Solutions Manual to accompany Quantitative Methods

An Introduction for Business Management

Provisional version of June 10, 2014

Paolo Brandimarte

A Wiley-Interscience Publication

JOHN WILEY & SONS, INC.

New York / Chichester / Weinheim / Brisbane / Singapore / Toronto

Contents

Preface *vii*

1 Quantitative Methods: Should We Bother? *1*
1.1 Solutions *1*
1.2 Computational supplement: How to solve the optimal mix problem *3*

2 Calculus *7*
2.1 Solutions *7*

3 Linear Algebra *15*
3.1 Solutions *15*

4 Descriptive Statistics: On the Way to Elementary Probability *25*
4.1 Solutions *25*

5 Probability Theories *29*
5.1 Solutions *29*
5.2 Additional problems *30*
5.3 Solutions of additional problems *31*

6 Discrete Random Variables *33*
6.1 Solutions *33*

7 Continuous Random Variables 37
7.1 Solutions 37

8 Dependence, Correlation, and Conditional Expectation 43
8.1 Solutions 43

9 Inferential Statistics 47
9.1 Solutions 47

10 Simple Linear Regression 63
10.1 Solutions 63

11 Time Series Models 69
11.1 Solutions 69

12 Deterministic Decision Models 75
12.1 Solutions 75

13 Decision Making Under Risk 91
13.1 Solutions 91

14 Advanced Regression Models 99
14.1 Solutions 99

Appendix A R – A software tool for statistics 103

Appendix B Introduction to MATLAB 105
B.1 Working with vectors and matrices in the MATLAB environment 105
B.2 MATLAB graphics 113
B.3 Solving equations and computing integrals 114
B.4 Statistics in MATLAB 116
B.5 Using MATLAB to solve linear and quadratic programming problems 118

Appendix C Introduction to AMPL 121
C.1 Running optimization models in AMPL 122
C.2 Mean-variance efficient portfolios in AMPL 123
C.3 The knapsack model in AMPL 125

Preface

This solutions manual contains

- worked-out solutions to end-of-chapter problems in the book
- additional problems (solved)
- computational supplements illustrating the application of the following tools:
  - Microsoft Excel
  - R
  - MATLAB
  - AMPL

Some software tools are introduced in the appendices, where I am giving you a few hints and clues about how they can be used to apply the methods described in the book. Some of these tools are free, some have free student demos, some can be obtained at a reduced price. Anyway, they are all widely available and I encourage you to try them. To the very least, they can provide us with quantiles from probability distributions, and are much more handy and precise than using old-style statistical tables.

The manual is work-in-progress, so be sure to check back every now and then whether a new version has been posted.

This version is dated June 10, 2014

As usual, for comments, suggestions, and criticisms, my e-mail address is:

Paolo Brandimarte
paolo.brandimarte@polito.it

1

Quantitative Methods: Should We Bother?

1.1 SOLUTIONS

Problem 1.1 We consider the strategy of trying Plan A first and then Plan B; a more complete solution approach should rely on the decision tree framework of Chapter 13 (see Problem 13.1).

Imagine that we are at the end of year 1, and say that the first movie has been a success. If we try Plan B for the second movie, we invest 4000 now, and at the end of the second year we will make 6600 with probability $0.5 + \alpha$ and 0 with probability $1 - (0.5 + \alpha) = 0.5 - \alpha$. The expected NPV for this part of the strategy is

$$\text{NPV}_h = \frac{6600}{1.1} \times (0.5 + \alpha) - 4000 = 6000 \times \alpha - 1000;$$

note that here we are discounting the cash flow at the end of year 2 back to the end of year 1. We go on with Plan B only if this NPV is positive, i.e.,

$$\alpha \geq \frac{1000}{6000} = 0.1667.$$

On the other branch of the tree, the first movie has been a flop. The second movie, if we adopt plan B, yields the following NPV (discounted back to the end of year 1):

$$\text{NPV}_f = \frac{6600}{1.1} \times (0.5 - \alpha) - 4000 = -6000 \times \alpha - 5000.$$

This will always be negative for $\alpha \geq 0$, which makes sense: With 50–50 probabilities, the bet is not quite promising, and the situation does not improve after a first flop if this makes the odds even less favorable, and we dismiss the possibility of producing the second movie.

Let us step back to the root node (beginning of year 1), where we apply Plan A. There, the expected NPV is

$$\begin{aligned} & -2500 + 0.5 \times \frac{4400 + \text{NPV}_h}{1.1} + 0.5 \times \frac{0}{1.1} \\ & = -954.5455 + \alpha \times 2727.273, \end{aligned}$$

which is positive if

$$\alpha \geq \frac{954.5455}{2727.273} = 0.35.$$

This condition is more stringent that the previous one. Thus, the conditional probability of a second hit after the first one should not be less than 0.85 (which also implies that the the conditional probability of a hit after a first flop is 0.15).

Problem 1.2 Rather than extending the little numerical example of Chapter 1, let us state the model in general form (also see Chapter 12):

$$\begin{aligned}
\max \quad & \sum_{i=1}^{N} (p_i - c_i)x_i, \\
\text{s.t.} \quad & \sum_{i=1}^{N} r_{im}x_i \leq R_m, && m = 1, \ldots, M, \\
& 0 \leq x_i \leq d_i, && i = 1, \ldots, N,
\end{aligned}$$

where:

- Items are indexed by $i = 1, \ldots, N$
- Resources are indexed by $m = 1, \ldots, M$
- d_i, p_i, and c_i are demand, selling price, and production cost, respectively, for item i
- r_{im} is the unit requirement of resource m for item i, and R_m is the total availability of resource m

In this model, we have a single decision variable, x_i, representing what we produce *and* sell. If we introduce the possibility of third-party production, we can no longer identify production and sales. We need to change decision variables as follows:

- x_i is what we produce
- y_i is what we buy

We could also introduce a variable z_i to denote what we sell, but since $z_i = x_i + y_i$, we can avoid this.[1] Let us denote by $g_i > c_i$ the cost of purchasing item i from the third-party supplier. The model is now

$$\begin{aligned}
\max \quad & \sum_{i=1}^{N} \left[(p_i - c_i)x_i + (p_i - g_i)y_i \right] \\
\text{s.t.} \quad & \sum_{i=1}^{N} r_{im}x_i \leq R_m && m = 1, \ldots, M \\
& x_i + y_i \leq d_i && i = 1, \ldots, N \\
& x_i, y_i \geq 0 && i = 1, \ldots, N
\end{aligned}$$

[1]However, in multiperiod problems involving inventory holding we do need such a variable; see Chapter 12.

If we allow for overtime work, we change the first model by introducing the amount of overtime O_m on resource m, with cost q_m:

$$\begin{aligned}
\max \quad & \sum_{i=1}^{N} (p_i - c_i) x_i - \sum_{m=1}^{M} q_m O_m \\
\text{s.t.} \quad & \sum_{i=1}^{N} r_{im} x_i \le R_m + O_m, && m = 1, \ldots, M, \\
& 0 \le x_i \le d_i, && i = 1, \ldots, N, \\
& O_m \ge 0, && m = 1, \ldots, M.
\end{aligned}$$

These are just naive models used for introductory purposes. In practice, we should (at the very least) account for limitations on overtime work, as well as fixed charges associated with purchasing activities.

1.2 COMPUTATIONAL SUPPLEMENT: HOW TO SOLVE THE OPTIMAL MIX PROBLEM

In this section we show how different software tools can be used to solve the optimal mix problem of Section 1.1.2.

A first alternative is using MATLAB (see Section B.5 in the Appendix):

```
>> m = [45, 60];
>> reqs = [ 15 10; 15 35; 15 5; 25 15];
>> res = 2400*ones(4,1);
>> d = [100, 50];
>> x = linprog(-m, reqs, res, [], [], zeros(2, 1), d)
Optimization terminated.
x =
   73.8462
   36.9231
```

Unfortunately, with MATLAB we cannot solve integer programming problems. To this aim, we may use AMPL (see Appendix C). We need a model file and a data file, whose content is illustrated in Fig. 1.1. Using the CPLEX solver with AMPL, we find

```
ampl: model ProdMix.mod;
ampl: data ProdMix.dat;
ampl: solve;
CPLEX 11.1.0: optimal integer solution; objective 5505
2 MIP simplex iterations
0 branch-and-bound nodes
1 Gomory cut
ampl: display x;
x [*] :=
1  73
2  37
;
```

If we omit the `integer` keyword in the definition of decision variable `x`, we obtain the same solution as MATLAB.

```
# ProdMix.mod
param NumItems > 0;
param NumResources > 0;
param ProfitContribution{1..NumItems};
param MaxDemand{1..NumItems};
param ResReqs{1..NumItems, 1..NumResources};
param ResAvail{1..NumResources};

var x{i in 1..NumItems} >= 0, <= MaxDemand[i], integer;

maximize profit:
      sum {i in 1..NumItems} ProfitContribution[i] * x[i];

subject to Capacity {j in 1..NumResources}:
      sum {i in 1..NumItems} ResReqs[i,j] * x[i] <= ResAvail[j];
```

```
param NumItems := 2;
param NumResources := 4;
param: ProfitContribution MaxDemand :=
    1 45 100
    2 60 50;
param ResAvail := default 2400;
param ResReqs:
     1  2  3  4 :=
1   15 15 15 25
2   10 35  5 15;
```

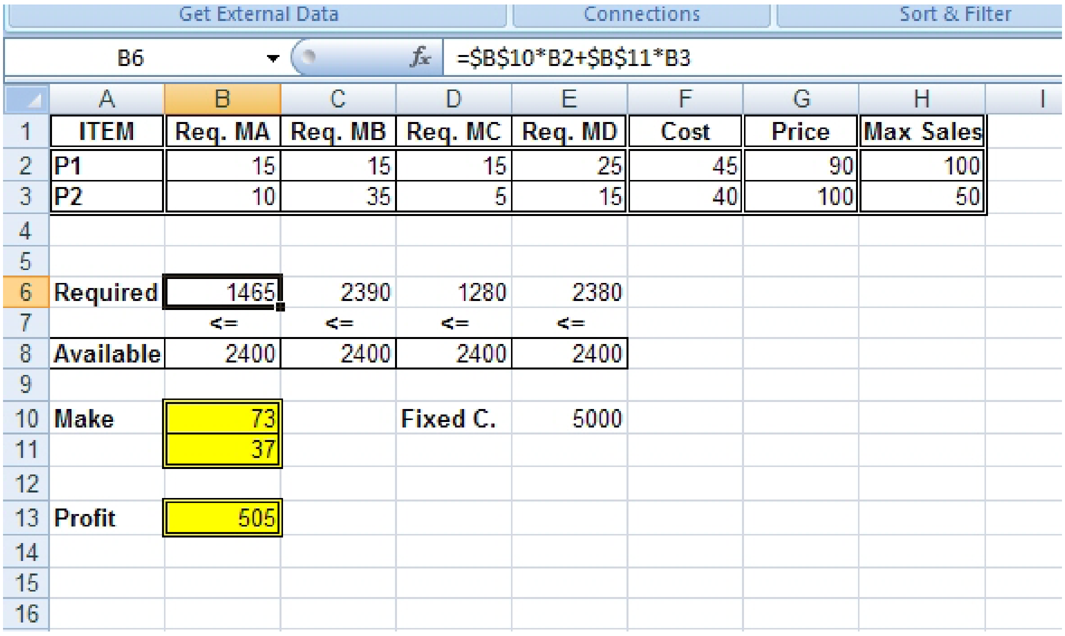
Fig. 1.1 AMPL model (`ProdMix.mod`) and data (`ProdMix.dat`) files for product mix optimization.

Another widespread tool that can be used to solve (small) LP and MILP models is Microsoft Excel, which is equipped by a Solver directly interfaced with the spreadsheet.[2] This is a double-edged sword, since it means that the model must be expressed in a two-dimensional array of cells, where data, constraints, and the objective function must be linked one to another by formulas.

The product mix problem can be represented as in the `ProdMix.xls` workbook, as shown in Fig. 1.2. The cell `Profit` contain the profit contribution and the cells `Required` are used to calculate the resource requirements as a function of the amounts produced, which are the contents of the cells `Make`. It is important to name ranges of cells to include them in the model in a readable way.

The model is described by opening the Solver window and specifying decision variables, constraints, and the objective cell as illustrated in Fig. fig:ProdMixExcel2. As you see, reference is made to named cell ranges; the cells containing `<=` in the worksheet have no meaning, actually, and are only included to clarify the model structure.

[2]You should make sure that the Solver was included in your Excel installation; sometimes, it is not included to save space on disk.



Get External Data | Connections | Sort & Filter

B6 =$B$10*B2+$B$11*B3

	A	B	C	D	E	F	G	H
1	ITEM	Req. MA	Req. MB	Req. MC	Req. MD	Cost	Price	Max Sales
2	P1	15	15	15	25	45	90	100
3	P2	10	35	5	15	40	100	50
4								
5								
6	Required	1465	2390	1280	2380			
7		<=	<=	<=	<=			
8	Available	2400	2400	2400	2400			
9								
10	Make	73		Fixed C.	5000			
11		37						
12								
13	Profit	505						
14								
15								
16								

Fig. 1.2 The `ProdMix.xls` workbook to solve the optimal product mix problem.

As one can imagine, describing and maintaining a large-scale model in this form may quickly turn into nightmare (not taking into account the fact that state-of-the-art solvers are needed to solve large problem instances). Nevertheless, Excel can be used to solve small scale models and is in fact the tool of choice of many Management Science books. We should mention, however, that the true power of Excel is its integration with VBA (Visual Basic for Application), a powerful programming language that can be used to develop remarkable applications. A clever strategy is to use Excel as a familiar and user-friendly interface, and VBA to build a link with state-of-the-art software libraries.

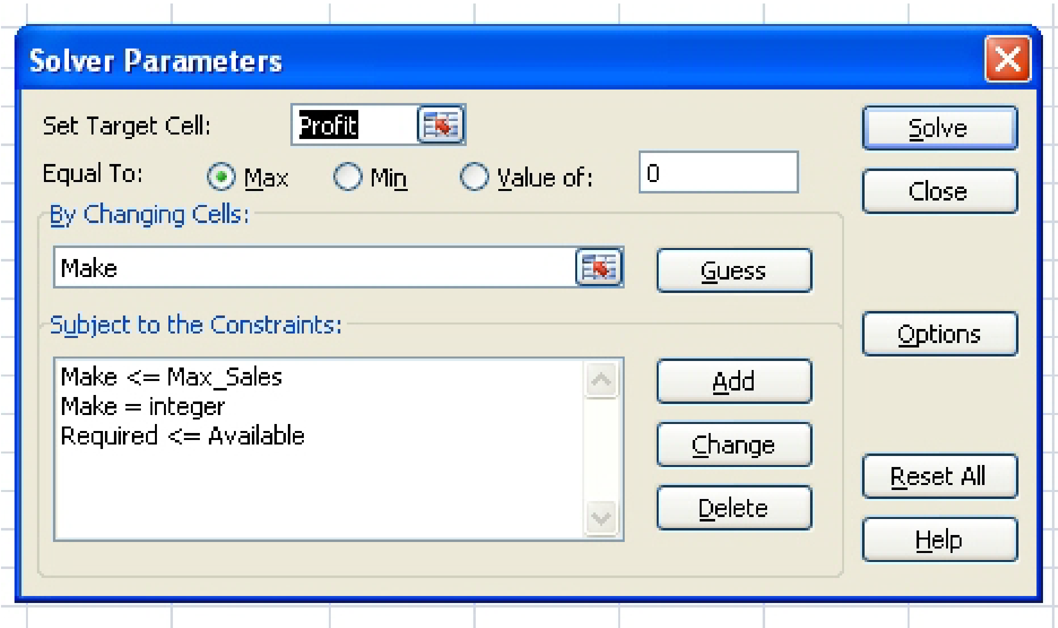


Fig. 1.3 Caption for ProdMixExcel2

2

Calculus

2.1 SOLUTIONS

Problem 2.1 A first requirement for function $f(x)$ is that the argument of the square root is positive:

$$1 - x^2 \geq 0 \quad \Rightarrow \quad -1 \leq x \leq 1$$

Then, the denominator of the ratio cannot be zero:

$$\sqrt{1 - x^2} \neq 1 \quad \Rightarrow \quad x \neq 0$$

Then, the domain of f is $[-1, 0) \cup (0, 1]$.

The square root in function $g(x)$ is not an issue, as $x^2 + 1 \neq 0$. We also observe that the denominator is never zero, since

$$\sqrt{x^2 + 1} = x \quad \Rightarrow \quad x^2 + 1 = x^2 \quad \Rightarrow \quad 1 = 0$$

which is false. Then, the domain of g is the whole real line.

Problem 2.2 The first line is easy to find using the form $y = mx + q$

$$y = -3x + 10$$

For the second one, we use the form $y - y_0 = m(x - x_0)$

$$y - 4 = 5(x + 2) \quad \Rightarrow \quad y = 5x + 14$$

For the third line, we observe that its slope is

$$m = \frac{3 - (-5)}{1 - 3} = -4$$

Then we have

$$y - 3 = -4(x - 1) \quad \Rightarrow \quad y = -4x + 7$$

Alternatively, we might also consider its parametric form

$$\begin{cases} y = \lambda y_a + (1-\lambda)y_b = 3\lambda - 5(1-\lambda) \\ x = \lambda x_a + (1-\lambda)x_b = \lambda + 3(1-\lambda) \end{cases}$$

and eliminate λ between the two equations. This approach is less handy, but it stresses the idea of a line as the set of *affine combinations* of two vectors. An affine combination of vectors is a linear combination whose weights add up to one (see Chapter 3).

Problem 2.3

$$f_1'(x) = \frac{3\cdot(x^2+1) - 3x\cdot 2x}{(x^2+1)^2} = \frac{3(3x^2+1)}{(x^2+1)^2}$$

$$f_2'(x) = (3x^2 - 2x + 5)e^{x^3-x^2+5x-3}$$

$$f_3'(x) = \frac{1}{2\sqrt{\exp\left(\frac{x+2}{x-1}\right)}} \cdot \exp\left(\frac{x+2}{x-1}\right) \cdot \frac{-3}{(x-1)^2}$$

Problem 2.4 Let us start with $f(x) = x^3 - x$. We observe the following:

- Limits:

$$\lim_{x\to-\infty} f(x) = -\infty, \qquad \lim_{x\to+\infty} f(x) = +\infty$$

- Roots: we have $f(x) = 0$ for

$$x(x^2-1) = 0 \Rightarrow x = 0,\ x = \pm 1$$

- The first order derivative $f'(x) = 3x^2 - 1$ is zero for $x = \pm 1/\sqrt{3} \approx \pm 0.5774$, positive for $x < -1/\sqrt{3}$ and $x > 1/\sqrt{3}$, negative otherwise. Hence, the function is increasing (from $-\infty$) for $x < -1/\sqrt{3}$, decreasing for $-1/\sqrt{3} < x < 1/\sqrt{3}$, and then it increases to $+\infty$.

- The second order derivative $f''(x) = 6x$ is negative for negative x and positive for positive x; hence, the function is concave for $x < 0$ (with a maximum at $x = -1/\sqrt{3}$) and convex for $x > 0$ (with a minimum at $x = 1/\sqrt{3}$).

For function $g(x) = x^3 + x$ the analysis is similar, but now the first-order derivative $f'(x) = 3x^2 + 1$ is always positive and the function has a unique root at $x = 0$ (and neither minima nor maxima).

See the plots in Fig. 2.1.

Problem 2.5

1. For function $f_1(x)$, we observe that the function is continuous at $x = 0$, as

$$f_1(0_-) = 0 = 0 = f_1(0_+)$$

   but not differentiable, as

$$f_1'(0_-) = -1 \neq 0 = f_1'(0_+)$$

2. For function $f_1(x)$, we observe that the function is not continuous at $x = 0$, as

$$f_2(0_-) = 1 \neq 0 = f_2(0_+)$$

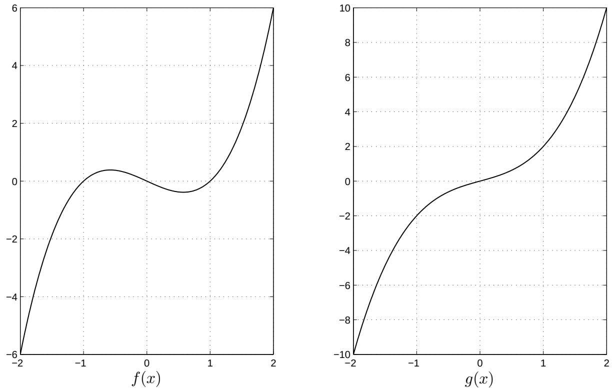


Fig. 2.1 Plots of functions in Problem 2.4.

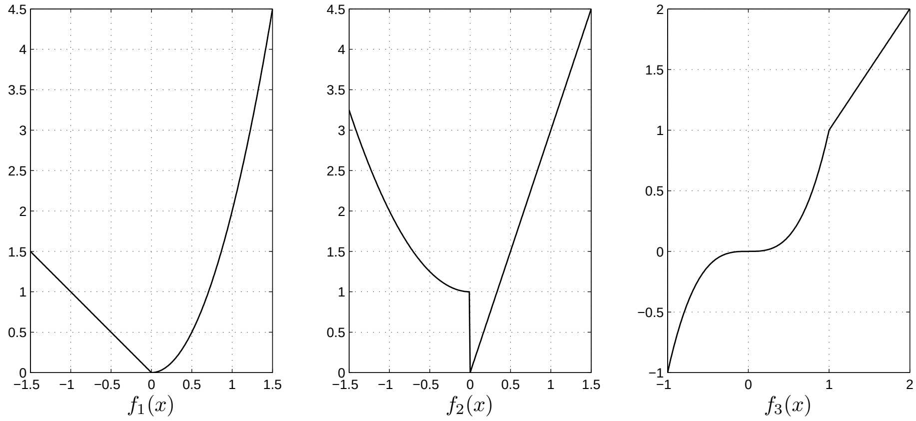


Fig. 2.2 Plots of functions in Problem 2.5.

Then, the function cannot be differentiable.

3. For function $f_3(x)$, we observe that the function is continuous at $x = 1$, as

$$f_3(1_-) = 1 = 1 = f_3(1_+)$$

but not differentiable, as

$$f_3'(1_-) = 3 \neq 1 = f_3'(1_+)$$

See the plots in Fig. 2.2.

Problem 2.6 Consider function

$$f(x) = \exp\left(-\frac{1}{1+x^2}\right)$$

and find linear (first-order) and quadratic (second-order) approximations around points $x_0 = 0$ and $x_0 = 10$. Check the quality of approximations around these points.

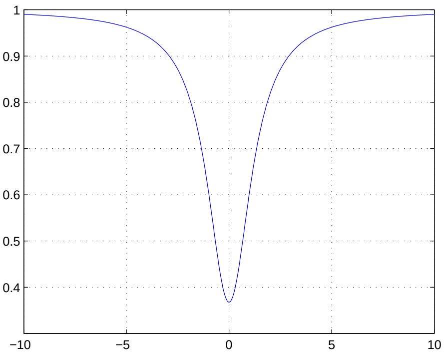


Fig. 2.3 Plot of function in Problem 2.6.

We have

$$f(0) = 0.3678794411714 42,\ f(10) = 0.990147863338053$$

The function is plotted in Fig. 2.3.

Let us find and evaluate the first-order derivative

$$f'(x) = \exp\left(-\frac{1}{1+x^2}\right)\frac{2x}{(1+x^2)^2},$$
$$f'(0) = 0,\ f'(10) = 0.001941276077518$$

Then the second-order derivative

$$\begin{aligned} f''(x) &= \exp\left(-\frac{1}{1+x^2}\right)\left[\frac{2x}{(1+x^2)^2}\right]^2 + \exp\left(-\frac{1}{1+x^2}\right)\frac{2\cdot(1+x^2)^2 - 2x\cdot 2(1+x^2)\cdot 2x}{(1+x^2)^4} \\ &= \exp\left(-\frac{1}{1+x^2}\right)\left[\frac{4x^2}{(1+x^2)^4} + \frac{2-6x^2}{(1+x^2)^3}\right] \\ &= \exp\left(-\frac{1}{1+x^2}\right)\frac{2-6x^4}{(1+x^2)^4} \end{aligned}$$
$$f''(0) = 0.735758882342885,\ f''(10) = -5.708885506270199\cdot 10^{-4}$$

The Taylor expansions $p_{n,x_0}(x)$, where n is the order and x_0 is where the approximation is built, are:

$$\begin{aligned} p_{1,0}(x) &= 0.367879441171442 \\ p_{2,0}(x) &= 0.367879441171442 + \frac{1}{2}\cdot 0.735758882342885\cdot x^2 \\ p_{1,10}(x) &= 0.990147863338053 + 0.001941276077518\cdot(x-10) \\ p_{2,10}(x) &= 0.990147863338053 + 0.001941276077518\cdot(x-10) \\ &\quad - \frac{1}{2}\cdot 0.0005708885506270199\cdot(x-10)^2 \end{aligned}$$

The four approximations are plotted in Figs. 2.4(a), (b), (c) and (d), respectively, where the function plot is the dashed line.

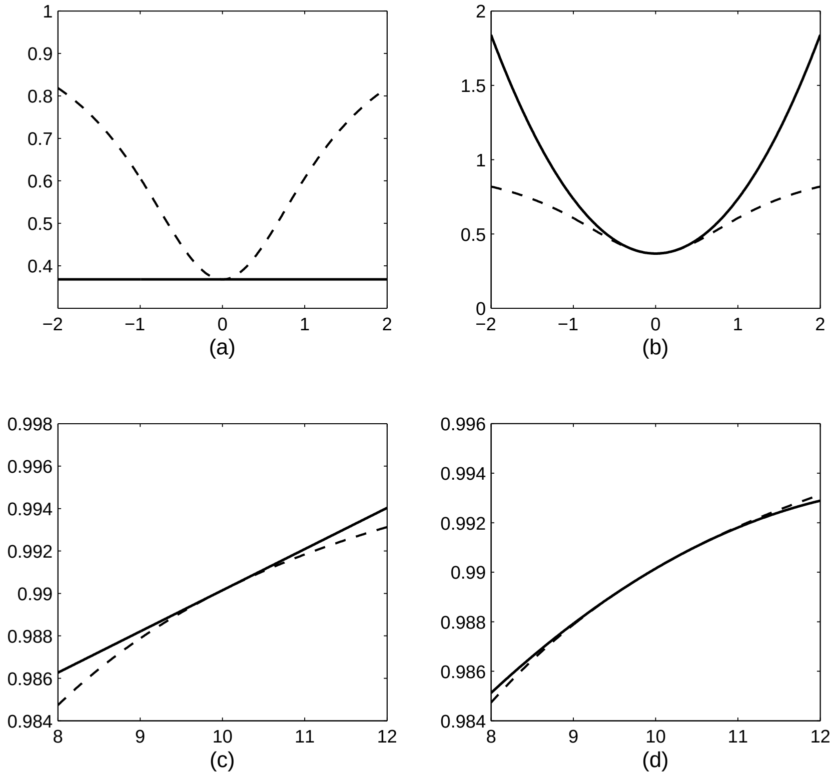


Fig. 2.4 Plot of approximations of Problem 2.6.

Problem 2.7 Let us express the bond price in terms of the continuous-time yield y_c:

$$P(y_c) = \sum_{t=1}^{T-1} Ce^{-y_c t} + (C+F)e^{-y_c T}$$

and take its first-order derivative

$$P'(y_c) = -\sum_{t=1}^{T-1} Cte^{-y_c t} - (C+F)Te^{-y_c T} = -\sum_{t=1}^{T} td_t$$

where d_t is the discounted cash flow at time t, i.e., $d_t = Ce^{-y_c t}$ for $t = 1, \ldots, T-1$ and $d_T = (C+F)e^{-y_c T}$. We see that this expression, unlike the case with discrete-time compounding, does not contain any extra-term involving yield. Hence, we may write

$$P'(y_c) = \frac{dP}{dy_c} = -\sum_{t=1}^{T} td_t \cdot \frac{\sum_{t=1}^{T} d_t}{\sum_{t=1}^{T} d_t} = -D \cdot P$$

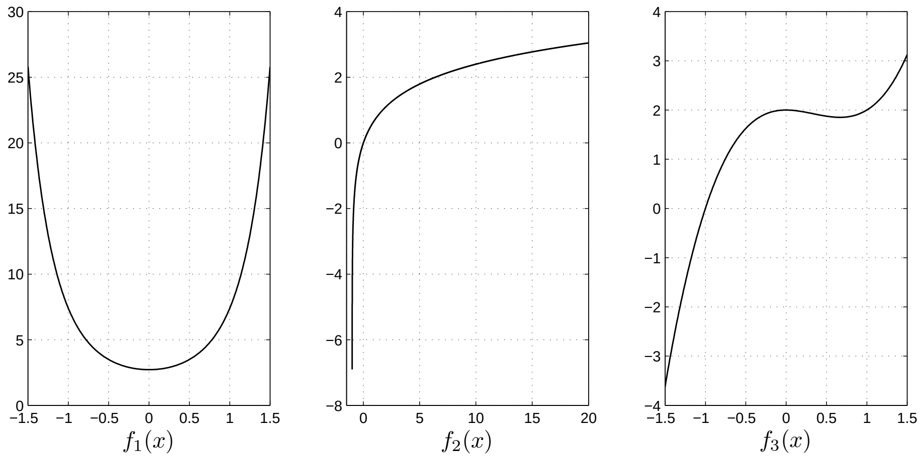


Fig. 2.5 Plots of functions in Problem 2.8.

where

$$D \equiv \frac{\sum_{t=1}^{T} t d_t}{\sum_{t=1}^{T} d_t}$$

is duration. Then, we may express relative price variations for small changes in yield as

$$\frac{\Delta P}{P} = -D \cdot \Delta y_c$$

where we use duration, rather than modified duration.

Problem 2.8 For function f_1 we have

$$f_1'(x) = 2xe^{x^2+1}, \quad f_1''(x) = 2e^{x^2+1} + 4x^2e^{x^2+1} = 2e^{x^2+1}(1+x^2) > 0$$

Then, the function is convex on the real line.

For function f_2, with domain $x > -1$, we have

$$f_2'(x) = \frac{1}{x+1}, \quad f_2''(x) = \frac{-1}{(x+1)^2} > 0$$

Then, the function is concave on its domain.

For function f_3 we have

$$f_3'(x) = 3x^2 - 2x, \quad f_3''(x) = 6x - 2$$

Since the second order derivative changes its sign at $x = 1/3$, the function neither convex nor concave.

See Fig. 2.5.

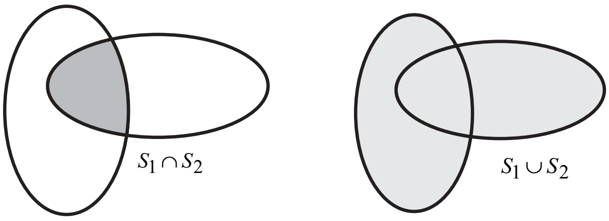


Fig. 2.6 Intersection and union of two convex sets.

Problem 2.9 We must prove that if $\mathbf{x}_a, \mathbf{x}_b \in S_1 \cap S_2$, then $\mathbf{x}_\lambda = \lambda \mathbf{x}_a + (1-\lambda)\mathbf{x}_b \in S_1 \cap S_2$, for any $\lambda \in [0, 1]$.

Now consider two elements $\mathbf{x}_a, \mathbf{x}_b \in S_1 \cap S_2$. Since S_1 and S_2 are both convex, we know that, for any $\lambda \in [0, 1]$,

$$\lambda \mathbf{x}_a + (1-\lambda)\mathbf{x}_b \in S_1$$
$$\lambda \mathbf{x}_a + (1-\lambda)\mathbf{x}_b \in S_2$$

But this shows $\mathbf{x}_\lambda \in S_1 \cap S_2$.

This property is visualized in Fig. 2.6; note that the union of convex sets need not be convex.

Problem 2.10 From Section 2.12 we know that the present value V_0, at time $t = 0$, of a stream of constant cash flows $C_t = C$, $t = 1, \ldots, T$, is

$$V_0 = \frac{C}{r}\left[1 - \frac{1}{(1+r)^T}\right]$$

To get the future value V_T, at time $t = T$, we just multiply by a factor $(1+r)^T$, which yields

$$V_T = \frac{C}{r}\left[(1+r)^T - 1\right]$$

Problem 2.11 You work for $T_s = 40$ years saving S per year; then you live $T_c = 20$ years, consuming $C = 20000$ per year. The cumulated wealth when you retire is

$$\frac{S}{r}\left[(1+r)^{T_s} - 1\right]$$

and the present value of the consumption stream is

$$\frac{C}{r}\left[1 - \frac{1}{(1+r)^{T_c}}\right]$$

Equating these two expressions we find

$$S = \frac{C\left[1 - \dfrac{1}{(1+r)^{T_c}}\right]}{(1+r)^{T_s} - 1} = \frac{20000 \times \left[1 - \frac{1}{1.05^{20}}\right]}{(1.05)^{40} - 1} = 2063.28$$

If $T_c = 10$, S is reduced to 1278.44

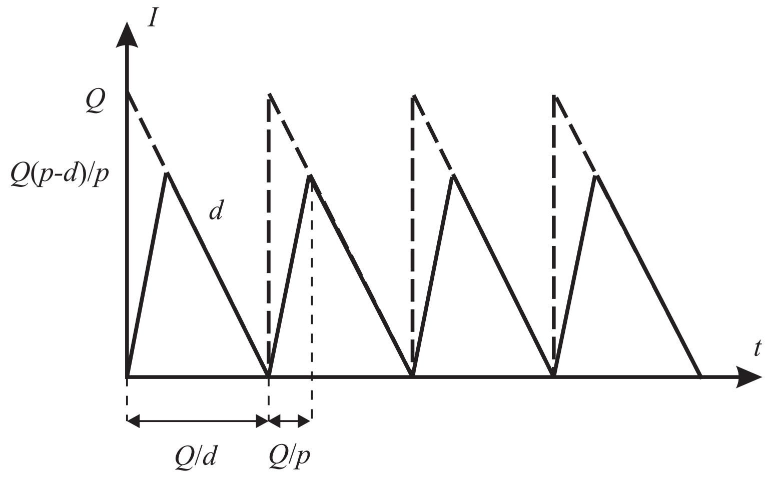


Fig. 2.7 Inventory with finite rate of replenishment.

Problem 2.12 The finite replenishment rate p is the number of items delivered per unit of time. When the inventory level reaches zero, it does not immediately increase by Q units but increases progressively at rate $p - d$, as shown in Fig. 2.7; this rate is the difference between the rates of item inflow and outflow. It takes Q/p time units to complete the production lot Q; hence, when the lot is completed, the inventory level has reached a level

$$(p - d)\, Q/p$$

which is the height of the triangle corresponding to one cycle. Then, inventory decreases between $(p - d)\, Q/p$ and 0 at rate d. With respect to the EOQ model, there is a difference in the average inventory level, which is now

$$\frac{(p - d)Q}{2p}$$

Then, the total cost function is

$$\frac{Ad}{Q} + h \cdot \frac{(p - d)Q}{2p}$$

and using the same drill as the EOQ case (set first-order derivative to 0) we find

$$Q^* = \sqrt{\frac{2Ad}{h} \cdot \frac{p}{p - d}}$$

It is interesting to note that when $p \to +\infty$ we get back to the EOQ formula.

3

Linear Algebra

3.1 SOLUTIONS

Problem 3.1 Gaussian elimination works as follows:

$$\left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 1 & 0 & 4 & 9 \\ 0 & 2 & 1 & 0 \end{array}\right] \overset{E_2 \leftarrow E_2 - E_1}{\Longrightarrow} \left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -2 & 5 & 12 \\ 0 & 2 & 1 & 0 \end{array}\right] \overset{E_3 \leftarrow E_3 + E_2}{\Longrightarrow} \left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -2 & 5 & 12 \\ 0 & 0 & 6 & 12 \end{array}\right]$$

Using backsubstitution:

$$\begin{aligned} 6x_3 &= 12 && \rightarrow && x_3 = 2 \\ -2x_2 + 5x_3 &= 12 && \rightarrow && x_2 = \frac{12 - 5x_3}{-2} = -1 \\ x_1 + 2x_2 - x_3 &= 3 && \rightarrow && x_1 = 3 - 2x_2 + x_3 = 1 \end{aligned}$$

To apply Cramer's rule we compute the determinant of the matrix

$$\Delta = \begin{vmatrix} 1 & 2 & -1 \\ 1 & 0 & 4 \\ 0 & 2 & 1 \end{vmatrix} = 1 \times \begin{vmatrix} 0 & 4 \\ 2 & 1 \end{vmatrix} - 2 \times \begin{vmatrix} 1 & 0 \\ 4 & 1 \end{vmatrix} - 1 \times \begin{vmatrix} 1 & 0 \\ 0 & 2 \end{vmatrix}$$

$$= 1 \times (-8) - 2 \times 1 - 2 \times 2 = -12$$

By a similar token

$$\Delta_1 = \begin{vmatrix} -3 & 2 & -1 \\ 9 & 0 & 4 \\ 0 & 2 & 1 \end{vmatrix} = -12$$

$$\Delta_2 = \begin{vmatrix} 1 & -3 & -1 \\ 1 & 9 & 4 \\ 0 & 0 & 1 \end{vmatrix} = 12$$

$$\Delta_3 = \begin{vmatrix} 1 & 2 & -3 \\ 1 & 0 & 9 \\ 0 & 2 & 0 \end{vmatrix} = -24$$

By the way, it is sometimes convenient to develop the determinant not using the first row, but any row or column with few nonzero entries; for instance

$$\Delta_2 = 1 \times \begin{vmatrix} 1 & -3 \\ 1 & 9 \end{vmatrix}$$

$$\Delta_3 = -2 \times \begin{vmatrix} 1 & -3 \\ 1 & 9 \end{vmatrix}$$

Then, we find

$$x_1 = \frac{\Delta_1}{\Delta} = 1, \quad x_2 = \frac{\Delta_2}{\Delta} = -1, \quad x_3 = \frac{\Delta_3}{\Delta} = 2$$

Problem 3.2 Let us consider polynomials of degree up to n

$$a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

Such a polynomial may be expressed as a vector $\mathbf{a} \in \mathbb{R}^{n+1}$

$$\mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$$

where we associate each monomial x^k, $k = 0, 1, 2, \ldots, n$ with a unit vector:

$$\mathbf{e}_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \ldots \mathbf{e}_n = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

We know that

$$(x^k)' = kx^{k-1}$$

Therefore, the mapping from a monomial to its derivative may be represented as follows

$$\mathbf{e}_k = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{e}'_k = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ k \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

If we align such vectors, we obtain the following matrix

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 2 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & n-1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & n \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

For instance, consider the polynomial

$$p(x) = 3 + 5x + 2x^2 - x^3 + 2x^4 \quad \Rightarrow \quad p'(x) = 5 + 4x - 3x^2 + 8x^3$$

The mapping is

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 5 \\ 2 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \\ -3 \\ 8 \\ 0 \end{bmatrix}$$

We may also observe that the matrix is not invertible, which makes sense, since two polynomials that differ only in the constant term have the same derivative.

Problem 3.3 Prove that the representation of a vector using a basis is unique.

Let us assume that, contrary to the statement, we have two representations of the same vector using the same basis (which is a set of linearly independent vectors:

$$\mathbf{v} = \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2 + \cdots + \alpha_n \mathbf{e}_n$$
$$\mathbf{v} = \beta_1 \mathbf{e}_1 + \beta_2 \mathbf{e}_2 + \cdots + \beta_n \mathbf{e}_n$$

Taking the difference, we have

$$\mathbf{0} = (\alpha_1 - \beta_1)\mathbf{e}_1 + (\alpha_2 - \beta_2)\mathbf{e}_2 + \cdots + (\alpha_n - \beta_n)\mathbf{e}_n$$

However, since the vectors $\mathbf{e}_k$, $k = 1, \ldots, n$, are a basis, there is no way to find a linear combination of them yielding the null vector, unless all of the coefficients in the linear combination are all zero, which implies

$$\alpha_k = \beta_k, \quad k = 1, \ldots, n$$

Problem 3.4 Let $\mathbf{B} = \mathbf{AD}$, where $\mathbf{B} \in \mathbb{R}^{m,n}$ and we denote its generic element b_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$. To obtain b_{ij}, we multiply elements of row i of $\mathbf{A}$ by elements of column j of $\mathbf{D}$:

$$b_{ij} = \sum_{k=1}^{n} a_{ik} d_{kj}$$

We may think of this element as the inner product between the row vector $\mathbf{a}_i^T$ and the column vector $\mathbf{d}_j$. However, we have $d_{kj} = 0$ for $j \neq k$, so

$$b_{ij} = a_{ik} d_{kk}$$

i.e., element k of row i of matrix $\mathbf{A}$ is multiplied by the corresponding element d_{kk} on the diagonal of $\mathbf{D}$:

$$\mathbf{AD} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 6 & 4 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 7 \end{bmatrix} = \begin{bmatrix} 2 & -9 & 35 \\ 4 & -18 & 28 \end{bmatrix}$$

Problem 3.5 Multiplying the matrices, we find:

$$\mathbf{AX} = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 1 \\ 2 & 0 & 2 \end{bmatrix} \begin{bmatrix} 6 & 5 & 7 \\ 2 & 2 & 4 \\ 3 & 3 & 6 \end{bmatrix} = \begin{bmatrix} 12 & 11 & 19 \\ 5 & 5 & 10 \\ 18 & 16 & 26 \end{bmatrix}$$
$$\mathbf{BX} = \begin{bmatrix} 1 & 3 & 0 \\ 0 & 4 & -1 \\ 2 & 3 & 0 \end{bmatrix} \begin{bmatrix} 6 & 5 & 7 \\ 2 & 2 & 4 \\ 3 & 3 & 6 \end{bmatrix} = \begin{bmatrix} 12 & 11 & 19 \\ 5 & 5 & 10 \\ 18 & 16 & 26 \end{bmatrix}$$

This implies that, unlike with the scalar case, we cannot simplify an equation $\mathbf{AX} = \mathbf{BX}$ to $\mathbf{A} = \mathbf{B}$. If matrix $\mathbf{X}$ is invertible, then we may postmultiply by its inverse $\mathbf{X}^{-1}$ and simplify. But in the example $\mathbf{X}$ is singular; to see this, observe that its second and third row are linearly dependent, as they are obtained multiplying the vector $[1, 1, 2]$ by 2 and 3, respectively.

Problem 3.6 Given the matrix $\mathbf{H} = \mathbf{I} - 2\mathbf{hh}^T$, we may check orthogonality directly:

$$\begin{aligned} \mathbf{H}^T\mathbf{H} &= (\mathbf{I} - 2\mathbf{hh}^T)(\mathbf{I} - 2\mathbf{hh}^T) \\ &= \mathbf{I} - 2\mathbf{hh}^T - 2\mathbf{hh}^T + 4\mathbf{h}(\mathbf{h}^T\mathbf{h})\mathbf{h}^T \\ &= \mathbf{I} - 4\mathbf{hh}^T + 4\mathbf{hh}^T \\ &= \mathbf{I} \end{aligned}$$

where we exploit the symmetry of $\mathbf{H}$ and the condition $\mathbf{h}^T\mathbf{h} = 1$.

This transformation is actually a reflection of a generic vector with respect to a hyperplane passing through the origin and characterized by an orthogonal vector $\mathbf{h}$. To see this, consider the application of $\mathbf{H}$ to vector $\mathbf{v}$:

$$\mathbf{Hv} = \mathbf{Iv} - 2\mathbf{hh}^T\mathbf{v} = \mathbf{v} - 2\alpha\mathbf{h}$$

where $\alpha = \mathbf{h}^T\mathbf{v}$ is the length of the projection of $\mathbf{v}$ on the unit vector $\mathbf{h}$. To illustrate in the plane, consider $\mathbf{v} = [3, 1]^T$ and $\mathbf{h} = [1, 0]^T$:

$$\mathbf{Hv} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} - 2 \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \end{bmatrix} \right) = \begin{bmatrix} 3 \\ 1 \end{bmatrix} - 2 \times 3 \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} -3 \\ 1 \end{bmatrix}$$

The resulting vector is indeed the reflection of $\mathbf{v}$ with respect to the horizontal axis, which is a line going through the origin and orthogonal to vector $\mathbf{h}$.

This implies that $\mathbf{H}$ is a rotation matrix, and we know that rotation matrices are orthogonal.

Problem 3.7 To prove these results, it is convenient to regard a matrix $\mathbf{J}_n$ whose elements are all 1 as the product

$$\mathbf{1}_n\mathbf{1}_n^T$$

where $\mathbf{1}_n = [1, 1, 1, \ldots, 1]^T \in \mathbb{R}^n$ is a column vector of ones. Then

$$\begin{aligned}
\mathbf{x}^T\mathbf{C} &= \mathbf{x}^T \left(\mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T \right) \\
&= \mathbf{x}^T\mathbf{I}_n - \frac{1}{n}\mathbf{x}^T\mathbf{1}_n\mathbf{1}_n^T \\
&= \mathbf{x}^T - \left(\frac{1}{n}\sum_{k=1}^{n} x_k \right) \mathbf{1}_n^T \\
&= [x_1, x_2, x_3, \ldots, x_n]^T - [\bar{x}, \bar{x}, \bar{x}, \ldots, \bar{x}]^T \\
&= [x_1 - \bar{x}, x_2 - \bar{x}, x_3 - \bar{x}, \ldots, x_n - \bar{x}]^T
\end{aligned}$$

where we exploit the fact

$$\mathbf{x}^T\mathbf{1}_n = \sum_{k=1}^{n} x_k = n\bar{x}$$

As to the second statement, we recall that

$$\sum_{k=1}^{n}(x_k - \bar{x})^2 = \sum_{k=1}^{n} x_k^2 - n\bar{x}^2$$

Moreover

$$\begin{aligned}
\mathbf{x}^T\mathbf{C}\mathbf{x} &= \mathbf{x}^T \left(\mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T \right) \mathbf{x} \\
&= \mathbf{x}^T\mathbf{x} - \frac{1}{n}(\mathbf{x}^T\mathbf{1}_n)(\mathbf{1}_n^T\mathbf{x}) \\
&= \sum_{k=1}^{n} x_k^2 - \frac{1}{n}\left(\sum_{k=1}^{n} x_k\right)\left(\sum_{k=1}^{n} x_k\right) \\
&= \sum_{k=1}^{n} x_k^2 - n\bar{x}^2
\end{aligned}$$

Problem 3.8 The determinant of a diagonal matrix $\mathbf{D}$ may be developed by rows; since only one element in the first row is nonzero, we have

$$\det(\mathbf{D}) = \begin{vmatrix} d_1 & & & & \\ & d_2 & 0 & & \\ & & d_3 & & \\ & & & \ddots & \\ & & & & d_n \end{vmatrix} = d_1 \cdot \begin{vmatrix} d_2 & 0 & & \\ & d_3 & & \\ & & \ddots & \\ & & & d_n \end{vmatrix}$$

Repeating the scheme recursively, we have

$$\det(\mathbf{D}) = \prod_{j=1}^{n} d_j$$

The reasoning is the same for a lower triangular matrix, whereas for an upper triangular triangular it is convenient to start from the last row:

$$\det(\mathbf{U}) = \begin{vmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1,n-1} & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2,n-1} & u_{2n} \\ & & u_{33} & \cdots & u_{3,n-1} & u_{3n} \\ & & & \ddots & \vdots & \vdots \\ & & & & u_{n-1,n-1} & u_{n-1,n} \\ & & & & & u_{nn} \end{vmatrix} = u_{nn} \cdot \begin{vmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1,n-1} \\ & u_{22} & u_{23} & \cdots & u_{2n-1} \\ & & u_{33} & \cdots & u_{3,n-1} \\ & & & \ddots & \vdots \\ & & & & u_{n-1,n-1} \end{vmatrix}$$

Going on recursively, we find

$$\det(\mathbf{U}) = \prod_{j=1}^{n} u_{jj}$$

Problem 3.9 Of course we may apply the standard approach based on minors, but sometimes shortcuts are possible. For instance, the case of a diagonal matrix is easy:

$$\mathbf{A}_1^{-1} = \begin{bmatrix} \frac{1}{6} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & -\frac{1}{5} \end{bmatrix}$$

To deal with the second case, observe that, if denote the generic element of $\mathbf{A}_2^{-1}$ by b_{ij}, we must have $\mathbf{A}_2\mathbf{A}_2^{-1} = \mathbf{I}$, i.e.,

$$\begin{bmatrix} 0 & 0 & 5 \\ 0 & 2 & 0 \\ 3 & 0 & 0 \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} 5b_{31} & 5b_{32} & 5b_{33} \\ 2b_{21} & 2b_{22} & 2b_{23} \\ 3b_{11} & 3b_{12} & 3b_{13} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

If we equate element by element, we find

$$5b_{31} = 1, \quad 2b_{22} = 1, \quad 3b_{13} = 1$$

whereas all the remaining elements of the inverse are zero. Hence

$$\mathbf{A}_2^{-1} = \begin{bmatrix} 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ \frac{1}{5} & 0 & 0 \end{bmatrix}$$

With respect to the diagonal case, we have a permutation of elements.

For the last case, let us apply the standard procedure. The first step is easy:

$$\det(\mathbf{A}_3) = 2$$

Then we must find the adjoint matrix $\tilde{\mathbf{A}}$ of $\mathbf{A}_3$. Since its element $\tilde{a}_{ij}$ is the cofactor C_{ji}, we may transpose the matrix and find a sequence of 2×2 determinants:

$$\mathbf{B} = \mathbf{A}_3^T = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Now, to find $\tilde{a}_{11}$ we just eliminate the first row and the first column of $\mathbf{B}$, obtaining a 2×2 determinant:

$$\tilde{a}_{11} = (-1)^{1+1} \begin{vmatrix} 1 & 0 \\ 1 & 1 \end{vmatrix} = 1$$

By a similar token:

$$\tilde{a}_{12} = (-1)^{1+2} \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} = -1$$

Here we cross the first row and the second column, and change the sign to the resulting determinant. In the second row of $\tilde{\mathbf{A}}$ there is a different pattern, as the first element has a change in sign, and so on. This yields

$$\tilde{\mathbf{A}} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix}$$

and

$$\mathbf{A}_3^{-1} = \frac{1}{2}\tilde{\mathbf{A}} = \begin{bmatrix} 0.5 & -0.5 & 0.5 \\ 0.5 & 0.5 & -0.5 \\ -0.5 & 0.5 & 0.5 \end{bmatrix}$$

Problem 3.10 Let us interpret $\mathbf{Ax}$ as a linear combination of columns $\mathbf{A}_j$ of $\mathbf{A}$ with weights x_j:

$$\mathbf{Ax} = \begin{bmatrix} \vdots & \vdots & & \vdots \\ \mathbf{A}_1 & \mathbf{A}_2 & \cdots & \mathbf{A}_n \\ \vdots & \vdots & & \vdots \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1\mathbf{A}_1 + x_2\mathbf{A}_2 + \cdots + x_n\mathbf{A}_n$$

If this linear combination yields the null vector $\mathbf{0}$, but $\mathbf{x} \neq \mathbf{0}$, then the columns of $\mathbf{A}$ are not linearly independent; hence, the matrix is singular.

We say that $\mathbf{x}$ is in the *null space* of $\mathbf{A}$; note that this implies that any vector $\lambda\mathbf{x}$, for any real number λ is in this null space, too. Now consider a vector $\mathbf{z}$ such that $\mathbf{Az} = \mathbf{b}$, and imagine that you wish to invert the mapping. It is easy to see that this is impossible, since

$$\mathbf{A}(\mathbf{z} + \lambda\mathbf{x}) = \mathbf{b}$$

as well, for any λ. The mapping cannot be inverted, matrix $\mathbf{A}$ cannot be inverted, and it is singular.

Problem 3.11 We should find a vector $\mathbf{x} \neq \mathbf{0}$ such that

$$(\mathbf{hh}^T - \mathbf{h}^T\mathbf{hI})\mathbf{x} = \mathbf{h}(\mathbf{h}^T\mathbf{x}) - (\mathbf{h}^T\mathbf{h})\mathbf{x} = \mathbf{0}$$

(see Problem 3.10). Note that the terms between parentheses in the above expression are actually scalars. We have

$$\mathbf{h}^T\mathbf{x} = \sum_{i=1}^{n} h_i x_i, \qquad \mathbf{h}(\mathbf{h}^T\mathbf{x}) = \begin{bmatrix} h_1\left(\sum_{i=1}^n h_i x_i\right) \\ h_2\left(\sum_{i=1}^n h_i x_i\right) \\ \vdots \\ h_n\left(\sum_{i=1}^n h_i x_i\right) \end{bmatrix}$$

$$\mathbf{h}^T\mathbf{h} = \sum_{i=1}^{n} h_i^2, \qquad (\mathbf{h}^T\mathbf{h})\mathbf{x} = \begin{bmatrix} x_1\left(\sum_{i=1}^n h_i^2\right) \\ x_2\left(\sum_{i=1}^n h_i^2\right) \\ \vdots \\ x_n\left(\sum_{i=1}^n h_i^2\right) \end{bmatrix}$$

Therefore, we have a system of linear equations with the following form:

$$h_j \left(\sum_{i=1}^{n} h_i x_i \right) - x_j \left(\sum_{i=1}^{n} h_i^2 \right), \quad j = 1, \ldots, n$$

Looking at this expression, we see that the non-zero vector

$$x_j = \frac{h_j}{\sum_{i=1}^{n} h_i^2}, \quad j = 1, \ldots, n$$

is in fact a solution of the system. Hence, the matrix is singular.

Problem 3.12 Consider two nonnull orthogonal vectors $\mathbf{x}$ and $\mathbf{y}$. If they are linearly dependent, then we may write $\mathbf{x} = \alpha \mathbf{y}$, for some real number $\alpha \neq 0$. Then, orthogonality implies

$$\mathbf{x}^T \mathbf{y} = \alpha \mathbf{y}^T \mathbf{y} = \alpha \parallel \mathbf{y} \parallel^2 = 0$$

But, given the properties of vector norms, this is possible only if $\mathbf{y} = \mathbf{0}$, which contradicts the hypotheses. Since we have a contradiction, we conclude that $\mathbf{x}$ and $\mathbf{y}$ are linearly independent.

Problem 3.13 Show that if λ is an eigenvalue of $\mathbf{A}$, then $1/(1+\lambda)$ is an eigenvalue of $(\mathbf{I} + \mathbf{A})^{-1}$.

We may prove the result in two steps:

1. If λ is an eigenvalue of $\mathbf{A}$, then $1 + \lambda$ is an eigenvalue of $\mathbf{I} + \mathbf{A}$.
2. If λ is an eigenvalue of $\mathbf{A}$, then $1/\lambda$ is an eigenvalue of $\mathbf{A}^{-1}$ (assuming that $\mathbf{A}$ is invertible, which implies that there is no eigenvalue $\lambda = 0$).

If λ is an eigenvalue of $\mathbf{A}$, it is a solution of the characteristic equation for matrix $\mathbf{A}$

$$\det(\mathbf{A} - \lambda \mathbf{I}) = 0$$

Now consider the characteristic equation for matrix $\mathbf{I} + \mathbf{A}$

$$\det(\mathbf{A} + \mathbf{I} - \mu \mathbf{I}) = 0$$

Say that μ solves the equation, which we may rewrite as follows

$$\det(\mathbf{A} + (1 - \mu)\mathbf{I}) = 0$$

which implies that $1 - \mu = \lambda$ is an eigenvalue of $\mathbf{A}$ or, in other words, that $\mu = 1 + \lambda$ is an eigenvalue of $\mathbf{I} + \mathbf{A}$.

By a similar token, let us consider the characteristic equation for $\mathbf{A}^{-1}$:

$$\det(\mathbf{A}^{-1} - \mu \mathbf{I}) = 0$$

We know that $\det(AB) = \det(A)\det(B)$, for square matrices $\mathbf{A}$ and $\mathbf{B}$. Then, let us multiply the last equation by $\det(\mathbf{A})/\mu$:

$$\frac{1}{\mu} \det(\mathbf{A}) \det(\mathbf{A}^{-1} - \mu \mathbf{I}) = \det \left(\frac{1}{\mu} \mathbf{A}\mathbf{A}^{-1} - \frac{\mu}{\mu} \mathbf{A}\mathbf{I} \right) = \det \left(\frac{1}{\mu} \mathbf{I} - \mathbf{A} \right) = 0$$

We see that $\lambda = 1/\mu$ is an eigenvalue of $\mathbf{A}$, which also implies that $\mu = 1/\lambda$ is an eigenvalue of $\mathbf{A}^{-1}$.

Putting the two results together, we find that $1/(1+\lambda)$ is an eigenvalue of $(\mathbf{I} + \mathbf{A})^{-1}$.

Problem 3.14 In the book[1] we show that a symmetric matrix $\mathbf{A}$ can be factored as

$$\mathbf{A} = \mathbf{P\Lambda P}^T$$

where $\mathbf{P}$ is an orthogonal matrix ($\mathbf{P}^T = \mathbf{P}^{-1}$) whose columns are normalized eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix consisting of (real) eigenvalues. It is also easy to see that

$$\mathbf{A}^{-1} = \mathbf{P\Lambda}^{-1}\mathbf{P}^T$$

To see this, recall that of $\mathbf{B}$ and $\mathbf{B}$ are square and invertible, $(\mathbf{BC})^{-1} = \mathbf{C}^{-1}\mathbf{B}^{-1}$, which implies

$$\mathbf{A}^{-1} = (\mathbf{P\Lambda P}^T)^{-1} = (\mathbf{P}^T)^{-1}\mathbf{\Lambda}^{-1}\mathbf{P}^{-1} = \mathbf{A}^{-1} = \mathbf{P\Lambda}^{-1}\mathbf{P}^T$$

Then we have

$$\mathbf{A} + \mathbf{A}^{-1} = \mathbf{A} = \mathbf{P\Lambda P}^T + \mathbf{P\Lambda}^{-1}\mathbf{P}^T = \mathbf{P}(\mathbf{\Lambda} + \mathbf{\Lambda}^{-1})\mathbf{P}^T$$

from which we see that the eigenvalues of $\mathbf{A} + \mathbf{A}^{-1}$ are given by $\lambda + 1/\lambda$, where λ is an eigenvalue of $\mathbf{A}$. Now take the minimum of this expression

$$\min \lambda + \frac{1}{\lambda} \quad \Rightarrow \quad 1 - \frac{1}{\lambda^2} = 0 \quad \Rightarrow \quad \lambda^* = 1 \quad \Rightarrow \quad \lambda^* + \frac{1}{\lambda^*} = 2$$

where we use the assumption that $\lambda > 0$.

For the general case, we take

$$\mathbf{AU} = \mathbf{U\Lambda}$$

where the columns of $\mathbf{U}$ are eigenvectors. If this matrix is invertible, we have

$$\mathbf{A} = \mathbf{U\Lambda U}^{-1}$$

and we may repeat the argument. However, we must assume that the matrix $\mathbf{U}^{-1}$ is in fact invertible, which amounts to saying that $\mathbf{A}$ is *diagonable.* Furthermore, in general, eigenvalues might be complex, which introduces further complication outside the scope of the book.

Problem 3.15 Prove that, for a symmetric matrix $\mathbf{A}$, we have

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2 = \sum_{k=1}^{n} \lambda_k^2$$

where λ_k, $k = 1, \ldots, n$, are the eigenvalues of $\mathbf{A}$.

This may a rather challenging problem, which is considerably simplified if we use a rather concept, the *trace* of a square matrix. The trace of the matrix is just the sum of the elements on its diagonal:

$$\mathrm{tr}(\mathbf{A}) = \sum_{k=1}^{n} a_{kk}$$

Two important properties of the trace are:

[1]Problems 3.14 and 3.15 are taken from the book by Searle, see end of chapter references.

1. The trace is equal to the sum of the eigenvalues:

$$\mathrm{tr}(\mathbf{A}) = \sum_{k=1}^{n} \lambda_k$$

2. If we have symmetric matrices $\mathbf{B}$ and $\mathbf{C}$

$$\mathrm{tr}(\mathbf{BC}) = \sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}c_{ij}$$

i.e., the trace of $\mathbf{BC}$ is the sum of the elementwise product of $\mathbf{B}$ and $\mathbf{C}$.

Since we have $\mathbf{A}^T = \mathbf{A}$

$$\mathrm{tr}(\mathbf{A}^T\mathbf{A}) = \mathrm{tr}(\mathbf{A}^2) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}a_{ij} = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2 \tag{3.1}$$

Furthermore, the eigenvalues of $\mathbf{A}^2$ are the squared eigenvalues of $\mathbf{A}$. Indeed, if $\mathbf{u}_i$ is an eigenvector corresponding to eigenvalue λ_i of $\mathbf{A}$, we have

$$\mathbf{A}^2\mathbf{u}_i = \mathbf{A}(\mathbf{A}\mathbf{u}_i) = \lambda_i\mathbf{A}\mathbf{u}_i = \lambda_i^2\mathbf{u}_i$$

But the trace of $\mathbf{A}^2$ is the sum of its eigenvalues

$$\mathrm{tr}(\mathbf{A}^2) = \sum_{i=1}^{n} \lambda_i^2 \tag{3.2}$$

Putting Eqs. (3.1) and (3.2) together we obtain the result.

4

Descriptive Statistics: On the Way to Elementary Probability

4.1 SOLUTIONS

Problem 4.1 You are carrying out a research about how many pizzas are consumed by teenagers, in the age range from 13 to 17. A sample of 20 boys/girls in that age range is taken, and the number of pizzas eaten per month is given in the following table:

If we work on the raw data

4	12	7	11	9	7	8	13	16	11
4	7	5	7	11	7	7	41	9	14

we obtain the mean as follows:

$$\overline{X} = \frac{4 + 12 + 7 + \cdots + 14}{20} = 10.5$$

We may also sort data

4	4	5	7	7	7	7	7	7	8
9	9	11	11	11	12	13	14	16	41

and find frequencies

X_i	4	5	7	8	9	11	12	13	14	16	41
f_i	2	1	6	1	2	3	1	1	1	1	1

and compute

$$\overline{X} = \frac{2 \times 4 + 1 \times 5 + 6 \times 7 + \cdots + 1 \times 16 + 1 \times 41}{20} = 10.5$$

Sorting data is useful to immediately spot the median:

$$m = \frac{8 + 9}{2} = 8.5$$

If we get rid of the largest observation (41), we obtain

$$\overline{X} = 8.8947, \quad m = 8$$

As we see, the median is less sensitive to outliers.

To find standard deviation:

$$\sum_{i=1}^{20} X_i^2 = 4^2 + 12^2 + \cdots + 14^2 = 3386$$
$$\overline{X}^2 = 110.25$$
$$S^2 = \frac{1}{19}(3386 - 20 \times 110.25) = 62.1579$$
$$S = \sqrt{62.1579} = 7.8840$$

Problem 4.2 The following table shows a set of observed values and their frequencies:

Value	1	2	3	4	5	6	7	8
Frequency	5	4	7	10	13	8	3	1

- Compute mean, variance, and standard deviation.
- Find the cumulated relative frequencies.

We have

$$\sum_{k=1}^{8} f_k = 5 + 4 + \cdots + 3 + 1 = 51$$

observations; the relative frequencies p_i are

0.0980, 0.0784, 0.1373, 0.1961, 0.2549, 0.1569, 0.0588, 0.0196

from which we immediately find the cumulative relative frequencies $P_k = \sum_{j=1}^{k} p_j$

0.0980, 0.1765, 0.3137, 0.5098, 0.7647, 0.9216, 0.9804, 1.0000

To find mean, variance, and standard deviation:

$$\overline{X} = \sum_{k=1}^{8} p_k X_k = 0.0980 \times 1 + 0.0784 \times 2 + \cdots + 0.0196 \times 8 = 4.2353$$
$$S^2 = \frac{1}{50}\left(\sum_{k=1}^{8} f_k X_k^2 - 51\overline{X}^2\right) = \frac{1}{50}\left(5 \times 1 + 4 \times 2^2 + \cdots + 1 \times 8^2 - 51 \times 4.2353^2\right) = 3.0635$$
$$S = \sqrt{3.0635} = 1.7503$$

Problem 4.3 First we find frequencies, relative frequencies, and cumulative frequencies:

X_k	f_k	p_k	P_k
2	3	0.1875	0.1875
3	2	0.125	0.3125
4	5	0.3125	0.625
5	2	0.125	0.75
6	3	0.1875	0.9375
8	1	0.0625	1

The mean is

$$\overline{X} = 3 \times 2 + 2 \times 3 + 5 \times 4 + \cdots + 1 \times 8 = 4.25$$

The largest frequency, 5, is associated to the value 4, which is the mode.

To compute median and quartiles, it may be convenient to just sort the data:

X_k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f_k	2	2	2	3	3	4	4	4	4	4	5	5	6	6	6	8

The median (which is also the second quartile) is the average of data in positions 8 and 9 (i.e., 4). The first quartile is the average of data in positions 4 and 5 (i.e., 3). The third quartile is the average of data in positions 12 and 13 (i.e., 5.5).

Problem 4.4 Let us sort the observations

13.60, 14.30, 15.00, 15.20, 15.60, 16.10, 19.20, 20.10, 21.00, 21.10, 21.30, 22.20

The descriptive statistics are

$$\overline{X} = 17.89, \quad S = 3.19, \quad m = \frac{16.10 + 19.20}{2} = 17.65$$

The mean and the median are not too different, so data are not much skewed.

We have $X_{(8)} = 20.1$, i.e., that person is in position 8 out of 12. The percentile rank can be calculated as

$$\frac{b+e}{n} = \frac{7+1}{12} = 67\%$$

where b is the number of observations "before" in the rank and e is the number of "equal" observations. We may find a different value if we take another definition

$$\frac{b}{b+a} = \frac{7}{7+4} = 63.63\%$$

where a is the number of observations "after" in the rank.

The quartiles are

$$Q_1 = \frac{15.00 + 15.10}{2} = 15.05, \quad Q_2 \equiv m = 17.65, \quad Q_3 = \frac{21.00 + 21.50}{2} = 21.05$$

To check the definition of the first quartile Q_1, we observe that

1. There are at least $\frac{25 \times 12}{100} = 3$ observations less than or equal to $Q_1 = 15.05$ (i.e, 13.60 , 14.30, 15.00)
2. There are at least $\frac{(100-25) \times 12}{100} = 9$ observations larger than or equal to $Q_1 = 15.05$ (i.e, 15.20 , 15.60 , 16.10 , 19.20 , 20.10 , 21.00, 21.10 , 21.30, 22.20)

We observe that the above statements apply to both $X_(3) = 15.00$ and $X_{(4)} = 15.20$, and we take their average.

If we use linear interpolations, the order statistic $X_{(i)}$ is associated with the percentile

$$100 \times \frac{(i - 0.5)}{12}, \qquad i = 1, \ldots, 12$$

For instance, the percentile corresponding to $X_{(1)} = 13.60$ is

$$100 \times \frac{0.5}{12} = 4.1667$$

By the same token, the next percentiles are

$$\begin{aligned}
X_{(2)} &\to 12.5000 \\
X_{(3)} &\to 20.8333 \\
X_{(4)} &\to 29.1667 \\
X_{(5)} &\to 37.5000 \\
X_{(6)} &\to 45.8333 \\
X_{(7)} &\to 54.1667 \\
X_{(8)} &\to 62.5000 \\
X_{(9)} &\to 70.8333 \\
X_{(10)} &\to 79.1667 \\
X_{(11)} &\to 87.5000 \\
X_{(12)} &\to 95.8333
\end{aligned}$$

To find the 90% percentile, we must interpolate between $X_{(11)} = 21.30$ and $X_{(12)} = 22.20$ as follows:

$$21.30 + \frac{90.0000 - 87.500}{95.8333 - 87.5000} \times (22.20 - 21.30) = 21.57$$

Problem 4.5 The first answer is obtained by taking the ratio between how many female professors who had a hangover twice or more and the total number of female professors:

$$\frac{36}{66 + 25 + 36} = 28.35\%$$

The second answer is obtained by taking the ratio between the number of male professors who had a hangover once or less and the total number of professors (male or female) who had a hangover once or less:

$$\frac{61 + 23}{61 + 23 + 66 + 25} = 48\%$$

In Chapter 5, where we introduce the language of probability theory, we learn how to express these questions in terms of conditional probabilities.

1. The first answer can also be written as

$$\mathrm{P}\{(\geq 2) \,|\, F\} = \frac{\mathrm{P}\{(\geq 2) \cap F\}}{\mathrm{P}(F)}$$

where (≥ 2) is the event "the professor had twice or more hangovers" and F is the event "the professor is female." Since

$$\mathrm{P}\{(\geq 2) \cap F\} = \frac{36}{251}, \quad \mathrm{P}(F) = \frac{66 + 25 + 36}{251}$$

where 251 is the total number of professors, we obtain the above result.

2. By the same token, the second answer can also be written as

$$\mathrm{P}\{M \,|\, (\leq 1)\} = \frac{\mathrm{P}\{M \cap (\leq 1)\}}{\mathrm{P}(M)}$$

where $(\leq 1) = (= 0) \cup (= 1)$ is the event "the professor had one or less hangover," which is the union of the event "the professor had no hangover" and "the professor had one hangover."

5

Probability Theories

5.1 SOLUTIONS

Problem 5.1 Consider two events E and G, such that $E \subseteq G$. Then prove that $\mathrm{P}(E) \leq \mathrm{P}(G)$.

We may express G as the union of E and the part of G that does not intersect with E:

$$G = E \cup (G \backslash E)$$

Then, of course the two components are disjoint:

$$E \cap (G \backslash E) = \emptyset$$

and we may apply additive probability

$$\mathrm{P}(G) = \mathrm{P}(E) + \mathrm{P}(G \backslash E) \geq \mathrm{P}(E)$$

since $\mathrm{P}(G \backslash E) \geq 0$.

Problem 5.2 From Bayes' theorem we have

$$\mathrm{P}(A \,|\, E) = \frac{\mathrm{P}(E \,|\, A)\mathrm{P}(A)}{\mathrm{P}(E)}, \quad \mathrm{P}(B \,|\, E) = \frac{\mathrm{P}(E \,|\, B)\mathrm{P}(B)}{\mathrm{P}(E)}$$

Taking ratios

$$\frac{\mathrm{P}(A \,|\, E)}{\mathrm{P}(B \,|\, E)} = \frac{\mathrm{P}(E \,|\, A)\mathrm{P}(A)}{\mathrm{P}(E)} \cdot \frac{\mathrm{P}(E)}{\mathrm{P}(E \,|\, B)\mathrm{P}(B)} = \frac{\mathrm{P}(E \,|\, A)}{\mathrm{P}(E \,|\, B)}$$

under the assumption that $\mathrm{P}(A) = \mathrm{P}(B)$.

This is an inversion formula in the following sense:

- The ratio $\frac{\mathrm{P}(E|A)\mathrm{P}(A)}{\mathrm{P}(E)}$ gives the relative likelihood of A and B given the occurrence of E.
- If we invert conditioning, we consider the probability of E given A or B, which is what we need when we may observe A or B, but not E.
- The formula states that when $\mathrm{P}(A) = \mathrm{P}(B)$ the second ratio is equal to the first one.

Problem 5.3 In this case, it is necessary to lay down all of the events and the pieces of information we have. We know that:

1. the event chooseA occurred, i.e., the participant selected box A
2. the event opC occurred, i.e., the presenter opened box C
3. the event notC occurred, the prize is not in box C (otherwise, the game would have stopped immediately, and the participant could not switch from box A to box B)

Hence, we need the conditional probability $\mathrm{P}(\mathsf{A} \mid \mathsf{chooseA} \cap \mathsf{opC} \cap \mathsf{notC})$. Using the definition of conditional probability:

$$\mathrm{P}(\mathsf{A} \mid \mathsf{chooseA} \cap \mathsf{opC} \cap \mathsf{notC}) = \frac{\mathrm{P}(\mathsf{A} \cap \mathsf{chooseA} \cap \mathsf{opC} \cap \mathsf{notC})}{\mathrm{P}(\mathsf{opC} \cap \mathsf{chooseA} \cap \mathsf{notC})}$$

However, the event $\mathsf{opC} \cap \mathsf{chooseA}$ is independent on the other events, as neither the participant nor the presenter has any clue, and so their behavior is not influenced by knowledge of the box containing the prize:

$$\mathrm{P}(\mathsf{A} \cap \mathsf{chooseA} \cap \mathsf{opC} \cap \mathsf{notC}) = \mathrm{P}(\mathsf{chooseA} \cap \mathsf{opC}) \cdot \mathrm{P}(\mathsf{A} \cap \mathsf{notC})$$
$$\mathrm{P}(\mathsf{opC} \cap \mathsf{chooseA} \cap \mathsf{notC}) = \mathrm{P}(\mathsf{chooseA} \cap \mathsf{opC}) \cdot \mathrm{P}(\mathsf{notC})$$

Therefore

$$\mathrm{P}(\mathsf{A} \mid \mathsf{chooseA} \cap \mathsf{opC} \cap \mathsf{notC}) = \frac{\mathrm{P}(\mathsf{A} \cap \mathsf{notC})}{\mathrm{P}(\mathsf{notC})} = \frac{\mathrm{P}(\mathsf{A})}{\mathrm{P}(\mathsf{notC})} = \frac{1/3}{2/3} = \frac{1}{2}$$

where we use the fact $\mathsf{A} \subseteq \mathsf{notC}$. Hence, the participant has no incentive to switch, as he cannot squeeze any useful information out of the presenter's behavior.

5.2 ADDITIONAL PROBLEMS

Problem 5.4 Each of two cabinets identical in appearance has two drawers. Cabinet A contains a silver coin in each drawer; cabinet B contains a silver coin in one of its drawers and a gold coin in the other. A cabinet is randomly selected, one of its drawers is opened, and a silver coin is found. What is the probability that there is a silver coin in the other drawer?

Problem 5.5 Disgustorama is a brand new food company producing cakes. Cakes may suffer from two defects: incomplete leavening (which affects rising) and/or excessive baking. The two defects are the result of two operations carried out at different stages, so they can be considered independent. Quality is checked only before packaging. We know that the first defect (bad leavening) occurs with probability 7% and the second one with probability 3%. Find:

- The probability that a cake is both dead flat and burned
- The probability that a cake is defective (i.e., it has at least one defect)
- The probability that a cake is burned, if we know that it is defective

5.3 SOLUTIONS OF ADDITIONAL PROBLEMS

Problem 5.4 One possible approach is based on Bayes' theorem. Let A be the event "we have picked cabinet A"; B is the event "we have picked cabinet B". Since we select the cabinet purely at random, a priori $\mathrm{P}(\mathsf{A}) = \mathrm{P}(\mathsf{B}) = 0.5$.

Now we have some additional information, and we should revise our belief by finding the conditional probabilities $\mathrm{P}(\mathsf{A} \,|\, \mathsf{S}_1)$ and $\mathrm{P}(\mathsf{B} \,|\, \mathsf{S}_1)$, where S_1 is the event "the first coin is silver".

Bayes' theorem yields:

$$\mathrm{P}(\mathsf{A} \,|\, \mathsf{S}_1) = \frac{\mathrm{P}(\mathsf{S}_1 \,|\, \mathsf{A})\mathrm{P}(\mathsf{A})}{\mathrm{P}(\mathsf{S}_1)} = \frac{\mathrm{P}(\mathsf{S} \,|\, \mathsf{A})\mathrm{P}(\mathsf{A})}{\mathrm{P}(\mathsf{S}_1 \,|\, \mathsf{A})\mathrm{P}(\mathsf{A}) + \mathrm{P}(\mathsf{S}_1 \,|\, \mathsf{B})\mathrm{P}(\mathsf{B})} = \frac{1 \times 0.5}{1 \times 0.5 + 0.5 \times 0.5} = \frac{2}{3}$$

By the way, it may be useful to observe that $\mathrm{P}(\mathsf{S}_1) = 0.75$, which may sense, as there are 3 silver coins and 1 gold coin. Then

$$\mathrm{P}(\mathsf{B} \,|\, \mathsf{S}_1) = 1 - \mathrm{P}(\mathsf{A} \,|\, \mathsf{S}_1) = \tfrac{1}{3}.$$

Now we may calculate

$$\mathrm{P}(\mathsf{S}_2 \,|\, \mathsf{S}_1) = \mathrm{P}(\mathsf{S}_2 \,|\, \mathsf{A}) \cdot \mathrm{P}(\mathsf{A} \,|\, \mathsf{S}_1) + \mathrm{P}(\mathsf{S}_2 \,|\, \mathsf{B}) \cdot \mathrm{P}(\mathsf{B} \,|\, \mathsf{S}_1) = 1 \times \tfrac{2}{3} + 0 \times \tfrac{1}{3} = \tfrac{2}{3}$$

The above solution is unnecessarily contrived, but it is a good illustration of a general framework based on the revision of unconditional probabilities. In our case, we know that the second coin is silver only if we picked cabinet A, so

$$\mathrm{P}(\mathsf{S}_2 \,|\, \mathsf{S}_1) = \mathrm{P}(\mathsf{A} \,|\, \mathsf{S}_1) = \tfrac{2}{3}$$

A quite straightforward solution is found by applying the definition of conditional probability:

$$\mathrm{P}(\mathsf{S}_2 \,|\, \mathsf{S}_1) = \frac{\mathrm{P}(\mathsf{S}_2 \cap \mathsf{S}_1)}{\mathrm{P}(\mathsf{S}_1)} = \frac{\mathrm{P}(\mathsf{A})}{\mathrm{P}(\mathsf{S}_1)} = \frac{0.5}{0.75} = \frac{2}{3}$$

This is a smarter solution, even though it works only in this peculiar case and misses the more general style of reasoning.

Problem 5.5 Let A_1 be the event "incomplete leavening" and let Let A_2 be the event "excessive baking":

$$\mathrm{P}(A_1) = 0.07, \quad \mathrm{P}(A_2) = 0.03$$

By assumption, these events are independent.

1. We want $\mathrm{P}(A_1 \cap A_2)$. Because of independence,

$$\mathrm{P}(A_1 \cap A_2) = \mathrm{P}(A_1) \cdot \mathrm{P}(A_2) = 0.07 \times 0.03 = 0.0021$$

2. We want $\mathrm{P}(A_1 \cup A_2)$. Note that the two events are not disjoint (if they were, they could *not* be independent!). Hence

$$\mathrm{P}(A_1 \cup A_2) = \mathrm{P}(A_1) + \mathrm{P}(A_2) - \mathrm{P}(A_1 \cap A_2) = 0.07 + 0.03 - 0.0021 = 0.0979$$

3. We apply Bayes' theorem to find the conditional probability

$$\mathrm{P}(A_2 \,|\, A_1 \cup A_2) = \frac{\mathrm{P}(A_1 \cup A_2 \,|\, A_2) \cdot \mathrm{P}(A_2)}{\mathrm{P}(A_1 \cup A_2)} = \frac{\mathrm{P}(A_2)}{\mathrm{P}(A_1 \cup A_2)} = \frac{0.03}{0.0979} = 0.3064$$

After all, this solution is rather intuitive if we interpret probabilities as relative frequencies; however, we prefer to use a more general and sound reasoning. Also note that, in this case, we do not apply the usual theorem of total probability to find the denominator of the ratio, as A_1 and S_2 are not independent.

6

Discrete Random Variables

6.1 SOLUTIONS

Problem 6.1 Clearly, variance is minimized when $p = 0$ or $p = 1$, i.e., when there is no variability at all. In Chapter 6, we show that when $x_1 = 1$ and $x_2 = 0$, i.e., when we deal with a standard Bernoulli variable, its variance is maximized for $p = 0.5$. Here we wonder whether this applies to general values x_1 and x_2 as well.

One possible approach is to repeat the drill, write variance explicitely, and maximize it with respect to p:

$$\begin{aligned}
\mathrm{E}[X] &= px_1 + (1-p)x_2 = p(x_1 - x_2) + x_2 \\
\mathrm{E}^2[X] &= p^2(x_1 - x_2)^2 + 2p(x_1 - x_2)x_2 + x_2^2 \\
\mathrm{E}[X^2] &= px_1^2 + (1-p)x_2^2 = p(x_1^2 - x_2^2) + x_2^2 \\
\mathrm{Var}(X) &= p(x_1^2 - x_2^2) + x_2^2 - p^2(x_1 - x_2)^2 - 2p(x_1 - x_2)x_2 - x_2^2 \equiv \sigma^2(p)
\end{aligned}$$

Applying the first-order condition

$$\frac{d\sigma^2(p)}{dp} = x_1^2 - x_2^2 - 2p(x_1 - x_2)^2 - 2(x_1 - x_2)x_2 = 0$$

we find

$$p = \frac{x_1^2 - x_2^2 - 2x_1x_2 + 2x_2^2}{2p(x_1 - x_2)^2} = \frac{(x_1 - x_2)^2}{2p(x_1 - x_2)^2} = 0.5$$

which is the same result as the standard Bernoulli variable. This is not surprising after all and can be obtained by a more straightforward approach. Let us consider random variable

$$Y = \frac{X - x_2}{x_1 - x_2}$$

We see that when $X = x_1$, $Y = 1$, and when $X = x_2$, $Y = 0$. Hence, Y is the standard Bernoulli variable, whose variance is maximized for $p = 0.5$. But by recalling the properties of variance we see that

$$\mathrm{Var}(Y) = \frac{\mathrm{Var}(X)}{(x_1 - x_2)^2}$$

Since the two variances differ by a positive constant, they are both maximized for the same value of p.

Problem 6.2 A much easier and insightful proof will be given in Chapter 8, based on conditioning, but if we observe the expected value of the geometric random variable

$$\mathrm{E}[X] = \sum_{i=1}^{\infty} i(1-p)^{i-1}p = pS_1$$

we see that the sum

$$S_1 \equiv \sum_{i=1}^{\infty} i(1-p)^{i-1}$$

looks like the derivative of a geometric series. More precisely, if we let

$$S(p) \equiv \sum_{i=0}^{\infty} (1-p)^i = \frac{1}{1-(1-p)} = \frac{1}{p}$$

term-by-term differentiation yields

$$S'(p) = -\sum_{i=0}^{\infty} i(1-p)^{i-1} = -\sum_{i=1}^{\infty} i(1-p)^{i-1} = -S_1$$

Then

$$S_1 = -\frac{d}{dp}\left(\frac{1}{p}\right) = \frac{1}{p^2}$$

and

$$\mathrm{E}[X] = p\frac{1}{p^2} = \frac{1}{p}$$

Problem 6.3 The binomial expansion formula is

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{n-k} a^{n-k} b^k$$

and we want to prove that

$$\sum_{k=1}^{n} p_k = 1$$

where

$$p_k \equiv \mathrm{P}(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$

Now it is easy to see that

$$\binom{n}{k} = \frac{n!}{(n-k)!k!} = \frac{n!}{(n-k)![n-(n-k)]!} = \binom{n}{n-k}$$

Hence

$$\sum_{k=1}^{n} p_k = \sum_{k=0}^{n} \binom{n}{n-k} p^k (1-p)^{n-k} = (1-p+p)^n = 1$$

Problem 6.4 Measuring profit in \$ millions, if the probability of success is p, expected profit is

$$p \times 16 - (1-p) \times 5 = p \times 21 - 5$$

If we launch the product immediately, then $p = 0.65$ and expected profit is \$8.65 million. If we delay, expected profit is, taking the additional cost and the time discount into account

$$-1 + \frac{p \times 21 - 5}{1.03}$$

We are indifferent between the two options if

$$-1 + \frac{p \times 21 - 5}{1.03} = 8.65 \quad \Rightarrow \quad p = \frac{(1 + 8.65) \times 1.03 + 5}{21} = 0.7114$$

Then, the minimal improvement in success probability is

$$\Delta = 0.7114 - 0.65 \approx 6.14\%$$

Problem 6.5 If we assume that the persons we test are independent (i.e., we do not consider groups of heavy drinkers...), we are dealing with a binomial random variable with parameters $p = 0.4$ and $n = 25$:

$$\begin{aligned}
\mathrm{P}(X \geq 4) &= 1 - \mathrm{P}(X \leq 3) \\
&= 1 - [\mathrm{P}(X=0) + \mathrm{P}(X=1) + \mathrm{P}(X=2) + \mathrm{P}(X=3)] \\
\mathrm{P}(X=0) &= 0.6^{25} = 2.8430 \cdot 10^{-6} \\
\mathrm{P}(X=1) &= \binom{25}{1} 0.4^1 \times 0.6^{24} = 25 \times 0.4^1 \times 0.6^{24} = 4.7384 \cdot 10^{-5} \\
\mathrm{P}(X=2) &= \binom{25}{2} 0.4^2 \times 0.6^{23} = \frac{25 \times 24}{2} \times 0.4^2 \times 0.6^{23} = 3.7907 \cdot 10^{-4} \\
\mathrm{P}(X=3) &= \binom{25}{3} 0.4^3 \times 0.6^{22} = \frac{25 \times 24 \times 23}{2 \times 3} \times 0.4^3 \times 0.6^{22} = 0.001937
\end{aligned}$$

which yields

$$\mathrm{P}(X \geq 4) = 0.99763$$

Problem 6.6 Assuming that batteries in a package are independent (which is in fact a debatable assumption), we are dealing with a binomial random variable with parameters $p = 0.02$ and $n = 8$. The probability that a package is returned is

$$P_{\mathsf{bad}} = 1 - \mathrm{P}(X=0) - \mathrm{P}(X=1) = 1 - 0.98^8 - 8 \times 0.02 \times 0.08^7 = 0.010337$$

If the consumer buys three packages, the number of returned packages is binomial with parameters $p = P_{\mathsf{bad}}$ and $n = 3$:

$$\mathrm{P}(Y=1) = 3 \times P_{\mathsf{bad}} \times (1 - P_{\mathsf{bad}})^2 = 0.030379$$

7

Continuous Random Variables

7.1 SOLUTIONS

Problem 7.1 We want to find $\mathrm{P}(X > 200)$, where $X \sim \mathcal{N}(250, 40^2)$. Using standardization:

$$\begin{aligned}\mathrm{P}(X > 200) &= \mathrm{P}\left(\frac{X-250}{40} > \frac{200-250}{40}\right) \\ &= \mathrm{P}\left(Z > -1.25\right)\end{aligned}$$

where Z is standard normal. Depending on the tool you have at hand, there are different ways of evaluating $\mathrm{P}\left(Z > -1.25\right)$:

- If you have software for evaluating the CDF $\Phi(z) = \mathrm{P}(Z \leq z)$

$$\mathrm{P}\left(Z > -1.25\right) = 1 - \Phi(-1.25) = 1 - 0.1056 = 0.8944$$

- If you have statistical tables, you will typically find $\Phi(z)$ only for nonnegative values of z; in such a case, take advantage of the symmetry of the PDF of the standard normal:

$$\mathrm{P}\left(Z > -1.25\right) = \mathrm{P}\left(Z \geq 1.25\right) = \Phi(1.25) = 0.8944$$

Whatever you do, check the sensibility of your result! Since 200 is smaller than the expected value 250, we must find a probability that is larger than 0.5.

Problem 7.2 Using standardization again:

$$\begin{aligned}\mathrm{P}(230 \leq X \leq 260) &= \mathrm{P}\left(\frac{230-250}{20} \leq \frac{X-250}{20} \leq \frac{260-250}{20}\right) \\ &= \mathrm{P}\left(-1 \leq Z \leq 0.5\right) \\ &= \Phi(0.5) - \Phi(-1) \\ &= 0.6915 - 0.1587 = 0.5328\end{aligned}$$

The considerations we stated for problem 1.1 apply here as well.

Problem 7.3 We should set the reorder point R for an item, whose demand during lead time is uncertain. We have a very rough model of uncertainty – the lead time demand is uniformly distributed between 5000 and 20000 pieces. Set the reorder point in such a way that the service level is 95%.

In this case, we must find a quantile from the uniform distribution on $[5000, 20000]$. Such a quantile can be easily obtained by finding a value "covering" 95% of the interval:

$$R = 5000 + (20000 - 5000) \times 0.95 = 19250$$

Problem 7.4 You are working in your office, and you would like to take a very short nap, say, 10 minutes. However, every now and then, your colleagues come to your office to ask you for some information; the interarrival time of your colleagues is exponentially distributed with expected value 15 minutes. What is the probability that you will not be caught asleep and reported to you boss?

The time until the next visit by a colleague is an exponentially distributed random variable X with rate $\lambda = 1/15$. The required probability is

$$\mathrm{P}(X \leq 10)$$

that may be obtained by recalling the CDF of the exponential distribution, $F_X(x) = 1 - e^{-\lambda x}$. In our case

$$\mathrm{P}(X \leq 10) = 1 - e^{-\frac{10}{15}} = 0.4866$$

As an equivalent approach, we may integrate the PDF $f_X(x) = \lambda e^{-\lambda x}$:

$$\mathrm{P}(X \leq 10) = \int_0^{10} \tfrac{1}{15} e^{-\frac{x}{15}}\, dx = -e^{-\frac{x}{15}} \Big|_0^{10} = -e^{-\frac{10}{15}} - (-e^{-\frac{0}{15}}) = 0.4866$$

Problem 7.5 We know that virtually all of the realizations of a normal variable are in the interval $(\mu - 3\sigma, \mu + 3\sigma)$. Since we are representing demand (which is supposed to be positive, unless you are *very bad* with marketing), we should make sure that the probability of a negative demand is a negligible modeling error.

In our case, $12000 - 3 \times 7000 = -9000$; hence, there is a nonnegligible probability of negative demand, according to the normal model. Indeed:

$$\mathrm{P}(D \leq 0) = \mathrm{P}\left(Z \leq \frac{0 - 12000}{7000} \right) = \Phi(-1.7143) = 4.32\%$$

Therefore, the model does not look quite reasonable. Please note that it is quite plausible that demand has expected value 12000 and standard deviation 7000; we are just observing that it cannot be normally distributed; for instance, it could well be skewed to the right.

Problem 7.6 Let X be a normal random variable with expected value $\mu = 6$ and standard deviation $\sigma = 1$. Consider random variable $W = 3X^2$. Find the expected value $\mathrm{E}[W]$ and the probability $\mathrm{P}(W > 120)$.

We recall that $\mathrm{Var}(X) = \mathrm{E}[X^2] - \mathrm{E}^2[X]$. Then

$$\mathrm{E}[W] = \mathrm{E}[3X^2] = 3(\mathrm{Var}(X) + \mathrm{E}^2[X]) = 3 \times (1^2 + 6^2) = 111$$

The probability is a little trickier because of a potential trap: It is *tempting* to write

$$\mathrm{P}(W > 120) = \mathrm{P}(3X^2 > 120) = \mathrm{P}(X > \sqrt{40})$$

where we take the square root of both sides of inequality. but this is not quite correct, as we are not considering the possibility of large negative values of X. Indeed, $\sqrt{X^2} = \mid X \mid$. Hence

$$\mathrm{P}(W > 120) = \mathrm{P}(X > \sqrt{40}) + \mathrm{P}(X < -\sqrt{40}).$$

Using the familiar drill:

$$\mathrm{P}(X > \sqrt{40}) = 1 - \Phi\left(\frac{6.3246 - 6}{1}\right) = 0.3728$$
$$\mathrm{P}(X < -\sqrt{40}) = \Phi\left(\frac{-6.3246 - 6}{1}\right) = \Phi(-12.3246) \approx 0$$

Hence, $\mathrm{P}(W > 120) = 37.28\%$. In this *lucky* case, the above error would be inconsequential.

Problem 7.7 Let us denote the inventory level after replenishment by I, and the demand of customer i by D_i. We know that $I \sim \mathcal{N}(400, 40^2)$, $\mathrm{E}[D_i] = 3$, and $\mathrm{Var}(D_i) = 0.3^2$. The total demand, if we receive N customer orders, is

$$D = \sum_{i=1}^{N} D_i$$

If we assume that customer demands are independent, and N is large enough, the central limit theorem states that

$$D \sim \mathcal{N}(3N, 0.3^2 N)$$

The probability of a stockout is given by

$$\mathrm{P}\left(D > I\right)$$

and we should find N such that

$$\mathrm{P}\left(\sum_{i=1}^{N} D_i > I\right) > 0.1$$

or, equivalently

$$\mathrm{P}\left(Y_N \le 0\right) < 0.9$$

where $Y_N = \sum_{i=1}^{N} D_i - I$.

We know that Y_N is normal, and we need its parameters:

$$\mu_N = 3N - 400$$
$$\sigma_N = \sqrt{0.3^2 N + 40^2}$$

To find N, which is integer, let us allow for real values and find n such that

$$\mathrm{P}\left(Y_n \le 0\right) = 0.9$$

which means that the 90% quantile of Y_n should be 0. Since $z_{0.9} = 1.2816$, we must solve the equation:

$$(3n - 400) + 1.2816\sqrt{0.3^2 n + 40^2} = 0$$

which is equivalent to

$$8.8522n^2 - 2400n + 157372.20 = 0$$

whose solutions are

$$n_1 = 116.19, \quad n_2 = 150.49$$

Hence, taking the smaller root and rounding it up, we find $N = 117$.

Problem 7.8 We know that a chi-square variable with n degrees of freedom is the sum of n independent standard normals squared:

$$X = \sum_{i=1}^{n} Z_i^2$$

Then

$$\mathrm{E}[X] = \sum_{i=1}^{n} \mathrm{E}[Z_i^2] = n\mathrm{E}[Z^2] = n(\mathrm{Var}(Z) + \mathrm{E}^2[Z]) = n,$$

since the expected value of the standard normal Z is 0 and its variance is 1.

Problem 7.9 This is a little variation on the classical newsvendor's problem. Usually, we are given the economics (profit margin m and cost of unsold items c_u), from which we obtain the optimal service level β, and then the order quantity Q. Here we go the other way around:

$$Q = \mu + z_\beta \sigma \quad \Rightarrow \quad z_\beta = \frac{Q - \mu}{\sigma} = \frac{15000 - 10000}{2500} = 2$$

From the CDF of the standard normal we find

$$\beta = \Phi(2) = 0.9772 = \frac{m}{m + c}$$

In our case $m = 14 - 10 = 4$ and $c_u = 10 - b$, where b is the buyback price. Hence

$$\frac{4}{4 + (10 - b)} = 0.9772 \quad \Rightarrow \quad b = 14 - \frac{4}{0.9772} = 9.9069$$

We see that the manufacturer should bear the whole risk, which is not surprising since she requires a very high service level (97.72%).

Problem 7.10

- If the probability that the competitor enters the market is assumed to be 50%, how many items should you order to maximize expected profit? (Let us assume that selling prices are the same in both scenarios.)
- What if this probability is 20%? Does purchased quantity increase or decrease?

The service level must be

$$\beta = \frac{m}{m + c_u} = \frac{16 - 10}{(16 - 10) + (10 - 7)} = \frac{2}{3}$$

The only difficulty in this problem is figuring out the PDF of the demand, which is given in Fig. 7.1.

Roughly speaking, we have two scenarios:

1. If the competition is strong, the PDF is the uniform distribution on the left
2. If the competition is weak, the PDF is the uniform distribution on the right

Since the probability of strong competition is 50%, the two areas are the same, which implies that the value of the density is 1/2000. Note that the two pieces do not overlap, which makes

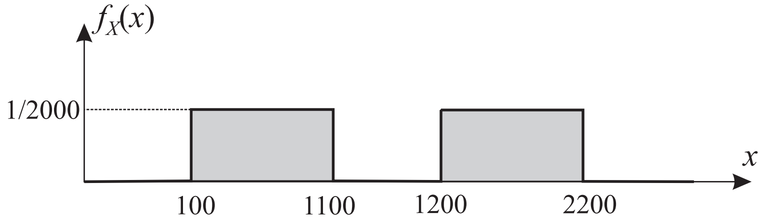


Fig. 7.1 PDF of demand in Problem 7.10.

the reasoning easy. Since $\beta > 0.5$, we must take a quantile in the second range, between 1200 and 2200: the area within the second rectangle must be

$$\tfrac{2}{3} - 0.5 = \tfrac{1}{6}$$

and we must be careful, as the area in this second rectangle is only 0.5 (the PDF is 1/2000 and not 1/1000):

$$Q = 1200 + \tfrac{1}{6} \times (2200 - 1200) \times \tfrac{1}{0.5} \approx 1533$$

In the second case, the reasoning is quite similar, but now the rectangle on the left has "weight" 0.2, rather than 0.8:

$$Q = 1200 + (\tfrac{2}{3} - 0.2) \times (2200 - 1200) \times \tfrac{1}{0.8} \approx 1783$$

A couple of observations are in order:

1. A common mistake is finding the optimal order quantities in the two scenarios and then taking their average. This is conceptually wrong, as it amounts to

   - finding the optimal solution *assuming that we know what the competition is going to do*
   - taking the average of the two optimal solutions for the two alternative scenarios

   This is not correct, as we have to make a decision *before* we discover what competitors choose.

2. We have found the PDF of the demand a bit informally, which worked well as the two intervals are disjoint. A sounder reasoning is based on finding the CDF first, by applying the total probability theorem and conditioning with respect to the competition level (strong or weak):

   $$F_X(x) \equiv \mathrm{P}(D \le x) = 0.5 \times \mathrm{P}(D \le x \,|\, \mathsf{strong}) + 0.5 \times \mathrm{P}(D \le x \,|\, \mathsf{weak})$$

   But

   $$\mathrm{P}(D \le x \,|\, \mathsf{strong}) = \begin{cases} 0 & x < 100 \\ \frac{x-100}{1000} & 0 \le x \le 1100 \\ 1 & x > 1100 \end{cases}$$

   $$\mathrm{P}(D \le x \,|\, \mathsf{weak}) = \begin{cases} 0 & x < 1200 \\ \frac{x-1200}{1000} & 1200 \le x \le 2200 \\ 1 & x > 2200 \end{cases}$$

   Adding everything up, we obtain the CDF in Fig. 7.2. Taking its derivative, we find the PDF above.

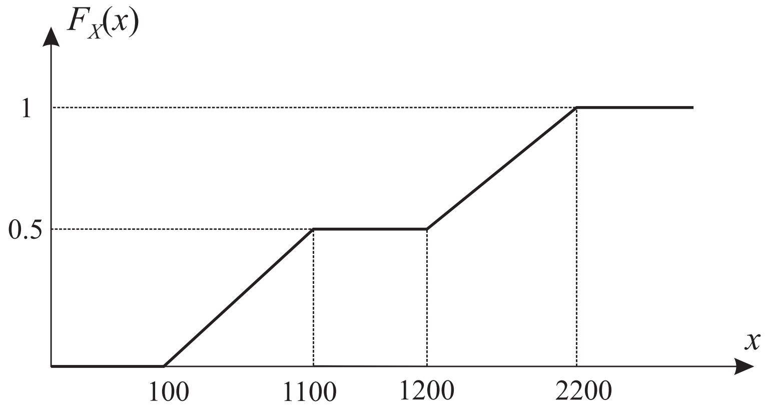


Fig. 7.2 CDF of demand in Problem 7.10.

Problem 7.11 Let us write the CDF $F_X(x)$:

$$\begin{aligned} F_X(x) &\equiv \mathrm{P}(X \le x) \\ &= \mathrm{P}(\max\{U_1, U_2, \ldots, U_n\} \le x) \\ &= \mathrm{P}(U_1 \le x, U_2 \le x, \ldots, U_n \le x) \\ &= \mathrm{P}(U_1 \le x) \cdot \mathrm{P}(U_2 \le x) \cdots \mathrm{P}(U_n \le x) \end{aligned}$$

where we take advantage of independency among the random variables U_i, $i = 1, \ldots, n$. We also know that all of them are uniform on the unit interval, and $\mathrm{P}(U \le x) = x$, for $x \in [0, 1]$.
Therefore

$$F_X(x) = [\mathrm{P}(U \le x)]^n = x^n$$

Observation: As you see, the statement of the problem is not quite correct, as $F_X(x) = x^n$ for $x \in [0, 1]$, whereas $F_X(x) = 1$ for $x > 1$ and $F_X(x) = 0$ for $x < 0$.

8

Dependence, Correlation, and Conditional Expectation

8.1 SOLUTIONS

Problem 8.1 To solve the problem we need to characterize the aggregate demand we see at the central warehouse:

$$D_C = D_1 + D_2$$

where D_1 and D_2 are not independent, since they are affected by a common risk factor X. Rather than computing their covariance, since we have independent factors X, ϵ_1, and ϵ_2, it is much easier to rewrite aggregate demand as

$$D_C = (100 + 120)X + \epsilon_1 + \epsilon_2$$

This demand is normal, with expected value

$$\mu_C = 220\mu_X + \mu_1 + \mu_2 = 220 \times 28 + 200 + 300 = 6660$$

and standard deviation

$$\sigma_C = \sqrt{220^2\sigma_X^2 + \sigma_1^2 + \sigma_2^2} = \sqrt{220^2 \times 16 + 100 + 150} = 880.14$$

The inventory level should be chosen as

$$Q = \mu_C + z_{0.95}\sigma_C = 6660 + 1.6449 \times 880.14 \approx 8108$$

If the two specific factors are positively correlated, standard deviation is larger:

$$\sigma_C = \sqrt{220^2\sigma_X^2 + \sigma_1^2 + \sigma_2^2 + 2\rho_{1,2}\sigma_1\sigma_2}$$

This implies a larger stocking level, needed to hedge against some more uncertainty; however, with these numbers, the effect would be negligible, as the most variability comes from the common factor X.

Problem 8.2 We just need to find the distribution of the demand D_C for component C, which depends on demands D_1 and D_2 for end items P_1 and P_2, respectively:

$$D_C = 2D_1 + 3D_2$$

Since D_1 and D_2 are normal, so is D_C, and its expected value and standard deviation are:

$$\begin{aligned} \mu_C &= \mathrm{E}[D_C] = 2\mathrm{E}[D_1] + 3\mathrm{E}[D_2] = 2 \times 1000 + 3 \times 700 = 4100 \\ \sigma_C &= \sqrt{\mathrm{Var}(D_C)} = \sqrt{4\mathrm{Var}(D_1) + 9\mathrm{Var}(D_2)} \\ &= \sqrt{4 \times 250^2 + 9 \times 180^2} = 735.9348 \end{aligned}$$

The inventory level is the 92% quantile of this distribution:

$$Q = \mu_C + z_{0.92}\sigma_C = 4100 + 1.4051 \times 735.9348 = 5134.04$$

Problem 8.3 Let R_{IFM} and R_{PM} be the rates of return from the two stocks, over one day. Loss is

$$L = -(10000 \times R_{\mathrm{IFM}} + 20000 \times R_{\mathrm{PM}})$$

where the sign is inconsequential as we assume that expected return over one day is 0% and the normal distribution is symmetric. We need the standard deviation of the distribution of loss:

$$\begin{aligned} \sigma_L &= \sqrt{(10000\sigma_{\mathrm{IFM}})^2 + (20000\sigma_{\mathrm{PM}})^2 + 2\rho \times 10000\sigma_{\mathrm{IFM}} \times 20000\sigma_{\mathrm{PM}}} \\ &= \sqrt{(10000 \times 0.02)^2 + (20000 \times 0.04)^2 + 2 \times 0.68 \times 10000 \times 0.02 \times 20000 \times 0.04} \\ &= 947.42 \end{aligned}$$

Since $z_0.95 = 1.6449$:

$$\mathrm{VaR}_{0.95} = 1.6449 \times 947.42 = \$1558.36$$

A little discussion. Note that for each individual position we have:

$$\begin{aligned} \mathrm{VaR}_{0.95,\mathrm{IFM}} &= 1.6449 \times 10000 \times 0.02 = \$328.97 \\ \mathrm{VaR}_{0.95,\mathrm{PM}} &= 1.6449 \times 20000 \times 0.04 = \$1315.88 \end{aligned}$$

We see that

$$\mathrm{VaR}_{0.95,\mathrm{IFM}} + \mathrm{VaR}_{0.95,\mathrm{PM}} = 328.97 + 1315.88 = 1644.85 > 1558.36$$

The sum of the two individual risks does exceed the joint risk, but not by much, since there is a fairly strong positive correlation. In real life, correlations do change dramatically when markets crash, going either to +1 or to -1. So, the power of diversification must not be overstated when managing tail risk, not to mention the fact that normal distribution do not feature the negative skewness and excess kurtosis that we observe in actual data.

Problem 8.4 **Note:** There is an error in the statement of the problem, as the two variables X and Y must be *identically distributed* but not necessarily independent.

One way of proving the claim is by taking advantage of the distributive property of covariance:

$$\begin{aligned} \mathrm{Cov}(X - Y, X + Y) &= \mathrm{Cov}(X, X + Y) - \mathrm{Cov}(Y, X + Y) \\ &= \mathrm{Cov}(X, X) + \mathrm{Cov}(X, Y) - \mathrm{Cov}(Y, X) - \mathrm{Cov}(Y, Y) \\ &= \mathrm{Var}(X) - \mathrm{Var}(Y) = 0 \end{aligned}$$

since the two variances are the same.

Alternatively, we may rewrite covariance as follows:

$$\begin{aligned}\mathrm{Cov}(X-Y, X+Y) &= \mathrm{E}[(X-Y)(X+Y)] - \mathrm{E}[X-Y]\mathrm{E}[X+Y] \\ &= \mathrm{E}[X^2 - Y^2] - \mathrm{E}[X-Y]\mathrm{E}[X+Y].\end{aligned}$$

But, since X and Y are identically distributed,

$$\mathrm{E}[X^2 - Y^2] = 0, \quad \mathrm{E}[X-Y] = 0,$$

and the claim follows.

9

Inferential Statistics

9.1 SOLUTIONS

Problem 9.1 Finding the confidence interval is easy. We first find the sample statistics

$$\overline{X} = \frac{1}{18}\sum_{i=1}^{18} X_i = 133.2222,$$

$$S = \sqrt{\frac{1}{17}\left(\sum_{i=1}^{18} X_i^2 - 18\overline{X}^2\right)} = 10.2128.$$

Then, given the relevant quantile $t_{1-\alpha/2,N-1} = t_{0.975,17} = 2.1098$, we find

$$\overline{X} \pm t_{1-\alpha/2,N-1}\frac{S}{\sqrt{N}} = (128.1435, 138.3009).$$

These calculations are easy to carry out in R:

```
> X = c(130,122,119,142,136,127,120,152,141,132,127,118,150,141,133,137,129,142)
> Xbar = mean(X); Xbar
[1] 133.2222
> S = sd(X); S
[1] 10.21277
> t = qt(0.975,17); t
[1] 2.109816
> Xbar-t*S/sqrt(18)
[1] 128.1435
> Xbar+t*S/sqrt(18)
[1] 138.3009
```

A much better approach is to use the following function:

```
> t.test(X)$conf.int
```

```
[1] 128.1435 138.3009
attr(,"conf.level")
[1] 0.95
```

This can also be carried out in MATLAB:

```
>> X=[130,122,119,142,136,127,120,152,141,132,127,118,150,141,133,137,129,142];
>> [Xbar,~,CI]=normfit(X)
Xbar =
  133.2222
CI =
  128.1435
  138.3009
```

It is interesting to note that the MATLAB function makes pretty explicit a hidden assumption: This way of calculating confidence intervals is correct for a normal sample. For other distributions, it is at best a good approximation, at least for a large sample size.

The second part of the problem requires a bit more thought, as it refers to a *single* realization of the random variable $X \sim \mathsf{N}(\mu, \sigma^2)$. We want to find

$$P\{X > 130\} = P\left\{\frac{X-\mu}{\sigma} > \frac{130-\mu}{\sigma}\right\} 1 - P\left\{Z \leq \frac{130-\mu}{\sigma}\right\} = 1 - \Phi\left(\frac{130-\mu}{\sigma}\right),$$

where $Z \sim \mathsf{N}(0,1)$ is standard normal and Φ is its CDF. One possibility is to plug the sample statistics above:

$$P\{X > 130\} = 1 - \Phi\left(\frac{130 - 133.2222}{10.21277}\right) = 1 - 0.3762 = 0.6238.$$

The calculation is straightforward in R:

```
> 1-pnorm(130,mean(X),sd(X))
[1] 0.6238125
```

Note, however, that we are plugging our estimates of μ and σ as if they were exact. However, the sample statistics are realizations of random variables; hence, strictly speaking, the above calculation is only an approximation (see Section 10.4 for a full illustration in the case of linear regression models used for forecasting). However, we use it for the sake of simplicity in order to illustrate the difference between a purely data-driven analysis and the fitting of a theoretical uncertainty model. Indeed, one could argue that the above probability can be estimated by just counting the number of observations that are larger than 130. There are ten such cases (not counting the first observation), therefore we could use the estimate:

$$\frac{10}{18} = 0.5556.$$

If we interpret "larger" in the non-strict sense and include the first observation, we find

$$\frac{11}{18} = 0.6111,$$

which is much closer to the result obtained by fitting a normal distribution. Clearly, with a small sample the discrete nature of data may have an impact, whereas $P\{X > 130\} = P\{X >= 130\}$ in the continuous case. Arguably, with a larger dataset, we would get a more satisfactory result, without relying on any distributional assumption (who said that IQs are normally distributed?).

However, what is the probability of observing an IQ larger than 152? Using the data, the answer is zero, since there is no larger observation than 152. However, using a theoretical distribution with an infinite support, like the normal, we may, in some sense, "append a tail" to the dataset. In our case, the probability is not quite negligible:

```
> 1-pnorm(max(X),mean(X),sd(X))
[1] 0.03298284
```

This kind of reasoning is essential when we deal with risk management, where we are concerned with *extreme* events.

Problem 9.2 The confidence interval is

$$\overline{X} \pm t_{1-\alpha/2,N-1}\frac{S}{\sqrt{N}} = 13.38 \pm 2.093024 \times \frac{4.58}{\sqrt{20}} = (11.23649, 15.52351).$$

The quantile is found in R by using the command

```
> qt(0.975,19)
[1] 2.093024
```

whereas in MATLAB we may use

```
>> tinv(0.975,19)
ans =
    2.0930
```

This is the same function name as in Excel, but if you are using Excel you need to use the total probability of the two tails: `TINV(0.05,19)`.

If we increase the sample size to $N' = 50$, both $\overline{X}$ and S will change, since we use a different sample. If we assume that S does not change significantly, we may say that the new half-length of the confidence interval is about

$$2.093024 \times \frac{4.58}{\sqrt{50}}.$$

This means that the half-length is shrunk by a factor

$$\sqrt{\frac{50}{20}} = 1.581139.$$

If we double the sample size, all other things being equal, we shrink the interval by a factor $\sqrt{2}$. If we aim at gaining one decimal digit of precision, which is obtained by shrinking the interval by a factor 10, we must multiply the sample size by 100. This explains why sampling methods, such as Monte Carlo methods, may be rather expensive (e.g., in terms of computational effort).

Problem 9.3 The confidence interval is

$$128.37 \pm 2.235124 \times \frac{37.3}{\sqrt{50}} = (116.5797, 140.1603),$$

where we use the quantile $t_{0.985,49} = 2.235124$. By the way, note that the corresponding standard normal quantile is $z_{0.985} = 2.17009$; this shows that the usual rule of thumb, stating that when we have more than 30 degrees of freedom, we can use the standard normal quantiles is just an approximation, possibly justified in the past, given the limitations of statistical tables.

The confidence interval length is

$$2t_{0.985,49}\frac{S}{\sqrt{N}} = 23.58063.$$

If we assume that the sample standard deviation S will not change too much when additional observations are collected, in order to reduce the confidence interval by 50% we should find N' such that

$$t_{0.985,N'-1}\frac{S}{\sqrt{N'}} = \frac{23.58063}{2\times 2} = 5.895157.$$

Solving for N' is complicated by the fact N' also defines the t quantile. If we use the standard normal quantile $z_{0.985} = 2.17009$, we get rid of this dependence on N' and find

$$N' = \left(\frac{2.17009\times 37.3}{5.895157}\right)^2 = 188.5309,$$

i.e., we need about 134 more observations. If we use the quantile $t_{0.985,49} = 2.235124$, we stay on the safe side and find

$$N' = \left(\frac{2.235124\times 37.3}{5.895157}\right)^2 = 200.$$

This result is obvious: If we keep all other factors unchanged, halving the confidence intervals requires $\sqrt{N'} = 2\sqrt{N}$, which implies $N' = 4N$.

The formula

$$N = \left(\frac{z_{1-\alpha/2}S}{H}\right)^2$$

where H is the half-lenght of the confidence interval is useful when we want to prescribe a maximum error β. More precisely, we might require that

$$|\overline{X} - \mu| < \beta,$$

with probability $1-\alpha$. This can be translated into the requirement $H = |\overline{X} - \mu|$. Clearly, we need a pilot sample in order to figure out S. Typically, a large sample is required, warranting the use of z quantiles. Then, we may check N and see if we can afford that sample size; if not, we must settle for a less precise estimate.

Problem 9.4 The estimator is in fact unbiased:

$$\begin{aligned}
\mathrm{E}\left[\widetilde{X}\right] &= \mathrm{E}\left[\frac{1}{N}\left(\frac{1}{2}X_1 + \frac{3}{2}X_2 + \frac{1}{2}X_3 + \frac{3}{2}X_4 + \cdots + \frac{1}{2}X_{N-1} + \frac{3}{2}X_N\right)\right] \\
&= \frac{1}{N}\sum_{k=1}^{N/2}\mathrm{E}\left[\frac{1}{2}X_k + \frac{3}{2}X_{k+1}\right] \\
&= \frac{1}{N}\left(\frac{1}{2}+\frac{3}{2}\right)\sum_{k=1}^{N/2}\mathrm{E}[X] \\
&= \frac{1}{N}\cdot 2\cdot\frac{N}{2}\cdot\mathrm{E}[X] = \mu,
\end{aligned}$$

where we have used the fact that the sample consists of an even number of i.i.d. observations. By the same token, we find

$$\begin{aligned}\mathrm{Var}\left(\widetilde{X}\right) &= \frac{1}{N^2}\sum_{k=1}^{N/2}\mathrm{Var}\left(\frac{1}{2}X_k + \frac{3}{2}X_{k+1}\right)\\ &= \frac{1}{N^2}\cdot\left(\frac{1}{4}+\frac{9}{4}\right)\cdot\frac{N}{2}\cdot\sigma^2 = 1.25\sigma^2.\end{aligned}$$

This variance is larger than the variance σ^2/N of the usual sample mean. Hence this is an unbiased estimator, but not ane efficient one (see Definition 9.16).

Problem 9.5 The key issue in this kind of problems is to get the null hypothesis right. A fairly common source of mistakes is to identify the null hypothesis with what we wish to prove. Per se, this is not a mistake. In nonparametric tests of normality, we have to take as the null hypothesis that the distribution is normal; otherwise, we have no idea of how to choose a test statistic and study its distribution to come up with a rejection region.

In this case, it is tempting to choose $H_0 : \mu \geq 1250$. However, this makes no sense, since the rejection region would be the left tail, where the test statistic

$$T = \frac{\overline{X} - 1250}{S/\sqrt{N}}$$

is *very* negative. In other words, we are lead to "accept" the hypothesis of an improvement even when the performance in the sample is worse than the old average. The point is that we can really only reject the null hypothesis (note that this is an intrinsic limitation of this simple testing framework). Thus, we should test the null hypothesis $H_0 : \mu \leq 1250$, i.e., the hypothesis that nothing interesting is happening, against the alternative $H_0 : \mu \leq 1250$.

Having said this, this problem is actually solved in the text (see Example 9.14) and was included in the problem list by mistake! See Problem 9.6 for another example.

Problem 9.6 We should test the null hypothesis that no improvement occurred

$$H_0 : \mu \geq 29$$

agains the alternative

$$H_a : \mu < 29.$$

Thus, we may say that there is strong evidence of an improvement if the test statistic

$$T = \frac{\overline{X} - \mu_0}{S/\sqrt{N}} = \frac{26.9 - 29}{8/\sqrt{20}} = -1.173936$$

is sufficiently negative, in such a way that the apparent improvement cannot be attributed to a lucky random sample. If we choose $\alpha = 0.05$, the relevant quantile for this one-tailed test, with rejection region on the left tail, is $t_{0.05,19} = -1.729133$. Unfortunately, the standardized test statistic is not negative enough to support the hypothesis of an improvement, as we fail to reject the null hypothesis.

To put it another way, a t distributed random variable T_{19} with 19 degrees of freedom has a probability

$$\mathrm{P}\{T_{19} \leq -1.173936\} = 0.1274632$$

of being less than or equal to the observed test statistic T. This is the p-value, and it is too large to safely reject the null hypothesis.

Problem 9.7 The two samples are independent and large enough to warrant application of the approach described in Example 9.16. The relevant statistics are

$$\overline{X}_1 - \overline{X}_2 = 8.2 - 9.4 = -1.2, \qquad S_{\overline{X}_1 - \overline{X}_2} = \sqrt{\frac{2.1^2}{46} + \frac{2.9^2}{54}} = 0.5016077.$$

The test statistic for the null hypothesis $H_0 : \mu_1 - \mu_2 = 0$ is

$$Z = \frac{-1.2}{0.5016077} = -2.392308.$$

We may compute the p-value

$$2\mathrm{P}\{Z \geq |-2.392308|\} = 0.0167428,$$

which is indeed fairly small (we reject the null hypothesis for any significance level larger than 2%) and justifies rejection of the null hypothesis. There seems to be a statistically significant difference between the two machines.

Problem 9.8 Here the two samples are related, and we should use a paired t-test. We take the difference between performance at the beginning and after 3 hours, which yields the sample

$$11, 11, -3, 25, -1, 30, 32, 36, 12, 29.$$

Note that positive values of these differences suggest a deterioration of performance (the larger, the index, the better). The sample mean and sample standard deviations of these differences are are

$$\overline{D} = 18.2000, \qquad S_D = 14.0222.$$

We test the null hypothesis $H_0 : \mu_D \leq 0$ against the alternative $H_a : \mu_D > 0$ (i.e., there is a significant effect of weariness). The test statistic is

$$T = \frac{18.2}{14.0222/\sqrt{10}} = 4.1045.$$

With $\alpha = 0.05$, we compare T against $t_{0.95,9} = 1.833113$ and reject the null hypothesis. We may also compute the p-value

$$\mathrm{P}\{T_9 \geq 4.1045\} = 0.001329383,$$

which is small enough to rekect the null hypothesis and conclude that the effect of weariness is statistically significant.

We may carry out the test quite conveniently in R:

```
> X1 = c(68, 64, 69, 88, 72, 80, 85, 116, 77, 78)
> X2 = c(57, 53, 72, 63, 73, 50, 53, 80, 65, 49)
> help(t.test)
starting httpd help server ... done
> t.test(X1,X2,paired=TRUE,alternative="greater")

        Paired t-test

data:  X1 and X2
t = 4.1045, df = 9, p-value = 0.001329
```

```
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 10.07159      Inf
sample estimates:
mean of the differences
                   18.2
```

Problem 9.9 We have to apply formula (9.20) in the text, which requires the following quantitities:

$$S = \sqrt{\frac{1}{10}\left(\sum_{k=1}^{10} X_i^2 - 10\overline{X}^2\right)} = 10.6192, \quad \chi^2_{0.025,9} = 2.7004, \quad \chi^2_{0.975,9} = 19.0228.$$

The confidence interval for variance is

$$\left(\frac{(n-1)S^2}{\chi^2_{0.975,9}}, \frac{(n-1)S^2}{\chi^2_{0.025,9}}\right) = (7.3042, 19.0228).$$

Taking square roots, we find the interval for standard deviation

$$(53.3518, 375.8342).$$

We may carry out the calculations in MATLAB, for instance:

```
>> X=[ 103.23; 111.00; 86.45; 105.17; 101.91;
92.15;  97.40; 102.06; 121.47; 116.62];
>> S=std(X)
S =
   10.6192
>> chi2=chi2inv(0.025,9)
chi2 =
    2.7004
>> chi1=chi2inv(0.975,9)
chi1 =
   19.0228
>> sqrt(9*S^2/chi1)
ans =
    7.3042
>> sqrt(9*S^2/chi2)
ans =
   19.3864
```

A more direct way relies on `normfit`, which yields a point estimator and a confidence interval for both expected value and standard deviation, assuming a normal sample:

```
>> [mu,sig,cm,cs]=normfit(X')
mu =
  103.7460
sig =
   10.6192
cm =
```

```
   96.1495
  111.3425
cs =
    7.3042
   19.3864
>>
```

Problem 9.10 The point estimator of the parameter p of the corresponding Bernoulli distribution is

$$\hat{p} = \frac{63}{1000} = 0.063,$$

and an approximate confidence interval is (using the normal quantile $z_{0.995} = 2.5758$)

$$\left(\hat{p} \pm z_{1-\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right) = \left(0.063 \pm 2.5758\sqrt{\frac{0.063 \times (1-0.063)}{1000}}\right) = (0.0432, 0.0828).$$

As stated in the text (Section 9.4.3), this is just an approximation, in that we are approximating a binomial distribution by a normal one.

A more sophisticated approach is actually taken in R, where a confidence interval for p is given by the function `prop.test`:

```
> prop.test(63,1000,conf.level=0.99)

        1-sample proportions test with continuity correction

data:  63 out of 1000, null probability 0.5
X-squared = 762.129, df = 1, p-value < 2.2e-16
alternative hypothesis: true p is not equal to 0.5
99 percent confidence interval:
 0.04552120 0.08638302
sample estimates:
    p
0.063
```

Problem 9.11 Let us recall the definitions first:

$$\overline{X}_{i\cdot} \equiv \frac{1}{n}\sum_{j=1}^{n} X_{ij}, \qquad i = 1, 2, \ldots, m,$$

$$\overline{X}_{\cdot\cdot} \equiv \frac{1}{nm}\sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij},$$

$$\text{SS}_b \equiv n\sum_{i=1}^{m}\left(\overline{X}_{i\cdot} - \overline{X}_{\cdot\cdot}\right)^2,$$

$$\text{SS}_w \equiv \sum_{i=1}^{m}\sum_{j=1}^{n}\left(X_{ij} - \overline{X}_{i\cdot}\right)^2,$$

where X_{ij} is observation j, $j = 1, \ldots, n$, of group i, $i = 1, \ldots, m$.

A typical trick of the trade when dealing with this kind of proofs is to add and subtract the same quantity in order to come up with different sums of squares (and possibly completing the squares if necessary). Let us rewrite SS_w as follows and develop the squares:

$$\begin{aligned}
\mathrm{SS}_w &= \sum_{i=1}^{m}\sum_{j=1}^{n}\left[(X_{ij}-\overline{X}_{\cdot\cdot})-(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})\right]^2 \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n}(X_{ij}-\overline{X}_{\cdot\cdot})^2+\sum_{i=1}^{m}\sum_{j=1}^{n}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})^2-2\sum_{i=1}^{m}\sum_{j=1}^{n}(X_{ij}-\overline{X}_{\cdot\cdot})(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})
\end{aligned}$$

The three terms of the sum can be rewritten as follows:

$$\begin{aligned}
\sum_{i=1}^{m}\sum_{j=1}^{n}(X_{ij}-\overline{X}_{\cdot\cdot})^2 &= \sum_{i=1}^{m}\sum_{j=1}^{n}X_{ij}^2+\sum_{i=1}^{m}\sum_{j=1}^{n}\overline{X}_{\cdot\cdot}^2-2\sum_{i=1}^{m}\sum_{j=1}^{n}X_{ij}\overline{X}_{\cdot\cdot} \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n}X_{ij}^2+nm\overline{X}_{\cdot\cdot}^2-2\overline{X}_{\cdot\cdot}\cdot nm\overline{X}_{\cdot\cdot} \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n}X_{ij}^2-nm\overline{X}_{\cdot\cdot}^2 \\
\sum_{i=1}^{m}\sum_{j=1}^{n}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})^2 &= n\sum_{i=1}^{m}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})^2 \\
&= \mathrm{SS}_b \\
\sum_{i=1}^{m}\sum_{j=1}^{n}(X_{ij}-\overline{X}_{\cdot\cdot})(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot}) &= \sum_{i=1}^{m}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})\left[\sum_{j=1}^{n}(X_{ij}-\overline{X}_{\cdot\cdot})\right] \\
&= \sum_{i=1}^{m}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})(n\overline{X}_{i\cdot}-n\overline{X}_{\cdot\cdot}) \\
&= n\sum_{i=1}^{m}(\overline{X}_{i\cdot}-\overline{X}_{\cdot\cdot})^2 \qquad = \mathrm{SS}_b.
\end{aligned}$$

By putting everything together, we prove the result:

$$\mathrm{SS}_w = \sum_{i=1}^{m}\sum_{j=1}^{n}X_{ij}^2-nm\overline{X}_{\cdot\cdot}^2-\mathrm{SS}_b.$$

Problem 9.12 We apply one-way ANOVA, as in example 9.24.

The three sample means for the three groups are

$$\begin{aligned}
\overline{X}1\cdot &= \frac{1445.44}{5} = 289.09 \\
\overline{X}2\cdot &= \frac{1198.12}{5} = 239.62 \\
\overline{X}3\cdot &= \frac{1358.81}{5} = 271.76,
\end{aligned}$$

and the overall sample mean is

$$\overline{X}_{\cdot\cdot} = 266.82.$$

Given the variability in the data, we may guess that the difference in the sample means is not enough to reject the null hypothesis $H_0 : \mu_1 = \mu_2 = \mu_3$, but a proper check is in order.

We calculate the sums of squares

$$\mathrm{SS}_b = 5 \times \left[(289.09 - 266.82)^2 + (239.62 - 266.82)^2 + (271.76 - 266.82)^2\right] = 6299.55$$
$$\mathrm{SS}_w = \sum_{i=1}^{5}\sum_{j=1}^{3} X_{ij}^2 - 5 \times 3 \times 266.82^2 - 6299.55 = 175226.23.$$

The two estimates of variance are then, accounting for the degrees of freedom,

$$\mathrm{SS}_b/2 = 3149.77, \qquad \mathrm{SS}_w/(15-3) = 14602.19,$$

and the test statistic is

$$TS = \frac{3149.77}{14602.19} = 0.2157.$$

The relevant quantile of the F distribution, if we assume a 5% significance level, is

$$F_{0.95,2,12} = 3.8853,$$

so that the test statistic is not in the rejection region. We cannot say that there is any statistically significant difference in the three groups. We may also find the p-value

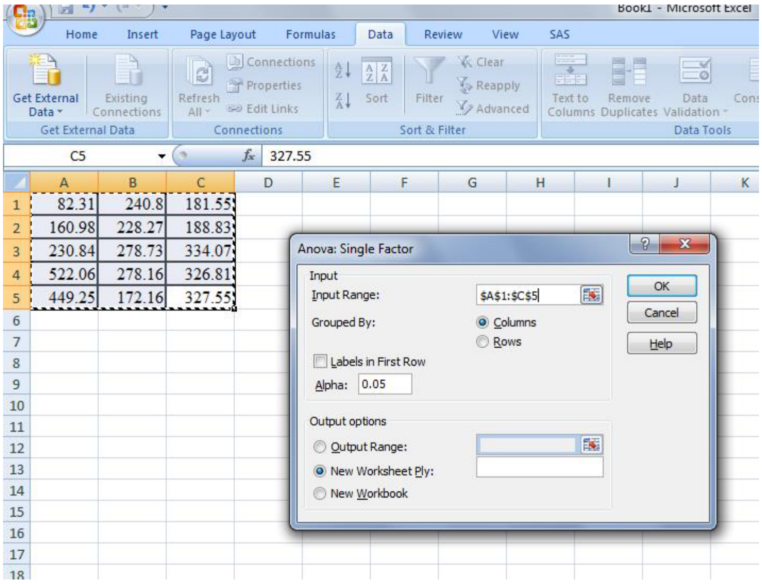
$$\mathrm{P}\{F_{2,12} \geq 0.2157\} = 1 - \mathrm{P}\{F_{2,12} \leq 0.2157\} = 1 - 0.1910 = 0.8090,$$

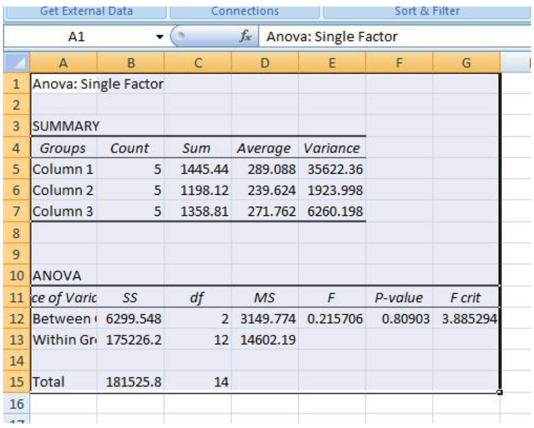
which is way too large to reject H_0.

All of this is conveniently carried out in MATLAB as follows:

```
X = [82.31,  240.80, 181.55
     160.98, 228.27, 188.83
     230.84, 278.73, 334.07
     522.06, 278.16, 326.81
     449.25, 172.16, 327.55];
>> [p,tab]=anova1(X)

p =
    0.8090

tab =
   'Source'     'SS'            'df'     'MS'             'F'          'Prob>F'
   'Columns'    [6.2995e+03]    [ 2]     [3.1498e+03]     [0.2157]     [0.8090]
   'Error'      [1.7523e+05]    [12]     [1.4602e+04]           []           []
   'Total'      [1.8153e+05]    [14]               []           []           []
```

In Excel, we may just place data in a worksheet and apply one-way ANOVA in the Data Analysis tool, as shown in the figures below:





Anova: Single Factor						
SUMMARY						
Groups	Count	Sum	Average	Variance		
Column 1	5	1445.44	289.088	35622.36		
Column 2	5	1198.12	239.624	1923.998		
Column 3	5	1358.81	271.762	6260.198		
ANOVA						
ce of Varic	SS	df	MS	F	P-value	F crit
Between	6299.548	2	3149.774	0.215706	0.80903	3.885294
Within Gr	175226.2	12	14602.19			
Total	181525.8	14				

Using R is a bit more cumbersome, as we must set up a dataframe and fit a model explaning output as a function of input factors:

```
> v1=c(82.31, 160.98, 230.84, 522.06, 449.25)
> v2=c(240.80, 228.27, 278.73, 278.16, 172.16)
> v3=c(181.55, 188.83, 334.07, 326.81, 327.55)
> output=c(v1,v2,v3)
> group=c(rep("g1",5),rep("g2",5),rep("g3",5))
> data=data.frame(output,group)
> summary(aov(output~group,data=data))
            Df  Sum Sq Mean Sq F value Pr(>F)
group        2    6300    3150   0.216  0.809
Residuals   12 175226   14602
```

On the other hand, R offers sophisticated tools for ANOVA.

Problem 9.13 Using the inverse transform method (see page 451), we generate an exponential random variable with rate λ as $X = -(\ln U)/\lambda$. Since an m-Erlang variable Y with rate λ is the sum of m independent exponentials, we can generate it as follows:

$$Y = -\frac{1}{\lambda}\sum_{k=1}^{m} \ln U_k = -\frac{1}{\lambda}\ln\left(\prod_{k=1}^{m} U_k\right).$$

By taking the product of the uniforms, we have to compute one logarithm.

Problem 9.14 The problem requires finding a way to simulate from a triangular distribution. Hence, we may find the CDF first and then invert it, using the inverse transform approach for random variate generation (see p. 451). The PDF if piecewise linear, and integrating it on the first subinterval $[1, 2]$ yields

$$F(x) = \int_1^x (t-1)dt = \left.\frac{(t-1)^2}{2}\right|_1^x = \frac{(x-1)^2}{2}.$$

We also observe that the PDF is symmetric; hence the mode should Inversion of this function for $U \leq 0.5$ yields

$$\frac{(X-1)^2}{2} = U \qquad \Rightarrow \qquad X = 1 + \sqrt{2U}. \tag{9.1}$$

Note that we choose the positive root in order to generate X in the correct interval. On the second interval $[2, 3]$ we find

$$F(x) = \frac{1}{2} + \int_2^x (3-t)dt = \frac{1}{2} - \left.\frac{(3-t)^2}{2}\right|_2^x = 1 + \frac{(3-x)^2}{2}.$$

Note that we must account for the cumulated probability of the first interval ($\frac{1}{2}$). The inverse transform yields

$$1 + \frac{(3-x)^2}{2} = U \qquad \Rightarrow \qquad X = 3 - \sqrt{2(1-U)}.$$

Again, we choose the appropriate root. As a quick check, also note that for $U = 0, 0.5, 1$ (respectively) we obtain $X = 1, 2, 3$. We urge the reader to draw the CDF, which is continuous, nondecreasing, and consists of portions of two parabolas.

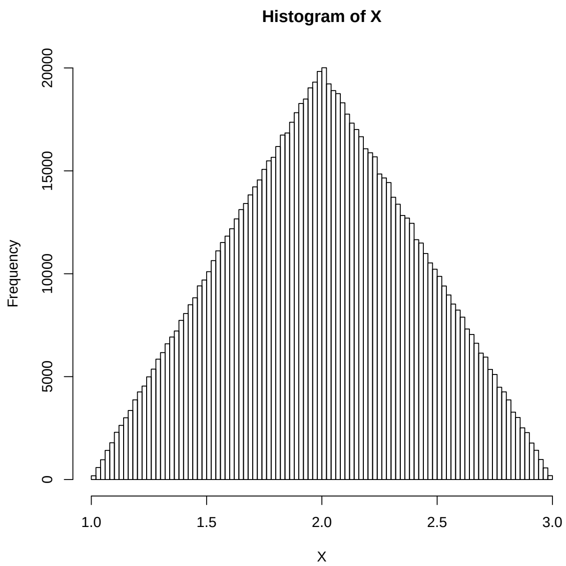
The algorithm requires generating $U \sim \mathsf{U}(0, 1)$ and returning

$$X = \begin{cases} 1 + \sqrt{2U}, & \text{if } U \leq 0.5, \\ 3 - \sqrt{2(1-U)}, & \text{if } U > 0.5. \end{cases}$$

We may use R to check the algorithm. The script

```
set.seed(55555)
howmany = 1000000
U = runif(howmany)
iBelow = (U <= 0.5)
iAbove = (U > 0.5)
X = numeric(howmany)
X[iBelow] = 1+sqrt(2*U[iBelow])
X[iAbove] = 3-sqrt(2*(1-U[iAbove]))
hist(X,nclass=100)
```

produces the following histogram:



Here is the same idea in MATLAB:

```
howmany = 1000000;
U = rand(howmany,1);
iBelow = find(U <= 0.5);
iAbove = find(U > 0.5);
X = zeros(howmany,1);
X(iBelow) = 1+sqrt(2*U(iBelow));
X(iAbove) = 3-sqrt(2*(1-U(iAbove)));
hist(X,100)
```

In general, to sample from a triangular distribution, it is better to think of standardazing a triangle with support (a, b) and mode $c \in (a, b)$, finding a triangle with support $(0, 1)$ and mode $c' \in (0, 1)$. Then, we apply the inverse transform method there and we destandardize.

Problem 9.15 According to this policy (see Section 2.1 and page 445), whenever the inventory position falls below the reorder level R, a fixed quantity Q is ordered, which will be received after the lead time LT, which is fixed under the problem assumptions. Note that, in principle, we might have more than one order outstanding in the pipeline.

The inventory position is $I_p = H + O$, where H is the inventory on hand (i.e., the physical inventory available), and O is the amount on order (i.e., the total amount ordered but not received yet). We are assuming impatient customers, i.e., demand that cannot be satisfied from stock immediately is lost. When backordering is allowed, the backlog would be another relevant variable.

Both H and O define the state of the system, which also includes the ordered list of events that will occur in the future.

These events are:

- Customer arrivals
- Supply order arrivals

When initializing the system, we should draw an exponential random variable with rate λ defining the time at which the first customer will arrive. We also have to initialize the other state variable, and statistical counters should be managed in order to collect the relevant performance measures (inventory holding cost, ordering cost, fraction of demand lost, number of stockouts, etc.).

The procedures for avent management can be outlined as follows:

Customer arrival at simulated time t.

- Draw an exponential random variable t_a and schedule the next arrival at time $t + t_a$.
- Draw the random demand X according to the selected distribution.
- If $H \geq X$, update on-hand inventory $H = H - X$ and inventory position $I_p = I_p - X$; otherwise record an unsatisfied demand. (We do not allow for partial order satisfaction.)
- If $I_p < R$, schedule the next supply order arrival at time $t + LT$.

Supply order arrival.

- Update on-hand inventory $H = H + Q$ and inventory position $I_p = I_p + Q$.

If we allow for backlog B, inventory position is defined as $I_p = H + O - B$. When demand cannot be satisfied, update B and I_p accordingly. When a supply order is received, use it to serve backlogged customers. Note that if we do not allow for partial order satisfaction, we must also manage the queue of waiting customers with their demand, as the aggregate backlog B does not tell the whole story.

Problem 9.16 This random variable, for large n, takes the "small" value 0 with high probability and the "large" value n with a vanishing probability. We may guess that the variable converges to zero, but this must be carefully checked.

Convergence in probability to a value c requires (see Definition 9.7):

$$\lim_{n \to +\infty} \mathrm{P}\{| X_n - c |> \epsilon\} = 0.$$

If we choose $c = 0$, we see that

$$\mathrm{P}\{X_n > 0 > \epsilon\} = \frac{1}{n^2}$$

indeed goes to zero. Hence, we have convergence in probability.

Convergence in quadratic mean to a number c, also said mean square convergence, requires (see Definition 9.10)

$$\lim_{n \to +\infty} \mathrm{E}\left[(X_n - c)^2\right] = 0.$$

Here we find

$$\begin{aligned} \mathrm{E}\left[(X_n - c)^2\right] &= (0 - c)^2 \left(1 - \frac{1}{n^2}\right) + (n - c)^2 \frac{1}{n^2} \\ &= c^2 - \frac{c^2}{n^2} + 1 - \frac{2c}{n^2} + \frac{c^2}{n^2} \\ &= 1 + c^2 - \frac{2c}{n}. \end{aligned}$$

The last term goes to zero for increasing n, but $1 + c^2$ is never zero, even if we plug the reasonable limit $c = 0$. Hence, we do not have convergence in quadratic mean. This also shows that this convergence is not implied by convergence in probability.

Problem 9.17 For such an exponential distribution, we know (see Section 7.6.3) that

$$\mathrm{E}[X] = \frac{1}{\lambda} \qquad \Rightarrow \qquad \lambda = \frac{1}{\mathrm{E}[X]}.$$

The estimator of the first moment is

$$M_1 = \overline{X} = \frac{1}{N}\sum_{k=1}^{N} X_k,$$

hence

$$\hat{\lambda} = \frac{1}{\frac{1}{N}\sum_{k=1}^{N} X_k}.$$

Thus, we find the same estimator as in Example 9.37, where we apply maximum likelihood.

Problem 9.18 The likelihood function for a sample of size n is

$$f_n(x \mid a, b) = \begin{cases} \dfrac{1}{(b-a)^n}, & \text{if } a \le x \le b, \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, since we want to maximize the likelihood, we must have

$$a \le x_k, \quad b \ge x_k, \qquad k = 1, \ldots, n,$$

which implies

$$a \le \min_{k=1,\ldots,n} x_k, \quad b \ge \max_{k=1,\ldots,n} x_k.$$

This just states that the bounds a and b must be compatible with the observations. The likelihood function, when those bounds hold, is increasing in b and decreasing in a:

$$\frac{\partial f_n}{\partial b} = \frac{1}{(b-a)^{n+1}}, \qquad \frac{\partial f_n}{\partial a} = \frac{-1}{(b-a)^{n+1}}.$$

Hence, we take the largest value of b and the smallest value of a compatible with the bounds:

$$\hat{a} \le \min_{k=1,\ldots,n} x_k, \qquad \hat{b} \ge \max_{k=1,\ldots,n} x_k.$$

Note that we find a maximum if the interval on which the uniform distribution is defined, i.e., its support, is the *closed* interval $[a, b]$. With an open interval, we are in trouble.

Problem 9.19 We want to prove that

$$\mathrm{E}\left[X_{(n)}\right] = \mathrm{E}\left[\max_{k=1,\ldots,n} X_k\right] = \frac{n}{n+1}\theta,$$

when $X_k \sim \mathsf{U}[0, \theta]$ and the observations are mutually independent. By the way, this is obvious for $n = 1$.

Let us define $Y = \max_{k=1,\ldots,n} X_k$. In problem 7.11 we show that, when $\theta = 1$,

$$F_Y(y) = \left[\mathrm{P}(X \le x)\right]^n = x^n.$$

It is easy to generalize the result to an arbitrary θ,

$$F_Y(y) = [\mathrm{P}(X \le x)]^n = \frac{x^n}{\theta^n},$$

and then find the density of Y:

$$f_Y(y) = \frac{F_Y(y)}{dy} = n\frac{y^{n-1}}{\theta^n}.$$

Now let us compute the expected value of Y:

$$\mathrm{E}[Y] = \int_0^\theta f_Y(y)\,dy = \int_0^\theta y \cdot n\frac{y^{n-1}}{\theta^n}\,dy = \frac{n}{\theta^n}\int_0^\theta y^n\,dy = \frac{n}{\theta^n}\frac{\theta^{n+1}}{n+1} = \frac{n}{n+1}\theta.$$

10

Simple Linear Regression

10.1 SOLUTIONS

Problem 10.1 Let us carry out the calculations in detail:

$$\sum_{i=1}^{10} x_i = 55, \qquad \sum_{i=1}^{10} Y_i = 420, \qquad \sum_{i=1}^{10} x_i Y_i = 2590, \qquad \sum_{i=1}^{10} x_i^2 = 385.$$

Using Eq. (10.3) we find

$$b = \frac{n \sum_{i=1}^{n} x_i Y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} Y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} = \frac{10 \times 2590 - 44 \times 420}{10 \times 385 - 55^2} = 3.393939.$$

Then

$$a = \overline{Y} - b\overline{x} = \frac{420 - 3.393939 \times 55}{10} = 23.33333.$$

To find R^2, we need the vector of forecasted values $\hat{Y}_i = a + bx_i$:

$$\begin{bmatrix} 26.72727, & 30.12121, & 33.51515, & 36.90909, & 40.30303, \\ 43.69697, & 47.09091, & 50.48485, & 53.87879, & 57.27273 \end{bmatrix}^{\mathsf{T}}.$$

Therefore

$$R^2 = \frac{\sum_{i=1}^{n} \left(\hat{Y}_i - \overline{Y}\right)^2}{\sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)^2} = \frac{950.303}{1760} = 0.5399.$$

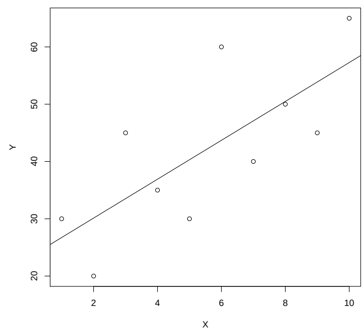
All of the above is easily carried out in R:

```
> X=1:10
> Y=c(30, 20, 45, 35, 30, 60, 40, 50, 45, 65)
> mod=lm(Y~X)
```

```
> summary(mod)
Call:
lm(formula = Y ~ X)
Residuals:
    Min      1Q  Median      3Q     Max
-10.303  -8.432  -1.197   6.614  16.303

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   23.333      6.873   3.395  0.00943 **
X              3.394      1.108   3.064  0.01548 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 10.06 on 8 degrees of freedom
Multiple R-squared: 0.5399,     Adjusted R-squared: 0.4824
F-statistic: 9.389 on 1 and 8 DF,  p-value: 0.01548
```

It is also easy to plot the data points and the regression line:

```
> plot(X,Y)
> abline(mod)
```



MATLAB is a bit less convenient, because:

1. We have to form a *matrix* of regressors, including a leading column of ones.
2. We must care about column vs. row vectors, as in MATLAB vectors are just matrices, whereas they are different data structures in R.
3. We have to collect output statistics in a variable (say, `stats`), whose first element is R^2.

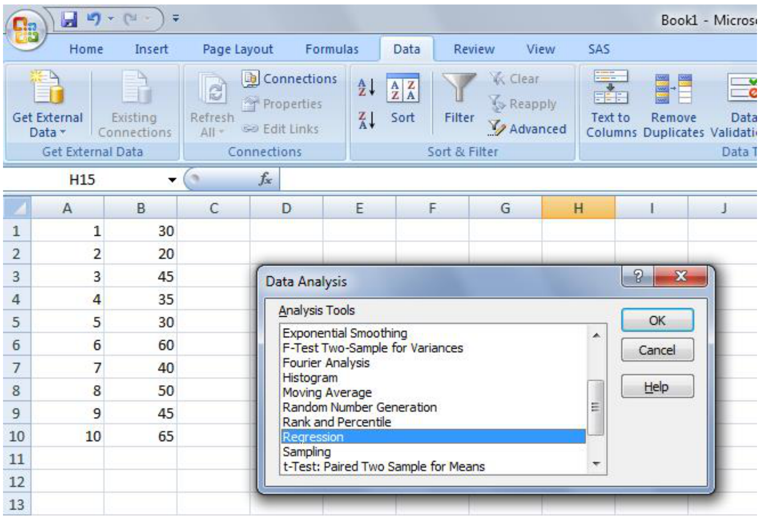
```
>> X=(1:10)';
>> Y=[30, 20, 45, 35, 30, 60, 40, 50, 45, 65]';
>> [b,~,~,~,stats]=regress(Y,[ones(10,1),X])
b =
   23.3333
```

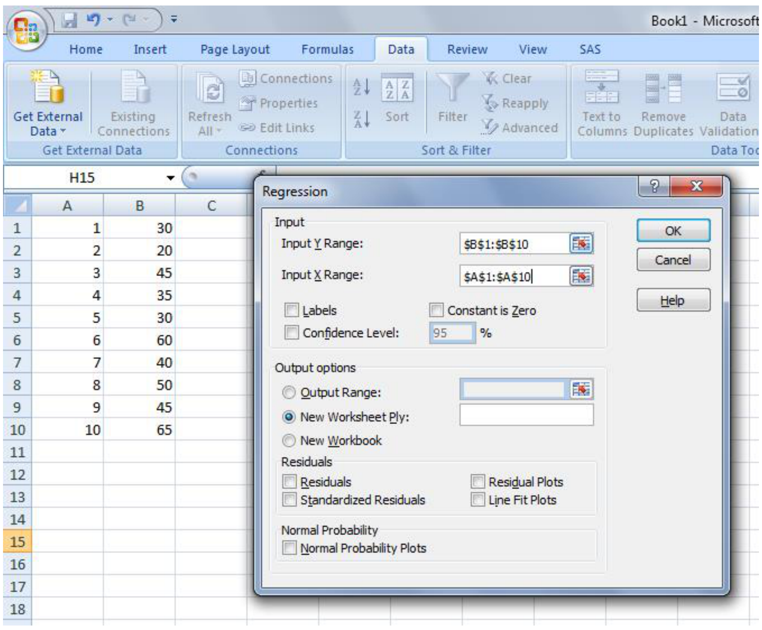
```
    3.3939

stats =
    0.5399    9.3892    0.0155   101.2121
```

Using Excel requires the installation of the Data analysis add-in, which includes a regression function. Then:

1. We place input data in cells.
2. We open the tool and select the input ranges (and output options).
3. We obtain the result in a new worksheet.

This is illustrated in the following screenshots.





	A	B	C	D	E	F	G	H	I
1	SUMMARY OUTPUT								
2									
3	*Regression Statistics*								
4	Multiple R	0.734809434							
5	R Square	0.539944904							
6	Adjusted R Squ	0.482438017							
7	Standard Error	10.06042351							
8	Observations	10							
9									
10	ANOVA								
11		*df*	*SS*	*MS*	*F*	*Significance F*			
12	Regression	1	950.3030303	950.3030303	9.389221557	0.015483959			
13	Residual	8	809.6969697	101.2121212					
14	Total	9	1760						
15									
16		*Coefficients*	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
17	Intercept	23.33333333	6.872577627	3.395135654	0.00942805	7.48514092	39.18152575	7.48514092	39.18152575
18	X Variable 1	3.393939394	1.107616175	3.064183669	0.015483959	0.839771917	5.948106871	0.839771917	5.948106871
19									
20									

Problem 10.2 The task is easily accomplished in R:

```
> X=c(45, 50, 55, 60, 65, 70, 75)
> Y=c(24.2, 25.0, 23.3, 22.0, 21.5, 20.6, 19.8)
> mod=lm(Y~X)
> summary(mod)
Call:
lm(formula = Y ~ X)
Residuals:
        1         2         3         4         5         6         7
-0.692857  0.957143  0.107143 -0.342857  0.007143 -0.042857  0.007143

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 32.54286    1.27059  25.612 1.69e-06 ***
X           -0.17000    0.02089  -8.138 0.000455 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.5527 on 5 degrees of freedom
Multiple R-squared: 0.9298,     Adjusted R-squared: 0.9158
F-statistic: 66.23 on 1 and 5 DF,  p-value: 0.0004548

> confint(mod)
                 2.5 %     97.5 %
(Intercept) 29.2766922 35.8090220
X           -0.2236954 -0.1163046
```

In MATLAB, we have to collect the relevant output from `regr`:

```
>> X=[45 50 55 60 65 70 75]';
>> Y=[24.2 25.0 23.3 22.0 21.5 20.6 19.8]';
>> [b,bint]=regress(Y,[ones(7,1),X])
b =
   32.5429
   -0.1700
bint =
   29.2767   35.8090
```

```
  -0.2237   -0.1163
```

Nevertheless, let us dig into the detail by applying Eq. (10.16), using R for convenience: We need the residual standard deviation of the residuals:

```
> sig=sqrt(sum((Y-mod$fitted)^2)/5)
> sig
[1] 0.5526559
```

Equation (10.16) yields the standard error on the slope:

```
> sigb=sig/sqrt(sum((X-mean(X))^2))
> sigb
[1] 0.02088843
```

To find the confidence interval, we also need the quantile $t_{1-\alpha/2,n-2}$:

```
> t=qt(0.975,5)
> t
[1] 2.570582
> -0.17-t*sigb
[1] -0.2236954
> -0.17+t*sigb
[1] -0.1163046
```

Problem 10.3 To solve the problem, we apply the concepts of Section 7.4.4. We first find the critical ratio involving the profit margin ($m = 14 - 10 = 4$) and the cost of unsold items ($c_u = 10 - 2 = 8$). The optimal service level (probability of not stocking out) is

$$\frac{m}{m + c_u} = \frac{4}{4 + 8} = 0.3333.$$

Thus, we need the quantile at level 33.33% of the sales distribution.
To find the distribution, we carry out a linear regression:

```
> X=1:4
> Y=c(102, 109, 123, 135)
> mod=lm(Y~X)
> summary(mod)

Call:
lm(formula = Y ~ X)

Residuals:
   1    2    3    4
 1.7 -2.6  0.1  0.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   89.000      2.779   32.02 0.000974 ***
X             11.300      1.015   11.13 0.007970 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Residual standard error: 2.269 on 2 degrees of freedom
Multiple R-squared: 0.9841,     Adjusted R-squared: 0.9762
F-statistic:   124 on 1 and 2 DF,  p-value: 0.00797
```

Thus, the point forecast for sales in May is

$$\hat{Y}_0 = a + bx_0 = 89 + 11.3 \times 5 = 145.5$$

We also need the residual standard error, which is $\hat{\sigma}_\epsilon = 2.269$. We can also find it directly:

```
> (sig=sqrt(sum((Y-mod$fitted)^2)/2))
[1] 2.269361
```

If we assume normal errors, the distribution of Y_0 is normal and characterized by expected value and standard deviation. To find the standard deviation of Y_0, also accounting for estimation uncertainty, we may apply Eq. (10.23):

```
> SigY0 = sig*sqrt(1 + 1/4 + (5-mean(X))^2/( sum( (X-mean(X))^2 )))
> SigY0
[1] 3.588175
```

Then, we find the quantity that should be ordered:

$$\hat{Y}_0 + z_{0.3333}\sigma_{Y0}.$$

Using R:

```
> 145.5 + qnorm(1/3)*SigY0
[1] 143.9545
```

We ordered just a bit less than the expected sales, since uncertainty seems low (the R^2 of the regression model is 98.41%).

11

Time Series Models

11.1 SOLUTIONS

Problem 11.1 The first step is finding the initial basis (level) $\hat{B}_0$ and trend $\hat{T}_0$. Using R, or any other tool, we find:

```
> X=1:3
> Y=c(35,50,60)
> mod=lm(Y~X); mod$coeff
(Intercept)           X
   23.33333    12.50000
```

Now we proceed with the parameter update process, starting from the first time bucket. Choosing $\alpha = 0.1$ and $\beta = 0.2$, we find

$$\begin{aligned}
\hat{B}_1 &= 0.1 \times 35 + 0.9 \times (23.33333 + 12.5) = 35.75 \\
\hat{T}_1 &= 0.2 \times (35.75 - 23.33333) + 0.2 \times 12.5 = 12.48333 \\
\hat{B}_2 &= 0.1 \times 50 + 0.9 \times (35.75 + 12.48333) = 48.41 \\
\hat{T}_2 &= 0.2 \times (48.41 - 35.75) + 0.8 \times 12.48333 = 12.51867 \\
\hat{B}_3 &= \ldots
\end{aligned}$$

All of the relevant results are reported in the following table:

t	Y_t	$\hat{B}_t$	$\hat{T}_t$	F'_t	e_t	$\lvert e_t \rvert / Y_t$	e_t^2
0	–	23.33333	12.5	–	–	–	–
1	35	35.75	12.48333	35.83333	–	–	–
2	50	48.41	12.51867	48.23333	–	–	–
3	60	60.8358	12.50009	60.92867	–	–	–
4	72	73.2023	12.47338	73.33589	-1.33589	0.018554	1.784611
5	83	85.40811	12.41986	85.67568	-2.67568	0.032237	7.159261
6	90	97.04518	12.2633	97.82797	-7.82797	0.086977	61.27717

Note that we do *not* compute forecasts and errors within the fit sample. The first forecast is calculated at the end of the fit sample, for time bucket 4:

$$F_{3,1} = F'_4 = \hat{B}_3 + \hat{T}_3 = 60.8358 + 12.50009 = 73.33589,$$

with forecast error

$$e_4 = Y_4 - F'_4 = 72 - 73.33589 = -1.33589.$$

Then we find the performance measures:

$$\text{MAPE} = \frac{1}{3} \times \left(\frac{\lvert e_4 \rvert}{Y_4} + \frac{\lvert e_5 \rvert}{Y_5} + \frac{\lvert e_6 \rvert}{Y_6} \right) = \frac{0.018554 + 0.032237 + 0.086977}{3} \approx 4.59\%$$

$$\text{RMSE} = \sqrt{\frac{e_4^2 + e_5^2 + e_6^2}{3}} = \sqrt{\frac{1.784611 + 7.159261 + 61.27717}{3}} = 4.8381.$$

The required forecasts are

$$F_{6,2} = \hat{B}_6 + 2\hat{T}_6 = 97.04518 + 2 \times 12.2633 = 121.5718$$
$$F_{6,3} = \hat{B}_6 + 3\hat{T}_6 = 97.04518 + 3 \times 12.2633 = 133.8351.$$

Problem 11.2 The fit sample consists of demand in time buckets $t = 1, 2, , \ldots, 8$. The average demand over the fit sample yields:

$$\hat{B}_0 = \frac{1}{8} \sum_{j=1}^{8} Y_j = \frac{21 + 27 + 41 + 13 + 19 + 32 + 42 + 12}{8} = 25.875$$

Then we estimate the seasonal factors (note they add up to 4):

$$\hat{S}_{-3} = \frac{Y_1 + Y_5}{2\hat{B}_0} = 0.77295$$
$$\hat{S}_{-2} = \frac{Y_2 + Y_6}{2\hat{B}_0} = 1.14010$$
$$\hat{S}_{-1} = \frac{Y_3 + Y_7}{2\hat{B}_0} = 1.60386$$
$$\hat{S}_{0} = \frac{Y_4 + Y_8}{2\hat{B}_0} = 0.48309.$$

Let us choose $\alpha = 0.1$ and $\gamma = 0.2$ and update parameters as follows:

$$\hat{B}_1 = 0.1 \times \frac{21}{0.77295} + 0.9 \times 25.875 = 26.004375$$
$$\hat{S}_1 = 0.2 \times \frac{21}{26.004375} + 0.8 \times 0.77295 = 0.77987$$
$$\hat{B}_2 = 0.1 \times \frac{27}{1.14010} + 0.9 \times 26.004375 = 25.772158$$
$$\hat{S}_2 = 0.2 \times \frac{27}{25.772158} + 0.8 \times 1.14010 = 1.121606$$
$$\hat{B}_3 = \ldots.$$

All of the relevant results are reported in the following table:

t	Y_t	$\hat{B}_t$	$\hat{S}_t$	F'_t	e_t	$\lvert e_t \rvert$	$\frac{e_t}{Y_t}$
-3	–	–	0.772947	–	–	–	–
-2	–	–	1.140097	–	–	–	–
-1	–	–	1.603865	–	–	–	–
0	–	25.875	0.483092	–	–	–	–
1	21	26.00438	0.779869	–	–	–	–
2	27	25.77216	1.121606	–	–	–	–
3	41	25.75127	1.601523	–	–	–	–
4	13	25.86714	0.486987	–	–	–	–
5	19	25.71673	0.771659	–	–	–	–
6	32	25.99811	1.143456	–	–	–	–
7	42	26.02081	1.604037	–	–	–	–
8	12	25.88286	0.482315	–	–	–	–
9	22	26.14557	0.785616	19.97273	2.027268	2.027268	0.092149
10	33	26.417	1.164604	29.89632	3.103681	3.103681	0.094051
11	38	26.14432	1.573923	42.37384	-4.37384	4.373842	-0.1151
12	10	25.60323	0.463967	12.6098	-2.6098	2.609803	-0.26098

The first forecast and error calculation is carried out after observing Y_8 for time bucket 9, the first one in the test sample:

$$F_{8,1} = F'_9 = \hat{B}_8 S_5 = 25.8829 \times 0.77166 = 19.9727$$
$$e_9 = 22 - 19.9727 = 2.0273.$$

The performance measures are:

$$\text{MAD} = \frac{|e_9| + |e_{10}| + |e_{11}| + |e_{12}|}{4} = \frac{2.0273 + 3.10368 + 4.373842 + 2.6098}{4} = 3.02865$$
$$\text{MPE} = \frac{1}{4}\left(\frac{e_9}{Y_9} + \frac{e_{10}}{Y_{10}} + \frac{e_{11}}{Y_{11}} + \frac{e_{12}}{Y_{12}}\right) = \frac{0.09215 + 0.094051 - 0.115101 - 0.26098}{4} = -4.747\%.$$

The required forecast is

$$F_{5,3} = \hat{B}_5 \hat{S}_4 = 25.7167 \times 0.486987 = 12.5237.$$

Problem 11.3 The main issue is that we *cannot* initialize B_0 with the average of the forst 7 time buckets. We must adjust for the fact that we use two observations for three seasons (quarters) within the seasonal cycle, and only one for the last one:

$$B_0 = \frac{1}{4}\left(\frac{40+46}{2} + \frac{28+30}{2} + \frac{21+29}{2} + 37\right) = 33.5,$$
$$S_{-3} = \frac{40+46}{2B_0} = 1.284,$$
$$S_{-2} = \frac{28+30}{2B_0} = 0.866,$$
$$S_{-1} = \frac{21+29}{2B_0} = 0.746,$$
$$S_0 = \frac{37}{B_0} = 1.104.$$

Check that the four seasonal factors add up to 4.

Problem 11.4 We want to apply the Holt–Winter method, assuming a cycle of one year and a quarterly time bucket, corresponding to ordinary seasons. We are at the beginning of summer and the current parameter estimates are

- Level 80
- Trend 10
- Seasonality factors: winter 0.8, spring 0.9, summer 1.1, autumn 1.2

On the basis of these estimates, what is your forecast for next summer? If the demand scenario (summer 88, autumn 121, winter 110) is realized, what are MAD and MAD%?

We recall from Section 11.5.5 the form of the forecast:

$$F_{t+h} = (\hat{B}_t + h\hat{T}_t) \cdot \hat{S}_{t+h-s}.$$

Hence, the forecast made at the end of spring (beginning of summer) for the incoming summer is:

$$F_{0,1} = F'_{\text{summer}} = (80 + 10) \times 1.1 = 99.$$

To calculate the required error measures, we first compute

$$e_{\text{summer}} = 88 - 99 = -11,$$

and then we go on updating level and trend estimates, using $\alpha = 0.1$ and $\beta = 0.2$ (we do not update the seasonal factors because we won't need them in this specific problem, since the horizon we analyze is less than one whole cycle):

$$\begin{aligned}
\hat{B}_1 &= 0.1 \times \frac{88}{1.1} + 0.9 \times (80 + 10) = 89\\
\hat{T}_1 &= 0.2 \times (89 - 80) + 0.8 \times 10 = 9.8\\
F_{1,1} &= F'_{\text{autumn}} = (89 + 9.8) \times 1.2 = 118.56\\
e_{\text{autumn}} &= 121 - 118.56 = 2.44\\
\hat{B}_2 &= 0.1 \times \frac{121}{1.2} + 0.9 \times (89 + 9.8) = 99.00333\\
\hat{T}_2 &= 0.2 \times (99.00333 - 89) + 0.8 \times 9.8 = 9.84067\\
F_{2,1} &= F'_{\text{winter}} = (99.00333 + 9.84067) \times 0.8 = 87.0752\\
e_{\text{winter}} &= 110 - 87.0752 = 22.9248.
\end{aligned}$$

Therefore:

$$\begin{aligned}
\text{MAD} &= \frac{11 + 2.44 + 22.9248}{3} = 12.1216,\\
\text{MAD\%} &= \frac{\text{MAD}}{\overline{Y}} = \frac{12.1216}{(88 + 121 + 110)/3} = 11.40\%.
\end{aligned}$$

Problem 11.5 We know that, if $|\beta| < 1$,

$$\sum_{k=0}^{+\infty} \beta^k = \frac{1}{1-\beta}.$$

Let us summ the weights in Eq. (11.18), where $(1-\alpha)$ plays the role of β:

$$\sum_{k=0}^{+\infty} \alpha(1-\alpha)^k = \alpha \cdot \frac{1}{1-(1-\alpha)} = 1.$$

Problem 11.6 The application of Eq. (11.30) is immediate, but let us prove the two results in a direct way:

$$\text{Cov}(Y_t, Y_{t+1}) = \text{Cov}(\mu + \epsilon_t - \theta_1 \epsilon_{t-1}, \mu + \epsilon_{t+1} - \theta_1 \epsilon_t) = -\theta_1 \sigma_\epsilon^2,$$

since the constant μ plays no role and the sequence $\{\epsilon_t\}$ consist of i.i.d. variables with variance σ_2. By the very same token

$$\text{Cov}(Y_t, Y_{t+2}) = 0$$

and the same applies to larger lags. Then

$$\rho_Y(1) = \frac{\text{Cov}(Y_t, Y_{t+1})}{\text{Var}(Y_t)} = \frac{-\theta_1 \sigma_\epsilon^2}{\sigma_\epsilon^2 + \theta_1^2 \sigma_\epsilon^2} = \frac{-\theta_1}{1 + \theta_1^2}.$$

Problem 11.7 The easy way to solve the problem is to take advantage of Eq. (11.30), which gives the autocorrelation function of the moving average process

$$Y_t = \mu + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q}.$$

as

$$\rho_Y(k) = \frac{\gamma_Y(k)}{\gamma_Y(0)} = \begin{cases} \dfrac{-\theta_k + \theta_1\theta_{k+1} + \cdots + \theta_{q-k}\theta_q}{1 + \theta_1^2 + \cdots + \theta_q^2}, & k = 1, 2, \ldots, q \\ 0, & k > q. \end{cases}$$

A moving average forecast is

$$F_{t,h} = \frac{1}{n} \sum_{\tau=t-n+1}^{t} X_\tau,$$

and in order to cast in the above framework we should set $q = n - 1$ and

$$\theta_j = -1, \qquad j = 1, \ldots, n-1.$$

Therefore, for forecasts that are no more than k time buckets apart, we have

$$\frac{-\theta_k + \theta_1\theta_{k+1} + \cdots + \theta_{n-1-k}\theta_{n-1}}{1 + \theta_1^2 + \cdots + \theta_{n-1}^2} = \frac{1 + (n-1-k)}{1 + (n-1)} = 1 - \frac{k}{n}.$$

To better get the intuition, consider the following representation of a moving average M_1 starting at $t = 1$ and a moving average M_{k+1} starting at $t = k + 1$:

$$\begin{array}{ccccccccc} X_1 & X_2 & \cdots & X_k & X_{k+1} & \ldots & X_{n-1} & X_n & \\ & & & & X_{k+1} & \ldots & X_{n-1} & X_n & \ldots X_{n+k+1} \end{array}$$

Clearly, only $n - k$ observations overlap. Therefore, taking into account of the variance σ^2 of each term and the division by n, we have

$$\begin{aligned} \mathrm{Cov}(M_1, M_{k+1}) &= \frac{1}{n^2} \sum_{\tau=k+1}^{n} \mathrm{Cov}(X_\tau, X_\tau) = \frac{n-k}{n^2}\sigma^2, \\ \mathrm{Var}(M_1) &= \mathrm{Var}(M_{k+1}) = \frac{1}{n}\sigma^2, \\ \rho(M_1, M_{k+1}) &= \frac{\sigma^2(n-k)/n^2}{\sigma^2/n} = \frac{n-k}{n} = 1 - \frac{k}{n}. \end{aligned}$$

12

Deterministic Decision Models

12.1 SOLUTIONS

Problem 12.1 We want to prove that, for $\lambda \in [0,1]$ and any $\mathbf{x}_1, \mathbf{x}_2$,

$$g(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) \leq \lambda g(\mathbf{x}_1) + (1-\lambda) g(\mathbf{x}_2).$$

Since each $f_i(\mathbf{x})$ is convex, we have

$$\begin{aligned}
g(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) &= \sum_{i=1}^{m} \alpha_i f_i(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) \\
&\leq \sum_{i=1}^{m} \alpha_i \left[\lambda f_i(\mathbf{x}_1) + (1-\lambda) f_i(\mathbf{x}_2)\right] \\
&= \lambda \sum_{i=1}^{m} \alpha_i f_i(\mathbf{x}_1) + (1-\lambda) \sum_{i=1}^{m} \alpha_i f_i(\mathbf{x}_2) \\
&= \lambda g(\mathbf{x}_1) + (1-\lambda) g(\mathbf{x}_2).
\end{aligned}$$

Also note that the above proof relies on non-negativity of each α_i as well.

Problem 12.2 Let us find the derivatives of f:

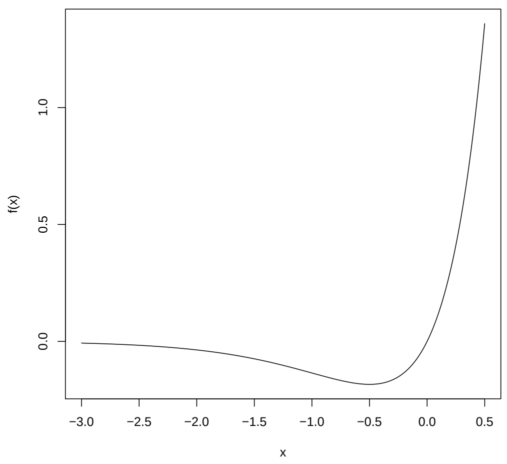
$$\begin{aligned}
f'(x) &= (1+2x)e^{2x} \\
f''(x) &= 4(1+x)e^{2x}.
\end{aligned}$$

We observe that

- The first-order derivative is negative for $x < 0.5$ and positive for $x > 0.5$; $x = 0.5$ is a stationary point.
- The second-order derivative is negative for $x < 1$ and positive for $x > 1$.

Thus, the function is not convex, as the second-order derivative is not always non-negative. However, $x = 0.5$ is globally optimal, as the function is decreasing to its left and increasing to its right (and locally convex around $x = 0.5$).

We may plot the function to get a clearer picture.



The plot may be obtained by the following R commands

```
> f <- function (x) x*exp(2*x)
> x <- seq(from=-3,to=0.5,by=0.01)
> plot(x,f(x),type="l")
```

Alternatively, you may use MATLAB:

```
>> f = @(x) x .* exp(2*x);
>> x = -3:0.01:0.5;
>> plot(x,f(x))
```

We observe that convexity is a sufficient condition to rule out issues with local optimality, but it is not necessary. Furthermore, the function f shares an important property with convex functions: The level sets, i.e., sets of the form

$$S_\alpha = \{x \mid f(x) \leq \alpha\},$$

are convex. In convex analysis, which is beyond the scope of the book, a function like f is called *pseudoconvex*.

Problem 12.3 The problem requires the minimization of the (convex) quadratic function

$$f(x, y, z) = x^2 + y^2 + z^2,$$

subject to two linear constraints. We associate Lagrange multipliers λ_1 and λ_2 with the two equality constraints. The Lagrangian function is

$$\mathcal{L}(x, y, z, \lambda_1, \lambda_2) = x^2 + y^2 + z^2 + \lambda_1(3x + y + z - 5) + \lambda_2(x + y + z - 1),$$

and the optimality conditions are a set of linear equations:

$$\frac{\partial \mathcal{L}}{\partial x} = 2x + 3\lambda_1 + \lambda_2 = 0$$
$$\frac{\partial \mathcal{L}}{\partial y} = 2y + \lambda_1 + \lambda_2 = 0$$
$$\frac{\partial \mathcal{L}}{\partial z} = 2z + \lambda_1 + \lambda_2 = 0$$
$$\frac{\partial \mathcal{L}}{\partial \lambda_1} = 3x + y + z - 5 = 0$$
$$\frac{\partial \mathcal{L}}{\partial \lambda_2} = x + y + z - 1 = 0.$$

Eliminating $y + z$ between the fourth and fifth equation we find $x = 2$. Eliminating $\lambda_1 + \lambda_2$ between the second and third equation we find $y = z$, which, plugged into the fifth equation yields immediately $x = y = -\frac{1}{2}$. This is enough to spot the optimal solution. If we also want to find the multipliers, subtracting the second equation from the first one yields

$$\lambda_1 = y - x = -2.5,$$

and then from the second equation we find

$$\lambda_2 = -\lambda_1 - 2 = 3.5.$$

The system of linear equations can be solved in MATLAB as follows:

```
>> A= [2 0 0 3 1; 0 2 0 1 1; 0 0 2 1 1;
3 1 1 0 0; 1 1 1 0 0];
>> b = [0;0;0;5;1];
>> A\b
ans =
    2.0000
   -0.5000
   -0.5000
   -2.5000
    3.5000
```

If you have the Optimization Toolbox, you may tackle the quadratic problem using `quadprog`:

```
>> H = 2*eye(3);
>> Aeq = [3 1 1; 1 1 1];
>> beq = [5;1];
>> [x,~,~,~,lambda] = quadprog(H,[],[],[],Aeq,beq);
>> x
x =
    2.0000
   -0.5000
   -0.5000
>> lambda.eqlin
ans =
```

```
 -2.5000
  3.5000
```

In R, these two tasks are carried out as follows:

```
> A <- rbind(c(2,0,0,3,1), c(0,2,0,1,1), c(0,0,2,1,1),
+       c(3,1,1,0,0), c(1,1,1,0,0))
> b <- c(0,0,0,5,1)
> solve(A,b)
[1]  2.0 -0.5 -0.5 -2.5  3.5
> library(quadprog)
> Qmat <- diag(c(2,2,2))
> bvet <- c(5,1)
> Amat <- cbind(c(3,1,1),c(1,1,1))
> result <- solve.QP(Qmat, c(0,0,0),Amat,bvet,meq=2)
> result
$solution
[1]  2.0 -0.5 -0.5
$Lagrangian
[1] 2.5 3.5
```

Note that, when dealing with equality constraints, there may be some ambiguity with the sign of multipliers.

Problem 12.4 Solve the optimization problem

$$\begin{aligned} \max \quad & xyz \\ \text{s.t.} \quad & x + y + z \le 1 \\ & x, y, z \ge 0 \end{aligned}$$

How can you justify intuitively the solution you find?

There is a clear symmetry in the problem, suggesting that there is an optimal solution such that

$$x^* = y^* = z^*.$$

On the domain $\mathbb{R}^3_+$ the objective function is increasing in all decision variables, suggesting that the inequality constraint is active in the optimal solution, implying

$$x^* = y^* = z^* = \frac{1}{3}.$$

This may be checked by the following MATLAB script

```
f = @(X) -X(1).*X(2).*X(3);
A = ones(1,3);
b = 1;
lb = zeros(3,1);
X0 = zeros(3,1);
out = fmincon(f,X0,A,b,[],[],lb)
```

which yields

```
out =
```

```
0.3333
0.3333
0.3333
```

However, we need more careful analysis to be sure that this is the actual global optimum, and not a local one.

We have four inequality constraints, and we should introduce four non-negative Lagrange multipliers. To streamline the task, a common strategy is to assume an interior solution, i.e., a solution in which all variables are strictly positive. This assumption may be checked a posteriori. Then, the Lagrangian function is

$$\mathcal{L}(x, y, z, \mu) = -xyz + \mu(x + y + z - 1),$$

and the stationarity conditions are

$$\frac{\partial \mathcal{L}}{\partial x} = -yz + \mu = 0$$
$$\frac{\partial \mathcal{L}}{\partial y} = -xz + \mu = 0$$
$$\frac{\partial \mathcal{L}}{\partial z} = -xy + \mu = 0,$$

which in fact confirms that at the optimum the decision variables should take the same value. Note that we change the sign of the objective function to cast the maximization problem in minimization form and apply the familiar conditions. Then we consider complementary slackness:

$$\mu(x + y + z - 1) = 0.$$

If we assume that the constraint is active, we in fact find the above solution,

$$x^* = y^* = z^* = \frac{1}{3},$$

and

$$\mu^* = \frac{1}{9} > 0.$$

If we assume that the constraint is not active, this implies $\mu = 0$, but this in turn implies

$$x^* = y^* = z^* = 0,$$

which is an alternative candidate solution, which is not interior. Then, we should consider a more complicated condition, but it is easy to see that solution is in fact feasible, but inferior to the alternative one.

Problem 12.5 Consider the constrained problem:

$$\begin{aligned} \min \quad & x^3 - 3xy \\ \text{s.t.} \quad & 2x - y = -5 \\ & 5x + 2y \geq 37 \\ & x, y \geq 0 \end{aligned}$$

- Is the objective function convex?

- Apply the KKT conditions; do we find the true minimizer?

It is easy to see that the objective function is not convex on $\mathbb{R}^2$. If we fix $y = 0$, i.e., if we imagine taking a section of the surface described by the objective function $f(x, y)$, we obtain the cubic function

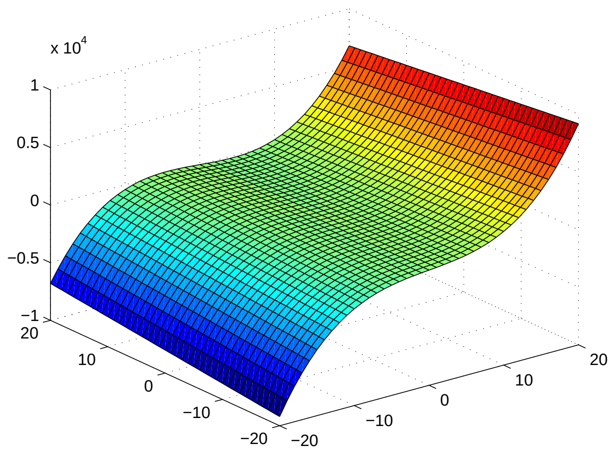
$$f(x, 0) = x^3,$$

which is not convex. However, this does not necessarily imply that the function is not convex over the feasible set. For instance, the cubic function is convex for $x \geq 0$.

We may use MATLAB to plot the function as follows:

```
f1 = @(x,y) x.^3 - 3*x.*y;
x = -20:1:20;
[X,Y] = meshgrid(x,x);
Z = f1(X,Y);
surf(X,Y,Z);
contour(X,Y,Z);
```

producing the following surface plot



We see that in fact the function is not convex, but it looks like it could be on a restricted domain. A formal check can be carried out by analyzing the Hessian matrix, which requires the calculation of the partial derivatives

$$\frac{\partial f}{\partial x} = 3x^2 - 3y, \qquad \frac{\partial^2 f}{\partial x^2} = 6x, \qquad \frac{\partial^2 f}{\partial y\, \partial x} = 0$$

$$\frac{\partial f}{\partial y} = -3x, \qquad \frac{\partial^2 f}{\partial y^2} = 0.$$

The Hessian matrix is

$$\begin{bmatrix} 6x & -3 \\ -3 & 0 \end{bmatrix},$$

and we should wonder if it is positive definite or at least semidefinite on $\mathbb{R}^2_+$. An easy way for doing so is to observe that the trace of the matrix is $6x \geq 0$ and its determinant is $9 > 0$. Since the trace is the sum of the eigenvalues and the determinant is the product, we see that eigenvalues are non-negative. This shows that the matrix is positive semidefinite, and the function is convex.

By the way, this is the same analysis that one carries out on the leading minors.

To find the optimal solution, let us introduce multipliers λ and $\mu \geq 0$ associated with the equality and inequality constraints and form the Lagrangian

$$\mathcal{L}(x, y, \lambda, \mu) = x^3 - 3xy + \lambda(2x - y + 5) + \mu(37 - 5x - 2y).$$

As in Problem 12.4 we streamline the Lagrangian by assuming an interior solution $x^*, y^* > 0$. The stationarity conditions are

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial x} &= 3x^2 - 3y + 2\lambda - 5\mu = 0 \\
\frac{\partial \mathcal{L}}{\partial y} &= -3x - \lambda - 2\mu = 0.
\end{aligned}$$

We should also take into account the conditions

$$\begin{aligned}
&2x - y = -5, \\
&\mu(37 - 5x - 2y) = 0, \\
&\mu \geq 0.
\end{aligned}$$

If we assume that the inequality constraint is active, we solve a system of two linear equations corresponding to the constraints and find

$$x = 3, \qquad y = 11.$$

If we plug this solution into the stationarity conditions, we find

$$\begin{aligned}
2\lambda - 5\mu &= -3x^2 + 3y = -27 + 33 = 6 \\
-\lambda - 2\mu &= 3x = 9,
\end{aligned}$$

which yields

$$\lambda = -\frac{11}{3}, \qquad \mu = -\frac{8}{3} < 0,$$

which is not acceptable. If we assume $\mu = 0$ and $5x + 2y > 37$, we have to solve the system

$$\begin{aligned}
&3x^2 - 3y + 2\lambda = 0 \\
&-3x - \lambda = 0 \\
&2x - y = -5.
\end{aligned}$$

From the second and third equation we find

$$\lambda = -3x, \qquad y = 2x + 5,$$

which, plugged into the first equation, yields

$$3x^2 - 3(2x + 5) - 6x = 3(x^2 - 4x - 5) = 0 \quad \Rightarrow \quad x_1 = 5, x_2 = -1.$$

The negative root must be discarded, and the positive one gives

$$y = 15, \qquad \lambda = -15.$$

The solution does satisfy the inequality constraint, as

$$5 \times 5 + 2 \times 15 => 37.$$

This may also be checked by the MATLAB script

```
f = @(X) X(1).^3 - 3*X(1).*X(2);
Aeq = [2, -1];
beq = -5;
A = [-5, -2];
b = -37;
lb = [0;0];
X0 = [Aeq;A]\[beq;b];
[X,~,~,~,lambda] = fmincon(f,X0,A,b,Aeq,beq,lb);
```

which yields

```
X =
    5.0000
   15.0000
lambda =
         eqlin: -15.0000
      eqnonlin: [0x1 double]
       ineqlin: 1.1112e-07
         lower: [2x1 double]
         upper: [2x1 double]
    ineqnonlin: [0x1 double]
```

Problem 12.6 Let us recall the single-period model, where we denote the amount of end product j sold by y_j, and the amount of raw material i used for j by x_{ij}, which is defined only for $i \in \mathcal{R}_j$:

$$\begin{aligned}
\max \quad & \sum_{j\in\mathcal{J}} p_j y_j - \sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{R}_j} c_i x_{ij} \\
\text{s.t.} \quad & y_j = \sum_{i\in\mathcal{R}_j} x_{ij}, && \forall j \in \mathcal{J}, \\
& L_j y_j \le \sum_{i\in\mathcal{R}_j} q_i x_{ij} \le U_j y_j, && \forall j \in \mathcal{J}, \\
& \sum_{j:i\in\mathcal{R}_j} x_{ij} \le I_i, && \forall i \in \mathcal{I}, \\
& 0 \le y_j \le d_j && \forall j, \in \mathcal{J}, \\
& x_{ij} \ge 0, && \forall j \in \mathcal{J}, i \in \mathcal{R}_j.
\end{aligned}$$

The first extension requires redefining variables in order to account for time. We use $t = 0, 1, 2, \dots, T$ to refer to time bucket t and let:

- y_{jt} be the amount of end item j blended and sold during time bucket $t = 1, \dots, T$
- x_{ijt} be the amount of raw material i used to blend end item j during time bucket $t = 1, \dots, T$

- I_{jt} be the on-hand inventory of raw material i *at the end* of time bucket $t = 1, \ldots, T$ and I_{j0} the *given* initial inventory (i.e., this is actually a parameter, not a decision variable)
- z_{jt} be the amount of raw material i purchased at the beginning of time bucket $t = 1, \ldots, T$

We also need introducing a few more parameters:

- d_{jt} the demand of end product i at time bucket t
- c_{jt} the unit cost of raw material j at time bucket t
- h_i the holding cost for raw material j

Then, the model reads as follows:

$$\begin{aligned}
\max \quad & \sum_{t=1}^{T} \sum_{j \in \mathcal{J}} p_j y_{jt} - \sum_{t=1}^{T} \sum_{j \in \mathcal{J}} \sum_{i \in \mathcal{R}_j} c_{it} x_{ijt} \\
\text{s.t.} \quad & y_{jt} = \sum_{i \in \mathcal{R}_j} x_{ijt}, && \forall j \in \mathcal{J},\, t = 1, \ldots, T, \\
& L_j y_{jt} \le \sum_{i \in \mathcal{R}_j} q_i x_{ijt} \le U_j y_{jt}, && \forall j \in \mathcal{J},\, t = 1, \ldots, T, \\
& I_{it} = I_{i,t-1} - \sum_{j : i \in \mathcal{R}_j} x_{ijt} + z_{jt}, && \forall i \in \mathcal{I},\, t = 1, \ldots, T, \\
& 0 \le y_{jt} \le d_{jt} && \forall j, \in \mathcal{J},\, t = 1, \ldots, T, \\
& x_{ijt} \ge 0, && \forall j \in \mathcal{J},\, i \in \mathcal{R}_j,\, t = 1, \ldots, T, \\
& z_{it}, I_{it} \ge 0, && \forall i \in \mathcal{I},\, t = 1, \ldots, T.
\end{aligned}$$

For the second part of the problem, let us assume that we have M tanks of capacity H. We also assume that the raw materials are measured by the same measurement unit and that we want to use one tank for each type of raw material. Then, we introduce binary variables

$$\delta_{it} = \begin{cases} 1, & \text{if we store raw material } i \text{ at the end of time bucket } t, \\ 0, & \text{otherwise.} \end{cases}$$

Then, we add an upper bound on the inventory holding,

$$I_{it} \le H \delta_{it}, \qquad \forall i \in \mathcal{I},\, t = 1, \ldots, T,$$

and the constraint

$$\sum_{i \in \mathcal{I}} \delta_{it} \le M, \qquad t = 1, \ldots, T.$$

If we relax the above assumptions, we may introduce an *integer* variable

$$\gamma_{it} \in \{0, 1, 2, 3, \ldots M\},$$

describing the *number* of tanks associated with raw material i at time bucket t and write

$$v_i I_{it} \le H \gamma_{it}, \qquad \forall i \in \mathcal{I},\, t = 1, \ldots, T,$$

where v_i is the unit volume of raw material i (if different measurement units are relevant), and

$$\sum_{i \in \mathcal{I}} \gamma_{it} \leq M, \qquad t = 1, \ldots, T.$$

Here we are not considering the need for cleaning a tank if we change the material stored; if necessary, we should introduce variables specifying which tank is used for which raw material, and accounting for a cleaning cost if this allocation is changed.

Problem 12.7 The model includes capacity constraints

$$\sum_{i=1}^{N} r_{im} x_{it} \leq R_{mt}, \qquad m = 1, \ldots, M, \;\; t = 1, \ldots, T,$$

where capacity may change over time according to a prescribed plan. In the problem we want to take this plan under control. Thus, R_m is given, but we introduce binary decision variables

$$\delta_{mt} = \begin{cases} 1, \text{if center } m \text{ is closed for maintenance during time bucket } t, \\ 0, \text{otherwise.} \end{cases}$$

The capacity constraint is now

$$\sum_{i=1}^{N} r_{im} x_{it} \leq R_m \delta_{mt}, \qquad m = 1, \ldots, M, \;\; t = 1, \ldots, T.$$

Maintenance of each center is enforced by requiring

$$\sum_{t=1}^{T} \delta_{mt} = 1, \qquad m = 1, \ldots, M,$$

and the limit on maintenance personnel is enforced by

$$\sum_{m=1}^{M} \delta_{mt} \leq 2, \qquad t = 1, \ldots, T.$$

If each center consists of multiple machines, it may be preferable not to shut down the whole center, but only a selected number of machines. If so, the above variable could be integer and represent the number of machines of center m subject to maintenance during time bucket t. If the time bucket is larger than the maintenance time, we may represent the loss of capacity in terms of actual availability reduction.

Problem 12.8 Let $\delta_i \in \{0, 1\}$ be a decision variable stating if activity δ_i, $i = 1, 2, 3, 4$, is started. The variable δ_1 can be activated only if *all* of the predecessor variables δ_2, δ_3, and δ_4 are started. This can be expressed in two ways:

$$3\delta_1 \leq \delta_2 + \delta_3 + \delta_4,$$

or

$$\delta_1 \leq \delta_i, \qquad i = 2, 3, 4.$$

They are both correct ways to represent the requirement, but it is easy to see that the first approach uses a single constraint that is just the sum of the three disaggregated constraints. In general, an aggregate constraint is a relaxation of the individual one. For instance, solutions satisfying both

$$g_1(\mathbf{x}) \le 0 \quad \text{and} \quad g_2(\mathbf{x}) \le 0$$

also satisfy the aggregate constraint

$$g_1(\mathbf{x}) + g_2(\mathbf{x}) \le 0,$$

but the converse is not true. When variables are restricted to binary values, as in the knapsack problem, the two feasible sets are the same, but when we run a commercial branch and bound algorithm based on LP relaxation, this is not true when variables $\delta \in \{0, 1\}$ are relaxed to $\delta \in [0, 1]$. This results in weaker lower bounds that are less effective in pruning the search tree. Therefore, the disaggregated form, even if it requires more constraints, is usually preferred.

Problem 12.9 We are looking for a way to model a logical `AND` of activities, rather than their exclusive or inclusive `OR`. In order to linearize the constraint $x_i = x_j x_k$, on the one hand x_i can be set to 1 only if x_j and x_k are:

$$x_i \le x_j, \qquad x_i \le x_k.$$

However, we must also *force* x_i to 1 only if x_j and x_k are set to 1:

$$x_i \ge x_j + x_k - 1.$$

This constraint is not relevant if $x_j x_k = 0$, but if $x_j x_k = 1$ it reads $x_i \ge 1$, forcing activity i to be started.

If we consider the `AND` of n variables,

$$y = \prod_{i=1}^{n} x_i$$

we may generalize the above idea as follows:

$$y \le x_i, \qquad i = 1, \ldots, n,$$
$$y \ge \sum_{i}^{n} x_i - n + 1.$$

Problem 12.10

- In the classical lot-sizing model, we implicitly assume that each customer order may be satisfied by items that were produced in different batches. In some cases, this is not acceptable; one possible reason is due to lot tracing; another possible reason is that there are little differences among batches (e.g., in color), that customers are not willing to accept. Then, we should explicitly account for individual order sizes and due dates. Build a model to maximize profit.

- As a final generalization, assume that customers are impatient and that they order different items together (each order consists of several lines, specifying item type and quantity). If you cannot satisfy the whole order immediately, it is lost. Build a model to maximize profit.

The first model requires the introduction of a "negative" inventory, typically called backlog, which is penalized more heavily than positive inventory holding. In other words, we associate a piecewise linear cost to inventory, which may be linearized by splitting I_{it} in two non-negative components:

$$I_{it} = I_{it}^{+} - I_{it}^{-}, \qquad I_{it}^{+}, I_{it}^{-} \geq 0.$$

The inventory balance constraint is

$$I_{it}^{+} - I_{it}^{-} = I_{i,t-1}^{+} - I_{i,t-1}^{-} + x_{it} - d_{it},$$

where x_{it} is the amount of item i produced during the time bucket t and d_{it} is demand. The inventory cost component of the cost function is

$$\sum_{t=1}^{T} \sum_{i=1}^{n} \left(h_i I_{it}^{+} + b_i I_{it}^{-} \right),$$

to which fixed charges related to setups should be added. The unit backlog cost $b_i > h_i$ limits the use of backlog.

If customers are willing to wait for at most two time buckets, we may apply the disaggregated formulation of page 683, where y_{itp} is the amount of item i produced during time bucket t to satisfy demand during time bucket $p \geq t$. We have just to define this variable only for $p \in \{t, t+1, t+2\}$. You may visualize the idea on the basis of Fig. 12.23: Rather than having all arcs outflowing from a node and moving toward South–East, we have at most three of them.

In the third model we need to introduce decision variables related to customer orders. Say that for each item i we have a collection of $\mathcal{O}_i$ orders indexed by j. Each order (i, j) is associated with a due date L_{ij}, which is the latest time at which we may produce items to satisfy that order. Then, we introduce a binary variable

$$\delta_{ijt} = \begin{cases} 1, & \text{if order } j,\ j = 1, \ldots, \mathcal{O}_i, \text{ of item } i \text{ is satisfied by production in } t \leq L_{ij}, \\ 0, & \text{otherwise.} \end{cases}$$

The variable is not defined for time buckets after the order due date. Satisfaction of demand is enforced by requiring

$$\sum_{t=1}^{L_{ij}} \delta_{ijt} = 1, \qquad \forall i, \forall j \in \mathcal{O}_i.$$

We also link these binary variables to the lot sizes

$$x_{it} = \sum_{j \in \mathcal{O}_i : t \leq L_{ij}} q_{ij} \delta_{ijt}, \qquad \forall t,$$

where q_{ij} is the order quantity and the sum is restricted to orders with a due date not earlier than t.

In the last case we must associate a binary variable to an order $j \in \mathcal{O}$ which is not associated with a single product. Rather, each order j has a due date L_j, a profit contribution

p_j, and order lines q_{ij} specifying the ordered amount for each end item (some of these quantities are zero). If we do not assume demand satisfaction, we have to rewrite the problem in maximization form, where we subtract inventory holding and setup costs from the total profit contribution

$$\sum_{j \in \mathcal{O}} p_j \delta_j,$$

where δ_j is 1 if the corresponding order is satisfied, 0 otherwise. We may also introduce decision variables z_{it} representing the amount of item i sold during time bucket t. These variables are related to the binaries as follows:

$$z_{it} = \sum_{j \in \mathcal{O}} q_{ij} \delta_j.$$

The inventory balance constraint is

$$I_{it} = I_{i,t-1} + x_{it} - z_{it}, \qquad \forall i, \forall t.$$

Problem 12.11 The essential decision variables in this model are not really changed and are the portfolio weights,

$$w_i \geq 0, \qquad i = 1, \ldots, n,$$

where non-negativity reflects the fact that sort-selling is forbidden. What is changed is that the approach is data-driven and we only rely on the time series r_{it}, $t = 1, \ldots, T$. The expected return of asset i is estimated by the sample mean

$$\hat{\mu}_i = \frac{1}{T} \sum_{t=1}^{T}, \qquad i = 1, \ldots, n.$$

By the same token, the value of the objective function,

$$\mathrm{E}\left\{ \left| \sum_{i=1}^{n} R_i w_i - \mathrm{E}\left[\sum_{k=1}^{n} R_k w_k \right] \right| \right\} = \mathrm{E}\left\{ \left| \sum_{i=1}^{n} (R_i - \mu_i) w_i \right| \right\},$$

is estimated as

$$\frac{1}{T} \sum_{t=1}^{T} \left| \sum_{i=1}^{n} (r_{it} - \hat{\mu}_i) w_i \right|.$$

This is a piecewise linear function that can be linearized by a common trick, based on non-negative deviation variables y_t and z_t related to surplus and shortfall with respect to the mean. By introducing the constraint

$$\sum_{i=1}^{n} (r_{it} - \hat{\mu}_i) w_i = y_t - z_t, \qquad t = 1, \ldots, T,$$

we may write the objective function as

$$\frac{1}{T} \sum_{t=1}^{T} (y_t + z_t).$$

Clearly, the leading $1/T$ can be disregarded when optimizing the portfolio.

The (conditional) lower bounds on asset holding and the cardinality constraint require the introduction of binary variables

$$\delta_i = \begin{cases} 1, \text{if asset } i \text{ is included in the portfolio,} \\ 0, \text{otherwise.} \end{cases}$$

Let q_i be the lower bound on the weight of asset i Let us also introduce a collection of asset subsets $\mathcal{S}_j$, $j = 1, \dots, m$, associated with lower and upper bounds L_j and U_j on holding. Then the model reads as follows:

$$\begin{aligned}
\min \quad & \sum_{t=1}^{T} (y_t + z_t) \\
st \quad & \sum_{i=1}^{n} (r_{it} - \hat{\mu}_i) w_i = y_t - z_t, \qquad t = 1, \dots, T, \\
& \sum_{i=1}^{n} \hat{\mu}_i w_i \geq R_{\min} \\
& \sum_{i=1}^{n} w_i = 1 \\
& L_j \leq \sum_{i \in \mathcal{S}_j} w_i \leq U_j, \qquad j = 1, \dots, m, \\
& \sum_{i=1}^{n} \delta_k \leq K, \\
& \delta_i q_i \leq w_i \leq \delta_i, \qquad i = 1, \dots, n, \\
& w_i, y_t, z_t \geq 0, \qquad \delta_i \in \{0, 1\},
\end{aligned}$$

where $R_{\min}$ is the minimum target on expected return and K is the maximum cardinality.

The danger of this modeling framework lies in the overfitting possibility with respect to available data.

Problem 12.12 The starting point is the network flow model of Section 12.2.4. Here we have to define a set of commodities defined by the source-destination pair (s, d). Let:

- $k = 1, \dots, M$ be the index of commodities
- s_k and d_k the source and the destination node of commodity k, respectively
- F_k the required flow (packets from s_k to d_k)

The flow variables should be redefined as x_{ij}^k, the amount of flow of commodity k on arc $(i, j) \in \mathcal{A}$. The equilibrium constraints are

$$\begin{aligned}
& \sum_{(i,j) \in \mathcal{A}} x_{ij}^k = \sum_{(j,i) \in \mathcal{A}} x_{ij}^k, \qquad k = 1, \dots, M; \forall i \in \mathcal{A}; i \notin \{s_k, d_k\}, \\
& \sum_{(i,j) \in \mathcal{A}} x_{ij}^k = F_k, \qquad k = 1, \dots, M; \ i = s_k, \\
& \sum_{(i,j) \in \mathcal{A}} x_{ij}^k = -F_k, \qquad k = 1, \dots, M; \ i = d_k.
\end{aligned}$$

The capacity constraint for each node $i \in \mathcal{N}$ is

$$\sum_{k=1}^{M} \sum_{(i,j)\in\mathcal{A}} x_{ij}^k \le (0.9 + 0.1\delta_i)C_i, \qquad \forall i \in \mathcal{N},$$

where C_i is the capacity of the node and δ_i is a binary variable set to 1 if we use more than 90% of the node capacity. By a similar token we define a binary variable γ_{ij} for arcs and write the capacity constraint

$$\sum_{k=1}^{M} x_{ij}^k \le (0.9 + 0.1\gamma_{ij})B_{ij}, \qquad \forall (i,j) \in \mathcal{A},$$

where B_{ij} is the arc capacity. The objective function is

$$\min \sum_{i\in\mathcal{N}} \delta_i + \sum_{(i,j)\in\mathcal{A}} \gamma_{ij}.$$

The capacity expansion problem, per se, is not difficult to deal with, as we could introduce binary variables, e.g., $z_{i,25}$ and $z_{i,70}$, representing the expansion of nodes, as well as $y_{ij,25}$ and $y_{ij,70}$ for arcs. These variables are mutually exclusive,

$$z_{i,25} + z_{i,70} \le 1, \qquad \forall i \in \mathcal{N}, y_{ij,25} + y_{ij,70} \qquad \le 1, \qquad \forall (i,j) \in \mathcal{A}.$$

The tradeoff may be explored by setting a budget for the overall expansion cost,

$$\sum_{i\in\mathcal{N}} (Q_{i,25}z_{i,25} + Q_{i,70}z_{i,70}) + \sum_{(i,j)\in\mathcal{A}} (P_{ij,25}y_{ij,25} + P_{ij,70}y_{ij,70}) \le B(i,j) \in \mathcal{A},$$

where Q and P are the node and arc expansion costs, respectively, for the two expansion levels, and B is the budget. By perturbing the budget and minimizing congestion we may explore their tradeoffs.

The tricky part is the interaction with congestion. For instance, the capacity constraint becomes nonlinear:

$$\sum_{k=1}^{M} \sum_{(i,j)\in\mathcal{A}} x_{ij}^k \le (0.9 + 0.1\delta_i)(1 + 0.25z_{i,25} + 0.70z_{i,70})C_i, \qquad \forall i \in \mathcal{N}.$$

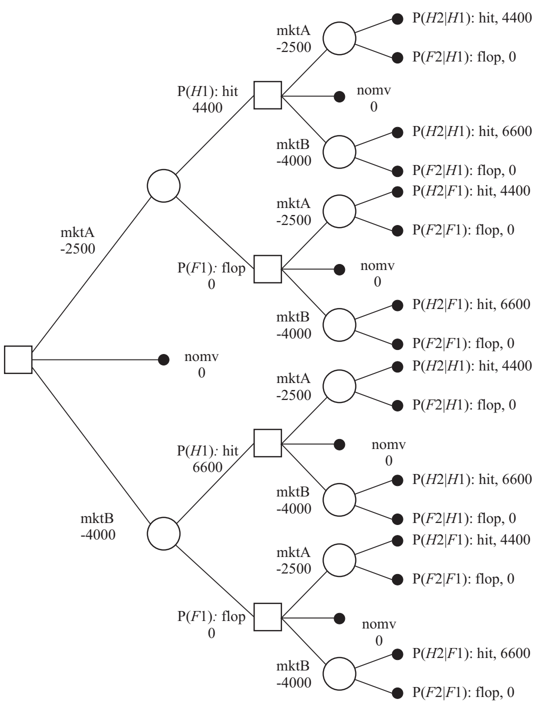
A similar issue arises for arc capacities. However, we may linearize the constraint as we have shown in Problem 12.9.

13

Decision Making Under Risk

13.1 SOLUTIONS

Problem 13.1



The above tree can be used to formalize the problem. Note that we associate cash flows and conditional probabilities to nodes as stated in the labels. A full-fledged software tool would use a more precise formalism, as it is also necessary to discount cash flows.

Problem 13.2 If we do not wait to improve the product, the expected profit is

$$\bar{\pi}_{\text{nowait}} = 0.6 \times 10 + 0.4 \times (-2) = €5.2 \text{ million.}$$

If we improve the product, there are alternative scenarios, depending on product success and competitor's entrance:

- **No competitor entrance**. In this case, the undiscounted expected profit is

  $$0.9 \times 10 + 0.1 \times (-2) = €8.8 \text{ million.}$$

  We must discount it and subtract the additional cost:

  $$\bar{\pi}_{\text{noentry}} = -0.5 + \frac{8.8}{1.05} = €7.880952 \text{ million.}$$

- **Competitor entrance**. In this case, the undiscounted expected profit is

  $$0.9 \times 5 + 0.1 \times (-4) = €4.1 \text{ million.}$$

  We must discount it and subtract the additional cost:

  $$\bar{\pi}_{\text{entry}} = -0.5 + \frac{4.1}{1.05} = €3.404762 \text{ million.}$$

The expected value of profit if we wait, and the entrance probability p is 50%, is

$$\bar{\pi}_{\text{wait}} = (1-p)\times 7.880952 + p\times 3.404762 = 0.5\times 7.880952 + 0.5\times 3.404762 = €5.642857 \text{ million.}$$

Since this is larger than €5.2 million, if we assume risk neutrality we should wait.

We are indifferent between waiting or not when $\bar{\pi}_{\text{wait}} = \bar{\pi}_{\text{nowait}}$, i.e., when

$$(1-p) \times 7.880952 + p \times 3.404762 = 5.2.$$

Solving for the entrance probability p, we find

$$p = 0.5989362.$$

Thus, for $p > 59.9\%$, it would be better not to wait (again, assuming risk neutrality).

Problem 13.3 If you play it safe, you earn \$100,000 for sure. Let X be the random fee if you choose the risky portfolio, with random return R. Then

$$\begin{aligned} \mathrm{E}[X] =& 0 \times \mathrm{P}\{R < 0\} + 50{,}000 \times \mathrm{P}\{0 \le R < 0.03\} + 100{,}000 \times \mathrm{P}\{0.03 \le R < 0.09\} \\ &+ 200{,}000 \times \mathrm{P}\{0.09 \le R\}. \end{aligned}$$

The probabilities can be calculated by standardization and use of statistical tables. Let us take a quicker route using R:

```
> p1=pnorm(0,mean=0.08,sd=0.1);p1
[1] 0.2118554
> p2=pnorm(0.03,mean=0.08,sd=0.1)-p1;p2
[1] 0.09668214
> p3=pnorm(0.09,mean=0.08,sd=0.1)-pnorm(0.03,mean=0.08,sd=0.1);p3
[1] 0.2312903
```

```
> p4=1-pnorm(0.09,mean=0.08,sd=0.1);p4
[1] 0.4601722
```

Then, we compute the weighted sum giving the expected value of the fee:

```
> probs=c(p1,p2,p3,p4)
> fee=c(0,50,100,200)*1000
> m=sum(probs*fee);m
[1] 119997.6
```

Since this is larger than the sure fee above, a risk-neutral manager would choose the active portfolio.

The standard deviation is

$$\mathrm{E}[X^2] - \mathrm{E}^2[X],$$

where

$$\begin{aligned}\mathrm{E}[X^2] =& 0^2 \times \mathrm{P}\{R < 0\} + 50{,}000^2 \times \mathrm{P}\{0 \le R < 0.03\} + 100{,}000^2 \times \mathrm{P}\{0.03 \le R < 0.09\} \\ &+ 200{,}000^2 \times \mathrm{P}\{0.09 \le R\}.\end{aligned}$$

Using R:

```
> stdev=sqrt(sum(probs*bonus^2)-m^2);stdev
[1] 81006.66
```

As we may notice, this is rather significant. Arguably, a fairly risk-averse manager would not take any chances.

Problem 13.4 Let

$$u(x) = x - \frac{\lambda}{2}x^2.$$

Then

$$\begin{aligned}U(X) &= 0.20 \times u(10000) + 0.50 \times u(50000) + 0.30 \times u(100000) \\ &= 0.20 \times 9666.67 + 0.50 \times 41666.67 + 0.30 \times 66666.67 = 42766.67.\end{aligned}$$

To find the certainty equivalent C, we solve the quadratic equation

$$u(C) = 42766.67,$$

i.e.,

$$-\frac{\lambda}{2}x^2 + x - 42766.67 = 0.$$

The two solutions are

$$c_1 = 248336.16, \quad c_2 = 51663.84.$$

The first solution is associated with the decreasing part of the quadratic utility function. Hence the certainty equivalent is given by the second root.

The expected value of the lottery is

$$\mathrm{E}[X] = 0.20 \times 10000 + 0.5050000 + 0.30 \times 100000 = 57000,$$

and the risk premium is

$$\rho = 57000 - 51663.84 = 5336.16.$$

Problem 13.5 Let c be the insurance premium. If you buy insurance, your wealth is surely

$$100{,}000 - c.$$

We should compare the utility of this sure amount with the expected utility if you do not buy insurance. The threshold premium is such that you are indifferent between the two possibilities:

$$\log(100{,}000 - c) = 0.95 \times \log 100{,}000 + 0.04 \times \log 50{,}000 + 0.01 \times \log 1 = 11.3701.$$

Now we solve for c and find

$$c = 100{,}000 - e^{11.3701} \approx 13{,}312.$$

Problem 13.6 The utility function is

$$u(x) = -e^{-\lambda x}.$$

Let Q be the amount of the initial wealth W_0 that is allocated to the risky asset. The future wealth W is given by

$$W = Q(1 + R) + (W_0 - Q)(1 + r_\mathrm{f}) = Q(R - r_\mathrm{f}) + W_0(1 + r_\mathrm{f}),$$

where r_f is the risk-free rate of return and R is the random rate of return from the risky asset. Given the assumed distribution, the expected utility is

$$\begin{aligned}\mathrm{E}\big[u(W)\big] &= -p_\mathrm{u} \exp\left\{-\alpha\left[Q(R_\mathrm{u} - r_\mathrm{f}) + W_0(1 + r_\mathrm{f})\right]\right\} - p_\mathrm{d} \exp\left\{-\alpha\left[Q(R_\mathrm{d} - r_\mathrm{f}) + W_0(1 + r_\mathrm{f})\right]\right\}\\ &= -\exp\left[-\alpha W_0(1 + r_\mathrm{f})\right] \cdot \left\{p_\mathrm{u} \exp\left[-\alpha Q(R_\mathrm{u} - r_\mathrm{f})\right] p_\mathrm{d} \exp\left[-\alpha Q(R_\mathrm{d} - r_\mathrm{f})\right]\right\}.\end{aligned}$$

We should take the derivative with respect to Q and enforce the stationarity condition, but we clearly see that W_0 occurs only within the leading exponential, which does not depend on the decision variable Q. Thus, whatever initial wealth W_0 we are endowed with, the optimal wealth allocated to the risky asset is the same, which is a rather weird conclusion. This makes the exponential utility a rather questionable way of modeling preferences.

Problem 13.7 The loss on the portfolio (in thousands of dollars) is related to the two rates of return by

$$L_p = -(150R_d + 200R_m).$$

Its standard deviation is

$$\begin{aligned}\sigma_p &= \sqrt{150^2 \times \sigma_d^2 + 200^2 \times \sigma_m^2 + 2 \times \rho \times 150 \times \sigma_d \times 200 \times \sigma_m}\\ &= \sqrt{150^2 \times 0.02^2 + 200^2 \times 0.03^2 + 2 \times 0.8 \times 150 \times 0.02 \times 200 \times 0.03}\\ &= 8.590693.\end{aligned}$$

Since $z_{0.99} = 2.326348$,

$$\mathrm{VaR}_{0.99} = 2.326348 \times 8590.693 = \$19984.94.$$

Problem 13.8 Let $W_p = W_1 + W_2$ the portfolio wealth, which is the sum of the wealth allocated to the two stock shares. The loss on the portfolio is just the sum of the loss on the two positions, depending on the random rate of return:

$$L_p = -W_p R_p = -W_1 R_1 - W_2 R_2 = -L_1 - L_2.$$

On a short-term horizon, we have

$$\mathrm{E}[L_1] = \mathrm{E}[L_2] = 0.$$

Let

$$\mathrm{Var}(L_p) = \sigma_p^2, \quad \mathrm{Var}(L_1) = \sigma_1^2, \quad \mathrm{Var}(L_2) = \sigma_2^2, \quad \mathrm{Cov}(L_1, L_2) = \rho\sigma_1\sigma_2.$$

The portfolio VaR at level $1-\alpha$, under a normality assumption, is just the corresponding quantile of loss,

$$\mathrm{VaR}_{p,1-\alpha} = z_{1-\alpha}\sigma_p.$$

By the same token, on the two positions we have

$$\mathrm{VaR}_{1,1-\alpha} = z_{1-\alpha}\sigma_1, \quad \mathrm{VaR}_{2,1-\alpha} = z_{1-\alpha}\sigma_2.$$

But we have

$$\begin{aligned}
\sigma_p &= \sqrt{\sigma_1^2 + 2\rho\sigma_1\sigma_2 + \sigma_2^2} \\
&\leq \sqrt{\sigma_1^2 + 2\sigma_1\sigma_2 + \sigma_2^2} \\
&= \sqrt{(\sigma_1 + \sigma_2)^2} \\
&= \sigma_1 + \sigma_2,
\end{aligned}$$

where the inequality depends on $\rho \leq 1$.

Thus we have

$$\mathrm{VaR}_{p,1-\alpha} = z_{1-\alpha}\sigma_p.$$

By the same token, on the two positions we have

$$\mathrm{VaR}_{p,1-\alpha} \geq \mathrm{VaR}_{1,1-\alpha} + \mathrm{VaR}_{2,1-\alpha},$$

with equality in the case of perfect positive correlation, so that VaR is subadditive in this case.

Clearly, in the normal case VaR does not tell a different story than standard deviation. The above reasoning also show that standard deviation is subadditive, but it does not necessarily tell the whole story for a generic distribution.

Problem 13.9 We recall the deterministic model for covenience:

$$\begin{aligned}
\min \quad & \sum_{i\in\mathcal{S}} f_i y_i + \sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{D}} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{i\in\mathcal{S}} x_{ij} = d_j, & \forall j \in \mathcal{D} \\
& \sum_{j\in\mathcal{D}} x_{ij} \leq R_i y_i, & \forall i \in \mathcal{S} \\
& x_{ij} \geq 0, \ y_i \in \{0, 1\}.
\end{aligned}$$

Let us introduce demand uncertainty, represented by a set of scenarios indexed by s, characterized by probability π^s and demand realization d_j^s for each destination node j. Then, we move the transportation decisions to the second decision stage and denote them by x_{ij}^s.

The minimization of the total plant cost plus the expected transportation cost is obtained by solving the following model:

$$
\begin{aligned}
\min \quad & \sum_{i\in\mathcal{S}} f_i y_i + \sum_s \pi^s \left(\sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{D}} c_{ij} x_{ij}^s \right), \\
\text{s.t.} \quad & \sum_{i\in\mathcal{S}} x_{ij}^s = d_j^s \qquad \forall s,\ \forall j \in \mathcal{D}, \\
& \sum_{j\in\mathcal{D}} x_{ij}^s \le R_i y_i \qquad \forall s,\ \forall i \in \mathcal{S}, \\
& x_{ij}^s \ge 0,\ y_i \in \{0,1\}.
\end{aligned}
$$

here, capacity constraints link the first-stage variables y_i with the second-stage variables x_{ij}^s.

Unfortunately this naive extension is hardly satisfactory, since it reqires demand satisfaction for *every* possible scenario. Thus, it may yield a very costly solution, if extreme but unlikely high-demand scenarios are included.

We should consider a more "elastic" formulation allowing for the possibility of leaving some demand unsatisfied (at least in some high-demand scenarios). Let $z_j^s \ge 0$ be the amount of unmet demand at node j under scenario s; these decision variables are included in the objective function multiplied by a penalty coefficient β_j, yielding the elastic model formulation:

$$
\begin{aligned}
\min \quad & \sum_{i\in\mathcal{S}} f_i y_i + \sum_s \pi^s \left(\sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{D}} c_{ij} x_{ij}^s \right) + \sum_s \pi^s \left(\sum_{j\in\mathcal{D}} \beta_j z_j^s \right), \\
\text{s.t.} \quad & \sum_{i\in\mathcal{S}} x_{ij}^s + z_j^s = d_j^s \qquad \forall s,\ \forall j \in \mathcal{D}, \\
& \sum_{j\in\mathcal{D}} x_{ij}^s \le R_i y_i \qquad \forall s,\ \forall i \in \mathcal{S}, \\
& x_{ij}^s, z_j^s \ge 0,\ y_i \in \{0,1\}.
\end{aligned}
$$

The penalty coefficient β_j could be quantified by taking the relative importance of different markets into account; alternatively, it could be related to the cost of meeting demand by resorting to external suppliers.

Problem 13.10 In this model, decision variables at the first stage express how to position containers on the network nodes. At the second stage, transportation demand is observed and recourse actons are taken. It is fundamental to clarify data and decision variables.

Data:

- I_i^0 the number of available containers at node i before positioning (the initial inventory)
- d_{ij}^s the transportation demand from node i to node j in scenario s
- π^s the probability of scenario s
- V the container capacity in volume

- c_{ij} the cost of moving a container from node i to node j
- K the fixed-charge to rent a container (we assume that it does not depend on node i and path $i \to j$)

Decision variables:

- $I_i \geq 0$ the non-negative number of available containers at node i after first-stage positioning
- $X_{ij} \in \mathbb{Z}_+$ the non-negative integer number of containers moved from i to j at the first stage to improve positioning
- $Y_{ij}^s \in \mathbb{Z}_+$ the non-negative integer number of containers moved from i to j in scenario s at the second stage to satisfy transportation demand
- $Z_{ij}^s \in \mathbb{Z}_+$ the non-negative integer number of containers that are rented at second stage, in scenario s, to satisfy demand from i to j

Now we write the model constraints:

- Flow balance of containers at each node i, at first stage:

$$I_i = I_i^0 + \sum_{j \neq i} X_{ji} - \sum_{j \neq i} X_{ij}, \qquad \forall i.$$

- Link between first- and second-stage decisions, i.e., we cannot use more containers than available at each node, in each scenario:

$$\sum_{j \neq i} Y_{ij}^s \leq I_i, \qquad \forall i, \forall s.$$

- Satisfaction of demand, for each node pair and each scenario, using rented containers at each node:

$$V\left(Y_{ij}^s + Z_{ij}^s\right) \geq d_{ij}^s, \qquad \forall i, \forall j \neq i, \forall s.$$

Finally, we write the objective function to be minimized, i.e., sum of the first-stage cost and expected second-stage cost:

$$\min \sum_{(i,j):i \neq j} c_{ij} X_{ij} + K \sum_s \pi^s \left[\sum_{(i,j):i \neq j} Z_{ij}^s \right].$$

14

Advanced Regression Models

14.1 SOLUTIONS

Problem 16.1 We recall the formula for the covariance matrix of the estimators:

$$\text{Cov}(\mathbf{b}) = \sigma^2 \left(\mathcal{X}^\mathsf{T}\mathcal{X}\right)^{-1}.$$

For a simple regression model, the components of **b** are the intercept a and the slope b, and the data matrix collects the n observations of x, as well as a leading column of ones:

$$\mathcal{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}.$$

Therefore,

$$\mathcal{X}^\mathsf{T}\mathcal{X} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} = \begin{bmatrix} n & n\bar{x} \\ n\bar{x} & \sum_{i=1}^{n} x_i^2 \end{bmatrix}.$$

To invert this 2×2 matrix, we may use the following shortcut:

- Swap the two elements on the diagonal
- Change the sign of the two elements on the antidiagonal
- Divide by the determinant of the matrix

Hence,

$$\left(\mathcal{X}^\mathsf{T}\mathcal{X}\right)^{-1} = \frac{1}{n\sum_{i=1}^{n} x_i^2 - n^2\bar{x}^2} \begin{bmatrix} \sum_{i=1}^{n} x_i^2 & -n\bar{x} \\ -n\bar{x} & n \end{bmatrix}.$$

Therefore, by considering the element in position $(2,2)$, we find the formula of Eq. (10.16),

$$\mathrm{Var}(b) = \frac{\sigma^2 n}{n\sum_{i=1}^{n} x_i^2 - n^2\bar{x}^2} = \frac{\sigma^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} = \frac{\sigma^2}{\sum_{i=1}^{n} \left(x_i^2 - \bar{x}\right)^2}.$$

To find the formula of Eq. (10.18), we just need a bit more of manipulation of the element in position $(1,1)$:

$$\begin{aligned}
\mathrm{Var}(a) &= \frac{\sigma^2 \sum_{i=1}^{n} x_i^2}{n\sum_{i=1}^{n} x_i^2 - n^2\bar{x}^2} = \frac{\sigma^2}{n}\left[\frac{\sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}\right] \\
&= \frac{\sigma^2}{n}\left[\frac{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2 + n\bar{x}^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}\right] = \frac{\sigma^2}{n}\left[1 + \frac{n\bar{x}^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}\right] \\
&= \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n} \left(x_i^2 - \bar{x}\right)^2}\right].
\end{aligned}$$

Problem 16.2 The logistic function

$$f(z) = \frac{\exp(z)}{1+\exp(z)}$$

is clearly non-negative, since $\exp(z) > 0$ for any z, and its domain is the entire real line.

The behavior for $z \to \pm\infty$ depends on the behavior of its building block:

$$\begin{aligned}
\lim_{z\to-\infty} \exp(z) = 0 \quad &\Rightarrow \quad \lim_{z\to-\infty} \frac{\exp(z)}{1+\exp(z)} = 0 \\
\lim_{z\to+\infty} \exp(z) = +\infty \quad &\Rightarrow \quad \lim_{z\to+\infty} \frac{\exp(z)}{1+\exp(z)} = \lim_{z\to+\infty} \frac{\exp(z)}{\exp(z)} = 1.
\end{aligned}$$

For $z = 0$, $f(0) = 0.5$, i.e., the function crosses the vertical axis of ordinates y at $y = 1$.

Let us find the first- and second-order derivatives. By using the formula for the derivative of a ratio of functions, we find

$$f'(z) = \frac{\exp(z)\bigl(1+\exp(z)\bigr) - \exp(z)\exp(z)}{\bigl(1+\exp(z)\bigr)^2} = \frac{\exp(z)}{\bigl(1+\exp(z)\bigr)^2} > 0.$$

Hence, the function is monotonically increasing. By a similar token,

$$f''(z) = \frac{\exp(z)\bigl(1+\exp(z)\bigr)^2 - \exp(z)\cdot 2\bigl(1+\exp(z)\bigr)\exp(z)}{\bigl(1+\exp(z)\bigr)^4} = \frac{\exp(z)\bigl(1-\exp(z)\bigr)}{\bigl(1+\exp(z)\bigr)^3}.$$

For $z < 0$, $\exp(z) < 1$, the second-order derivative is positive, and the function is convex. For $z > 0$, $\exp(z) > 1$, the second-order derivative is negative, and the function is concave. There is an inflection point at $z = 0$.

All of this is consistent with the plot of Fig. 16.1.

Appendix A
R – A software tool for statistics

R is a statistical computing which can be downloaded for free.[1] To install the software, you just have to donwload the installer from the web site and follow the instructions.

There is a wide and growing set of libraries implementing an array of quite sophisticated methods, but a minimal application is finding quantiles of normal distributions, which is obtained by the function `qnorm`:

```
> qnorm(0.95)
[1] 1.644854
> qnorm(0.95,20,10)
[1] 36.44854
```

In this snapshot you see the R prompt (¿) which is displayed in the command window when you start the software. The first command returns $z_{0.95}$, i.e., the 95% quantile for the standard normal distribution. In the second case, we provide additional parameters corresponding to $\mu = 20$ and $\sigma = 10$. If you need the CDF, use `pnorm`:

```
> pnorm(0)
[1] 0.5
> pnorm(3)
[1] 0.9986501
> pnorm(20,15,10)
[1] 0.6914625
```

[1] R Development Core Team (2010). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org`

Generally speaking, given a distribution name, such as `norm`, the prefix `p` selects the CDF, and the prefix `q` selects the quantile function. The same applies to the `t` distribution, in which case we need to specifies the degrees of freedom:

```
> qt(0.95,5)
[1] 2.015048
> qt(0.95,500)
[1] 1.647907
```

If needed, we may also generate random samples, with prefix `r`:

```
> rnorm(5,20,30)
[1]  26.67444  18.84493 -24.30770  26.03461  35.93114
```

Finally, to get quantiles from the chi-square distribution:

```
> qchisq(0.95,4)
[1] 9.487729
```

or the F distribution:

```
> qf(0.95,3,5)
[1] 5.409451
```

Appendix B
Introduction to MATLAB

MATLAB[1] is a powerful environment for numerical computing, originally developed as an outgrowth of a package for linear algebra. This explains why MATLAB was built around vectors and matrices, even though now it has much more sophisticated data structures. For the purposes of the book, a grasp of the basics is more than enough.

B.1 WORKING WITH VECTORS AND MATRICES IN THE MATLAB ENVIRONMENT

- MATLAB is an interactive computing environment. You may enter expressions and obtain an immediate evaluation:

  ```
  >> rho = 1+sqrt(5)/2
  rho =
      2.1180
  ```

  By entering a command like this, you also define a variable `rho` which is added to the current environment and may be referred to in any other expression.

- There is a rich set of predefined functions. Try typing `help elfun`, `help elmat`, and `help ops` to get information on elementary mathematical functions, matrix manipulation, and operators, respectively. For each predefined function there is an online help:

  ```
  >> help sqrt
  ```

[1] See `http://www.mathworks.com`.

```
 SQRT   Square root.
    SQRT(X) is the square root of the elements of X. Complex
    results are produced if X is not positive.

    See also sqrtm, realsqrt, hypot.

    Reference page in Help browser
       doc sqrt
```

The `help` command should be used when you know the name of the function you are interested in, but you need additional information. Otherwise, `lookfor` may be tried:

```
>> lookfor sqrt
REALSQRT Real square root.
SQRT   Square root.
SQRTM     Matrix square root.
```

We see that `lookfor` searches for functions whose online help documentation includes a given string. Recent MATLAB releases include an extensive online documentation which can be accessed by the command `doc`.

- MATLAB is case sensitive (`Pi` and `pi` are different).

```
>> pi
ans =
    3.1416
>> Pi
??? Undefined function or variable 'Pi'.
```

- MATLAB is a matrix-oriented environment and programming language. Vectors and matrices are the basic data structures, and more complex ones have been introduced in the more recent MATLAB versions. Functions and operators are available to deal with vectors and matrices directly. You may enter row and column vectors as follows:

```
>> V1=[22, 5, 3]
V1 =
    22     5     3

>> V2 = [33; 7; 1]
V2 =
    33
     7
     1
```

We may note the difference between comma and semicolon; the latter is used to terminate a row. In the example above, commas are optional, as we could enter the same vector by typing `V1=[22 5 3]`.

- The `who` and `whos` commands may be used to check the user defined variables in the current environment, which can be cleared by the `clear command`.

```
>> who
```

```
Your variables are:
V1        V2
>> whos
  Name      Size                    Bytes  Class
  V1        1x3                        24  double array
  V2        3x1                        24  double array
Grand total is 6 elements using 48 bytes
>> clear V1
>> whos
  Name      Size                    Bytes  Class
  V2        3x1                        24  double array
Grand total is 3 elements using 24 bytes
>> clear
>> whos
>>
```

- You may also use the semicolon to suppress output from the evaluation of an expression:

```
>> V1=[22, 5, 3];
>> V2 = [33; 7; 1];
>>
```

  Using semicolon to suppress output is important when we deal with large matrices (and in MATLAB programming as well).

- You may also enter matrices (note again the difference between '`;`' and '`,`'):

```
>> A=[1 2 3; 4 5 6]
A =
     1     2     3
     4     5     6
>> B=[V2 , V2]
B =
    33    33
     7     7
     1     1
>> C=[V2 ; V2]
C =
    33
     7
     1
    33
     7
     1
```

  Also note the effect of the following commands:

```
>> M1=zeros(2,2)
M1 =
     0     0
     0     0
>> M1=rho
```

```
M1 =
    2.1180
>> M1=zeros(2,2);
>> M1(:,:)=rho
M1 =
    2.1180    2.1180
    2.1180    2.1180
```

- The colon (`:`) is used to spot subranges of an index in a matrix.

```
>> M1=zeros(2,3)
M1 =
     0     0     0
     0     0     0
>> M1(2,:)=4
M1 =
     0     0     0
     4     4     4
>> M1(1,2:3)=6
M1 =
     0     6     6
     4     4     4
```

- The dots (`...`) may be used to write multiline commands.

```
>> M=ones(2,
??? M=ones(2,

Missing variable or function.
>> M=ones(2,...
2)
M =
     1     1
     1     1
```

- The `zeros` and `ones` commands are useful to initialize and preallocate matrices. This is recommended for efficiency. In fact, matrices are resized automatically by MATLAB whenever you assign a value to an element beyond the current row or column range, but this may be time consuming and should be avoided when possible.

```
>> M = [1 2; 3 4];
>> M(3,3) = 5
M =
     1     2     0
     3     4     0
     0     0     5
```

It should be noted that this flexible management of memory is a double-edged sword: It may increase flexibility, but it may make debugging difficult.

- `[]` is the empty vector. You may also use it to delete submatrices:

```
>> M1
```

```
M1 =
     0     6     6
     4     4     4
>> M1(:,2)=[]
M1 =
     0     6
     4     4
```

- Another use of the empty vector is to pass default values to MATLAB functions. Unlike other programming languages, MATLAB is rather flexible in its processing of input arguments to functions. Suppose we have a function `f` taking three input parameters. The standard call would be something like `f(x1, x2, x3)`. If we call the function with one input arguments, `f(x1)`, the missing ones are given default values. Of course this does not happen automatically; the function must be programmed that way, and the reader is urged to see how this is accomplished by opening predefined MATLAB functions with the editor.

  Now suppose that we want to pass only the first and the third argument. We obviously cannot simply call the function like `f(x1, x3)`, since `x3` would be assigned to the second input argument of the function. To obtain what we want, we should use the empty vector: `f(x1, [], x3)`.

- Matrices can be transposed and multiplied easily (if dimensions fit):

```
>> M1'
ans =
     0     4
     6     4
>> M2=rand(2,3)
M2 =
    0.9501    0.6068    0.8913
    0.2311    0.4860    0.7621
>> M1*M2
ans =
    1.3868    2.9159    4.5726
    4.7251    4.3713    6.6136
>> M1+1
ans =
     1     7
     5     5
```

  The `rand` command yields a matrix with random entries, uniformly distributed in the (0,1) interval.

- Note the use of the dot `.` to operate element by element on a matrix:

```
>> A=0.5*ones(2,2)
A =
    0.5000    0.5000
    0.5000    0.5000
>> M1
M1 =
```

```
     0     6
     4     4
>> M1*A
ans =
     3     3
     4     4
>> M1.*A
ans =
     0     3
     2     2

>> I=[1 2; 3 4]
I =
     1     2
     3     4
>> I^2
ans =
     7    10
    15    22
>> I.^2
ans =
     1     4
     9    16
```

- Subranges may be used to build vectors. For instance, to compute the factorial:

```
>> 1:10
ans =
     1     2     3     4     5     6     7     8     9    10
>> prod(1:10)
ans =
     3628800
>> sum(1:10)
ans =
    55
```

  You may also specify an optional increment step in these expressions:

```
>> 1:0.8:4
ans =
    1.0000    1.8000    2.6000    3.4000
```

  The step can be negative too:

```
>> 5:-1:0
ans =
     5     4     3     2     1     0
```

- One more use of the colon operator is to make sure that a vector is a column vector:

```
>> V1 = 1:3
V1 =
     1     2     3
```

```
>> V2 = (1:3)'
V2 =
     1
     2
     3
>> V1(:)
ans =
     1
     2
     3
>> V2(:)
ans =
     1
     2
     3
```

The same effect cannot be obtained by transposition, unless one writes code using the function `size` to check matrix dimensions:

```
>> [m,n] = size(V2)
m =
     3
n =
     1
```

- Note the use of the special quantities `Inf` (infinity) and `NaN` (not a number):

```
>> l=1/0
Warning: Divide by zero.
l =
   Inf
>> l
l =
   Inf
>> prod(1:200)
ans =
   Inf
>> 1/0 - prod(1:200)
Warning: Divide by zero.
ans =
   NaN
```

- Useful functions to operate on matrices are: `eye`, `inv`, `eig`, `det`, `rank`, and `diag`:

```
>> eye(3)
ans =
     1     0     0
     0     1     0
     0     0     1
>> K=eye(3)*[1 2 3]'
K =
     1
```

```
     2
     3
>> K=inv(K)
K =
    1.0000         0         0
         0    0.5000         0
         0         0    0.3333
>> eig(K)
ans =
    1.0000
    0.5000
    0.3333
>> rank(K)
ans =
     3
>> det(K)
ans =
    0.1667
>> K=diag([1 2 3])
K =
     1     0     0
     0     2     0
     0     0     3
```

We should note a sort of dual nature in `diag`. If it receives a vector, it builds a matrix; if it receives a matrix, it returns a vector:

```
>> A = [1:3 ; 4:6 ; 7:9];
>> diag(A)
ans =
     1
     5
     9
```

- Some functions operate on matrices columnwise:

```
>> A = [1 3 5 ; 2 4 6 ];
>> sum(A)
ans =
     3     7    11
>> mean(A)
ans =
    1.5000    3.5000    5.5000
```

The last example may help to understand the rationale behind this choice. If the matrix contains samples from multiple random variables, and we want to compute the sample mean, we should arrange data in such a way that variables corresponds to columns, and joint realizations corresponds to rows. However, it is possible to specify the dimension along which these functions should work:

```
>> sum(A,2)
ans =
```

```
     9
    12
>> mean(A,2)
ans =
     3
     4
```

Another useful function in this vein computes cumulative sums:

```
>> cumsum(1:5)
ans =
     1     3     6    10    15
```

B.2 MATLAB GRAPHICS

Most plots in the book have been obtained using the following MATLAB commands.

- Plotting a function of a single variable is easy. Try the following commands:

```
>> x = 0:0.01:2*pi;
>> plot(x,sin(x))
>> axis([0 2*pi -1 1])
```

The `axis` command may be used to resize plot axes at will. There is also a rich set of ways to annotate a plot.

- Different types of plots may be obtained by using optional parameters of the `plot` command. Try with

```
>> plot(0:20, rand(1,21), 'o')
>> plot(0:20, rand(1,21), 'o-')
```

- To obtain a tridimensional surface, the `surf` command may be used.

```
>> f = @(x,y) exp(-3*(x.^2 + y.^2)).*(sin(5*pi*x)+ cos(10*pi*y));
>> [X Y] = meshgrid(-1:0.01:1 , -1:0.01:1);
>> surf(X,Y,f(X,Y))
```

Some explanation is in order here. The function `surf` must receive three matrices, corresponding to the x and y coordinates in the plane, and to the function value (the 'z' coordinate). A first requirement is that the function we want to draw should be encoded in such a way that it can receive matrix inputs; use of the dot operator is essential: Without the dots '`.`', input matrices would be multiplied row by column, as in linear algebra, rather than element by element. To build the two matrices of coordinates, `meshgrid` is used. To understand what this function accomplishes, let us consider a small scale example:

```
>> [X,Y] = meshgrid(1:4,1:4)
X =
     1     2     3     4
```

```
     1     2     3     4
     1     2     3     4
     1     2     3     4
Y =
     1     1     1     1
     2     2     2     2
     3     3     3     3
     4     4     4     4
```

We see that, for each point in the plane, we obtain matrices containing each coordinate.

B.3 SOLVING EQUATIONS AND COMPUTING INTEGRALS

- Systems of linear equations are easily solved:

```
>> A = [3 5 -1; 9 2 4;  4 -2 -9];
>> b = (1:3)';
>> X = A\b
X =
    0.3119
   -0.0249
   -0.1892
>> A*X
ans =
    1.0000
    2.0000
    3.0000
```

- To solve a nonlinear equation, we must write a piece of code evaluating the function. This can be done by writing a full-fledged program in the MATLAB programming language. However, when the function is a relatively simple expression it may be preferable to define functions in a more direct way, based on the function handle operator `@`:

```
>> f = @(x,y) exp(2*x).*sin(y)
f =
    @(x,y) exp(2*x).*sin(y)
```

We see that the operator is used to "abstract" a function from an expression. The `@` operator is also useful to define anonymous functions which may be passed to higher-order functions, i.e., functions which receive functions as inputs (e.g., to compute integrals or to solve non-linear equations).

We may also fix some input parameters to obtain function of the remaining arguments:

```
>> g = @(y) f(2,y)
g =
    @(y) f(2,y)
>> g(3)
ans =
```

```
    7.7049
```

- As an example, let us solve the equation

$$x^3 - x = 0$$

To this aim, we may use the `fzero` function, which needs to input arguments: the function f defining the equation $f(x) = 0$; a starting point x_0. From the following snapshot, we see that the function returns a zero close to the starting point:

```
>> f = @(x) x^3 - x
f =
    @(x)x^3-x
>> fzero(f, 2)
ans =
     1
>> fzero(f, -2)
ans =
    -1
>> fzero(f, 0.4)
ans =
  1.0646e-017
```

In general, finding all the roots of a nonlinear equation is a difficult problem. Polynomial equations are an exception. In MATLAB, we may solve a polynomial equation by representing a polynomial with a vector collecting its coefficients and passing it to the function `roots`:

```
>> p = [1 0 -1 0]
p =
     1     0    -1     0
>> roots(p)
ans =
     0
    -1
     1
```

- The `quad` function can be used for numerical quadrature, i.e., the numerical approximation of integrals. Consider the integral

$$I = \int_0^{2\pi} e^{-x} \sin(10x)\, dx$$

This integral can be calculated as follows

$$I = -\frac{1}{101} e^{-x} \left[\sin(10x) + 10\cos(10x)\right]\Big|_0^{2\pi} \approx 0.0988$$

but let us pretend we do not know it. To use `quad`, we have to define the function using the anonymous handle trick:

```
>> f=@(x) exp(-x).*sin(10*x)
```

```
f =
    @(x) exp(-x).*sin(10*x)
>> quad(f,0,2*pi)
ans =
    0.0987
```

Precision may be improved by specifying a tolerance parameter:

```
>> quad(f,0,2*pi, 10e-6)
ans =
    0.0987
>> quad(f,0,2*pi, 10e-8)
ans =
    0.0988
```

B.4 STATISTICS IN MATLAB

MATLAB, like R, can be used to carry out common tasks in statistics, such as generating pseudorandom variates, calculating descriptive statistics, and finding quantiles.

The following snapshot shows how to generate a column vector of 10 observations from a normal distribution with expected value 10 and standard deviation 20; then, we compute sample mean, sample variance, and sample deviation:[2]

```
>> X = normrnd(10,20,10,1)
X =
   20.7533
   46.6777
  -35.1769
   27.2435
   16.3753
  -16.1538
    1.3282
   16.8525
   81.5679
   65.3887
>> mean(X)
ans =
   22.4856
>> var(X)
ans =
  1.2530e+003
>> std(X)
ans =
   35.3977
```

We may also estimate the covariance matrix for a joint distribution:

```
>> mu = [10, 20, -5]
mu =
```

[2]Given the nature of random number generators, you will find different results.

```
    10    20    -5
>> rho = [1 0.9 -0.4
0.9 1 -0.2
-0.4 -0.2 1]
rho =
    1.0000    0.9000   -0.4000
    0.9000    1.0000   -0.2000
   -0.4000   -0.2000    1.0000
>> sigma = [20 30 9]
sigma =
    20    30     9
>> Sigma = corr2cov(sigma,rho)
Sigma =
   400   540   -72
   540   900   -54
   -72   -54    81
>> X = mvnrnd(mu, Sigma, 1000);
>> mean(X)
ans =
    9.2523   19.5316   -4.5473
>> cov(X)
ans =
  389.4356  522.0261  -71.6224
  522.0261  868.4043  -49.5950
  -71.6224  -49.5950   84.0933
```

In this snapshot, we have given the correlation matrix `rho` and the vector of standard deviations `sigma`, which have been transformed into the covariance matrix `Sigma` by the function `corr2cov`; the function `mvnrnd` generates a sample from a multivariate normal.

If we need quantiles of the normal distribution, we use `norminv`:

```
>> norminv(0.95)
ans =
    1.6449
>> norminv(0.95, 20, 10)
ans =
   36.4485
```

Just like with R, by default we find quantiles of the standard normal distribution; providing MATLAB with additional parameters, we may specify expected value and standard deviation. If we need the normal CDF, we use `normcdf`:

```
>> normcdf(0)
ans =
    0.5000
>> normcdf(3)
ans =
    0.9987
>> normcdf(20,15,10)
ans =
    0.6915
```

Using `inv`, `chi2inv`, and `finv` we find quantiles of the t, chi-square, and F distribution:

```
>> tinv(0.95,5)
```

```
ans =
    2.0150
>> chi2inv(0.95,4)
ans =
    9.4877
>> finv(0.95,3,5)
ans =
    5.4095
```

The first argument is always the probability level, and the remaining ones specify the parameters of each distribution.

B.5 USING MATLAB TO SOLVE LINEAR AND QUADRATIC PROGRAMMING PROBLEMS

The Optimization toolbox includes a function, `linprog`, which solves LP problems of the form

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \le \mathbf{b} \\ & \mathbf{A}_{\text{eq}}\mathbf{x} = \mathbf{b}_{\text{eq}} \\ & \mathbf{l} \le \mathbf{x} \le \mathbf{u} \end{aligned}$$

The call to the function is `x = linprog(f,A,b,Aeq,beq,lb,ub)`. t As an example, let us solve the problem

$$\begin{aligned} \max \quad & x_1 + x_2 \\ \text{s.t.} \quad & x_1 + 3x_2 \le 100 \\ & 2x_1 + x_2 \le 80 \\ & x_1 \ge 0,\ 0 \le x_2 \le 40 \end{aligned}$$

```
>> c = [-1, -1];
>> A = [1 3; 2 1];
>> b = [100; 80];
>> lb = zeros(2,1);
>> ub = [inf, 40];
>> x = linprog(c, A, b, [], [], lb, ub)
Optimization terminated.
x =
   28.0000
   24.0000
```

Note the use of "infinity" to specify the upper bound on x_1 and the empty vector `[]` as an empty placeholder for the arguments associated with equality constraints; since `linprog` solves a minimization problem, we have to change the sign of the coefficients of the objective function.

To solve quadratic programming problems, such as

$$\begin{aligned} \min \quad & \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \le \mathbf{b} \\ & \mathbf{A}_{\text{eq}}\mathbf{x} = \mathbf{b}_{\text{eq}} \\ & \mathbf{l} \le \mathbf{x} \le \mathbf{u} \end{aligned}$$

we may use `x = quadrog(H,f,A,b,Aeq,beq,lb,ub)`.

As an example, let us find the minimum variance portfolio consisting of 3 assets with expected return and covariance matrix given by[3]

$$\boldsymbol{\mu} = \begin{bmatrix} 0.15 \\ 0.20 \\ 0.08 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} 0.200 & 0.050 & -0.010 \\ 0.050 & 0.300 & 0.015 \\ -0.010 & 0.015 & 0.100 \end{bmatrix}$$

Let $r_T = 0.10$ be the target return:

```
>> Sigma =  [0.200  0.050 -0.010
  0.050  0.300  0.015
  -0.010  0.015  0.100 ]
Sigma =
    0.2000    0.0500   -0.0100
    0.0500    0.3000    0.0150
   -0.0100    0.0150    0.1000
>> mu = [ 0.15; 0.20; 0.08];
>> rt = 0.10;
>> Aeq = [ones(1,3); mu'];
>> beq = [1; rt];
>> w = quadprog(Sigma, [], [], [], Aeq, beq, zeros(3,1))
Optimization terminated.
w =
    0.2610
    0.0144
    0.7246
```

MATLAB cannot be used to solve mixed-integer programming problems; an excellent solver for this purpose is CPLEX, which can be invoked from AMPL (see Appendix C).

[3]See Example 12.5 in the book; also see Section C.2 for a solution using AMPL.

Appendix C
Introduction to AMPL

In this brief appendix, we want to introduce the basic syntax of AMPL. Since its syntax is almost self explanatory, we will just describe a few basic examples, so that the reader can get a grasp of the basic language elements.[1] AMPL is not a language to write procedures; there is a part of the language which is aimed at writing scripts, which behave like any program based on a sequence of control statements and instructions. But the core of AMPL is a *declarative* syntax to describe a mathematical programming model and the data to instantiate it. The optimization solver is separate: You can write a model in AMPL, and solve it with different solvers, possibly implementing different algorithms. Actually, AMPL interfaces have been built for many different solvers; in fact, AMPL is more of a language standard which has been implemented and is sold by a variety of providers.

A demo version is currently available on the web site `http://www.ampl.com`. The reader with no access to a commercial implementation can get the student demo and install it following the instructions. This student demo comes with two solvers: MINOS and CPLEX. MINOS is a solver for linear and nonlinear programming models with continuous variables, developed at Stanford University. CPLEX is a solver for linear and mixed-integer programming models. Originally, CPLEX was am academic product, but it is now developed and distributed by IBM. Recent CPLEX versions are able to cope with quadratic programming models, both continuous and mixed-integer. All the examples in this book have been solved using CPLEX.

[1]For more information, the reader is referred to the original reference for more information: R. Fourer, D.M. Gay, B.W. Kernighan, *AMPL: A Modeling Language for Mathematical Programming* (2nd ed.), Duxbury Press, 2002.

C.1 RUNNING OPTIMIZATION MODELS IN AMPL

Typically, optimization models in AMPL are written using two separate files.

- A **model** file, with standard extension `*.mod`, contains the description of parameters (data), decision variables, constraints, and the objective function.
- A separate **data** file, with standard extension `*.dat`, contains data values for a specific model instance. These data must match the description provided in the model file.

Both files are normal ASCII files which can be created using any text editor, including MATLAB editor (if you are using word processors, be sure you are creating plain text files, with no hidden control characters for formatting). It is also possible to describe a model in one file, but separating structure and data is a good practice, enabling to solve multiple instances of the same model easily.

When you start AMPL, you get a DOS-like window[2] with a prompt like:

```
ampl:
```

To load a model file, you must enter a command like:

```
ampl: model mymodel.mod;
```

where the semicolon must not be forgotten, as it marks the end of a command (otherwise AMPL waits for more input by issuing a prompt like `ampl?`).[3] To load a data file, the command is

```
ampl: data mymodel.dat;
```

Then we may solve the model by issuing the command:

```
ampl: solve;
```

To change data without loading a new model, you should do something like:

```
ampl: reset data;
ampl: data mymodel.dat;
```

Using `reset;` unloads the model too, and it must be used if you want to load and solve a different model. This is also important if you get error messages because of syntax errors in the model description. If you just correct the model file and load the new version, you will get a lot of error messages about duplicate definitions.

The solver can be select using the `option` command. For instance, you may choose

```
ampl: option solver minos;
```

or

```
ampl: option solver cplex;
```

Many more options are actually available, as well as ways to display the solution and to save output to files. We will cover only the essential in the following. We should also mention that the commercial AMPL versions include a powerful script language, which can be used to write complex applications in which several optimization models are dealt with, whereby one model provides input to another one.

[2]The exact look of the window and the way you start AMPL depend on the AMPL version you use.
[3]Here we are assuming that the model and data files are in the same directory as the AMPL executable, which is not good practice. It is much better to place AMPL on the DOS path and to launch it from the directory where the files are stored. See the manuals for details.

```
param NAssets > 0;
param ExpRet{1..NAssets};
param CovMat{1..NAssets, 1..NAssets};
param TargetRet;

var W{1..NAssets} >= 0;

minimize Risk:
      sum {i in 1..NAssets, j in 1..NAssets} W[i]*CovMat[i,j]*W[j];

subject to SumToOne:
      sum {i in 1..NAssets} W[i] = 1;

subject to MinReturn:
      sum {i in 1..NAssets} ExpRet[i]*W[i] = TargetRet;
```

```
param NAssets := 3;
param ExpRet :=
    1 0.15
    2 0.2
    3 0.08;
param CovMat:
     1          2          3        :=
1    0.2000     0.0500    -0.0100
2    0.0500     0.3000     0.0150
3   -0.0100     0.0150     0.1000;

param TargetRet := 0.1;
```

Fig. C.1 AMPL model (`MeanVar.mod`) and data (`MeanVar.dat`) files for mean-variance efficient portfolios.

C.2 MEAN-VARIANCE EFFICIENT PORTFOLIOS IN AMPL

To get acquainted with AMPL syntax, we represent the mean-variance portfolio optimization problem (see Example 12.5 in the book):

$$\begin{aligned} \min \quad & \mathbf{w}'\boldsymbol{\Sigma}\mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}'\bar{\mathbf{r}} = \bar{r}_T \\ & \sum_{i=1}^{n} w_i = 1 \\ & w_i \geq 0. \end{aligned}$$

AMPL syntax for this model is given in figure C.1. First we define model parameters: the number of assets `NAssets`, the vector of expected return (one per asset), the covariance matrix, and the target return. Note that each declaration must be terminated by a semicolon, as AMPL does not consider end of line characters. The restriction `NAssets > 0` is *not* a constraint of the model: It is an optional consistency check that is carried out when data are loaded, *before* issuing the solve command. Catching data inconsistencies as early as possible may be very helpful. Also note that in AMPL it is typical (but not required) to assign long names to parameters and variables, which are more meaningful than the terse names we use in mathematical models.

Then the decision variable W is declared; this variable must be non-negative to prevent short-selling, and this bound is associated to the variable, rather than being declared as a constraint. Finally, the objective function and the two constraints are declared. In both cases we use the sum operator, with a fairly natural syntax. We should note that braces ({}) are used when declaring vectors and matrices, whereas squares brackets ([]) are used to access elements. Objectives and constraints are always given a name, so that later we can access information such as the objective value and dual variables. Expressions for constraints and objective can be entered freely. There is no natural order in the declarations: We may interleave any type of model elements, provided what is used has already been declared.

In the second part of figure C.1 we show the data file. The syntax is fairly natural, but you should notice its basic features:

- Blank and newline characters do not play any role: We must assign vector data by giving both the index and the value; this may look a bit involved, but it allows quite general indexing.

- Each declaration must be closed by a semicolon.

- To assign a matrix, a syntax has been devised that allows to write data as a table, with rows and columns arranged in a visually clear way.

Now we are ready to load and solve the model, and to display the solution:

```
ampl: model MeanVar.mod;
ampl: data MeanVar.dat;
ampl: solve;
    CPLEX 9.1.0: optimal solution; objective 0.06309598494
    18 QP barrier iterations; no basis.
ampl: display W;
    W [*] :=
       1  0.260978
       2  0.0144292
       3  0.724592
    ;
```

We can also evaluate expressions based on the output from the optimization models, as well as checking the shadow prices (Lagrange multipliers) associated with the constraints:

```
ampl: display Risk;
  Risk = 0.063096
ampl: display sqrt(Risk);
  sqrt(Risk) = 0.251189
ampl: display MinReturn.dual;
  MinReturn.dual = -0.69699
ampl: display sum {k in 1..NAssets} W[k]*ExpRet[k];
  sum{k in 1 .. NAssets} W[k]*ExpRet[k] = 0.1
```

```
param NItems > 0;
param Value{1..NItems} >= 0;
param Cost{1..NItems} >= 0;
param Budget >= 0;

var x{1..NItems} binary;

maximize TotalValue:
      sum {i in 1..NItems} Value[i]*x[i];

subject to AvailableBudget:
      sum {i in 1..NItems} Cost[i]*x[i] <= Budget;
```

```
param NItems = 4;

param: Value Cost :=
   1      10    2
   2       7    1
   3      25    6
   4      24    5;

param Budget := 7;
```

Fig. C.2 AMPL model (`Knapsack.mod`) and data (`Knapsack.dat`) files for the knapsack model.

C.3 THE KNAPSACK MODEL IN AMPL

As another example, we consider the knapsack problem (see Section 12.4.1):

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} R_i x_i \\ \text{s.t.} \quad & \sum_{i=1}^{N} C_i x_i \leq W \\ & x_i \in \{0, 1\}. \end{aligned}$$

The corresponding AMPL model is displayed in figure C.2. Again, the syntax is fairly natural, and we should just note a couple of points:

- The decision variables are declared as `binary`.
- In the data file, the two vectors of parameters are assigned at the same time to save on writing; you should compare carefully the syntax used here against the syntax used to assign a matrix (see the covariance matrix in the previous example).

Now we may solve the model and check the solution (we must use `reset` to unload the previous model):

```
ampl: reset;
```

```
ampl: model Knapsack.mod;
ampl: data Knapsack.dat;
ampl: solve;
  CPLEX 9.1.0: optimal integer solution; objective 34
  3 MIP simplex iterations
  0 branch-and-bound nodes
ampl: display x;
x [*] :=
1  1
2  0
3  0
4  1
;
```

In this case, branch and bound is invoked (see Chapter 12). In fact, if you are using the student demo, you cannot solve this model with MINOS; CPLEX must be selected using

```
ampl: option solver cplex;
```

If you use MINOS, you will get the solution for the continuous relaxation of the model above, i.e., a model in which the binary decision variables are relaxed: $x \in [0, 1]$, instead of $x \in \{0, 1\}$. The same can be achieved in ILOG AMPL/CPLEX by issuing appropriate commands:

```
ampl: option cplex_options 'relax';
ampl: solve;
  CPLEX 9.1.0: relax
  Ignoring integrality of 4 variables.
  CPLEX 9.1.0: optimal solution; objective 36.2
  1 dual simplex iterations (0 in phase I)
ampl: display x;
  x [*] :=
  1  1
  2  1
  3  0
  4  0.8
;
```

Here we have used the `relax` option to solve the relaxed model. We may also use other options to gain some insights on the solution process:

```
ampl: option cplex_options 'mipdisplay 2';
ampl: solve;
CPLEX 9.1.0: mipdisplay 2
MIP start values provide initial solution with objective 34.0000.
Clique table members: 2
MIP emphasis: balance optimality and feasibility
Root relaxation solution time =    0.00 sec.

     Nodes                                      Cuts/
Node  Left   Objective  IInf  Best Integer    Best Node  ItCnt   Gap

   0     0     36.2000     1       34.0000     36.2000      1  6.47%
                cutoff             34.0000    Cuts:  2      2  0.00%
```

```
Cover cuts applied:  1
CPLEX 9.1.0: optimal integer solution; objective 34
2 MIP simplex iterations
0 branch-and-bound nodes
```

To interpret this output, the reader should have a look at Section 12.6.2., where the branch and bound method is explained.