2.e Course selection

```
1 # Page No. 61
2
3 library(MASS)
4
5 P_C = 1/2
6 P_AgC = 2/3
7
8 ans = P_C * P_AgC
9
10 print(fractions(ans))
```

R code Exa 2.f.a Drawing red balls from urn

```
1 # Page No. 62
2
3 library(MASS)
4
5 no_of_red_balls = 8
6 no_of_blue_balls = 4
7
8 P_R1 = no_of_red_balls / (no_of_blue_balls + no_of_
    red_balls)
9 P_R2gR1 = (no_of_red_balls - 1 ) / (no_of_blue_balls
    + no_of_red_balls - 1)
10 ans = P_R1 * P_R2gR1
11
12 print(fractions(ans))
```

R code Exa 2.h Each pile contains 1 ace

```
1 # Page No. 64
2
3 P_E1 = 1
4 P_E2gE1 = 39/51
5 P_E3gE1E2 = 26/50
6 P_E4gE1E2E3 = 13/49
7
8 ans = P_E1 * P_E2gE1 * P_E3gE1E2 * P_E4gE1E2E3
9
10 print(ans)
```

R code Exa 3.d Laboratory blood test

```
1 # Page No. 67
2
3 P_D = 0.005
4 P_EgD = 0.95
5 P_EgDc = 0.01
6
7 ans = P_EgD * P_D / (P_EgD * P_D + P_EgDc * (1 - P_D
8     ))
9 print(ans)
```

R code Exa 3.e Surgery dilemma

```
1 # Page No. 68
2
3 P_D = 0.6
4 P_EgD = 1
5 P_EgDc = 0.3
```

```

6
7 ans = P_D * P_EgD / (P_EgD * P_D + P_EgDc * (1 - P_D
  ))
8
9 print(ans)

```

R code Exa 3.f Criminal investigation

```

1 # Page No. 69
2
3 P_G = 0.6
4 P_CgG = 1
5 P_CgGc = 0.2
6
7 ans = P_G * P_CgG / (P_G * P_CgG + (1 - P_G) * P_
  CgGc)
8
9 print(ans)

```

R code Exa 3.i Urn and coins

```

1 # Page No. 72
2
3 library(MASS)
4
5 P_A = 2/3
6 P_headsgA = 1/4
7 P_B = 1/3
8 P_headsgB = 3/4
9
10 ans = P_A * P_headsgA / (P_B * P_headsgB)
11
12 print(fractions(ans))

```

R code Exa 3.1 Coloured cards

```
1 # Page No. 75
2
3 library(MASS)
4
5 P_RgRB = 1/2
6 P_RB = 1/3
7 P_RgRR = 1
8 P_RR = 1/3
9 P_RgBB = 0
10 P_BB = 1/3
11
12 ans = P_RgRB * P_RB / (P_RgBB * P_BB + P_RgRB * P_RB
13     + P_RgRR * P_RR)
14 print(fractions(ans))
```

R code Exa 3.n.a Flashlight gives more than 100 hours of use

```
1 # Page No. 77
2
3 P_F1 = 0.2
4 P_F2 = 0.3
5 P_F3 = 0.5
6 P_AgF1 = 0.7
7 P_AgF2 = 0.4
8 P_AgF3 = 0.3
9
10 ans = P_AgF1 * P_F1 + P_AgF2 * P_F2 + P_AgF3 * P_F3
11
12 print(ans)
```

R code Exa 3.n.b Type of flashlight

```
1 # Page No. 77
2
3 library(MASS)
4
5 P_FgA <- function(P_F, P_AgF)
6 {
7   P_A = 0.41
8   return(P_F * P_AgF / P_A)
9 }
10
11 P_AgF1 = 0.7
12 P_F1 = 0.2
13 P_AgF2 = 0.4
14 P_F2 = 0.3
15 P_AgF3 = 0.3
16 P_F3 = 0.5
17
18 cat("P(F1 | A) = ")
19 print(fractions(P_FgA(P_F1, P_AgF1)))
20 cat("P(F2 | A) = ")
21 print(fractions(P_FgA(P_F2, P_AgF2)))
22 cat("P(F3 | A) = ")
23 print(fractions(P_FgA(P_F3, P_AgF3)))
```

R code Exa 3.o Crime probability

```
1 # Page No. 77
2
3 ex_criminals = 10000
```



```

4 tot_pop = 1000000
5 P_hair_match = 10^-5
6
7 alpha = function(c)
8 {
9   c / (ex_criminals * c + tot_pop - ex_criminals)
10 }
11 P_MgG = (1 - P_hair_match)^(ex_criminals - 1)
12 P_all_aj = function(c)
13 {
14   (1 - ex_criminals * alpha(c)) / (1 - alpha(c))
15 }
16 P_MgGc = function(c)
17 {
18   P_hair_match * P_all_aj(c) * (1 - P_hair_match)^(
19     ex_criminals - 1)
20 }
21 P_G = alpha
22 P_GgM = function(c)
23 {
24   (P_MgG * P_G(c)) / (P_MgG * P_G(c) + P_MgGc(c) *
25     (1 - P_G(c)))
26 }
27 c1 = 100
28 ans1 = P_GgM(c1)
29 cat("For c =", c1, "alpha =", alpha(c1), "and P(G|M)
30   =", P_GgM(c1), "\n")
31 c2 = 10
32 ans2 = P_GgM(c2)
33
34 cat("For c =", c2, "alpha =", alpha(c2), "and P(G|M)
35   =", P_GgM(c2), "\n")
36 c3 = 1
37 ans3 = P_GgM(c3)

```

```

38
39 cat("For c =", c3, "alpha =", alpha(c3), "and P(G|M)
    =", P_GgM(c3), "\n")
40
41 # The answer may vary due to difference in
    representation.

```

R code Exa 4.h Independent die trials

```

1 # Page No. 83
2
3 library(MASS)
4
5 P_En <- function(n)
6 {
7     ans = ((13/18)^(n - 1)) * 1/9
8     return(ans)
9 }
10
11 ans = 0
12 i = 2
13
14 ix = P_En(1)
15
16 while(ix != 0)
17 {
18     ans = ans + ix
19     ix = P_En(i)
20     i = i + 1
21 }
22
23 print(fractions(ans))

```

R code Exa 5.a Conditional probability of accident prone policy holder

```
1 # Page No. 94
2
3 library(MASS)
4
5 P_A1gA = 0.4
6 P_A = 0.3
7 P_A1 = 0.26
8
9 P_AgA1 = P_A1gA * P_A / P_A1
10
11 P_A2gAA1 = 0.4
12 P_A2gAcA1 = 0.2
13
14 P_A2gA1 = P_A2gAA1 * P_AgA1 + P_A2gAcA1 * (1 - P_
    AgA1)
15
16 print(P_A2gA1)
```

R code Exa 3.a.1 Accident within a year of purchasing policy

```
1 # Page No. 66
2
3 P_A1gA = 0.4
4 P_A = 0.3
5
6 P_A1gAc = 0.2
7 ans = P_A1gA * P_A + P_A1gAc * (1 - P_A)
8
9 print(ans)
```

R code Exa 3.a.2 Policy holder is accident prone

```
1 # Page No. 66
2
3 library(MASS)
4
5 P_A = 0.3
6 P_A1 = 0.26
7 P_A1gA = 0.4
8
9 ans = P_A*P_A1gA/P_A1
10
11 print(fractions(ans))
```

Chapter 4

Random Variables

R code Exa 1.b Urn with replacement

```
1 # Page No. 118
2
3 P <- function(i)
4 {
5   x = choose(i - 1,2) / choose(20,3)
6   return(x)
7 }
8
9 ans = 0
10
11 for(i in 17:20)
12 {
13   ans = ans + P(i)
14 }
15
16 print(ans)
```

R code Exa 1.d Winnings from urn experiment

```

1 # Page No. 119
2
3 library(MASS)
4
5 P <- rep(4)
6
7 P[0] <- (choose(5,3) + choose(3,1) * choose(3,1) *
           choose(5,1)) / choose(11,3)
8 P[1] <- (choose(3,1) * choose(5,2) + choose(3,2) *
           choose(3,1)) / choose(11,3)
9 P[2] <- (choose(3,2) * choose(5,1)) / choose(11,3)
10 P[3] <- choose(3,3) / choose(11,3)
11
12 print(fractions(sum(P)))

```

R code Exa 10.a Properties of random variables

```

1 # Page No. 169
2
3 library(MASS)
4
5 sentinel = exp(-12)
6
7 F_x = function(x)
8 {
9   if(x < 0)
10   {
11     return(0)
12   }
13   if(x < 1)
14   {
15     return(x/2)
16   }
17   if(x < 2)
18   {

```

```

19     return(2/3)
20 }
21 if(x < 3)
22 {
23     return(11/12)
24 }
25 else
26 {
27     return(1)
28 }
29 }
30 P_X = function(F, b)
31 {
32     return(F(b - sentinal))
33 }
34 ans1 = P_X(F_x, 3)
35
36 cat("a: ")
37 print(fractions(ans1))
38
39 ans2 = F_x(1) - P_X(F_x, 1)
40
41 cat("b: ")
42 print(fractions(ans2))
43
44 ans3 = 1 - F_x(1/2)
45
46 cat("c: ")
47 print(fractions(ans3))
48
49 ans4 = F_x(4) - F_x(2)
50
51 cat("d: ")
52 print(fractions(ans4))

```

R code Exa 3.a Expectation of a die roll

```
1 # Page No. 126
2
3 library(MASS)
4
5 X = 1:6
6 w = rep(1/6, 6)
7
8 ans = weighted.mean(X, w)
9
10 print(fractions(ans))
```

R code Exa 3.d Expectation of number of students

```
1 # Page No. 127
2
3 P_Xe36 = 36/120
4 P_Xe40 = 40/120
5 P_Xe44 = 44/120
6 w = c(P_Xe36, P_Xe40, P_Xe44)
7 X = c(36, 40, 44)
8
9 ans = weighted.mean(X, w)
10
11 print(ans)
```

R code Exa 4.a Expectation of a random variable squared

```
1 # Page No. 128
2
3 P_Ye1 = 0.3 + 0.2
4 P_Ye0 = 0.5
```



```

5 w = c(P_Ye0, P_Ye1)
6 X = 0:1
7
8 ans = weighted.mean(X, w)
9
10 print(ans)

```

R code Exa 5.a Variance of a die roll

```

1 # Page No. 133
2
3 library(MASS)
4
5 X = 1:6
6 w = rep(1/6, 6)
7
8 E_X = weighted.mean(X, w)
9 X1 = X^2
10 E_X1 = weighted.mean(X1, w)
11 Var_X = E_X1 - E_X^2
12
13 print(fractions(Var_X))

```

R code Exa 6.a Binomial random variable coin experiment

```

1 # Page No. 135
2
3 library("MASS")
4
5 n = 5
6 p = 1/2
7 x = 0:5
8

```

```

9 ans = dbinom(x, size = n, prob = p)
10
11 for(i in x)
12 {
13   cat("P{ X =", i, " } =")
14   print(fractions(ans[i + 1]))
15 }
16
17 # The answer may vary due to difference in
   representation.

```

R code Exa 6.b Package replacement

```

1 # Page No. 135
2
3 p = 0.01
4 n = 10
5
6 ans = 1 - pbinom(1, size = n, prob = p)
7
8 print(ans)

```

R code Exa 6.c Wheel of fortune

```

1 # Page No. 136
2
3 library(MASS)
4
5 n = 3
6 p = 1/6
7 x = 0:3
8 X = c(-1, 1, 2, 3)
9

```

```

10 P_X = dbinom(x, size = n, prob = p)
11 ans = weighted.mean(X, P_X)
12
13 print(fractions(ans))

```

R code Exa 6.d Genetic traits

```

1 # Page No. 136
2
3 library(MASS)
4
5 n = 4
6 p = 3/4
7
8 ans = dbinom(3, size = n, prob = p)
9
10 print(fractions(ans))

```

R code Exa 6.h Computing binomial distribution function

```

1 # Page No. 142
2
3 n = 6
4 p = 0.4
5 x = 0:6
6
7 P_X = dbinom(x, size = n, prob = p)
8
9 for(i in 0:6)
10 {
11   cat("P{ X =", i, " } =", P_X[i + 1], "\n")
12 }

```

R code Exa 6.i Binomial distribution generation

```
1 # Page No. 143
2
3 n = 100
4 p = 0.75
5
6 ans1 = dbinom(70, size = n, prob = p)
7
8 cat("P{X = 70} = ", ans1, "\n")
9
10 ans2 = pbinom(70, size = n, prob = p)
11
12 cat("P{X <= 70} = ", ans2, "\n")
```

R code Exa 7.a Typos in a page

```
1 # Page No. 144
2
3 l = 1/2
4
5 ans = 1 - dpois(0, lambda = 1)
6
7 print(ans)
```

R code Exa 7.b Defective item

```
1 # Page No. 145
2
```

```

3 n = 10
4 p = 0.1
5 x = 1
6
7 binom_ans = pbinom(x, size = n, prob = p)
8
9 cat(binom_ans, "\n")
10
11 l = n * p
12 pois_ans = ppois(x, l)
13
14 cat(pois_ans)

```

R code Exa 7.c Radioactive particles

```

1 # Page No. 145
2
3 l = 3.2
4 q = 2
5
6 ans = ppois(q, lambda = l)
7
8 print(ans)

```

R code Exa 7.e.a Earthquake occurrence

```

1 # Page No. 154
2
3 no_of_weeks = 2
4 x = 2
5
6 l = 2 * no_of_weeks
7 ans = 1 - ppois(x, lambda = l)

```

```
8
9 print(ans)
10
11 # The answer may vary due to difference in
    representation.
```

R code Exa 7.f Generating poisson distribution

```
1 # Page No. 155
2
3 l1 = 100
4 x1 = 90
5
6 ans1 = ppois(x1, l1)
7
8 cat("a) ", ans1, "\n")
9
10 l2 = 1000
11 x2 = 1075
12
13 ans2 = ppois(x2, l2)
14
15 cat("b) ", ans2)
16
17 # The answer may slightly vary due to rounding off
    values.
```

R code Exa 8.g Throws of die required

```
1 # Page No. 160
2
3 r = 4
4 p = 1/6
```

```

5
6 E_X = r / p
7 Var_X = r * (1 - p) / p^2
8
9 cat(E_X, "\n")
10 cat(Var_X, "\n")

```

R code Exa 8.i Electrical components

```

1 # Page No. 161
2
3 lot_size = 10
4
5 P_4defectives = 0.3
6 P_1defective = 0.7
7
8 inspect_size = 3
9
10 P_acceptance = choose(4,0) * choose(6,3) * P_4
    defectives / choose(lot_size, inspect_size) +
    choose(1,0) * choose(9,3) * P_1defective / choose
    (lot_size, inspect_size)
11
12 ans = 1 - P_acceptance
13
14 print(ans*100)

```

Chapter 5

Continuous Random Variables

R code Exa 1.a.a Finding the constant in "fx"

```
1 # Page No. 187
2
3 library(MASS)
4
5 integrand = function(x)
6 {
7   4 * x - 2 * x^2
8 }
9 C = 1 / integrate(integrand, lower = 0, upper = 2)$
   value
10
11 print(fractions(C))
```

R code Exa 1.a.b Probability of a continuous random variable

```
1 # Page No. 187
2
3 library("MASS")
```



```

4
5 integrand = function(x)
6 {
7   C = 3/8
8   C * (4 * x - 2 * x^2)
9 }
10 ans = integrate(integrand, lower = 1, upper = 2)$
    value
11
12 print(fractions(ans))

```

R code Exa 1.b.a Computer functioning "a"

```

1 # Page No. 188
2
3 integrand = function(x)
4 {
5   exp(-x/100)
6 }
7 l = 1 / integrate(integrand, lower = 0, upper = Inf)
    $value
8 ans = l * integrate(integrand, lower = 50, upper =
    150)$value
9
10 print(ans)

```

R code Exa 1.b.b Computer functioning "b"

```

1 # Page No. 188
2
3 l = 0.01
4
5 integrand = function(x)

```

```

6 {
7   exp(-x/100)
8 }
9 ans = 1 * integrate(integrand, lower = 0, upper =
    100)$value
10
11 print(ans)

```

R code Exa 1.c Lifetime of radio tube

```

1 # Page No. 188
2
3 library(MASS)
4
5 n = 5
6 x = 2
7
8 integrand = function(x)
9 {
10   100 / x^2
11 }
12 P_Ei = integrate(integrand, lower = 100, upper =
    150)$value
13 ans = dbinom(x, size = n, prob = P_Ei)
14
15 print(fractions(ans))

```

R code Exa 2.a Expectation of a continuous random variable

```

1 # Page No. 190
2
3 library("MASS")
4

```

```

5 integrand = function(x)
6 {
7   2 * x^2
8 }
9 ans = integrate(integrand, lower = 0, upper = 1)$
    value
10
11 print(fractions(ans))

```

R code Exa 2.e Variance of a continuous random variable

```

1 # Page No. 194
2
3 library("MASS")
4
5 E_X = 2/3
6
7 integrand = function(x)
8 {
9   2 * x^3
10 }
11 E_X2 = integrate(integrand, lower = 0, upper = 1)$
    value
12 ans = E_X2 - E_X^2
13
14 print(fractions(ans))

```

R code Exa 3.b Uniform distribution

```

1 # Page No. 196
2
3 library("MASS")
4

```

```

5 min = 0
6 max = 10
7
8 ans1 = punif(3, min = min, max = max)
9
10 cat("Ans to a)")
11 print(fractions(ans1))
12
13 ans2 = 1 - punif(6, min = min, max = max)
14
15 cat("Ans to b)")
16 print(fractions(ans2))
17
18 ans3 = punif(8, min = min, max = max) - punif(3, min
      = min, max = max)
19
20 cat("Ans to c)")
21 print(fractions(ans3))
22
23 # The answer may vary due to difference in
    representation.

```

R code Exa 3.c Waiting time for bus

```

1 # Page No. 196
2
3 library("MASS")
4
5 uni = function(lower, upper)
6 {
7   alpha = 0
8   beta = 30
9
10   P_X = punif(upper, min = alpha, max = beta) -
      punif(lower, min = alpha, max = beta)

```

```

11   return(P_X)
12 }
13 ans1 = uni(10, 15) + uni(25, 30)
14
15 cat("Ans for a)")
16 print(fractions(ans1))
17
18 ans2 = uni(0, 5) + uni(15, 20)
19
20 cat("Ans for b)")
21 print(fractions(ans2))

```

R code Exa 4.b Normal random variable

```

1 # Page No. 202
2
3 mu = 3
4 sigma = sqrt(9)
5
6 ans1 = pnorm(5, mu, sigma) - pnorm(2, mu, sigma)
7
8 cat("Ans to a) ", ans1, "\n")
9
10 ans2 = 1 - pnorm(0, mu, sigma)
11
12 cat("Ans to b) ", ans2, "\n")
13
14 ans3 = 1 - pnorm(9, mu, sigma) + pnorm(-3, mu, sigma)
15
16 cat("Ans to c) ", ans3, "\n")

```

R code Exa 4.c Grading on the curve

```

1 # Page No. 202
2
3 cat("A:", (1 - pnorm(1)) * 100, "%\n")
4 cat("B:", (pnorm(1) - pnorm(0)) * 100, "%\n")
5 cat("C:", (pnorm(0) - pnorm(-1)) * 100, "%\n")
6 cat("D:", (pnorm(2) - pnorm(1)) * 100, "%\n")
7 cat("E:", (pnorm(-2)) * 100, "%\n")

```

R code Exa 4.d Paternity suit

```

1 # Page No. 203
2
3 mu = 270
4 sigma = sqrt(100)
5
6 ans = 1 - pnorm(290, mu, sigma) + pnorm(240, mu,
      sigma)
7
8 print(ans)

```

R code Exa 4.e Signals

```

1 # Page No. 203
2
3 P_EgOne = 1 - pnorm(1.5)
4
5 cat("P{1} =", P_EgOne, "\n")
6
7 P_EgZero = 1 - pnorm(2.5)
8
9 cat("P{0} =", P_EgZero, "\n")

```

R code Exa 4.f Normal approximation

```
1 # Page No. 204
2
3 n = 40
4 p = 1/2
5 x = 20
6
7 mu = n * p
8 sigma = sqrt(n * p * (1 - p))
9 ans1 = pnorm(x + 0.5, mean = mu, sd = sigma) - pnorm
      (x - 0.5, mean = mu, sd = sigma)
10
11 cat("Ans via approx:", ans1, "\n")
12
13 ans2 = dbinom(x, size = n, prob = p)
14
15 cat("Ans:", ans2)
16
17 # The answer may slightly vary due to rounding off
    values.
```

R code Exa 4.g Probability of attendance in a college

```
1 # Page No. 205
2
3 n = 450
4 p = 0.3
5
6 mu = n * p
7 sigma = sqrt(n * p * (1 - p))
8 ans = 1 - pnorm(150.5, mu, sigma)
```

```
9
10 print(ans)
```

R code Exa 4.h Effectiveness of diet

```
1 # Page No. 206
2
3 n = 100
4 p = 1/2
5
6 mu = n * p
7 sigma = sqrt(n * p * (1 - p))
8 ans = 1 - pnorm(64.5, mu, sigma)
9
10 print(ans)
```

R code Exa 4.i Outlawing cigarettes

```
1 # Page No. 206
2
3 P_Sn = function(n)
4 {
5   p = 0.52
6   mu = p * n
7   sigma = sqrt(n * p * (1 - p))
8   ans = 1 - pnorm(0.5 * n, mu, sigma)
9   return(ans)
10 }
11
12 N = c(11, 101, 1001)
13
14 for(i in N)
15 {
```



```

16   cat("For n =", i, "P(Sn > 0.5 * n) =", P_Sn(i), "\n"
      n")
17 }
18
19 i = 1
20
21 while(P_Sn(i) < 0.95)
22 {
23   i = i + 1
24 }
25
26 cat("For at least 95%, n =", i, "\n")

```

R code Exa 5.b Exponential random variable

```

1 # Page No. 209
2
3 lambda = 1/10
4
5 ans1 = 1 - pexp(10, lambda)
6
7 cat("Ans to a)", ans1, "\n")
8
9 ans2 = pexp(20, lambda) - pexp(10, lambda)
10
11 cat("Ans to b)", ans2, "\n")

```

R code Exa 5.e Laplace distribution

```

1 # Page No. 212
2
3 library(rmutil)
4

```

```
5 P_EgOne = plaplace(-1.5)
6 P_EgZero = plaplace(-2.5)
7
8 cat("P{1} =", P_EgOne, "\n")
9 cat("P{0} =", P_EgZero, "\n")
```

Chapter 6

Jointly Distributed Random Variables

R code Exa 1.a Joint PMF of drawing balls from urn

```
1 # Page No. 233
2
3 library(MASS)
4
5 P_XY = function(x, y)
6 {
7   blue = 5
8   red = 3
9   white = 4
10  lot_size = 3
11
12  ans = choose(blue, lot_size - x - y) * choose(red,
13           x) * choose(white, y) / choose(blue + white +
14           red, lot_size)
15 }
16
17 lot_size = 3
18
19 for(i in 0:lot_size)
```

```

18 {
19   for(j in 0:(lot_size - i))
20   {
21     cat("p(", i, ", ", j, ") =")
22     print(fractions(P_XY(i, j)))
23   }
24 }
25
26 # The answer may vary due to difference in
    representation.

```

R code Exa 1.b Joint PMF of number of boys and girls in a family

```

1 # Page No. 234
2
3 P = c(.15, .2, .35, .3)
4
5 P_BG <- function(i, j)
6 {
7   if((i + j) >= 4)
8   {
9     return(0)
10  }else
11  {
12    ans = P[(i + j) + 1] * ((factorial(i + j) / (
13      factorial(i) * factorial(j))) / 2^(i + j))
14    return(ans)
15  }
16 }
17 max_children = 3
18
19 for(i in 0:max_children)
20 {
21   for(j in 0:max_children)

```

```

22     {
23         cat("p(", i, ", ", j, ") =", P_BG(i, j), "\n")
24     }
25 }

```

R code Exa 1.c.a Joint density function "a"

```

1  # Page No. 236
2
3  myfun = function(x, y) (2 * exp(-x) * exp(-2 * y))
4  llimx = 1
5  ulimx = Inf
6  llimy = 0
7  ulimy = 1
8
9  f = function()
10 {
11     return(integrate(function(y)
12         {
13             sapply(y, function(y)
14                 {
15                     integrate(function(x) myfun(x,y), llimx, ulimx
16                         )$value
17                 })
18             }, llimy, ulimy))
19     ans = f()$value
20
21     print(ans)

```

R code Exa 1.c.b Joint density function "b"

```

1  # Page No. 236

```

```

2
3 library("MASS")
4
5 myfun <- function(x,y) (2 * exp(-x) * exp(-2 * y))
6 llimx <- 0
7 llimy <- 0
8 ulimy <- Inf
9
10 f <- function()
11 {
12   return(integrate(function(y)
13     {
14       sapply(y, function(y)
15         {
16           integrate(function(x) myfun(x,y), llimx, y)$
17             value
18         }, llimy, ulimy))
19   }
20   ans = f()$value
21
22   print(fractions(ans))

```

R code Exa 2.c Waiting time

```

1 # Page No. 243
2
3 library('pracma')
4 library(MASS)
5
6 integrand = function(x, y)
7 {
8   1/60^2
9 }
10 xm = function(y)

```

```

11 {
12   y - 10
13 }
14 ans = 2 * integral2(integrand, xmin = 10, xmax =
      60, ymin = 0, ymax = xm)$Q
15
16 print(fractions(ans))

```

R code Exa 3.c.a Basketball wins "a"

```

1 # Page No. 257
2
3 na = 26
4 pa = 0.4
5
6 E_Xa = na * pa
7 Var_Xa = na * pa * (1 - pa)
8
9 nb = 18
10 pb = 0.7
11
12 E_Xb = nb * pb
13 Var_Xb = nb * pb * (1 - pb)
14 E_Xab = E_Xa + E_Xb
15 Var_Xab = Var_Xa + Var_Xb
16 ans = 1 - pnorm(25 - 1/2, E_Xab, sqrt(Var_Xab))
17
18 print(ans)

```

R code Exa 3.c.b Basketball wins "b"

```

1 # Page No. 257
2

```

```

3 na = 26
4 pa = 0.4
5
6 E_Xa = na * pa
7 Var_Xa = na * pa * (1 - pa)
8
9 nb = 18
10 pb = 0.7
11
12 E_Xb = nb * pb
13 Var_Xb = nb * pb * (1 - pb)
14 E_Xab = E_Xa - E_Xb
15 Var_Xab = Var_Xa + Var_Xb
16 ans = 1 - pnorm(1 - 1/2, E_Xab, sqrt(Var_Xab))
17
18 print(ans)

```

R code Exa 3.d.a Price of security "a"

```

1 # Page No. 258
2
3 mu = 0.0165
4 sigma = 0.0730
5
6 p = 1 - pnorm(0, mu, sigma)
7 ans = p^2
8
9 print(ans)

```

R code Exa 3.d.b Price of security "b"

```

1 # Page No. 258
2

```



```

3 mu = 0.0165
4 sigma = 0.0730
5
6 mu = mu * 2
7 sigma = sqrt(2 * sigma^2)
8 ans = 1 - pnorm(0, mu, sigma)
9
10 print(ans)

```

R code Exa 4.a Conditional probability on joint PMF

```

1 # Page No. 264
2
3 library("MASS")
4
5 p00 = 0.4
6 p01 = 0.2
7 p10 = 0.1
8 p11 = 0.3
9
10 py1 = p01 + p11
11 px0 = p01 / py1
12 px1 = p11 / py1
13
14 print(fractions(px0))
15 print(fractions(px1))

```

R code Exa 6.b Probability of sample median

```

1 # Page No. 272
2
3 library("MASS")
4

```

```
5 integrand = function(x)
6 {
7   x * (1 - x)
8 }
9 ans = factorial(3) / factorial(1)^2 * integrate(
    integrand, lower = 1/4, upper = 3/4)$value
10
11 print(fractions(ans))
```

Chapter 7

Properties of Expectation

R code Exa 2.r Chipmunks and groves

```
1 # Page No. 312
2
3 no_of_chipmunks = 15
4 total_trees = 52
5 p = 7/52
6
7 Xi = rep(1, no_of_chipmunks)
8 E_Xi = Xi * p
9 E_X = sum(E_Xi)
10
11 print(E_X)
```

R code Exa 5.e.a Game of carps "a"

```
1 # Page No. 335
2
3 Pi = function(i)
4 {
```

```

5   if(i > 7)
6   {
7       i = 14 - i
8   }
9   ans = (i - 1) / 36
10  }
11  E_RgSi = function(i)
12  {
13      if(i == 2 || i == 3 || i == 7 || i == 11 || i ==
14          12)
15          {
16              return(1)
17          }
18      else
19          {
20              ans = 1 + 1 / (Pi(i) + Pi(7))
21              return(ans)
22          }
23  }
24  E_R = 0
25
26  for(i in 2:12)
27  {
28      E_R = E_R + E_RgSi(i) * Pi(i)
29  }
30
31  print(E_R)

```

R code Exa 5.e.b Game of carps "b"

```

1  # Page No. 335
2
3  Pi = function(i)
4  {

```

```

5   if(i > 7)
6   {
7       i = 14 - i
8   }
9   ans = (i - 1) / 36
10  }
11  P_Si = function(c)
12  {
13      if(i == 7 || i == 11)
14      {
15          return(Pi(i))
16      }
17      if( i == 2 || i == 3 || i == 12)
18      {
19          return(0)
20      }
21      else
22      {
23          ans = Pi(i)^2 / (Pi(i) + Pi(7))
24      }
25  }
26
27  p = 0
28
29  for(i in 2:12)
30  {
31      p = p + P_Si(i)
32  }
33
34  print(p)

```

R code Exa 5.e.c Game of carps "c"

1 # Page No. 335
2

```

3 p = 0.493
4 E_R = 3.376
5
6 E_RgSi = function(i)
7 {
8     if(i == 2 || i == 3 || i == 7 || i == 11 || i ==
        12)
9     {
10         return(1)
11     }
12     else
13     {
14         ans = 1 + 1 / (Pi(i) + Pi(7))
15         return(ans)
16     }
17 }
18 Pi = function(i)
19 {
20     if(i > 7)
21     {
22         i = 14 - i
23     }
24     ans = (i - 1) / 36
25 }
26 Qi = function(i)
27 {
28     if( i == 2 || i == 3 || i == 12)
29     {
30         return(0)
31     }
32     if( i == 7 || i == 11)
33     {
34         return(Pi(i) / p)
35     }
36     else
37     {
38         ans = Pi(i)^2 / (p * (Pi(i) + Pi(7)))
39         return(ans)

```

```
40     }
41 }
42
43 E_Rgwin = 0
44
45 for(i in 2:12)
46 {
47     E_Rgwin = E_Rgwin + E_RgSi(i) * Qi(i)
48 }
49
50 E_Rglose = (E_R - E_Rgwin * p) / (1 - p)
51
52 print(E_Rglose)
```

Chapter 8

Limit Theorems

R code Exa 2.b Chebyshev inequality

```
1 # Page No. 390
2
3 alpha = 0
4 beta = 10
5
6 E_X = (alpha + beta) / 2
7 Var_X = (beta - alpha)^2 / 12
8
9 ans1 = Var_X / 4^2
10
11 cat("From Chebyshev's inequality ,  $P\{|X - 5| > 4\} \leq$ "
    , ans1, "\n")
12
13 ans2 = punif(1, min = alpha, max = beta) + 1 - punif
    (9, min = 0, max = 10)
14
15 cat("Exact ans ,  $P\{|X - 5| > 4\} =$ ", ans2, "\n")
```

R code Exa 3.a Astronomical distances


```

1 # Page No. 393
2
3 P = function(n)
4 {
5   acc = 0.5
6   ans = pnorm(0.5 * sqrt(n) / 2, 0, 1) - pnorm(-0.5
7     * sqrt(n) / 2, 0, 1)
8   return(ans)
9 }
10 i = 1
11
12 while(P(i) < 0.95)
13 {
14   i = i + 1
15 }
16
17 print(i)

```

R code Exa 3.b Probability of a professor teaching two sections

```

1 # Page No. 396
2
3 mu = 100
4 var = mu
5
6 ans = 1 - pnorm(120 - 1/2, mu, sqrt(var))
7
8 print(ans)

```

R code Exa 3.c Sum of dice rolls

```

1 # Page No. 397

```

```

2
3 p = 1/6
4 E_X = 0
5 E_X2 = 0
6 no_of_die_rolls = 10
7
8 for(i in 1:6)
9 {
10   E_X = E_X + p * i
11   E_X2 = E_X2 + p * i^2
12 }
13
14 print(E_X)
15 print(E_X2)
16
17 Var_X = (E_X2 - E_X^2) * no_of_die_rolls
18 E_X = E_X * no_of_die_rolls
19 ans = pnorm(40 + 1/2, E_X, sqrt(Var_X)) - pnorm(30 -
      1/2, E_X, sqrt(Var_X))
20
21 print(ans)

```

R code Exa 3.d Approximation of uniform random variables

```

1 E_X = 1/2
2 Var_X = 1/12
3
4 n = 10
5
6 E_X = 1/2*10
7 Var_X = 1/12*10
8
9 ans = 1 - pnorm(6, E_X, sqrt(Var_X))
10
11 print(ans)

```

R code Exa 3.e Time for grading exams

```
1 # Page No. 398
2
3 mu = 20
4 sd = 4
5 n = 25
6
7 mu = mu * n
8 sd = sd * sqrt(n)
9 ans = pnorm(450, mu, sd)
10
11 print(ans)
```

R code Exa 5.a Factory production

```
1 # Page No. 405
2
3 mu = 100
4 var = 400
5 a = 20
6
7 ans = var / (var + a^2)
8
9 print(ans)
```

R code Exa 5.b Pairs

```
1 # Page No. 405
```

```

2
3 tot_men = 100
4 tot_people = 200
5
6 E_Xi = tot_men / (tot_people - 1)
7 E_X = E_Xi * tot_men
8 Var_X = E_X * (tot_men - 1) / (tot_people - 1) + 2 *
    choose(tot_men, 2) * ((tot_men*(tot_men - 1)) /
        ((tot_people - 1) * (tot_people - 3)) - (tot_men
            / (tot_people - 1))^2)
9 a = E_X - 30
10 ans = Var_X / (Var_X + a^2)
11
12 print(ans)

```
