AN INTRODUCTION TO OPTIMIZATION

SOLUTIONS MANUAL

Fourth Edition

Edwin K. P. Chong and Stanislaw H. Żak

A JOHN WILEY & SONS, INC., PUBLICATION

1. Methods of Proof and Some Notation

1.1

A	B	not A	not B	A⇒B	(not B)⇒(not A)
F	F	T	T	T	T
F	T	T	F	T	T
T	F	F	T	F	F
T	T	F	F	T	T

1.2

A	B	not A	not B	A⇒B	not (A and (not B))
F	F	T	T	T	T
F	T	T	F	T	T
T	F	F	T	F	F
T	T	F	F	T	T

1.3

A	B	not (A and B)	not A	not B	(not A) or (not B))
F	F	T	T	T	T
F	T	T	T	F	T
T	F	T	F	T	T
T	T	F	F	F	F

1.4

A	B	A and B	A and (not B)	(A and B) or (A and (not B))
F	F	F	F	F
F	T	F	F	F
T	F	F	T	T
T	T	T	F	T

1.5

The cards that you should turn over are 3 and A. The remaining cards are irrelevant to ascertaining the truth or falsity of the rule. The card with S is irrelevant because S is not a vowel. The card with 8 is not relevant because the rule does not say that if a card has an even number on one side, then it has a vowel on the other side.

Turning over the A card directly verifies the rule, while turning over the 3 card verifies the contraposition.

2. Vector Spaces and Matrices

2.1

We show this by contradiction. Suppose $n < m$. Then, the number of columns of $\boldsymbol{A}$ is n. Since $\operatorname{rank} \boldsymbol{A}$ is the maximum number of linearly independent columns of $\boldsymbol{A}$, then $\operatorname{rank} \boldsymbol{A}$ cannot be greater than $n < m$, which contradicts the assumption that $\operatorname{rank} \boldsymbol{A} = m$.

2.2

$\Rightarrow$: Since there exists a solution, then by Theorem 2.1, $\operatorname{rank} \boldsymbol{A} = \operatorname{rank}[\boldsymbol{A}\dot{:}\boldsymbol{b}]$. So, it remains to prove that $\operatorname{rank} \boldsymbol{A} = n$. For this, suppose that $\operatorname{rank} \boldsymbol{A} < n$ (note that it is impossible for $\operatorname{rank} \boldsymbol{A} > n$ since $\boldsymbol{A}$ has only n columns). Hence, there exists $\boldsymbol{y} \in \mathbb{R}^n$, $\boldsymbol{y} \neq \boldsymbol{0}$, such that $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{0}$ (this is because the columns of

$\boldsymbol{A}$ are linearly dependent, and $\boldsymbol{Ay}$ is a linear combination of the columns of $\boldsymbol{A}$). Let $\boldsymbol{x}$ be a solution to $\boldsymbol{Ax} = \boldsymbol{b}$. Then clearly $\boldsymbol{x} + \boldsymbol{y} \neq \boldsymbol{x}$ is also a solution. This contradicts the uniqueness of the solution. Hence, $\operatorname{rank} \boldsymbol{A} = n$.

$\Leftarrow$: By Theorem 2.1, a solution exists. It remains to prove that it is unique. For this, let $\boldsymbol{x}$ and $\boldsymbol{y}$ be solutions, i.e., $\boldsymbol{Ax} = \boldsymbol{b}$ and $\boldsymbol{Ay} = \boldsymbol{b}$. Subtracting, we get $\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{y}) = \boldsymbol{0}$. Since $\operatorname{rank} \boldsymbol{A} = n$ and $\boldsymbol{A}$ has n columns, then $\boldsymbol{x} - \boldsymbol{y} = \boldsymbol{0}$ and hence $\boldsymbol{x} = \boldsymbol{y}$, which shows that the solution is unique.

2.3

Consider the vectors $\bar{\boldsymbol{a}}_i = [1, \boldsymbol{a}_i^\top]^\top \in \mathbb{R}^{n+1}$, $i = 1, \ldots, k$. Since $k \geq n + 2$, then the vectors $\bar{\boldsymbol{a}}_1, \ldots, \bar{\boldsymbol{a}}_k$ must be linearly independent in $\mathbb{R}^{n+1}$. Hence, there exist $\alpha_1, \ldots \alpha_k$, not all zero, such that

$$\sum_{i=1}^{k} \alpha_i \boldsymbol{a}_i = \boldsymbol{0}.$$

The first component of the above vector equation is $\sum_{i=1}^{k} \alpha_i = 0$, while the last n components have the form $\sum_{i=1}^{k} \alpha_i \boldsymbol{a}_i = \boldsymbol{0}$, completing the proof.

2.4

a. We first postmultiply $\boldsymbol{M}$ by the matrix

$$\begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ -\boldsymbol{M}_{m-k,k} & \boldsymbol{I}_{m-k} \end{bmatrix}$$

to obtain

$$\begin{bmatrix} \boldsymbol{M}_{m-k,k} & \boldsymbol{I}_{m-k} \\ \boldsymbol{M}_{k,k} & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ -\boldsymbol{M}_{m-k,k} & \boldsymbol{I}_{m-k} \end{bmatrix} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_{m-k} \\ \boldsymbol{M}_{k,k} & \boldsymbol{O} \end{bmatrix}.$$

Note that the determinant of the postmultiplying matrix is 1. Next we postmultiply the resulting product by

$$\begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_k \\ \boldsymbol{I}_{m-k} & \boldsymbol{O} \end{bmatrix}$$

to obtain

$$\begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_{m-k} \\ \boldsymbol{M}_{k,k} & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_k \\ \boldsymbol{I}_{m-k} & \boldsymbol{O} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{M}_{k,k} \end{bmatrix}.$$

Notice that

$$\det \boldsymbol{M} = \det \left(\begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{M}_{k,k} \end{bmatrix} \right) \det \left(\begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_k \\ \boldsymbol{I}_{m-k} & \boldsymbol{O} \end{bmatrix} \right),$$

where

$$\det \left(\begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_k \\ \boldsymbol{I}_{m-k} & \boldsymbol{O} \end{bmatrix} \right) = \pm 1.$$

The above easily follows from the fact that the determinant changes its sign if we interchange columns, as discussed in Section 2.2. Moreover,

$$\det \left(\begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{M}_{k,k} \end{bmatrix} \right) = \det(\boldsymbol{I}_k) \det(\boldsymbol{M}_{k,k}) = \det(\boldsymbol{M}_{k,k}).$$

Hence,

$$\det \boldsymbol{M} = \pm \det \boldsymbol{M}_{k,k}.$$

b. We can see this on the following examples. We assume, without loss of generality that $\boldsymbol{M}_{m-k,k} = \boldsymbol{O}$ and let $\boldsymbol{M}_{k,k} = 2$. Thus $k = 1$. First consider the case when $m = 2$. Then we have

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_{m-k} \\ \boldsymbol{M}_{k,k} & \boldsymbol{O} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 2 & 0 \end{bmatrix}.$$

Thus,

$$\det \boldsymbol{M} = -2 = \det\left(-\boldsymbol{M}_{k,k}\right).$$

Next consider the case when $m = 3$. Then

$$\det \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_{m-k} \\ \boldsymbol{M}_{k,k} & \boldsymbol{O} \end{bmatrix} = \det \begin{bmatrix} 0 & \vdots & 1 & 0 \\ 0 & \vdots & 0 & 1 \\ \cdots & \cdots & \cdots & \cdots \\ 2 & \vdots & 0 & 0 \end{bmatrix} = 2 \neq \det\left(-\boldsymbol{M}_{k,k}\right).$$

Therefore, in general,

$$\det \boldsymbol{M} \neq \det\left(-\boldsymbol{M}_{k,k}\right)$$

However, when $k = m/2$, that is, when all sub-matrices are square and of the same dimension, then it is true that

$$\det \boldsymbol{M} = \det\left(-\boldsymbol{M}_{k,k}\right).$$

See [121].

2.5 ———————————————————————————————————————

Let

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix}$$

and suppose that each block is $k \times k$. John R. Silvester [121] showed that if at least one of the blocks is equal to $\boldsymbol{O}$ (zero matrix), then the desired formula holds. Indeed, if a row or column block is zero, then the determinant is equal to zero as follows from the determinant's properties discussed Section 2.2. That is, if $\boldsymbol{A} = \boldsymbol{B} = \boldsymbol{O}$, or $\boldsymbol{A} = \boldsymbol{C} = \boldsymbol{O}$, and so on, then obviously $\det \boldsymbol{M} = 0$. This includes the case when any three or all four block matrices are zero matrices.

If $\boldsymbol{B} = \boldsymbol{O}$ or $\boldsymbol{C} = \boldsymbol{O}$ then

$$\det \boldsymbol{M} = \det \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix} = \det(\boldsymbol{A}\boldsymbol{D}).$$

The only case left to analyze is when $\boldsymbol{A} = \boldsymbol{O}$ or $\boldsymbol{D} = \boldsymbol{O}$. We will show that in either case,

$$\det \boldsymbol{M} = \det(-\boldsymbol{B}\boldsymbol{C}).$$

Without loss of generality suppose that $\boldsymbol{D} = \boldsymbol{O}$. Following arguments of John R. Silvester [121], we premultiply $\boldsymbol{M}$ by the product of three matrices whose determinants are unity:

$$\begin{bmatrix} \boldsymbol{I}_k & -\boldsymbol{I}_k \\ \boldsymbol{O} & \boldsymbol{I}_k \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_k & \boldsymbol{O} \\ \boldsymbol{I}_k & \boldsymbol{I}_k \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_k & -\boldsymbol{I}_k \\ \boldsymbol{O} & \boldsymbol{I}_k \end{bmatrix} \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{O} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{C} & \boldsymbol{O} \\ \boldsymbol{A} & \boldsymbol{B} \end{bmatrix}.$$

Hence,

$$\begin{aligned} \det \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{O} \end{bmatrix} &= \begin{bmatrix} -\boldsymbol{C} & \boldsymbol{O} \\ \boldsymbol{A} & \boldsymbol{B} \end{bmatrix} \\ &= \det(-\boldsymbol{C}) \det \boldsymbol{B} \\ &= \det(-\boldsymbol{I}_k) \det \boldsymbol{C} \det \boldsymbol{B}. \end{aligned}$$

Thus we have

$$\det \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{O} \end{bmatrix} = \det(-\boldsymbol{B}\boldsymbol{C}) = \det(-\boldsymbol{C}\boldsymbol{B}).$$

2.6

We represent the given system of equations in the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 & 2 & 1 \\ 1 & -2 & 0 & -1 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \qquad \text{and} \qquad \boldsymbol{b} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}.$$

Using elementary row operations yields

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 & 2 & 1 \\ 1 & -2 & 0 & -1 \end{bmatrix} \to \begin{bmatrix} 1 & 1 & 2 & 1 \\ 0 & -3 & -2 & -2 \end{bmatrix}, \qquad \text{and}$$

$$[\boldsymbol{A}, \boldsymbol{b}] = \begin{bmatrix} 1 & 1 & 2 & 1 & 1 \\ 1 & -2 & 0 & -1 & -2 \end{bmatrix} \to \begin{bmatrix} 1 & 1 & 2 & 1 & 1 \\ 0 & -3 & -2 & -2 & -3 \end{bmatrix},$$

from which $\operatorname{rank} \boldsymbol{A} = 2$ and $\operatorname{rank}[\boldsymbol{A}, \boldsymbol{b}] = 2$. Therefore, by Theorem 2.1, the system has a solution.

We next represent the system of equations as

$$\begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 - 2x_3 - x_4 \\ -2 + x_4 \end{bmatrix}$$

Assigning arbitrary values to x_3 and x_4 ($x_3 = d_3$, $x_4 = d_4$), we get

$$\begin{aligned} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}^{-1} \begin{bmatrix} 1 - 2x_3 - x_4 \\ -2 + x_4 \end{bmatrix} \\ &= -\frac{1}{3} \begin{bmatrix} -2 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 - 2x_3 - x_4 \\ -2 + x_4 \end{bmatrix} \\ &= \begin{bmatrix} -\frac{4}{3}d_3 - \frac{1}{3}d_4 \\ 1 - \frac{2}{3}d_3 - \frac{2}{3}d_4 \end{bmatrix}. \end{aligned}$$

Therefore, a general solution is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -\frac{4}{3}d_3 - \frac{1}{3}d_4 \\ 1 - \frac{2}{3}d_3 - \frac{2}{3}d_4 \\ d_3 \\ d_4 \end{bmatrix} = \begin{bmatrix} -\frac{4}{3} \\ -\frac{2}{3} \\ 1 \\ 0 \end{bmatrix} d_3 + \begin{bmatrix} -\frac{1}{3} \\ -\frac{2}{3} \\ 0 \\ 1 \end{bmatrix} d_4 + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

where d_3 and d_4 are arbitrary values.

2.7

1. Apply the definition of $|-a|$:

$$\begin{aligned} |-a| &= \begin{cases} -a & \text{if } -a > 0 \\ 0 & \text{if } -a = 0 \\ -(-a) & \text{if } -a < 0 \end{cases} \\ &= \begin{cases} -a & \text{if } a < 0 \\ 0 & \text{if } a = 0 \\ a & \text{if } a > 0 \end{cases} \\ &= |a|. \end{aligned}$$

2. If $a \geq 0$, then $|a| = a$. If $a < 0$, then $|a| = -a > 0 > a$. Hence $|a| \geq a$. On the other hand, $|-a| \geq -a$ (by the above). Hence, $a \geq -|-a| = -|a|$ (by property 1).

3. We have four cases to consider. First, if $a, b \geq 0$, then $a + b \geq 0$. Hence, $|a + b| = a + b = |a| + |b|$.
Second, if $a, b \geq 0$, then $a + b \leq 0$. Hence $|a + b| = -(a + b) = -a - b = |a| + |b|$.
Third, if $a \geq 0$ and $b \leq 0$, then we have two further subcases:

1. If $a + b \geq 0$, then $|a + b| = a + b \leq |a| + |b|$.
2. If $a + b \leq 0$, then $|a + b| = -a - b \leq |a| + |b|$.

The fourth case, $a \leq 0$ and $b \geq 0$, is identical to the third case, with a and b interchanged.
4. We first show $|a - b| \leq |a| + |b|$. We have

$$\begin{aligned} |a-b| &= |a+(-b)| \\ &\leq |a| + |-b| \qquad \text{by property 3} \\ &= |a| + |b| \qquad \text{by property 1.} \end{aligned}$$

To show $||a|-|b|| \leq |a-b|$, we note that $|a| = |a-b+b| \leq |a-b|+|b|$, which implies $|a|-|b| \leq |a-b|$. On the other hand, from the above we have $|b| - |a| \leq |b-a| = |a-b|$ by property 1. Therefore, $||a| - |b|| \leq |a-b|$.
5. We have four cases. First, if $a, b \geq 0$, we have $ab \geq 0$ and hence $|ab| = ab = |a||b|$. Second, if $a, b \leq 0$, we have $ab \geq 0$ and hence $|ab| = ab = (-a)(-b) = |a||b|$. Third, if $a \leq 0$, $b \leq 0$, we have $ab \leq 0$ and hence $|ab| = -ab = a(-b) = |a||b|$. The fourth case, $a \leq 0$ and $b \geq 0$, is identical to the third case, with a and b interchanged.
6. We have

$$\begin{aligned} |a+b| &\leq |a| + |b| \qquad \text{by property 3} \\ &\leq c + d. \end{aligned}$$

7. $\Rightarrow$: By property 2, $-a \leq |a|$ and $a \leq |a$. Therefore, $|a| < b$ implies $-a \leq |a| < b$ and $a \leq |a| < b$.
$\Leftarrow$: If $a \geq 0$, then $|a| = a < b$. If $a < 0$, then $|a| = -a < b$.
For the case when "$<$" is replaced by "$\leq$", we simply repeat the above proof with "$<$" replaced by "$\leq$".
8. This is simply the negation of property 7 (apply DeMorgan's Law).

2.8

Observe that we can represent $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_2$ as

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle_2 = \boldsymbol{x}^\top \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} \boldsymbol{y} = (\boldsymbol{Q}\boldsymbol{x})^\top (\boldsymbol{Q}\boldsymbol{y}) = \boldsymbol{x}^\top \boldsymbol{Q}^2 \boldsymbol{y},$$

where

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}.$$

Note that the matrix $\boldsymbol{Q} = \boldsymbol{Q}^\top$ is nonsingular.
1. Now, $\langle \boldsymbol{x}, \boldsymbol{x} \rangle_2 = (\boldsymbol{Q}\boldsymbol{x})^\top (\boldsymbol{Q}\boldsymbol{x}) = \|\boldsymbol{Q}\boldsymbol{x}\|^2 \geq 0$, and

$$\begin{aligned} \langle \boldsymbol{x}, \boldsymbol{x} \rangle_2 = 0 &\Leftrightarrow \|\boldsymbol{Q}\boldsymbol{x}\|^2 = 0 \\ &\Leftrightarrow \boldsymbol{Q}\boldsymbol{x} = \boldsymbol{0} \\ &\Leftrightarrow \boldsymbol{x} = \boldsymbol{0} \end{aligned}$$

since $\boldsymbol{Q}$ is nonsingular.
2. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_2 = (\boldsymbol{Q}\boldsymbol{x})^\top (\boldsymbol{Q}\boldsymbol{y}) = (\boldsymbol{Q}\boldsymbol{y})^\top (\boldsymbol{Q}\boldsymbol{x}) = \langle \boldsymbol{y}, \boldsymbol{x} \rangle_2$.
3. We have

$$\begin{aligned} \langle \boldsymbol{x}+\boldsymbol{y}, \boldsymbol{z} \rangle_2 &= (\boldsymbol{x}+\boldsymbol{y})^\top \boldsymbol{Q}^2 \boldsymbol{z} \\ &= \boldsymbol{x}^\top \boldsymbol{Q}^2 \boldsymbol{z} + \boldsymbol{y}^\top \boldsymbol{Q}^2 \boldsymbol{z} \\ &= \langle \boldsymbol{x}, \boldsymbol{z} \rangle_2 + \langle \boldsymbol{y}, \boldsymbol{z} \rangle_2. \end{aligned}$$

4. $\langle r\boldsymbol{x}, \boldsymbol{y}\rangle_2 = (r\boldsymbol{x})^\top \boldsymbol{Q}^2\boldsymbol{y} = r\boldsymbol{x}^\top \boldsymbol{Q}^2\boldsymbol{y} = r\langle \boldsymbol{x}, \boldsymbol{y}\rangle_2$.

2.9

We have $\|\boldsymbol{x}\| = \|(\boldsymbol{x}-\boldsymbol{y})+\boldsymbol{y}\| \leq \|\boldsymbol{x}-\boldsymbol{y}\| + \|\boldsymbol{y}\|$ by the Triangle Inequality. Hence, $\|\boldsymbol{x}\| - \|\boldsymbol{y}\| \leq \|\boldsymbol{x}-\boldsymbol{y}\|$. On the other hand, from the above we have $\|\boldsymbol{y}\| - \|\boldsymbol{x}\| \leq \|\boldsymbol{y}-\boldsymbol{x}\| = \|\boldsymbol{x}-\boldsymbol{y}\|$. Combining the two inequalities, we obtain $|\|\boldsymbol{x}\| - \|\boldsymbol{y}\|| \leq \|\boldsymbol{x}-\boldsymbol{y}\|$.

2.10

Let $\epsilon > 0$ be given. Set $\delta = \epsilon$. Hence, if $\|\boldsymbol{x}-\boldsymbol{y}\| < \delta$, then by Exercise 2.9, $|\|\boldsymbol{x}\| - \|\boldsymbol{y}\|| \leq \|\boldsymbol{x}-\boldsymbol{y}\| < \delta = \epsilon$.

3. Transformations

3.1

Let $\boldsymbol{v}$ be the vector such that $\boldsymbol{x}$ are the coordinates of $\boldsymbol{v}$ with respect to $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_n\}$, and $\boldsymbol{x}'$ are the coordinates of $\boldsymbol{v}$ with respect to $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \ldots, \boldsymbol{e}'_n\}$. Then,

$$\boldsymbol{v} = x_1\boldsymbol{e}_1 + \cdots + x_n\boldsymbol{e}_n = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]\boldsymbol{x},$$

and

$$\boldsymbol{v} = x'_1\boldsymbol{e}'_1 + \cdots + x'_n\boldsymbol{e}'_n = [\boldsymbol{e}'_1, \ldots, \boldsymbol{e}'_n]\boldsymbol{x}'.$$

Hence,

$$[\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]\boldsymbol{x} = [\boldsymbol{e}'_1, \ldots, \boldsymbol{e}'_n]\boldsymbol{x}'$$

which implies

$$\boldsymbol{x}' = [\boldsymbol{e}'_1, \ldots, \boldsymbol{e}'_n]^{-1}[\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]\boldsymbol{x} = \boldsymbol{T}\boldsymbol{x}.$$

3.2

a. We have

$$[\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3] = [\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3]\begin{bmatrix} 1 & 2 & 4 \\ 3 & -1 & 5 \\ -4 & 5 & 3 \end{bmatrix}.$$

Therefore,

$$\boldsymbol{T} = [\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3]^{-1}[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3] = \begin{bmatrix} 1 & 2 & 4 \\ 3 & -1 & 5 \\ -4 & 5 & 3 \end{bmatrix}^{-1} = \frac{1}{42}\begin{bmatrix} 28 & -14 & -14 \\ 29 & -19 & -7 \\ -11 & 13 & 7 \end{bmatrix}.$$

b. We have

$$[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3] = [\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3]\begin{bmatrix} 1 & 2 & 3 \\ 1 & -1 & 0 \\ 3 & 4 & 5 \end{bmatrix}.$$

Therefore,

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

3.3

We have

$$[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3] = [\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3]\begin{bmatrix} 2 & 2 & 3 \\ 1 & -1 & 0 \\ -1 & 2 & 1 \end{bmatrix}.$$

Therefore, the transformation matrix from $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3\}$ to $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ is

$$\boldsymbol{T} = \begin{bmatrix} 2 & 2 & 3 \\ 1 & -1 & 0 \\ -1 & 2 & 1 \end{bmatrix},$$

Now, consider a linear transformation $L : \mathbb{R}^3 \to \mathbb{R}^3$, and let $\boldsymbol{A}$ be its representation with respect to $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$, and $\boldsymbol{B}$ its representation with respect to $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3\}$. Let $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{y}' = \boldsymbol{B}\boldsymbol{x}'$. Then,

$$\boldsymbol{y}' = \boldsymbol{T}\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{A}\boldsymbol{x}) = \boldsymbol{T}\boldsymbol{A}(\boldsymbol{T}^{-1}\boldsymbol{x}') = (\boldsymbol{T}\boldsymbol{A}\boldsymbol{T}^{-1})\boldsymbol{x}'.$$

Hence, the representation of the linear transformation with respect to $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3\}$ is

$$\boldsymbol{B} = \boldsymbol{T}\boldsymbol{A}\boldsymbol{T}^{-1} = \begin{bmatrix} 3 & -10 & -8 \\ -1 & 8 & 4 \\ 2 & -13 & -7 \end{bmatrix}.$$

3.4

We have

$$[\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3, \boldsymbol{e}'_4] = [\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3, \boldsymbol{e}_4] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Therefore, the transformation matrix from $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3, \boldsymbol{e}_4\}$ to $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3, \boldsymbol{e}'_4\}$ is

$$\boldsymbol{T} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Now, consider a linear transformation $L : \mathbb{R}^4 \to \mathbb{R}^4$, and let $\boldsymbol{A}$ be its representation with respect to $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3, \boldsymbol{e}_4\}$, and $\boldsymbol{B}$ its representation with respect to $\{\boldsymbol{e}'_1, \boldsymbol{e}'_2, \boldsymbol{e}'_3, \boldsymbol{e}'_4\}$. Let $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{y}' = \boldsymbol{B}\boldsymbol{x}'$. Then,

$$\boldsymbol{y}' = \boldsymbol{T}\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{A}\boldsymbol{x}) = \boldsymbol{T}\boldsymbol{A}(\boldsymbol{T}^{-1}\boldsymbol{x}') = (\boldsymbol{T}\boldsymbol{A}\boldsymbol{T}^{-1})\boldsymbol{x}'.$$

Therefore,

$$\boldsymbol{B} = \boldsymbol{T}\boldsymbol{A}\boldsymbol{T}^{-1} = \begin{bmatrix} 5 & 3 & 4 & 3 \\ -3 & -2 & -1 & -2 \\ -1 & 0 & -1 & -2 \\ 1 & 1 & 1 & 4 \end{bmatrix}.$$

3.5

Let $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3, \boldsymbol{v}_4\}$ be a set of linearly independent eigenvectors of $\boldsymbol{A}$ corresponding to the eigenvalues λ_1, λ_2, λ_3, and λ_4. Let $\boldsymbol{T} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3, \boldsymbol{v}_4]$. Then,

$$\begin{aligned} \boldsymbol{A}\boldsymbol{T} &= \boldsymbol{A}[\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3, \boldsymbol{v}_4] = [\boldsymbol{A}\boldsymbol{v}_1, \boldsymbol{A}\boldsymbol{v}_2, \boldsymbol{A}\boldsymbol{v}_3, \boldsymbol{A}\boldsymbol{v}_4] \\ &= [\lambda_1\boldsymbol{v}_1, \lambda_2\boldsymbol{v}_2, \lambda_3\boldsymbol{v}_3, \lambda_4\boldsymbol{v}_4] = [\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3, \boldsymbol{v}_4] \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 \\ 0 & 0 & 0 & \lambda_4 \end{bmatrix}. \end{aligned}$$

Hence,

$$\boldsymbol{A}\boldsymbol{T} = \boldsymbol{T} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix},$$

or

$$\boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}.$$

Therefore, the linear transformation has a diagonal matrix form with respect to the basis formed by a linearly independent set of eigenvectors.

Because

$$\det(\boldsymbol{A}) = (\lambda - 2)(\lambda - 3)(\lambda - 1)(\lambda + 1),$$

the eigenvalues are $\lambda_1 = 2$, $\lambda_2 = 3$, $\lambda_3 = 1$, and $\lambda_4 = -1$.

From $\boldsymbol{A}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$, where $\boldsymbol{v}_i \neq 0$ $(i = 1, 2, 3)$, the corresponding eigenvectors are

$$\boldsymbol{v}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \qquad \boldsymbol{v}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \qquad \boldsymbol{v}_3 = \begin{bmatrix} 0 \\ 2 \\ -9 \\ 1 \end{bmatrix}, \text{and} \qquad \boldsymbol{v}_4 = \begin{bmatrix} 24 \\ -12 \\ 1 \\ 9 \end{bmatrix}.$$

Therefore, the basis we are interested in is

$$\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3\} = \left\{ \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \\ -9 \\ 1 \end{bmatrix}, \begin{bmatrix} 24 \\ -12 \\ 1 \\ 9 \end{bmatrix} \right\}.$$

3.6

Suppose $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ are eigenvectors of $\boldsymbol{A}$ corresponding to $\lambda_1, \ldots, \lambda_n$, respectively. Then, for each $i = 1, \ldots, n$, we have

$$(\boldsymbol{I}_n - \boldsymbol{A})\boldsymbol{v}_i = \boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_i = \boldsymbol{v}_i - \lambda_i \boldsymbol{v}_i = (1 - \lambda_i)\boldsymbol{v}_i$$

which shows that $1 - \lambda_1, \ldots, 1 - \lambda_n$ are the eigenvalues of $\boldsymbol{I}_n - \boldsymbol{A}$.

Alternatively, we may write the characteristic polynomial of $\boldsymbol{I}_n - \boldsymbol{A}$ as

$$\pi_{\boldsymbol{I}_n - \boldsymbol{A}}(1 - \lambda) = \det((1 - \lambda)\boldsymbol{I}_n - (\boldsymbol{I}_n - \boldsymbol{A})) = \det(-[\lambda \boldsymbol{I}_n - \boldsymbol{A}]) = (-1)^n \pi_{\boldsymbol{A}}(\lambda),$$

which shows the desired result.

3.7

Let $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{V}^\perp$, and $\alpha, \beta \in \mathbb{R}$. To show that $\mathcal{V}^\perp$ is a subspace, we need to show that $\alpha\boldsymbol{x} + \beta\boldsymbol{y} \in \mathcal{V}^\perp$. For this, let $\boldsymbol{v}$ be any vector in $\mathcal{V}$. Then,

$$\boldsymbol{v}^\top(\alpha\boldsymbol{x} + \beta\boldsymbol{y}) = \alpha\boldsymbol{v}^\top\boldsymbol{x} + \beta\boldsymbol{v}^\top\boldsymbol{y} = 0,$$

since $\boldsymbol{v}^\top\boldsymbol{x} = \boldsymbol{v}^\top\boldsymbol{y} = 0$ by definition.

3.8

The null space of $\boldsymbol{A}$ is $\mathcal{N}(\boldsymbol{A}) = \{\boldsymbol{x} \in \mathbb{R}^3 : \boldsymbol{A}\boldsymbol{x} = 0\}$. Using elementary row operations and back-substitution, we can solve the system of equations:

$$\begin{bmatrix} 4 & -2 & 0 \\ 2 & 1 & -1 \\ 2 & -3 & 1 \end{bmatrix} \to \begin{bmatrix} 4 & -2 & 0 \\ 0 & 2 & -1 \\ 0 & -2 & 1 \end{bmatrix} \to \begin{bmatrix} 4 & -2 & 0 \\ 0 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \qquad \Rightarrow \qquad \begin{aligned} 4x_1 - 2x_2 &= 0 \\ 2x_2 - x_3 &= 0 \end{aligned}$$

$$\Rightarrow \qquad x_2 = \frac{1}{2}x_3, \qquad x_1 = \frac{1}{2}x_2 = \frac{1}{4}x_3 \qquad \Rightarrow \qquad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} \\ \frac{1}{2} \\ 1 \end{bmatrix} x_3.$$

Therefore,

$$\mathcal{N}(\boldsymbol{A}) = \left\{ \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix} c : c \in \mathbb{R} \right\}.$$

3.9

Let $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{R}(\boldsymbol{A})$, and $\alpha, \beta \in \mathbb{R}$. Then, there exists $\boldsymbol{v}, \boldsymbol{u}$ such that $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{v}$ and $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{u}$. Thus,

$$\alpha\boldsymbol{x} + \beta\boldsymbol{y} = \alpha\boldsymbol{A}\boldsymbol{v} + \beta\boldsymbol{A}\boldsymbol{u} = \boldsymbol{A}(\alpha\boldsymbol{v} + \beta\boldsymbol{u}).$$

Hence, $\alpha\boldsymbol{x} + \beta\boldsymbol{y} \in \mathcal{R}(\boldsymbol{A})$, which shows that $\mathcal{R}(\boldsymbol{A})$ is a subspace.

Let $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{N}(\boldsymbol{A})$, and $\alpha, \beta \in \mathbb{R}$. Then, $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ and $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{0}$. Thus,

$$\boldsymbol{A}(\alpha\boldsymbol{x} + \beta\boldsymbol{y}) = \alpha\boldsymbol{A}\boldsymbol{x} + \beta\boldsymbol{A}\boldsymbol{y} = \boldsymbol{0}.$$

Hence, $\alpha\boldsymbol{x} + \beta\boldsymbol{y} \in \mathcal{N}(\boldsymbol{A})$, which shows that $\mathcal{N}(\boldsymbol{A})$ is a subspace.

3.10

Let $\boldsymbol{v} \in \mathcal{R}(\boldsymbol{B})$, i.e., $\boldsymbol{v} = \boldsymbol{B}\boldsymbol{x}$ for some $\boldsymbol{x}$. Consider the matrix $[\boldsymbol{A}\ \boldsymbol{v}]$. Then, $\mathcal{N}(\boldsymbol{A}^\top) = \mathcal{N}([\boldsymbol{A}\ \boldsymbol{v}]^\top)$, since if $\boldsymbol{u} \in \mathcal{N}(\boldsymbol{A}^\top)$, then $\boldsymbol{u} \in \mathcal{N}(\boldsymbol{B}^\top)$ by assumption, and hence $\boldsymbol{u}^\top\boldsymbol{v} = \boldsymbol{u}^\top\boldsymbol{B}\boldsymbol{x} = \boldsymbol{x}^\top\boldsymbol{B}^\top\boldsymbol{u} = \boldsymbol{0}$. Now,

$$\dim \mathcal{R}(\boldsymbol{A}) + \dim \mathcal{N}(\boldsymbol{A}^\top) = m$$

and

$$\dim \mathcal{R}([\boldsymbol{A}\ \boldsymbol{v}]) + \dim \mathcal{N}([\boldsymbol{A}\ \boldsymbol{v}]^\top) = m.$$

Since $\dim \mathcal{N}(\boldsymbol{A}^\top) = \dim \mathcal{N}([\boldsymbol{A}\ \boldsymbol{v}]^\top)$, then we have $\dim \mathcal{R}(\boldsymbol{A}) = \dim \mathcal{R}([\boldsymbol{A}\ \boldsymbol{v}])$. Hence, $\boldsymbol{v}$ is a linear combination of the columns of $\boldsymbol{A}$, i.e., $\boldsymbol{v} \in \mathcal{R}(\boldsymbol{A})$, which completes the proof.

3.11

We first show $\boldsymbol{V} \subset (\boldsymbol{V}^\perp)^\perp$. Let $\boldsymbol{v} \in \boldsymbol{V}$, and $\boldsymbol{u}$ any element of $\boldsymbol{V}^\perp$. Then $\boldsymbol{u}^\top\boldsymbol{v} = \boldsymbol{v}^\top\boldsymbol{u} = 0$. Therefore, $\boldsymbol{v} \in (\boldsymbol{V}^\perp)^\perp$.

We now show $(\boldsymbol{V}^\perp)^\perp \subset \boldsymbol{V}$. Let $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ be a basis for $\boldsymbol{V}$, and $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_l\}$ a basis for $(\boldsymbol{V}^\perp)^\perp$. Define $\boldsymbol{A} = [\boldsymbol{a}_1 \cdots \boldsymbol{a}_k]$ and $\boldsymbol{B} = [\boldsymbol{b}_1 \cdots \boldsymbol{b}_l]$, so that $\boldsymbol{V} = \mathcal{R}(\boldsymbol{A})$ and $(\boldsymbol{V}^\perp)^\perp = \mathcal{R}(\boldsymbol{B})$. Hence, it remains to show that $\mathcal{R}(\boldsymbol{B}) \subset \mathcal{R}(\boldsymbol{A})$. Using the result of Exercise 3.10, it suffices to show that $\mathcal{N}(\boldsymbol{A}^\top) \subset \mathcal{N}(\boldsymbol{B}^\top)$. So let $\boldsymbol{x} \in \mathcal{N}(\boldsymbol{A}^\top)$, which implies that $\boldsymbol{x} \in \mathcal{R}(\boldsymbol{A})^\perp = \boldsymbol{V}^\perp$, since $\mathcal{R}(\boldsymbol{A})^\perp = \mathcal{N}(\boldsymbol{A}^\top)$. Hence, for all $\boldsymbol{y}$, we have $(\boldsymbol{B}\boldsymbol{y})^\top\boldsymbol{x} = \boldsymbol{0} = \boldsymbol{y}^\top\boldsymbol{B}^\top\boldsymbol{x}$, which implies that $\boldsymbol{B}^\top\boldsymbol{x} = \boldsymbol{0}$. Therefore, $\boldsymbol{x} \in \mathcal{N}(\boldsymbol{B}^\top)$, which completes the proof.

3.12

Let $\boldsymbol{w} \in \mathcal{W}^\perp$, and $\boldsymbol{y}$ be any element of $\mathcal{V}$. Since $\mathcal{V} \subset \mathcal{W}$, then $\boldsymbol{y} \in \mathcal{W}$. Therefore, by definition of $\boldsymbol{w}$, we have $\boldsymbol{w}^\top\boldsymbol{y} = 0$. Therefore, $\boldsymbol{w} \in \mathcal{V}^\perp$.

3.13

Let $r = \dim \mathcal{V}$. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$ be a basis for $\mathcal{V}$, and $\boldsymbol{V}$ the matrix whose ith column is $\boldsymbol{v}_i$. Then, clearly $\mathcal{V} = \mathcal{R}(\boldsymbol{V})$.

Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-r}$ be a basis for $\mathcal{V}^\perp$, and $\boldsymbol{U}$ the matrix whose ith row is $\boldsymbol{u}_i^\top$. Then, $\mathcal{V}^\perp = \mathcal{R}(\boldsymbol{U}^\top)$, and $\mathcal{V} = (\mathcal{V}^\perp)^\perp = \mathcal{R}(\boldsymbol{U}^\top)^\perp = \mathcal{N}(\boldsymbol{U})$ (by Exercise 3.11 and Theorem 3.4).

3.14

a. Let $\boldsymbol{x} \in \mathcal{V}$. Then, $\boldsymbol{x} = \boldsymbol{P}\boldsymbol{x} + (\boldsymbol{I} - \boldsymbol{P})\boldsymbol{x}$. Note that $\boldsymbol{P}\boldsymbol{x} \in \mathcal{V}$, and $(\boldsymbol{I} - \boldsymbol{P})\boldsymbol{x} \in \mathcal{V}^\perp$. Therefore, $\boldsymbol{x} = \boldsymbol{P}\boldsymbol{x} + (\boldsymbol{I} - \boldsymbol{P})\boldsymbol{x}$ is an orthogonal decomposition of $\boldsymbol{x}$ with respect to $\mathcal{V}$. However, $\boldsymbol{x} = \boldsymbol{x} + \boldsymbol{0}$ is also an orthogonal decomposition of $\boldsymbol{x}$ with respect to $\mathcal{V}$. Since the orthogonal decomposition is unique, we must have $\boldsymbol{x} = \boldsymbol{P}\boldsymbol{x}$.

b. Suppose $\boldsymbol{P}$ is an orthogonal projector onto $\mathcal{V}$. Clearly, $\mathcal{R}(\boldsymbol{P}) \subset \mathcal{V}$ by definition. However, from part a, $\boldsymbol{x} = \boldsymbol{P}\boldsymbol{x}$ for all $\boldsymbol{x} \in \mathcal{V}$, and hence $\mathcal{V} \subset \mathcal{R}(\boldsymbol{P})$. Therefore, $\mathcal{R}(\boldsymbol{P}) = \mathcal{V}$.

3.15

To answer the question, we have to represent the quadratic form with a symmetric matrix as

$$\boldsymbol{x}^\top \left(\frac{1}{2} \begin{bmatrix} 1 & -8 \\ 1 & 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -8 & 1 \end{bmatrix} \right) \boldsymbol{x} = \boldsymbol{x}^\top \begin{bmatrix} 1 & -7/2 \\ -7/2 & 1 \end{bmatrix} \boldsymbol{x}.$$

The leading principal minors are $\Delta_1 = 1$ and $\Delta_2 = -45/4$. Therefore, the quadratic form is indefinite.

3.16

The leading principal minors are $\Delta_1 = 2$, $\Delta_2 = 0$, $\Delta_3 = 0$, which are all nonnegative. However, the eigenvalues of $\boldsymbol{A}$ are $0, -1.4641, 5.4641$ (for example, use Matlab to quickly check this). This implies that the matrix $\boldsymbol{A}$ is indefinite (by Theorem 3.7). An alternative way to show that $\boldsymbol{A}$ is not positive semidefinite is to find a vector $\boldsymbol{x}$ such that $\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x} < 0$. So, let $\boldsymbol{x}$ be an eigenvector of $\boldsymbol{A}$ corresponding to its negative eigenvalue $\lambda = -1.4641$. Then, $\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x} = \boldsymbol{x}^\top(\lambda \boldsymbol{x}) = \lambda \boldsymbol{x}^\top \boldsymbol{x} = \lambda \|\boldsymbol{x}\|^2 < 0$. For this example, we can take $\boldsymbol{x} = [0.3251, 0.3251, -0.8881]^\top$, for which we can verify that $\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x} = -1.4643$.

3.17

a. The matrix $\boldsymbol{Q}$ is indefinite, since $\Delta_2 = -1$ and $\Delta_3 = 2$.

b. Let $\boldsymbol{x} \in \mathcal{M}$. Then, $x_2 + x_3 = -x_1$, $x_1 + x_3 = -x_2$, and $x_1 + x_2 = -x_3$. Therefore,

$$\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} = x_1(x_2 + x_3) + x_2(x_1 + x_3) + x_3(x_1 + x_2) = -(x_1^2 + x_2^2 + x_3^2).$$

This implies that the matrix $\boldsymbol{Q}$ is negative definite on the subspace $\mathcal{M}$.

3.18

a. We have

$$f(x_1, x_2, x_3) = x_2^2 = [x_1, x_2, x_3] \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

Then,

$$\boldsymbol{Q} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and the eigenvalues of $\boldsymbol{Q}$ are $\lambda_1 = 0$, $\lambda_2 = 1$, and $\lambda_3 = 0$. Therefore, the quadratic form is positive semidefinite.

b. We have

$$f(x_1, x_2, x_3) = x_1^2 + 2x_2^2 - x_1 x_3 = [x_1, x_2, x_3] \begin{bmatrix} 1 & 0 & -\frac{1}{2} \\ 0 & 2 & 0 \\ -\frac{1}{2} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

Then,

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 0 & -\frac{1}{2} \\ 0 & 2 & 0 \\ -\frac{1}{2} & 0 & 0 \end{bmatrix}$$

and the eigenvalues of $\boldsymbol{Q}$ are $\lambda_1 = 2$, $\lambda_2 = (1 - \sqrt{2})/2$, and $\lambda_3 = (1 + \sqrt{2})/2$. Therefore, the quadratic form is indefinite.

c. We have

$$f(x_1, x_2, x_3) = x_1^2 + x_3^2 + 2x_1 x_2 + 2x_1 x_3 + 2x_2 x_3 = [x_1, x_2, x_3] \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

Then,

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

and the eigenvalues of $\boldsymbol{Q}$ are $\lambda_1 = 0$, $\lambda_2 = 1 - \sqrt{3}$, and $\lambda_3 = 1 + \sqrt{3}$. Therefore, the quadratic form is indefinite.

3.19

We have

$$\begin{aligned} f(x_1,x_2,x_3) &= 4x_1^2+x_2^2+9x_3^2-4x_1x_2-6x_2x_3+12x_1x_3 \\ &= [x_1,x_2,x_3]\begin{bmatrix}4&-2&6\\-2&1&-3\\6&-3&9\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}. \end{aligned}$$

Let

$$\boldsymbol{Q}=\begin{bmatrix}4&-2&6\\-2&1&-3\\6&-3&9\end{bmatrix}, \qquad \boldsymbol{x}=\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=x_1\boldsymbol{e}_1+x_2\boldsymbol{e}_2+x_3\boldsymbol{e}_3,$$

where $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, and $\boldsymbol{e}_3$ form the natural basis for $\mathbb{R}^3$.

Let $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ be another basis for $\mathbb{R}^3$. Then, the vector $\boldsymbol{x}$ is represented in the new basis as $\tilde{\boldsymbol{x}}$, where $\boldsymbol{x}=[\boldsymbol{v}_1,\boldsymbol{v}_2,\boldsymbol{v}_3]\tilde{\boldsymbol{x}}=\boldsymbol{V}\tilde{\boldsymbol{x}}$.

Now, $f(\boldsymbol{x})=\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}=(\boldsymbol{V}\tilde{\boldsymbol{x}})^\top\boldsymbol{Q}(\boldsymbol{V}\tilde{\boldsymbol{x}})=\tilde{\boldsymbol{x}}^\top(\boldsymbol{V}^\top\boldsymbol{Q}\boldsymbol{V})\tilde{\boldsymbol{x}}=\tilde{\boldsymbol{x}}^\top\tilde{\boldsymbol{Q}}\tilde{\boldsymbol{x}}$, where

$$\tilde{\boldsymbol{Q}}=\begin{bmatrix}\tilde{q}_{11}&\tilde{q}_{12}&\tilde{q}_{13}\\\tilde{q}_{21}&\tilde{q}_{22}&\tilde{q}_{23}\\\tilde{q}_{31}&\tilde{q}_{32}&\tilde{q}_{33}\end{bmatrix}$$

and $\tilde{q}_{ij}=\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{v}_j$ for $i,j=1,2,3$.

We will find a basis $\{\boldsymbol{v}_1,\boldsymbol{v}_2,\boldsymbol{v}_3\}$ such that $\tilde{q}_{ij}=0$ for $i\neq j$, and is of the form

$$\begin{aligned}\boldsymbol{v}_1 &= \alpha_{11}\boldsymbol{e}_1\\ \boldsymbol{v}_2 &= \alpha_{21}\boldsymbol{e}_1+\alpha_{22}\boldsymbol{e}_2\\ \boldsymbol{v}_3 &= \alpha_{31}\boldsymbol{e}_1+\alpha_{32}\boldsymbol{e}_2+\alpha_{33}\boldsymbol{e}_3\end{aligned}$$

Because

$$\tilde{q}_{ij}=\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{v}_j=\boldsymbol{v}_i\boldsymbol{Q}(\alpha_{j1}\boldsymbol{e}_1+\ldots+\alpha_{jj}\boldsymbol{e}_j)=\alpha_{j1}(\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_1)+\ldots+\alpha_{jj}(\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_j),$$

we deduce that if $\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_j=0$ for $j<i$, then $\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{v}_j=0$. In this case,

$$\tilde{q}_{ii}=\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{v}_i=\boldsymbol{v}_i\boldsymbol{Q}(\alpha_{i1}\boldsymbol{e}_1+\ldots+\alpha_{ii}\boldsymbol{e}_i)=\alpha_{i1}(\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_1)+\ldots+\alpha_{ii}(\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_i)=\alpha_{ii}(\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_i).$$

Our task therefore is to find $\boldsymbol{v}_i$ $(i=1,2,3)$ such that

$$\begin{aligned}\boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_j &= 0, \qquad j<i\\ \boldsymbol{v}_i\boldsymbol{Q}\boldsymbol{e}_i &= 1,\end{aligned}$$

and, in this case, we get

$$\tilde{\boldsymbol{Q}}=\begin{bmatrix}\alpha_{11}&0&0\\0&\alpha_{22}&0\\0&0&\alpha_{33}\end{bmatrix}.$$

- Case of $i=1$.

  From $\boldsymbol{v}_1^\top\boldsymbol{Q}\boldsymbol{e}_1=1$,

  $$(\alpha_{11}\boldsymbol{e}_1)^\top\boldsymbol{Q}\boldsymbol{e}_1=\alpha_{11}(\boldsymbol{e}_1^\top\boldsymbol{Q}\boldsymbol{e}_1)=\alpha_{11}q_{11}=1.$$

  Therefore,

  $$\alpha_{11}=\frac{1}{q_{11}}=\frac{1}{\Delta_1}=\frac{1}{4} \qquad \Rightarrow \qquad \boldsymbol{v}_1=\alpha_{11}\boldsymbol{e}_1=\begin{bmatrix}\frac{1}{4}\\0\\0\\0\end{bmatrix}.$$

- Case of $i = 2$.

  From $\boldsymbol{v}_2^\top \boldsymbol{Q}\boldsymbol{e}_1 = 0$,

  $$(\alpha_{21}\boldsymbol{e}_1 + \alpha_{22}\boldsymbol{e}_2)^\top \boldsymbol{Q}\boldsymbol{e}_1 = \alpha_{21}(\boldsymbol{e}_1^\top \boldsymbol{Q}\boldsymbol{e}_1) + \alpha_{22}(\boldsymbol{e}_2^\top \boldsymbol{Q}\boldsymbol{e}_1) = \alpha_{21}q_{11} + \alpha_{22}q_{21} = 0.$$

  From $\boldsymbol{v}_2^\top \boldsymbol{Q}\boldsymbol{e}_2 = 1$,

  $$(\alpha_{21}\boldsymbol{e}_1 + \alpha_{22}\boldsymbol{e}_2)^\top \boldsymbol{Q}\boldsymbol{e}_2 = \alpha_{21}(\boldsymbol{e}_1^\top \boldsymbol{Q}\boldsymbol{e}_2) + \alpha_{22}(\boldsymbol{e}_2^\top \boldsymbol{Q}\boldsymbol{e}_2) = \alpha_{21}q_{12} + \alpha_{22}q_{22} = 1.$$

  Therefore,

  $$\begin{bmatrix} q_{11} & q_{21} \\ q_{12} & q_{22} \end{bmatrix} \begin{bmatrix} \alpha_{21} \\ \alpha_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

  But, since $\Delta_2 = 0$, this system of equations is inconsistent. Hence, in this problem $\boldsymbol{v}_2^\top \boldsymbol{Q}\boldsymbol{e}_2 = 0$ should be satisfied instead of $\boldsymbol{v}_2^\top \boldsymbol{Q}\boldsymbol{e}_2 = 1$ so that the system can have a solution. In this case, the diagonal matrix becomes

  $$\tilde{\boldsymbol{Q}} = \begin{bmatrix} \alpha_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \alpha_{33} \end{bmatrix},$$

  and the system of equations become

  $$\begin{bmatrix} q_{11} & q_{21} \\ q_{12} & q_{22} \end{bmatrix} \begin{bmatrix} \alpha_{21} \\ \alpha_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} \alpha_{21} \\ \alpha_{22} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix} \alpha_{22},$$

  where α_{22} is an arbitrary real number. Thus,

  $$\boldsymbol{v}_2 = \alpha_{21}\boldsymbol{e}_1 + \alpha_{22}\boldsymbol{e}_2 = \begin{bmatrix} \frac{1}{2} \\ 1 \\ 0 \end{bmatrix} a,$$

  where a is an arbitrary real number.

- Case of $i = 3$.

  Since in this case $\Delta_3 = \det(\boldsymbol{Q}) = 0$, we will have to apply the same reasoning of the previous case and use the condition $\boldsymbol{v}_3^\top \boldsymbol{Q}\boldsymbol{e}_3 = 0$ instead of $\boldsymbol{v}_3^\top \boldsymbol{Q}\boldsymbol{e}_3 = 1$. In this way the diagonal matrix becomes

  $$\tilde{\boldsymbol{Q}} = \begin{bmatrix} \alpha_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

  Thus, from $\boldsymbol{v}_3^\top \boldsymbol{Q}\boldsymbol{e}_1 = 0$, $\boldsymbol{v}_3^\top \boldsymbol{Q}\boldsymbol{e}_2 = 0$ and $\boldsymbol{v}_3^\top \boldsymbol{Q}\boldsymbol{e}_3 = 0$,

  $$\begin{aligned} \begin{bmatrix} q_{11} & q_{21} & q_{31} \\ q_{12} & q_{22} & q_{32} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} &= \boldsymbol{Q}^\top \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} = \boldsymbol{Q} \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} \\ &= \begin{bmatrix} 4 & -2 & 6 \\ -2 & 1 & -3 \\ 6 & -3 & 9 \end{bmatrix} \begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \end{aligned}$$

  Therefore,

  $$\begin{bmatrix} \alpha_{31} \\ \alpha_{32} \\ \alpha_{33} \end{bmatrix} = \begin{bmatrix} \alpha_{31} \\ 2\alpha_{31} + 3\alpha_{33} \\ \alpha_{33} \end{bmatrix},$$

where α_{31} and α_{33} are arbitrary real numbers. Thus,

$$\boldsymbol{v}_3 = \alpha_{31}\boldsymbol{e}_1 + \alpha_{32}\boldsymbol{e}_2 + \alpha_{33}\boldsymbol{e}_3 = \begin{bmatrix} b \\ 2b+3c \\ c \end{bmatrix},$$

where b and c are arbitrary real numbers.

Finally,

$$\boldsymbol{V} = [x_1, x_2, x_3] = \begin{bmatrix} \frac{1}{4} & \frac{a}{2} & b \\ 0 & a & 2b+3c \\ 0 & 0 & c \end{bmatrix},$$

where a, b, and c are arbitrary real numbers.

3.20

We represent this quadratic form as $f(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}$, where

$$\boldsymbol{Q} = \begin{bmatrix} 1 & \xi & -1 \\ \xi & 1 & 2 \\ -1 & 2 & 5 \end{bmatrix}.$$

The leading principal minors of $\boldsymbol{Q}$ are $\Delta_1 = 1$, $\Delta_2 = 1 - \xi^2$, $\Delta_3 = -5\xi^2 - 4\xi$. For the quadratic form to be positive definite, all the leading principal minors of $\boldsymbol{Q}$ must be positive. This is the case if and only if $\xi \in (-4/5, 0)$.

3.21

The matrix $\boldsymbol{Q} = \boldsymbol{Q}^\top > 0$ can be represented as $\boldsymbol{Q} = \boldsymbol{Q}^{1/2}\boldsymbol{Q}^{1/2}$, where $\boldsymbol{Q}^{1/2} = (\boldsymbol{Q}^{1/2})^\top > 0$.

1. Now, $\langle \boldsymbol{x}, \boldsymbol{x} \rangle_Q = (\boldsymbol{Q}^{1/2}\boldsymbol{x})^\top (\boldsymbol{Q}^{1/2}\boldsymbol{x}) = \|\boldsymbol{Q}^{1/2}\boldsymbol{x}\|^2 \geq 0$, and

$$\begin{aligned} \langle \boldsymbol{x}, \boldsymbol{x} \rangle_Q = 0 &\Leftrightarrow \|\boldsymbol{Q}^{1/2}\boldsymbol{x}\|^2 = 0 \\ &\Leftrightarrow \boldsymbol{Q}^{1/2}\boldsymbol{x} = \boldsymbol{0} \\ &\Leftrightarrow \boldsymbol{x} = \boldsymbol{0} \end{aligned}$$

since $\boldsymbol{Q}^{1/2}$ is nonsingular.

2. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_Q = \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{y} = \boldsymbol{y}^\top \boldsymbol{Q}^\top \boldsymbol{x} = \boldsymbol{y}^\top \boldsymbol{Q}\boldsymbol{x} = \langle \boldsymbol{y}, \boldsymbol{x} \rangle_Q$.
3. We have

$$\begin{aligned} \langle \boldsymbol{x} + \boldsymbol{y}, \boldsymbol{z} \rangle_Q &= (\boldsymbol{x} + \boldsymbol{y})^\top \boldsymbol{Q}\boldsymbol{z} \\ &= \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{z} + \boldsymbol{y}^\top \boldsymbol{Q}\boldsymbol{z} \\ &= \langle \boldsymbol{x}, \boldsymbol{z} \rangle_Q + \langle \boldsymbol{y}, \boldsymbol{z} \rangle_Q. \end{aligned}$$

4. $\langle r\boldsymbol{x}, \boldsymbol{y} \rangle_Q = (r\boldsymbol{x})^\top \boldsymbol{Q}\boldsymbol{y} = r\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{y} = r\langle \boldsymbol{x}, \boldsymbol{y} \rangle_Q$.

3.22

We have

$$\|\boldsymbol{A}\|_\infty = \max\{\|\boldsymbol{A}\boldsymbol{x}\|_\infty : \|\boldsymbol{x}\|_\infty = 1\}.$$

We first show that $\|\boldsymbol{A}\|_\infty \leq \max_i \sum_{k=1}^n |a_{ik}|$. For this, note that for each $\boldsymbol{x}$ such that $\|\boldsymbol{x}\|_\infty = 1$, we have

$$\begin{aligned} \|\boldsymbol{A}\boldsymbol{x}\|_\infty &= \max_i \left| \sum_{k=1}^n a_{ik}x_k \right| \\ &\leq \max_i \sum_{k=1}^n |a_{ik}||x_k| \\ &\leq \max_i \sum_{k=1}^n |a_{ik}|, \end{aligned}$$

since $|x_k| \leq \max_k |x_k| = \|\boldsymbol{x}\|_\infty = 1$. Therefore,

$$\|\boldsymbol{A}\|_\infty \leq \max_i \sum_{k=1}^n |a_{ik}|.$$

To show that $\|\boldsymbol{A}\|_\infty = \max_i \sum_{k=1}^n |a_{ik}|$, it remains to find a $\tilde{\boldsymbol{x}} \in \mathbb{R}^n$, $\|\tilde{\boldsymbol{x}}\|_\infty = 1$, such that $\|\boldsymbol{A}\tilde{\boldsymbol{x}}\|_\infty = \max_i \sum_{k=1}^n |a_{ik}|$. So, let j be such that

$$\sum_{k=1}^n |a_{jk}| = \max_i \sum_{k=1}^n |a_{ik}|.$$

Define $\tilde{\boldsymbol{x}}$ by

$$\tilde{x}_k = \begin{cases} |a_{jk}|/a_{jk} & \text{if } a_{jk} \neq 0 \\ 1 & \text{otherwise} \end{cases}.$$

Clearly $\|\tilde{\boldsymbol{x}}\|_\infty = 1$. Furthermore, for $i \neq j$,

$$\left|\sum_{k=1}^n a_{ik}\tilde{x}_k\right| \leq \sum_{k=1}^n |a_{ik}| \leq \max_i \sum_{k=1}^n |a_{ik}| = \sum_{k=1}^n |a_{jk}|$$

and

$$\left|\sum_{k=1}^n a_{jk}\tilde{x}_k\right| = \sum_{k=1}^n |a_{jk}|.$$

Therefore,

$$\|\boldsymbol{A}\tilde{\boldsymbol{x}}\|_\infty = \max_i \left|\sum_{k=1}^n a_{ik}\tilde{x}_k\right| = \sum_{k=1}^n |a_{jk}| = \max_i \sum_{k=1}^n |a_{ik}|.$$

3.23

We have

$$\|\boldsymbol{A}\|_1 = \max\{\|\boldsymbol{A}\boldsymbol{x}\|_1 : \|\boldsymbol{x}\|_1 = 1\}.$$

We first show that $\|\boldsymbol{A}\|_1 \leq \max_k \sum_{i=1}^m |a_{ik}|$. For this, note that for each $\boldsymbol{x}$ such that $\|\boldsymbol{x}\|_1 = 1$, we have

$$\begin{aligned} \|\boldsymbol{A}\boldsymbol{x}\|_1 &= \sum_{i=1}^m \left|\sum_{k=1}^n a_{ik}x_k\right| \\ &\leq \sum_{i=1}^m \sum_{k=1}^n |a_{ik}||x_k| \\ &\leq \sum_{k=1}^n |x_k| \sum_{i=1}^m |a_{ik}| \\ &\leq \left(\max_k \sum_{i=1}^m |a_{ik}|\right) \sum_{k=1}^n |x_k| \\ &\leq \max_k \sum_{i=1}^m |a_{ik}|, \end{aligned}$$

since $\sum_{k=1}^n |x_k| = \|\boldsymbol{x}\|_1 = 1$. Therefore,

$$\|\boldsymbol{A}\|_1 \leq \max_k \sum_{i=1}^m |a_{ik}|.$$

To show that $\|\boldsymbol{A}\|_1 = \max_k \sum_{i=1}^m |a_{ik}|$, it remains to find a $\tilde{\boldsymbol{x}} \in \mathbb{R}^m$, $\|\tilde{\boldsymbol{x}}\|_1 = 1$, such that $\|\boldsymbol{A}\tilde{\boldsymbol{x}}\|_1 = \max_k \sum_{i=1}^m |a_{ik}|$. So, let j be such that

$$\sum_{i=1}^m |a_{ij}| = \max_k \sum_{i=1}^m |a_{ik}|.$$

Define $\tilde{\boldsymbol{x}}$ by

$$\tilde{x}_k = \begin{cases} 1 & \text{if } k = j \\ 0 & \text{otherwise} \end{cases}.$$

Clearly $\|\tilde{\boldsymbol{x}}\|_1 = 1$. Furthermore,

$$\|\boldsymbol{A}\tilde{\boldsymbol{x}}\|_1 = \sum_{i=1}^m \left| \sum_{k=1}^n a_{ik}\tilde{x}_k \right| = \sum_{i=1}^m |a_{ij}| = \max_k \sum_{i=1}^m |a_{ik}|.$$

4. Concepts from Geometry

4.1

$\Rightarrow$: Let $S = \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$ be a linear variety. Let $\boldsymbol{x}, \boldsymbol{y} \in S$ and $\alpha \in \mathbb{R}$. Then,

$$\boldsymbol{A}(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) = \alpha\boldsymbol{A}\boldsymbol{x} + (1-\alpha)\boldsymbol{A}\boldsymbol{y} = \alpha\boldsymbol{b} + (1-\alpha)\boldsymbol{b} = \boldsymbol{b}.$$

Therefore, $\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \in S$.

$\Leftarrow$: If S is empty, we are done. So, suppose $\boldsymbol{x}_0 \in S$. Consider the set $S_0 = S - \boldsymbol{x}_0 = \{\boldsymbol{x} - \boldsymbol{x}_0 : \boldsymbol{x} \in S\}$. Clearly, for all $\boldsymbol{x}, \boldsymbol{y} \in S_0$ and $\alpha \in \mathbb{R}$, we have $\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \in S_0$. Note that $\boldsymbol{0} \in S_0$. We claim that S_0 is a subspace. To see this, let $\boldsymbol{x}, \boldsymbol{y} \in S_0$, and $\alpha \in \mathbb{R}$. Then, $\alpha\boldsymbol{x} = \alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{0} \in S_0$. Furthermore, $\frac{1}{2}\boldsymbol{x} + \frac{1}{2}\boldsymbol{y} \in S_0$, and therefore $\boldsymbol{x} + \boldsymbol{y} \in S_0$ by the previous argument. Hence, S_0 is a subspace. Therefore, by Exercise 3.13, there exists $\boldsymbol{A}$ such that $S_0 = \mathcal{N}(\boldsymbol{A}) = \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}\}$. Define $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x}_0$. Then,

$$\begin{aligned} S &= S_0 + \boldsymbol{x}_0 = \{\boldsymbol{y} + \boldsymbol{x}_0 : \boldsymbol{y} \in \mathcal{N}(\boldsymbol{A})\} \\ &= \{\boldsymbol{y} + \boldsymbol{x}_0 : \boldsymbol{A}\boldsymbol{y} = \boldsymbol{0}\} \\ &= \{\boldsymbol{y} + \boldsymbol{x}_0 : \boldsymbol{A}(\boldsymbol{y} + \boldsymbol{x}_0) = \boldsymbol{b}\} \\ &= \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}. \end{aligned}$$

4.2

Let $\boldsymbol{u}, \boldsymbol{v} \in \Theta = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\| \leq r\}$, and $\alpha \in [0,1]$. Suppose $\boldsymbol{z} = \alpha\boldsymbol{u} + (1-\alpha)\boldsymbol{v}$. To show that Θ is convex, we need to show that $\boldsymbol{z} \in \Theta$, i.e., $\|\boldsymbol{z}\| \leq r$. To this end,

$$\begin{aligned} \|\boldsymbol{z}\|^2 &= (\alpha\boldsymbol{u}^\top + (1-\alpha)\boldsymbol{v}^\top)(\alpha\boldsymbol{u} + (1-\alpha)\boldsymbol{v}) \\ &= \alpha^2\|\boldsymbol{u}\|^2 + 2\alpha(1-\alpha)\boldsymbol{u}^\top\boldsymbol{v} + (1-\alpha)^2\|\boldsymbol{v}\|^2. \end{aligned}$$

Since $\boldsymbol{u}, \boldsymbol{v} \in \Theta$, then $\|\boldsymbol{u}\|^2 \leq r^2$ and $\|\boldsymbol{v}\|^2 \leq r^2$. Furthermore, by the Cauchy-Schwarz Inequality, we have $\boldsymbol{u}^\top\boldsymbol{v} \leq \|\boldsymbol{u}\|\|\boldsymbol{v}\| \leq r^2$. Therefore,

$$\|\boldsymbol{z}\|^2 \leq \alpha^2 r^2 + 2\alpha(1-\alpha)r^2 + (1-\alpha)^2 r^2 = r^2.$$

Hence, $\boldsymbol{z} \in \Theta$, which implies that Θ is a convex set, i.e., the any point on the line segment joining $\boldsymbol{u}$ and $\boldsymbol{v}$ is also in Θ.

4.3

Let $\boldsymbol{u}, \boldsymbol{v} \in \Theta = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$, and $\alpha \in [0,1]$. Suppose $\boldsymbol{z} = \alpha\boldsymbol{u} + (1-\alpha)\boldsymbol{v}$. To show that Θ is convex, we need to show that $\boldsymbol{z} \in \Theta$, i.e., $\boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}$. To this end,

$$\begin{aligned} \boldsymbol{A}\boldsymbol{z} &= \boldsymbol{A}(\alpha\boldsymbol{u} + (1-\alpha)\boldsymbol{v}) \\ &= \alpha\boldsymbol{A}\boldsymbol{u} + (1-\alpha)\boldsymbol{A}\boldsymbol{v}. \end{aligned}$$

Since $\boldsymbol{u}, \boldsymbol{v} \in \Theta$, then $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b}$ and $\boldsymbol{A}\boldsymbol{v} = \boldsymbol{b}$. Therefore,

$$\boldsymbol{A}\boldsymbol{z} = \alpha\boldsymbol{b} + (1-\alpha)\boldsymbol{b} = \boldsymbol{b},$$

and hence $\boldsymbol{z} \in \Theta$.

4.4

Let $\boldsymbol{u}, \boldsymbol{v} \in \Theta = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} \geq \boldsymbol{0}\}$, and $\alpha \in [0,1]$. Suppose $\boldsymbol{z} = \alpha\boldsymbol{u} + (1-\alpha)\boldsymbol{v}$. To show that Θ is convex, we need to show that $\boldsymbol{z} \in \Theta$, i.e., $\boldsymbol{z} \geq \boldsymbol{0}$. To this end, write $\boldsymbol{x} = [x_1, \ldots, x_n]^\top$, $\boldsymbol{y} = [y_1, \ldots, y_n]^\top$, and $\boldsymbol{z} = [z_1, \ldots, z_n]^\top$. Then, $z_i = \alpha x_i + (1-\alpha)y_i$, $i = 1, \ldots, n$. Since $x_i, y_i \geq 0$, and $\alpha, 1-\alpha \geq 0$, we have $z_i \geq 0$. Therefore, $\boldsymbol{z} \geq \boldsymbol{0}$, and hence $\boldsymbol{z} \in \Theta$.

5. Elements of Calculus

5.1

Observe that

$$\|\boldsymbol{A}^k\| \leq \|\boldsymbol{A}^{k-1}\|\|\boldsymbol{A}\| \leq \|\boldsymbol{A}^{k-2}\|\|\boldsymbol{A}\|^2 \leq \cdots \leq \|\boldsymbol{A}\|^k.$$

Therefore, if $\|\boldsymbol{A}\| < 1$, then $\lim_{k\to\infty} \|\boldsymbol{A}^k\| = \boldsymbol{O}$ which implies that $\lim_{k\to\infty} \boldsymbol{A}^k = \boldsymbol{O}$.

5.2

For the case when $\boldsymbol{A}$ has all real eigenvalues, the proof is simple. Let λ be the eigenvalue of $\boldsymbol{A}$ with largest absolute value, and $\boldsymbol{x}$ the corresponding (normalized) eigenvector, i.e., $\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x}$ and $\|\boldsymbol{x}\| = 1$. Then,

$$\|\boldsymbol{A}\| \geq \|\boldsymbol{A}\boldsymbol{x}\| = \|\lambda\boldsymbol{x}\| = |\lambda|\|\boldsymbol{x}\| = |\lambda|,$$

which completes the proof for this case.

In general, the eigenvalues of $\boldsymbol{A}$ and the corresponding eigenvectors may be complex. In this case, we proceed as follows (see [41]). Consider the matrix

$$\boldsymbol{B} = \frac{\boldsymbol{A}}{\|\boldsymbol{A}\| + \varepsilon},$$

where ε is a positive real number. We have

$$\|\boldsymbol{B}\| = \frac{\|\boldsymbol{A}\|}{\|\boldsymbol{A}\| + \varepsilon} < 1.$$

By Exercise 5.1, $\boldsymbol{B}^k \to \boldsymbol{O}$ as $k \to \infty$, and thus by Lemma 5.1, $|\lambda_i(\boldsymbol{B})| < 1$, $i = 1, \ldots, n$. On the other hand, for each $i = 1, \ldots, n$,

$$\lambda_i(\boldsymbol{B}) = \frac{\lambda_i(\boldsymbol{A})}{\|\boldsymbol{A}\| + \varepsilon},$$

and thus

$$|\lambda_i(\boldsymbol{B})| = \frac{|\lambda_i(\boldsymbol{A})|}{\|\boldsymbol{A}\| + \varepsilon} < 1.$$

which gives

$$|\lambda_i(\boldsymbol{A})| < \|\boldsymbol{A}\| + \varepsilon.$$

Since the above arguments hold for any $\varepsilon > 0$, we have $|\lambda_i(\boldsymbol{A})| \leq \|\boldsymbol{A}\|$.

5.3

a. $\nabla f(\boldsymbol{x}) = (\boldsymbol{a}\boldsymbol{b}^\top + \boldsymbol{b}\boldsymbol{a}^\top)\boldsymbol{x}$.

b. $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{a}\boldsymbol{b}^\top + \boldsymbol{b}\boldsymbol{a}^\top$.

5.4

We have

$$Df(\boldsymbol{x}) = [x_1/3, x_2/2],$$

and

$$\frac{d\boldsymbol{g}}{dt}(t) = \begin{bmatrix} 3 \\ 2 \end{bmatrix}.$$

By the chain rule,

$$\begin{aligned} \frac{d}{dt}F(t) &= Df(\boldsymbol{g}(t))\frac{d\boldsymbol{g}}{dt}(t) \\ &= [(3t+5)/3, (2t-6)/2] \begin{bmatrix} 3 \\ 2 \end{bmatrix} \\ &= 5t - 1. \end{aligned}$$

5.5

We have

$$Df(\boldsymbol{x}) = [x_2/2, x_1/2],$$

and

$$\frac{\partial \boldsymbol{g}}{\partial s}(s,t) = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \qquad \frac{\partial \boldsymbol{g}}{\partial t}(s,t) = \begin{bmatrix} 3 \\ 1 \end{bmatrix}.$$

By the chain rule,

$$\begin{aligned} \frac{\partial}{\partial s}f(\boldsymbol{g}(s,t)) &= Df(\boldsymbol{g}(t))\frac{\partial \boldsymbol{g}}{\partial s}(s,t) \\ &= \frac{1}{2}[2s+t, 4s+3t] \begin{bmatrix} 4 \\ 2 \end{bmatrix} \\ &= 8s + 5t, \end{aligned}$$

and

$$\begin{aligned} \frac{\partial}{\partial t}f(\boldsymbol{g}(s,t)) &= Df(\boldsymbol{g}(t))\frac{\partial \boldsymbol{g}}{\partial t}(s,t) \\ &= \frac{1}{2}[2s+t, 4s+3t] \begin{bmatrix} 3 \\ 1 \end{bmatrix} \\ &= 5s + 3t. \end{aligned}$$

5.6

We have

$$Df(\boldsymbol{x}) = [3x_1^2x_2x_3^2 + x_2,\ x_1^3x_3^2 + x_1,\ 2x_1^3x_2x_3 + 1]$$

and

$$\frac{d\boldsymbol{x}}{dt}(t) = \begin{bmatrix} e^t + 3t^2 \\ 2t \\ 1 \end{bmatrix}.$$

By the chain rule,

$$\begin{aligned}
&\frac{d}{dt} f(\boldsymbol{x}(t)) \\
&\quad = Df(\boldsymbol{x}(t)) \frac{d\boldsymbol{x}}{dt}(t) \\
&\quad = [3x_1(t)^2 x_2(t) x_3(t)^2 + x_2(t),\ x_1(t)^3 x_3(t)^2 + x_1(t),\ 2x_1(t)^3 x_2(t) x_3(t) + 1] \begin{bmatrix} e^t + 3t^2 \\ 2t \\ 1 \end{bmatrix} \\
&\quad = 12t(e^t + 3t^2)^3 + 2te^t + 6t^2 + 2t + 1.
\end{aligned}$$

5.7

Let $\varepsilon > 0$ be given. Since $\boldsymbol{f}(\boldsymbol{x}) = o(g(\boldsymbol{x}))$, then

$$\lim_{\boldsymbol{x} \to \boldsymbol{0}} \frac{\|\boldsymbol{f}(\boldsymbol{x})\|}{g(\boldsymbol{x})} = 0.$$

Hence, there exists $\delta > 0$ such that if $\|\boldsymbol{x}\| < \delta$, then

$$\frac{\|\boldsymbol{f}(\boldsymbol{x})\|}{g(\boldsymbol{x})} < \varepsilon,$$

which can be rewritten as

$$\|\boldsymbol{f}(\boldsymbol{x})\| \leq \varepsilon g(\boldsymbol{x}).$$

5.8

By Exercise 5.7, there exists $\delta > 0$ such that if $\|\boldsymbol{x}\| < \delta$, then $|o(g(\boldsymbol{x}))| < g(\boldsymbol{x})/2$. Hence, if $\|\boldsymbol{x}\| < \delta$, $\boldsymbol{x} \neq \boldsymbol{0}$, then

$$f(\boldsymbol{x}) \leq -g(\boldsymbol{x}) + |o(g(\boldsymbol{x}))| < -g(\boldsymbol{x}) + g(\boldsymbol{x})/2 = -\frac{1}{2} g(\boldsymbol{x}) < 0.$$

5.9

We have that

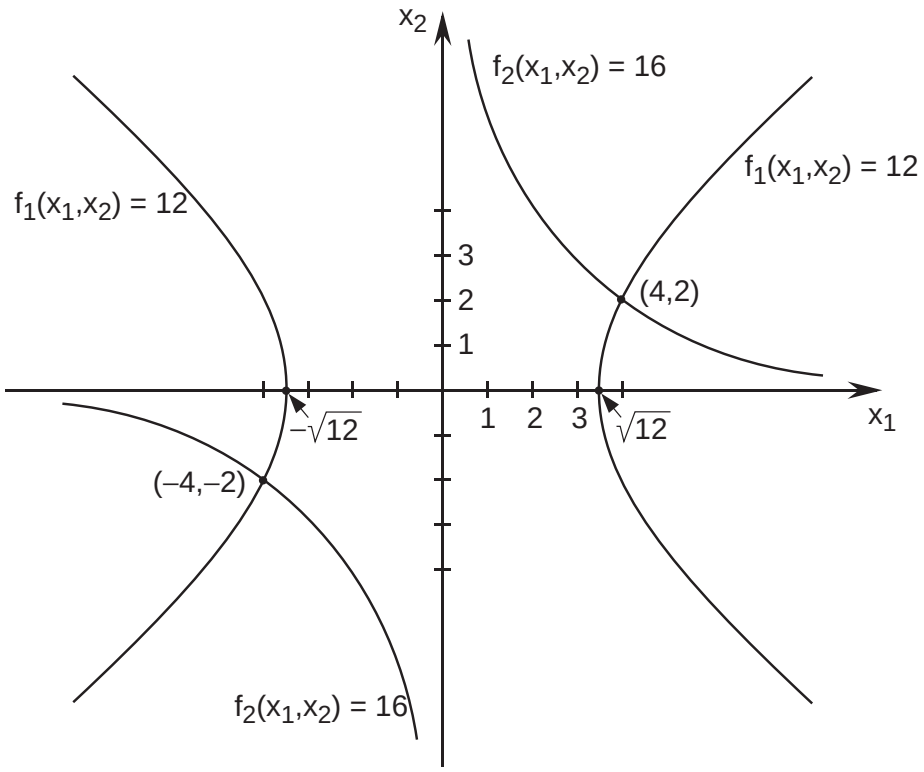
$$\{\boldsymbol{x} : f_1(\boldsymbol{x}) = 12\} = \{\boldsymbol{x} : x_1^2 - x_2^2 = 12\},$$

and

$$\{\boldsymbol{x} : f_2(\boldsymbol{x}) = 16\} = \{\boldsymbol{x} : x_2 = 8/x_1\}.$$

To find the intersection points, we substitute $x_2 = 8/x_1$ into $x_1^2 - x_2^2 = 12$ to get $x_1^4 - 12x_1^2 - 64 = 0$. Solving gives $x_1^2 = 16, -4$. Clearly, the only two possibilities for x_1 are $x_1 = +4, -4$, from which we obtain $x_2 = +2, -2$. Hence, the intersection points are located at $[4, 2]^\top$ and $[-4, -2]^\top$.

The level sets associated with $f_1(x_1, x_2) = 12$ and $f_2(x_1, x_2) = 16$ are shown as follows.



5.10

a. We have

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_o) + Df(\boldsymbol{x}_o)(\boldsymbol{x} - \boldsymbol{x}_o) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_o)^\top D^2 f(\boldsymbol{x}_o)(\boldsymbol{x} - \boldsymbol{x}_o) + \cdots .$$

We compute

$$\begin{aligned} Df(\boldsymbol{x}) &= [e^{-x_2}, -x_1 e^{-x_2} + 1], \\ D^2 f(\boldsymbol{x}) &= \begin{bmatrix} 0 & -e^{-x_2} \\ -e^{-x_2} & x_1 e^{-x_2} \end{bmatrix}. \end{aligned}$$

Hence,

$$\begin{aligned} f(\boldsymbol{x}) &= 2 + [1,0]\begin{bmatrix} x_1 - 1 \\ x_2 \end{bmatrix} + \frac{1}{2}[x_1 - 1, x_2]\begin{bmatrix} 0 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} x_1 - 1 \\ x_2 \end{bmatrix} + \cdots \\ &= 1 + x_1 + x_2 - x_1 x_2 + \frac{1}{2}x_2^2 + \cdots . \end{aligned}$$

b. We compute

$$\begin{aligned} Df(\boldsymbol{x}) &= [4x_1^3 + 4x_1 x_2^2, 4x_1^2 x_2 + 4x_2^3], \\ D^2 f(\boldsymbol{x}) &= \begin{bmatrix} 12x_1^2 + 4x_2^2 & 8x_1 x_2 \\ 8x_1 x_2 & 4x_1^2 + 12x_2^2 \end{bmatrix}. \end{aligned}$$

Expanding f about the point $\boldsymbol{x}_o$ yields

$$\begin{aligned} f(\boldsymbol{x}) &= 4 + [8,8]\begin{bmatrix} x_1 - 1 \\ x_2 - 1 \end{bmatrix} + \frac{1}{2}[x_1 - 1, x_2 - 1]\begin{bmatrix} 16 & 8 \\ 8 & 16 \end{bmatrix}\begin{bmatrix} x_1 - 1 \\ x_2 - 1 \end{bmatrix} + \cdots \\ &= 8x_1^2 + 8x_2^2 - 16x_1 - 16x_2 + 8x_1 x_2 + 12 + \cdots . \end{aligned}$$

c. We compute

$$\begin{aligned} Df(\boldsymbol{x}) &= [e^{x_1-x_2}+e^{x_1+x_2}+1, -e^{x_1-x_2}+e^{x_1+x_2}+1], \\ D^2f(\boldsymbol{x}) &= \begin{bmatrix} e^{x_1-x_2}+e^{x_1+x_2} & -e^{x_1-x_2}+e^{x_1+x_2} \\ -e^{x_1-x_2}+e^{x_1+x_2} & e^{x_1-x_2}+e^{x_1+x_2} \end{bmatrix}. \end{aligned}$$

Expanding f about the point $\boldsymbol{x}_o$ yields

$$\begin{aligned} f(\boldsymbol{x}) &= 2+2e+[2e+1,1]\begin{bmatrix} x_1-1 \\ x_2 \end{bmatrix} + \frac{1}{2}[x_1-1, x_2]\begin{bmatrix} 2e & 0 \\ 0 & 2e \end{bmatrix}\begin{bmatrix} x_1-1 \\ x_2 \end{bmatrix} + \cdots \\ &= 1+x_1++x_2+e(1+x_1^2+x_2^2)+\cdots. \end{aligned}$$

6. Basics of Unconstrained Optimization

6.1

a. In this case, $\boldsymbol{x}^*$ is definitely not a local minimizer. To see this, note that $\boldsymbol{d} = [1, -2]^\top$ is a feasible direction at $\boldsymbol{x}^*$. However, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = -1$, which violates the FONC.

b. In this case, $\boldsymbol{x}^*$ satisfies the FONC, and thus is possibly a local minimizer, but it is impossible to be definite based on the given information.

c. In this case, $\boldsymbol{x}^*$ satisfies the SOSC, and thus is definitely a (strict) local minimizer.

d. In this case, $\boldsymbol{x}^*$ is definitely not a local minimizer. To see this, note that $\boldsymbol{d} = [0, 1]^\top$ is a feasible direction at $\boldsymbol{x}^*$, and $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = 0$. However, $\boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} = -1$, which violates the SONC.

6.2

Because there are no constraints on x_1 or x_2, we can utilize conditions for unconstrained optimization. To proceed, we first compute the function gradient and find the critical points, that is, the points that satisfy the FONC,

$$\nabla f(x_1, x_2) = \mathbf{0}.$$

The components of the gradient $\nabla f(x_1, x_2)$ are

$$\frac{\partial f}{\partial x_1} = x_1^2 - 4 \quad \text{and} \quad \frac{\partial f}{\partial x_2} = x_2^2 - 16.$$

Thus there are four critical points:

$$\boldsymbol{x}^{(1)} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}, \quad \boldsymbol{x}^{(2)} = \begin{bmatrix} 2 \\ -4 \end{bmatrix}, \quad \boldsymbol{x}^{(3)} = \begin{bmatrix} -2 \\ 4 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{x}^{(4)} = \begin{bmatrix} -2 \\ -4 \end{bmatrix}.$$

We next compute the Hessian matrix of the function f:

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} 2x_1 & 0 \\ 0 & 2x_2 \end{bmatrix}.$$

Note that $\boldsymbol{F}(\boldsymbol{x}^{(1)}) > 0$ and therefore, $\boldsymbol{x}^{(1)}$ is a strict local minimizer. Next, $\boldsymbol{F}(\boldsymbol{x}^{(4)}) < 0$ and therefore, $\boldsymbol{x}^{(4)}$ is a strict local maximizer. The Hessian is indefinite at $\boldsymbol{x}^{(2)}$ and $\boldsymbol{x}^{(3)}$ and so these points are neither maximizer nor minimizers.

6.3

Suppose $\boldsymbol{x}^*$ is a global minimizer of f over Ω, and $\boldsymbol{x}^* \in \Omega' \subset \Omega$. Let $\boldsymbol{x} \in \Omega'$. Then, $\boldsymbol{x} \in \Omega$ and therefore $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$. Hence, $\boldsymbol{x}^*$ is a global minimizer of f over Ω'.

6.4

Suppose $\boldsymbol{x}^*$ is an interior point of Ω. Therefore, there exists $\varepsilon > 0$ such that $\{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}^*\| < \varepsilon\} \subset \Omega$. Since $\boldsymbol{x}^*$ is a local minimizer of f over Ω, there exists $\varepsilon' > 0$ such that $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x} \in \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}^*\| < \varepsilon'\}$. Take $\varepsilon'' = \min(\varepsilon, \varepsilon')$. Then, $\{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}^*\| < \varepsilon''\} \subset \Omega'$, and $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x} \in \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}^*\| < \varepsilon''\}$. Thus, $\boldsymbol{x}^*$ is a local minimizer of f over Ω'.

To show that we cannot make the same conclusion if $\boldsymbol{x}^*$ is not an interior point, let $\Omega = \{0\}$, $\Omega' = [-1, 1]$, and $f(x) = x$. Clearly, $0 \in \Omega$ is a local minimizer of f over Ω. However, $0 \in \Omega'$ is not a local minimizer of f over Ω'.

6.5

a. The TONC is: if $f''(0) = 0$, then $f'''(0) = 0$. To prove this, suppose $f''(0) = 0$. Now, by the FONC, we also have $f'(0) = 0$. Hence, by Taylor's theorem,

$$f(x) = f(0) + \frac{f'''(0)}{3!}x^3 + o(x^3).$$

Since 0 is a local minimizer, $f(x) \geq f(0)$ for all x sufficiently close to 0. Hence, for all such x,

$$\frac{f'''(0)}{3!}x^3 \geq o(x^3).$$

Now, if $x > 0$, then

$$f'''(0) \geq 3! \frac{o(x^3)}{x^3},$$

which implies that $f'''(0) \geq 0$. On the other hand, if $x < 0$, then

$$f'''(0) \leq 3! \frac{o(x^3)}{x^3},$$

which implies that $f'''(0) \leq 0$. This implies that $f'''(0) = 0$, as required.

b. Let $f(x) = -x^4$. Then, $f'(0) = 0$, $f''(0) = 0$, and $f'''(0) = 0$, which means that the FONC, SONC, and TONC are all satisfied. However, 0 is not a local minimizer: $f(x) < 0$ for all $x \neq 0$.

c. The answer is yes. To see this, we first write

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2}x^2 + \frac{f'''(0)}{3!}x^3.$$

Now, if the FONC is satisfied, then

$$f(x) = f(0) + \frac{f''(0)}{2}x^2 + \frac{f'''(0)}{3!}x^3.$$

Moreover, if the SONC is satisfied, then either (i) $f''(0) > 0$ or (ii) $f''(0) = 0$. In the case (i), it is clear from the above equation that $f(x) \geq f(0)$ for all x sufficiently close to 0 (because the third term on the right-hand side is $o(x^2)$). In the case (ii), the TONC implies that $f(x) = f(0)$ for all x. In either case, $f(x) \geq f(0)$ for all x sufficiently close to 0. This shows that 0 is a local minimizer.

6.6

a. The TONC is: if $f'(0) = 0$ and $f''(0) = 0$, then $f'''(0) \geq 0$. To prove this, suppose $f'(0) = 0$ and $f''(0) = 0$. By Taylor's theorem, for $x \geq 0$,

$$f(x) = f(0) + \frac{f'''(0)}{3!}x^3 + o(x^3).$$

Since 0 is a local minimizer, $f(x) \geq f(0)$ for sufficiently small $x \geq 0$. Hence, for all $x \geq 0$ sufficiently small,

$$f'''(0) \geq 3! \frac{o(x^3)}{x^3}.$$

This implies that $f'''(0) \geq 0$, as required.

b. Let $f(x) = -x^4$. Then, $f'(0) = 0$, $f''(0) = 0$, and $f'''(0) = 0$, which means that the FONC, SONC, and TONC are all satisfied. However, 0 is not a local minimizer: $f(x) < 0$ for all $x > 0$.

6.7

For convenience, let $\boldsymbol{z}_0 = \boldsymbol{x}_0 + \arg\min_{\boldsymbol{x} \in \Omega} f(\boldsymbol{x})$. Thus we want to show that $\boldsymbol{z}_0 = \arg\min_{\boldsymbol{y} \in \Omega'} f(\boldsymbol{y})$; i.e., for all $\boldsymbol{y} \in \Omega'$, $f(\boldsymbol{y} - \boldsymbol{x}_0) \geq f(\boldsymbol{z}_0 - \boldsymbol{x}_0)$. So fix $\boldsymbol{y} \in \Omega'$. Then, $\boldsymbol{y} - \boldsymbol{x}_0 \in \Omega$. Hence,

$$\begin{aligned} f(\boldsymbol{y} - \boldsymbol{x}_0) &\geq \min_{\boldsymbol{x} \in \Omega} f(\boldsymbol{x}) \\ &= f\left(\arg\min_{\boldsymbol{x} \in \Omega} f(\boldsymbol{x})\right) \\ &= f(\boldsymbol{z}_0 - \boldsymbol{x}_0), \end{aligned}$$

which completes the proof.

6.8

a. The gradient and Hessian of f are

$$\begin{aligned} \nabla f(\boldsymbol{x}) &= 2\begin{bmatrix} 1 & 3 \\ 3 & 7 \end{bmatrix}\boldsymbol{x} + \begin{bmatrix} 3 \\ 5 \end{bmatrix} \\ \boldsymbol{F}(\boldsymbol{x}) &= 2\begin{bmatrix} 1 & 3 \\ 3 & 7 \end{bmatrix}. \end{aligned}$$

Hence, $\nabla f([1,1]^\top) = [11,25]^\top$, and $\boldsymbol{F}([1,1]^\top)$ is as shown above.

b. The direction of maximal rate of increase is the direction of the gradient. Hence, the directional derivative with respect to a unit vector in this direction is

$$\left(\frac{\nabla f(\boldsymbol{x})}{\|\nabla f(\boldsymbol{x})\|}\right)^\top \nabla f(\boldsymbol{x}) = \frac{(\nabla f(\boldsymbol{x}))^\top \nabla f(\boldsymbol{x})}{\|\nabla f(\boldsymbol{x})\|} = \|\nabla f(\boldsymbol{x})\|.$$

At $\boldsymbol{x} = [1,1]^\top$, we have $\|\nabla f([1,1]^\top)\| = \sqrt{11^2 + 25^2} = 27.31$.

c. The FONC in this case is $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$. Solving, we get

$$\boldsymbol{x} = \begin{bmatrix} 3/2 \\ -1 \end{bmatrix}.$$

The point above does not satisfy the SONC because the Hessian is not positive semidefinite (its determinant is negative).

6.9

a. A differentiable function f decreases most rapidly in the direction of the negative gradient. In our problem,

$$\nabla f(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix}^\top = \begin{bmatrix} 2x_1x_2 + x_2^3 & x_1^2 + 3x_1x_2^2 \end{bmatrix}^\top.$$

Hence, the direction of most rapid decrease is

$$-\nabla f\left(\boldsymbol{x}^{(0)}\right) = -\begin{bmatrix} 5 & 10 \end{bmatrix}^\top.$$

b. The rate of increase of f at $\boldsymbol{x}^{(0)}$ in the direction $-\nabla f\left(\boldsymbol{x}^{(0)}\right)$ is

$$\nabla f\left(\boldsymbol{x}^{(0)}\right)^\top \frac{-\nabla f\left(\boldsymbol{x}^{(0)}\right)}{\|\nabla f\left(\boldsymbol{x}^{(0)}\right)\|} = -\|\nabla f\left(\boldsymbol{x}^{(0)}\right)\| = -\sqrt{125} = -5\sqrt{5}.$$

c. The rate of increase of f at $\boldsymbol{x}^{(0)}$ in the direction $\boldsymbol{d}$ is

$$\nabla f\left(\boldsymbol{x}^{(0)}\right)^\top \frac{\boldsymbol{d}}{\|\boldsymbol{d}\|} = \begin{bmatrix} 5 & 10 \end{bmatrix} \begin{bmatrix} 3 \\ 4 \end{bmatrix} \frac{1}{5} = 11.$$

6.10

a. We can rewrite f as

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \begin{bmatrix} 4 & 4 \\ 4 & 2 \end{bmatrix} \boldsymbol{x} + \boldsymbol{x}^\top \begin{bmatrix} 3 \\ 4 \end{bmatrix} + 7.$$

The gradient and Hessian of f are

$$\begin{aligned} \nabla f(\boldsymbol{x}) &= \begin{bmatrix} 4 & 4 \\ 4 & 2 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 3 \\ 4 \end{bmatrix}, \\ \boldsymbol{F}(\boldsymbol{x}) &= \begin{bmatrix} 4 & 4 \\ 4 & 2 \end{bmatrix}. \end{aligned}$$

Hence $\nabla f([0,1]^\top) = [7,6]^\top$. The directional derivative is

$$[1,0]^\top \nabla f([0,1]^\top) = 7.$$

b. The FONC in this case is $\nabla f(\boldsymbol{x}) = \mathbf{0}$. The only point satisfying the FONC is

$$\boldsymbol{x}^* = \frac{1}{4}\begin{bmatrix} -5 \\ 2 \end{bmatrix}.$$

The point above does not satisfy the SONC because the Hessian is not positive semidefinite (its determinant is negative). Therefore, f does not have a minimizer.

6.11

a. Write the objective function as $f(\boldsymbol{x}) = -x_2^2$. In this problem the only feasible directions at $\mathbf{0}$ are of the form $\boldsymbol{d} = [d_1, 0]^\top$. Hence, $\boldsymbol{d}^\top \nabla f(\mathbf{0}) = 0$ for all feasible directions $\boldsymbol{d}$ at $\mathbf{0}$.

b. The point $\mathbf{0}$ is a local maximizer, because $f(\mathbf{0}) = 0$, while any feasible point $\boldsymbol{x}$ satisfies $f(\boldsymbol{x}) \leq 0$.

The point $\mathbf{0}$ is not a strict local maximizer because for any $\boldsymbol{x}$ of the form $\boldsymbol{x} = [x_1, 0]^\top$, we have $f(\boldsymbol{x}) = 0 = f(\mathbf{0})$, and there are such points in any neighborhood of $\mathbf{0}$.

The point $\mathbf{0}$ is not a local minimizer because for any point $\boldsymbol{x}$ of the form $\boldsymbol{x} = [x_1, x_1^2]$ with $x_1 > 0$, we have $f(\boldsymbol{x}) = -x_1^4 < 0$, and there are such points in any neighborhood of $\mathbf{0}$. Since $\mathbf{0}$ is not a local minimizer, it is also not a strict local minimizer.

6.12

a. We have $\nabla f(\boldsymbol{x}^*) = [0, 5]^\top$. The only feasible directions at $\boldsymbol{x}^*$ are of the form $\boldsymbol{d} = [d_1, d_2]^\top$ with $d_2 \geq 0$. Therefore, for such feasible directions, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = 5d_2 \geq 0$. Hence, $\boldsymbol{x}^* = [0, 1]^\top$ satisfies the first order necessary condition.

b. We have $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{O}$. Therefore, for any $\boldsymbol{d}$, $\boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} \geq 0$. Hence, $\boldsymbol{x}^* = [0, 1]^\top$ satisfies the second order necessary condition.

c. Consider points of the form $\boldsymbol{x} = [x_1, -x_1^2 + 1]^\top$, $x_1 \in \mathbb{R}$. Such points are in Ω, and are arbitrarily close to $\boldsymbol{x}^*$. However, for such points $\boldsymbol{x} \neq \boldsymbol{x}^*$,

$$f(\boldsymbol{x}) = 5(-x_1^2 + 1) = 5 - 5x_1^2 < 5 = f(\boldsymbol{x}^*).$$

Hence, $\boldsymbol{x}^*$ is not a local minimizer.

6.13

a. We have $\nabla f(\boldsymbol{x}^*) = -[3, 0]^\top$. The only feasible directions at $\boldsymbol{x}^*$ are of the form $\boldsymbol{d} = [d_1, d_2]^\top$ with $d_1 \leq 0$. Therefore, for such feasible directions, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = 3d_1 \geq 0$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ satisfies the first order necessary condition.

b. We have $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{O}$. Therefore, for any $\boldsymbol{d}$, $\boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} \geq 0$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ satisfies the second order necessary condition.

c. Yes, $\boldsymbol{x}^*$ is a local minimizer. To see this, notice that any feasible point $\boldsymbol{x} = [x_1, x_2]^\top \neq \boldsymbol{x}^*$ is such that $x_1 < 2$. Hence, for such points $\boldsymbol{x} \neq \boldsymbol{x}^*$,

$$f(\boldsymbol{x}) = -3x_1 > -6 = f(\boldsymbol{x}^*).$$

In fact, $\boldsymbol{x}^*$ is a strict local minimizer.

6.14

a. We have $\nabla f(\boldsymbol{x}) = [0, 1]$, which is nonzero everywhere. Hence, no interior point satisfies the FONC. Moreover, any boundary point with a feasible direction $\boldsymbol{d}$ such that $d_2 < 0$ cannot be satisfy the FONC, because for such a $\boldsymbol{d}$, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = d_2 < 0$. By drawing a picture, it is easy to see that the only boundary point remaining is $\boldsymbol{x}^* = [0, 1]^\top$. For this point, any feasible direction satisfies $d_2 \geq 0$. Hence, for any feasible direction, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = d_2 \geq 0$. Hence, $\boldsymbol{x}^* = [0, 1]^\top$ satisfies the FONC, and is the only such point.

b. We have $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{O}$. So any point (and in particular $\boldsymbol{x}^* = [0, 1]^\top$) satisfies the SONC.

c. The point $\boldsymbol{x}^* = [0, 1]^\top$ is not a local minimizer. To see this, consider points of the form $\boldsymbol{x} = [\sqrt{1 - x_2^2}, x_2]^\top$ where $x_2 \in [1/2, 1)$. It is clear that such points are feasible, and are arbitrarily close to $\boldsymbol{x}^* = [0, 1]^\top$. However, for such points, $f(\boldsymbol{x}) = x_2 < 1 = f(\boldsymbol{x}^*)$.

6.15

a. We have $\nabla f(\boldsymbol{x}^*) = [3, 0]^\top$. The only feasible directions at $\boldsymbol{x}^*$ are of the form $\boldsymbol{d} = [d_1, d_2]^\top$ with $d_1 \geq 0$. Therefore, for such feasible directions, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = 3d_1 \geq 0$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ satisfies the first order necessary condition.

b. We have $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{O}$. Therefore, for any $\boldsymbol{d}$, $\boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} \geq 0$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ satisfies the second order necessary condition.

c. Consider points of the form $\boldsymbol{x} = [-x_2^2 + 2, x_2]^\top$, $x_2 \in \mathbb{R}$. Such points are in Ω, and could be arbitrarily close to $\boldsymbol{x}^*$. However, for such points $\boldsymbol{x} \neq \boldsymbol{x}^*$,

$$f(\boldsymbol{x}) = 3(-x_2^2 + 2) = 6 - 6x_2^2 < 6 = f(\boldsymbol{x}^*).$$

Hence, $\boldsymbol{x}^*$ is not a local minimizer.

6.16

a. We have $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$. Therefore, for any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^*$, we have $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = 0$. Hence, $\boldsymbol{x}^*$ satisfies the first-order necessary condition.

b. We have

$$\boldsymbol{F}(\boldsymbol{x}^*) = \begin{bmatrix} 8 & 0 \\ 0 & -2 \end{bmatrix}.$$

Any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^*$ has the form $\boldsymbol{d} = [d_1, d_2]^\top$ where $d_2 \leq 2d_1$, $d_1, d_2 \geq 0$. Therefore, for any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^*$, we have

$$\boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} = 8d_1^2 - 2d_2^2 \geq 8d_1^2 - 2(2d_1)^2 = 0.$$

Hence, $\boldsymbol{x}^*$ satisfies the second-order necessary condition.

c. We have $f(\boldsymbol{x}^*) = 0$. Any point of the form $\boldsymbol{x} = [x_1, x_1^2 + 2x_1]$, $x_1 > 0$, is feasible and has objective function value given by

$$f(\boldsymbol{x}) = 4x_1^2 - (x_1^2 + 2x_1)^2 = -(x_1^4 + 4x_1^3) < 0 = f(\boldsymbol{x}^*),$$

Moreover, there are such points in any neighborhood of $\boldsymbol{x}^*$. Therefore, the point $\boldsymbol{x}^*$ is not a local minimizer.

6.17

a. We have $\nabla f(\boldsymbol{x}^*) = [1/x_1^*, 1/x_2^*]^\top$. If $\boldsymbol{x}^*$ were an interior point, then $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$. But this is clearly impossible. Therefore, $\boldsymbol{x}^*$ cannot possibly be an interior point.

b. We have $\boldsymbol{F}(\boldsymbol{x}) = -\operatorname{diag}[1/x_1^2, 1/x_2^2]$, which is negative definite everywhere. Therefore, the second-order necessary condition is satisfied everywhere. (Note that because we have a maximization problem, *negative* definiteness is the relevant condition.)

6.18

Given $x \in \mathbb{R}$, let

$$f(x) = \sum_{i=1}^{n} (x - x_i)^2,$$

so that $\bar{x}$ is the minimizer of f. By the FONC,

$$f'(\bar{x}) = 0,$$

and hence

$$\sum_{i=1}^{n} 2(\bar{x} - x_i) = 0,$$

which on solving gives

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i.$$

6.19

Let θ_1 be the angle from the horizontal to the bottom of the picture, and θ_2 the angle from the horizontal to the top of the picture. Then, $\tan(\theta) = (\tan(\theta_2) - \tan(\theta_1))/(1 + \tan(\theta_2)\tan(\theta_1))$. Now, $\tan(\theta_1) = b/x$ and $\tan(\theta_2) = (a+b)/x$. Hence, the objective function that we wish to maximize is

$$f(x) = \frac{(a+b)/x - b/x}{1 + b(a+b)/x^2} = \frac{a}{x + b(a+b)/x}.$$

We have

$$f'(x) = -\frac{a^2}{(x + b(a+b)/x)^2}\left(1 - \frac{b(a+b)}{x^2}\right).$$

Let x^* be the optimal distance. Then, by the FONC, we have $f'(x^*) = 0$, which gives

$$\begin{aligned} 1 - \frac{b(a+b)}{(x^*)^2} &= 0 \\ \Rightarrow \qquad x^* &= \sqrt{b(a+b)}. \end{aligned}$$

6.20

The squared distance from the sensor to the baby's heart is $1 + x^2$, while the squared distance from the sensor to the mother's heart is $1 + (2-x)^2$. Therefore, the signal to noise ratio is

$$f(x) = \frac{1 + (2-x)^2}{1 + x^2}.$$

We have

$$\begin{aligned} f'(x) &= \frac{-2(2-x)(1+x^2) - 2x(1+(2-x)^2)}{(1+x^2)^2} \\ &= \frac{4(x^2 - 2x - 1)}{(1+x^2)^2}. \end{aligned}$$

By the FONC, at the optimal position x^*, we have $f'(x^*) = 0$. Hence, either $x^* = 1 - \sqrt{2}$ or $x^* = 1 + \sqrt{2}$. From the figure, it easy to see that $x^* = 1 - \sqrt{2}$ is the optimal position.

6.21

a. Let x be the decision variable. Write the total travel time as $f(x)$, which is given by

$$f(x) = \frac{\sqrt{1+x^2}}{v_1} + \frac{\sqrt{1+(d-x)^2}}{v_2}.$$

Differentiating the above expression, we get

$$f'(x) = \frac{x}{v_1\sqrt{1+x^2}} - \frac{d-x}{v_2\sqrt{1+(d-x)^2}}.$$

By the first order necessary condition, the optimal path satisfies $f'(x^*) = 0$, which corresponds to

$$\frac{x^*}{v_1\sqrt{1+(x^*)^2}} = \frac{d-x^*}{v_2\sqrt{1+(d-x^*)^2}},$$

or $\sin\theta_1/v_1 = \sin\theta_2/v_2$. Upon rearranging, we obtain the desired equation.

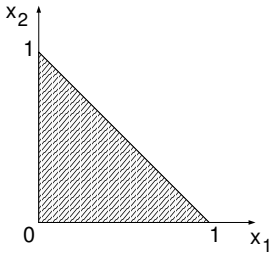
b. The second derivative of f is given by

$$f''(x) = \frac{1}{v_1(1+x^2)^{3/2}} + \frac{1}{v_2(1+(d-x)^2)^{3/2}}.$$

Hence, $f''(x^*) > 0$, which shows that the second order sufficient condition holds.

6.22

a. We have $f(\boldsymbol{x}) = U_1(x_1) + U_2(x_2)$ and $\Omega = \{\boldsymbol{x} : x_1, x_2 \geq 0,\ x_1 + x_2 \leq 1\}$. A picture of Ω looks like:



b. We have $\nabla f(\boldsymbol{x}) = [a_1, a_2]^\top$. Because $\nabla f(\boldsymbol{x}) \neq \boldsymbol{0}$, for all $\boldsymbol{x}$, we conclude that no interior point satisfies the FONC. Next, consider any feasible point $\boldsymbol{x}$ for which $x_2 > 0$. At such a point, the vector $\boldsymbol{d} = [1, -1]^\top$ is a feasible direction. But then $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = a_1 - a_2 > 0$ which means that FONC is violated (recall that the problem is to *maximize* f). So clearly the remaining candidates are those $\boldsymbol{x}$ for which $x_2 = 0$. Among these, if $x_1 < 1$, then $\boldsymbol{d} = [0, 1]^\top$ is a feasible direction, in which case we have $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = a_2 > 0$. This leaves the point $\boldsymbol{x} = [1, 0]^\top$. At this point, any feasible direction $\boldsymbol{d}$ satisfies $d_1 \leq 0$ and $d_2 \leq -d_1$. Hence, for any feasible direction $\boldsymbol{d}$, we have

$$\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = d_1 a_1 + d_2 a_2 \leq d_1 a_1 + (-d_1) a_2 = d_1(a_1 - a_2) \leq 0.$$

So, the only feasible point that satisfies the FONC is $[1, 0]^\top$.

c. We have $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{O} \leq 0$. Hence, any point satisfies the SONC (again, recall that the problem is to *maximize* f).

6.23

We have

$$\nabla f(\boldsymbol{x}) = \begin{bmatrix} 4(x_1 - x_2)^3 + 2x_1 - 2 \\ -4(x_1 - x_2)^3 - 2x_2 + 2 \end{bmatrix}.$$

Setting $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$ we get

$$\begin{aligned} 4(x_1 - x_2)^3 + 2x_1 - 2 &= 0 \\ -4(x_1 - x_2)^3 - 2x_2 + 2 &= 0. \end{aligned}$$

Adding the two equations, we obtain $x_1 = x_2$, and substituting back yields

$$x_1 = x_2 = 1.$$

Hence, the only point satisfying the FONC is $[1, 1]^\top$.

We have

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} 12(x_1 - x_2)^2 + 2 & -12(x_1 - x_2)^2 \\ -12(x_1 - x_2)^2 & 12(x_1 - x_2)^2 - 2 \end{bmatrix}.$$

Hence

$$\boldsymbol{F}([1, 1]^\top) = \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix}$$

Since $\boldsymbol{F}([1, 1]^\top)$ is not positive semidefinite, the point $[1, 1]^\top$ does not satisfy the SONC.

6.24

Suppose $\boldsymbol{d}$ is a feasible direction at $\boldsymbol{x}$. Then, there exists $\alpha_0 > 0$ such that $\boldsymbol{x} + \alpha \boldsymbol{d} \in \Omega$ for all $\alpha \in [0, \alpha_0]$. Let $\beta > 0$ be given. Then, $\boldsymbol{x} + \alpha(\beta \boldsymbol{d}) \in \Omega$ for all $\alpha \in [0, \alpha_0/\beta]$. Since $\alpha_0/\beta > 0$, by definition $\beta \boldsymbol{d}$ is also a feasible direction at $\boldsymbol{x}$.

6.25

$\Rightarrow$: Suppose $\boldsymbol{d}$ is feasible at $\boldsymbol{x} \in \Omega$. Then, there exists $\alpha > 0$ such that $\boldsymbol{x} + \alpha \boldsymbol{d} \in \Omega$, that is, $\boldsymbol{A}(\boldsymbol{x} + \alpha \boldsymbol{d}) = \boldsymbol{b}$. Since $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ and $\alpha \neq 0$, we conclude that $\boldsymbol{A}\boldsymbol{d} = \boldsymbol{0}$.

$\Leftarrow$: Suppose $\boldsymbol{A}\boldsymbol{d} = \boldsymbol{0}$. Then, for any $\alpha \in [0,1]$, we have $\alpha\boldsymbol{A}\boldsymbol{d} = \boldsymbol{0}$. Adding this equation to $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, we obtain $\boldsymbol{A}(\boldsymbol{x} + \alpha\boldsymbol{d}) = \boldsymbol{b}$, that is, $\boldsymbol{x} + \alpha\boldsymbol{d} \in \Omega$ for all $\alpha \in [0,1]$. Therefore, $\boldsymbol{d}$ is a feasible direction at $\boldsymbol{x}$.

6.26

The vector $\boldsymbol{d} = [1,1]^\top$ is a feasible direction at $\boldsymbol{0}$. Now,

$$\boldsymbol{d}^\top \nabla f(\boldsymbol{0}) = \frac{\partial f}{\partial x_1}(\boldsymbol{0}) + \frac{\partial f}{\partial x_2}(\boldsymbol{0}).$$

Since $\nabla f(\boldsymbol{0}) \leq \boldsymbol{0}$ and $\nabla f(\boldsymbol{0}) \neq \boldsymbol{0}$, then

$$\boldsymbol{d}^\top \nabla f(\boldsymbol{0}) < 0.$$

Hence, by the FONC, $\boldsymbol{0}$ is not a local minimizer.

6.27

We have $\nabla f(\boldsymbol{x}) = \boldsymbol{c} \neq \boldsymbol{0}$. Therefore, for any $\boldsymbol{x} \in \mathring{\Omega}$, we have $\nabla f(\boldsymbol{x}) \neq \boldsymbol{0}$. Hence, by Corollary 6.1, $\boldsymbol{x} \in \mathring{\Omega}$ cannot be a local minimizer (and therefore it cannot be a solution).

6.28

The objective function is $f(\boldsymbol{x}) = -c_1 x_1 - c_2 x_2$. Therefore, $\nabla f(\boldsymbol{x}) = [-c_1, -c_2]^\top \neq \boldsymbol{0}$ for all $\boldsymbol{x}$. Thus, by FONC, the optimal solution $\boldsymbol{x}^*$ cannot lie in the interior of the feasible set. Next, for all $\boldsymbol{x} \in L_1 \bigcup L_2$, $\boldsymbol{d} = [1,1]^\top$ is a feasible direction. Therefore, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = -c_1 - c_2 < 0$. Hence, by FONC, the optimal solution $\boldsymbol{x}^*$ cannot lie in $L_1 \bigcup L_2$. Lastly, for all $\boldsymbol{x} \in L_3$, $\boldsymbol{d} = [1,-1]^\top$ is a feasible direction. Therefore, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) = c_2 - c_1 < 0$. Hence, by FONC, the optimal solution $\boldsymbol{x}^*$ cannot lie in L_3. Therefore, by elimination, the unique optimal feasible solution must be $[1,0]^\top$.

6.29

a. We write

$$\begin{aligned}
f(a,b) &= \frac{1}{n}\sum_{i=1}^n a^2 x_i^2 + b^2 + y_i^2 + 2x_i ab - 2x_i y_i a - 2y_i b \\
&= a^2\left(\frac{1}{n}\sum_{i=1}^n x_i^2\right) + b^2 + 2\left(\frac{1}{n}\sum_{i=1}^n x_i\right)ab \\
&\quad - 2\left(\frac{1}{n}\sum_{i=1}^n x_i y_i\right)a - 2\left(\frac{1}{n}\sum_{i=1}^n y_i\right)b + \left(\frac{1}{n}\sum_{i=1}^n y_i^2\right) \\
&= [a\ b]\begin{bmatrix} \frac{1}{n}\sum_{i=1}^n x_i^2 & \frac{1}{n}\sum_{i=1}^n x_i \\ \frac{1}{n}\sum_{i=1}^n x_i & 1 \end{bmatrix}\begin{bmatrix} a \\ b \end{bmatrix} \\
&\quad - 2\left[\frac{1}{n}\sum_{i=1}^n x_i y_i, \frac{1}{n}\sum_{i=1}^n y_i\right]\begin{bmatrix} a \\ b \end{bmatrix} + \frac{1}{n}\sum_{i=1}^n y_i^2 \\
&= \boldsymbol{z}^\top \boldsymbol{Q}\boldsymbol{z} - 2\boldsymbol{c}^\top \boldsymbol{z} + d,
\end{aligned}$$

where $\boldsymbol{z}$, $\boldsymbol{Q}$, $\boldsymbol{c}$ and d are defined in the obvious way.

b. If the point $\boldsymbol{z}^* = [a^*, b^*]^\top$ is a solution, then by the FONC, we have $\nabla f(\boldsymbol{z}^*) = 2\boldsymbol{Q}\boldsymbol{z}^* - 2\boldsymbol{c} = \boldsymbol{0}$, which means $\boldsymbol{Q}\boldsymbol{z}^* = \boldsymbol{c}$. Now, since $\overline{X^2} - (\overline{X})^2 = \frac{1}{n}\sum_{i=1}^n (x_i - \overline{X})^2$, and the x_i are not all equal, then $\det \boldsymbol{Q} = \overline{X^2} - (\overline{X})^2 \neq 0$. Hence, $\boldsymbol{Q}$ is nonsingular, and hence

$$\boldsymbol{z}^* = \boldsymbol{Q}^{-1}\boldsymbol{c} = \frac{1}{\overline{X^2} - (\overline{X})^2}\begin{bmatrix} 1 & -\overline{X} \\ -\overline{X} & \overline{X^2} \end{bmatrix}\begin{bmatrix} \overline{XY} \\ \overline{Y} \end{bmatrix} = \begin{bmatrix} \frac{\overline{XY} - (\overline{X})(\overline{Y})}{\overline{X^2} - (\overline{X})^2} \\ \frac{(\overline{X^2})(\overline{Y}) - (\overline{X})(\overline{XY})}{\overline{X^2} - (\overline{X})^2} \end{bmatrix}.$$

Since $\boldsymbol{Q} > 0$, then by the SOSC, the point $\boldsymbol{z}^*$ is a strict local minimizer. Since $\boldsymbol{z}^*$ is the only point satisfying the FONC, then $\boldsymbol{z}^*$ is the only local minimizer.

c. We have

$$a^*\overline{X} + b^* = \left(\frac{\overline{XY} - (\overline{X})(\overline{Y})}{\overline{X^2} - (\overline{X})^2}\right)\overline{X} + \frac{(\overline{X^2})(\overline{Y}) - (\overline{X})(\overline{XY})}{\overline{X^2} - (\overline{X})^2} = \overline{Y}.$$

6.30

Given $\boldsymbol{x} \in \mathbb{R}^n$, let

$$f(\boldsymbol{x}) = \frac{1}{p}\sum_{i=1}^{p} \|\boldsymbol{x} - \boldsymbol{x}^{(i)}\|^2$$

be the average squared error between $\boldsymbol{x}$ and $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(p)}$. We can rewrite f as

$$\begin{aligned} f(\boldsymbol{x}) &= \frac{1}{p}\sum_{i=1}^{p} (\boldsymbol{x} - \boldsymbol{x}^{(i)})^\top (\boldsymbol{x} - \boldsymbol{x}^{(i)}) \\ &= \boldsymbol{x}^\top \boldsymbol{x} - 2\left(\frac{1}{p}\sum_{i=1}^{p} \boldsymbol{x}^{(i)}\right)^\top \boldsymbol{x} + \frac{1}{p}\|\boldsymbol{x}^{(i)}\|^2. \end{aligned}$$

So f is a quadratic function. Since $\bar{\boldsymbol{x}}$ is the minimizer of f, then by the FONC, $\nabla f(\bar{\boldsymbol{x}}) = \boldsymbol{0}$, i.e.,

$$2\bar{\boldsymbol{x}} - 2\frac{1}{p}\sum_{i=1}^{p} \boldsymbol{x}^{(i)} = \boldsymbol{0}.$$

Hence, we get

$$\bar{\boldsymbol{x}} = \frac{1}{p}\sum_{i=1}^{p} \boldsymbol{x}^{(i)},$$

i.e., $\bar{\boldsymbol{x}}$ is just the average, or centroid, or center of gravity, of $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(p)}$.

The Hessian of f at $\bar{\boldsymbol{x}}$ is

$$\boldsymbol{F}(\bar{\boldsymbol{x}}) = 2\boldsymbol{I}_n,$$

which is positive definite. Hence, by the SOSC, $\bar{\boldsymbol{x}}$ is a strict local minimizer of f (in fact, it is a strict global minimizer because f is a convex quadratic function).

6.31

Fix any $\boldsymbol{x} \in \Omega$. The vector $\boldsymbol{d} = \boldsymbol{x} - \boldsymbol{x}^*$ is feasible at $\boldsymbol{x}^*$ (by convexity of Ω). By Taylor's formula, we have

$$f(\boldsymbol{x}) = f(\boldsymbol{x}^*) + \boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) + o(\|\boldsymbol{d}\|) \geq f(\boldsymbol{x}^*) + c\|\boldsymbol{d}\| + o(\|\boldsymbol{d}\|).$$

Therefore, for all $\boldsymbol{x}$ sufficiently close to $\boldsymbol{x}^*$, we have $f(\boldsymbol{x}) > f(\boldsymbol{x}^*)$. Hence, $\boldsymbol{x}^*$ is a strict local minimizer.

6.32

Since $f \in \mathcal{C}^2$, $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{F}^\top(\boldsymbol{x}^*)$. Let $\boldsymbol{d} \neq \boldsymbol{0}$ be a feasible directions at $\boldsymbol{x}^*$. By Taylor's theorem,

$$f(\boldsymbol{x}^* + \boldsymbol{d}) - f(\boldsymbol{x}^*) = \frac{1}{2}\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) + \boldsymbol{d}^\top \boldsymbol{F}(\boldsymbol{x}^*)\boldsymbol{d} + o(\|\boldsymbol{d}\|^2).$$

Using conditions a and b, we get

$$f(\boldsymbol{x}^* + \boldsymbol{d}) - f(\boldsymbol{x}^*) \geq c\|\boldsymbol{d}\|^2 + o(\|\boldsymbol{d}\|^2),$$

Therefore, for all $\boldsymbol{d}$ such that $\|\boldsymbol{d}\|$ is sufficiently small,

$$f(\boldsymbol{x}^* + \boldsymbol{d}) > f(\boldsymbol{x}^*),$$

and the proof is completed.

6.33

Necessity follows from the FONC. To prove sufficiency, we write f as

$$f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^\top \boldsymbol{Q}(\boldsymbol{x} - \boldsymbol{x}^*) - \frac{1}{2}\boldsymbol{x}^{*\top}\boldsymbol{Q}\boldsymbol{x}^*$$

where $\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{b}$ is the unique vector satisfying the FONC. Clearly, since $\frac{1}{2}\boldsymbol{x}^{*\top}\boldsymbol{Q}\boldsymbol{x}^*$ is a constant, and $\boldsymbol{Q} > 0$, then

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) = -\frac{1}{2}\boldsymbol{x}^{*\top}\boldsymbol{Q}\boldsymbol{x}^*,$$

and $f(\boldsymbol{x}) = f(\boldsymbol{x}^*)$ if and only if $\boldsymbol{x} = \boldsymbol{x}^*$.

6.34

Write $\boldsymbol{u} = [u_1, \ldots, u_n]$. We have

$$\begin{aligned} x_n &= ax_{n-1} + bu_n \\ &= a(ax_{n-2} + bu_{n-1}) + bu_n \\ &= a^2 x_{n-2} + abu_{n-1} + bu_n \\ &\vdots \\ &= a^n x_0 + a^{n-1} bu_1 + \cdots + abu_{n-1} + bu_n \\ &= \boldsymbol{c}^\top \boldsymbol{u}, \end{aligned}$$

where $\boldsymbol{c} = [a^{n-1}b, \ldots, ab, b]^\top$. Therefore, the problem can be written as

$$\text{minimize}\, r\boldsymbol{u}^\top \boldsymbol{u} - q\boldsymbol{c}^\top \boldsymbol{u},$$

which is a positive definite quadratic in $\boldsymbol{u}$. The solution is therefore

$$\boldsymbol{u} = \frac{q}{2r}\boldsymbol{c},$$

or, equivalently, $u_i = qa^{n-i}b/(2r)$, $i = 1, \ldots, n$.

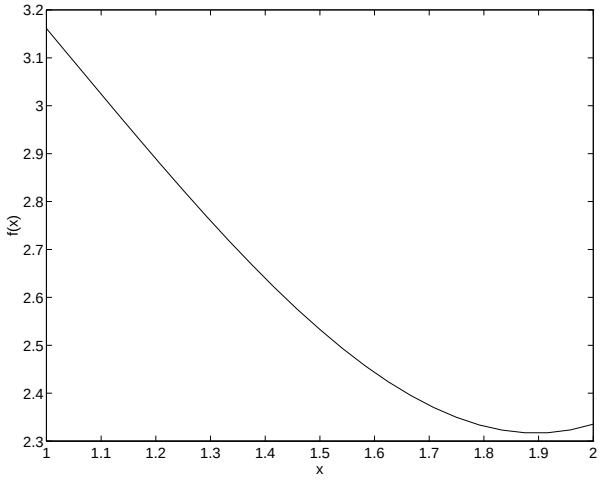
7. One Dimensional Search Methods

7.1

The range reduction factor for 3 iterations of the Golden Section method is $((\sqrt{5}-1/2)^3 = 0.236$, while that of the Fibonacci method (with $\varepsilon = 0$) is $1/F_{3+1} = 0.2$. Hence, if the desired range reduction factor is anywhere between 0.2 and 0.236 (e.g., 0.21), then the Golden Section method requires at least 4 iterations, while the Fibonacci method requires only 3. So, an example of a desired final uncertainty range is $0.21 \times (8-5) = 0.63$.

7.2

a. The plot of $f(x)$ versus x is as below:



b. The number of steps needed for the Golden Section method is computed from the inequality:

$$0.61803^N \leq \frac{0.2}{2-1} \qquad \Rightarrow \qquad N \geq 3.34.$$

Therefore, the fewest possible number of steps is 4. Applying 4 steps of the Golden Section method, we end up with an uncertainty interval of $[a_4, b_0] = [1.8541, 2.000]$. The table with the results of the intermediate steps is displayed below:

Iteration k	a_k	b_k	$f(a_k)$	$f(b_k)$	New uncertainty interval
1	1.3820	1.6180	2.6607	2.4292	[1.3820,2]
2	1.6180	1.7639	2.4292	2.3437	[1.6180,2]
3	1.7639	1.8541	2.3437	2.3196	[1.7639,2]
4	1.8541	1.9098	2.3196	2.3171	[1.8541,2]

c. The number of steps needed for the Fibonacci method is computed from the inequality:

$$\frac{1+2\varepsilon}{F_{N+1}} \leq \frac{0.2}{2-1} \qquad \Rightarrow \qquad N \geq 4.$$

Therefore, the fewest possible number of steps is 4. Applying 4 steps of the Fibonacci method, we end up with an uncertainty interval of $[a_4, b_0] = [1.8750, 2.000]$. The table with the results of the intermediate steps is displayed below:

Iteration k	ρ_k	a_k	b_k	$f(a_k)$	$f(b_k)$	New unc. int.
1	0.3750	1.3750	1.6250	2.6688	2.4239	[1.3750,2]
2	0.4	1.6250	1.7500	2.4239	2.3495	[1.6250,2]
3	0.3333	1.7500	1.8750	2.3495	2.3175	[1.7500,2]
4	0.45	1.8750	1.8875	2.3175	2.3169	[1.8750,2]

d. We have $f'(x) = 2x - 4\sin x$, $f''(x) = 2 - 4\cos x$. Hence, Newton's algorithm takes the form:

$$x^{(k+1)} = x^{(k)} - \frac{x^{(k)} - 2\sin x^{(k)}}{1 - 2\cos x^{(k)}}.$$

Applying 4 iterations with $x^{(0)} = 1$, we get $x^{(1)} = -7.4727$, $x^{(2)} = 14.4785$, $x^{(3)} = 6.9351$, $x^{(4)} = 16.6354$. Apparently, Newton's method is not effective in this case.

7.3

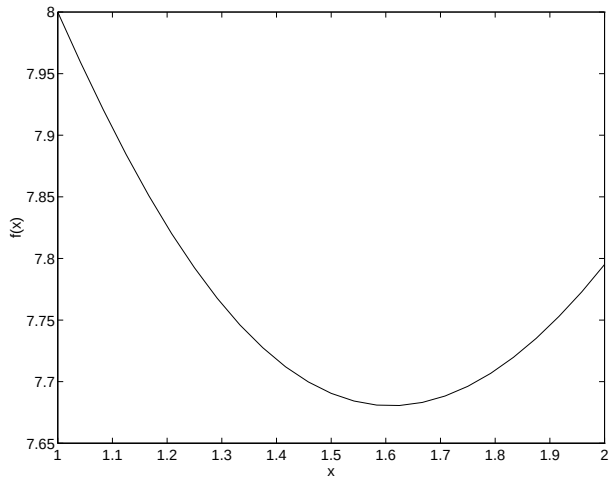
a. We first create the M-file `f.m` as follows:

```
% f.m
function y=f(x)
y=8*exp(1-x)+7*log(x);
```

The MATLAB commands to plot the function are:

```
fplot('f',[1 2]);
xlabel('x');
ylabel('f(x)');
```

The resulting plot is as follows:



b. The MATLAB routine for the Golden Section method is:

```
%Matlab routine for Golden Section Search

left=1;
right=2;
uncert=0.23;

rho=(3-sqrt(5))/2;

N=ceil(log(uncert/(right-left))/log(1-rho)) %print N

lower='a';
a=left+(1-rho)*(right-left);
f_a=f(a);
for i=1:N,
  if lower=='a'
    b=a
    f_b=f_a
    a=left+rho*(right-left)
    f_a=f(a)
  else
    a=b
    f_a=f_b
    b=left+(1-rho)*(right-left)
    f_b=f(b)
  end %if
  if f_a<f_b
    right=b;
    lower='a'
  else
    left=a;
    lower='b'
  end %if
  New_Interval = [left,right]
end %for i
%-----------------------------------------------------------------
```

Using the above routine, we obtain $N = 4$ and a final interval of $[1.528, 1.674]$. The table with the results of the intermediate steps is displayed below:

Iteration k	a_k	b_k	$f(a_k)$	$f(b_k)$	New uncertainty interval
1	1.382	1.618	7.7247	7.6805	[1.382,2]
2	1.618	1.764	7.6805	7.6995	[1.382,1.764]
3	1.528	1.618	7.6860	7.6805	[1.528,1.764]
4	1.618	1.674	7.6805	7.6838	[1.528,1.674]

c. The MATLAB routine for the Fibonacci method is:

```
%Matlab routine for Fibonacci Search technique

left=1;
right=2;
uncert=0.23;
epsilon=0.05;

F(1)=1;
F(2)=1;

N=0;
while F(N+2) < (1+2*epsilon)*(right-left)/uncert
  F(N+3)=F(N+2)+F(N+1);
  N=N+1;
end %while

N %print N

lower='a';
a=left+(F(N+1)/F(N+2))*(right-left);
f_a=f(a);

for i=1:N,
  if i~=N
    rho=1-F(N+2-i)/F(N+3-i)
  else
    rho=0.5-epsilon
  end %if
  if lower=='a'
    b=a
    f_b=f_a
    a=left+rho*(right-left)
    f_a=f(a)
  else
    a=b
    f_a=f_b
    b=left+(1-rho)*(right-left)
    f_b=f(b)
  end %if
  if f_a<f_b
    right=b;
    lower='a'
  else
    left=a;
    lower='b'
  end %if
  New_Interval = [left,right]
end %for i
%-----------------------------------------------------------------
```

Using the above routine, we obtain $N = 3$ and a final interval of $[1.58, 1.8]$. The table with the results of the intermediate steps is displayed below:

Iteration k	ρ_k	a_k	b_k	$f(a_k)$	$f(b_k)$	New uncertainty interval
1	0.4	1.4	1.6	7.7179	7.6805	[1.4,2]
2	0.333	1.6	1.8	7.6805	7.7091	[1.4,1.8]
3	0.45	1.58	1.6	7.6812	7.6805	[1.58,1.8]

7.4

Now, $\rho_k = 1 - \frac{F_{N-k+1}}{F_{N-k+2}}$. Hence,

$$\begin{aligned} 1 - \frac{\rho_k}{1-\rho_k} &= 1 - \frac{1 - F_{N-k+1}/F_{N-k+2}}{F_{N-k+1}/F_{N-k+2}} \\ &= 1 - \frac{F_{N-k+2} - F_{N-k+1}}{F_{N-k+1}} \\ &= 1 - \frac{F_{N-k}}{F_{N-k+1}} \\ &= \rho_{k+1} \end{aligned}$$

To show that $0 \leq \rho_k \leq 1/2$, we proceed by induction. Clearly $\rho_1 = 1/2$ satisfies $0 \leq \rho_1 \leq 1/2$. Suppose $0 \leq \rho_k \leq 1/2$, where $k \in \{1, \ldots, N-1\}$. Then,

$$\frac{1}{2} \leq 1 - \rho_k \leq 1$$

and hence

$$1 \leq \frac{1}{1-\rho_k} \leq 2.$$

Therefore,

$$\frac{1}{2} \leq \frac{\rho_k}{1-\rho_k} \leq 1.$$

Since $\rho_{k+1} = 1 - \frac{\rho_k}{1-\rho_k}$, then

$$0 \leq \rho_{k+1} \leq \frac{1}{2}$$

as required.

7.5

We proceed by induction. For $k = 2$, we have $F_0F_3 - F_1F_2 = (1)(3) - (1)(2) = 1 = (-1)^2$. Suppose $F_{k-2}F_{k+1} - F_{k-1}F_k = (-1)^k$. Then,

$$\begin{aligned} F_{k-1}F_{k+2} - F_kF_{k+1} &= F_{k-1}(F_{k+1} + F_k) - (F_{k-1} + F_{k-2})F_{k+1} \\ &= F_{k-1}F_k - F_{k-2}F_{k+1} \\ &= -(-1)^k \\ &= (-1)^{k+1}. \end{aligned}$$

7.6

Define $y_k = F_k$ and $z_k = F_{k-1}$. Then, we have

$$\begin{bmatrix} y_{k+1} \\ z_{k+1} \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} y_k \\ z_k \end{bmatrix},$$

where

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},$$

with initial condition

$$\begin{bmatrix} y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

We can write

$$F_n = y_n = [1,0] \begin{bmatrix} y_n \\ z_n \end{bmatrix} = [1,0]\boldsymbol{A}^n \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Since $\boldsymbol{A}$ is symmetric, it can be diagonalized as

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{u}^\top \\ \boldsymbol{v}^\top \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{u} & \boldsymbol{v} \end{bmatrix}$$

where

$$\lambda_1 = \frac{1+\sqrt{5}}{2}, \qquad \lambda_2 = \frac{1-\sqrt{5}}{2},$$

and

$$\boldsymbol{u} = -\frac{1}{5^{1/4}} \begin{bmatrix} \sqrt{2/(\sqrt{5}-1)} \\ \sqrt{(\sqrt{5}-1)/2} \end{bmatrix}, \qquad \boldsymbol{v} = \frac{1}{5^{1/4}} \begin{bmatrix} \sqrt{(\sqrt{5}-1)/2} \\ -\sqrt{2/(\sqrt{5}-1)} \end{bmatrix}.$$

Therefore, we have

$$\begin{aligned} F_n &= \boldsymbol{u}^\top \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}^n \boldsymbol{u} \\ &= u_1^2 \lambda_1^n + u_2^2 \lambda_2^n \\ &= \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^{n+1} - \left(\frac{1-\sqrt{5}}{2} \right)^{n+1} \right). \end{aligned}$$

7.7

The number $\log(2)$ is the root of the equation $g(x) = 0$, where $g(x) = \exp(x) - 2$. The derivative of g is $g'(x) = \exp(x)$. Newton's method applied to this root finding problem is

$$x^{(k+1)} = x^{(k)} - \frac{\exp(x^{(k)}) - 2}{\exp(x^{(k)})} = x^{(k)} - 1 + 2\exp(-x^{(k)}).$$

Performing two iterations, we get $x^{(1)} = 0.7358$ and $x^{(2)} = 0.6940$.

7.8

a. We compute $g'(x) = 2e^x/(e^x+1)^2$. Therefore Newton's method of tangents for this problem takes the form

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - \frac{(e^{x^{(k)}} - 1)/(e^{x^{(k)}} + 1)}{2e^{x^{(k)}}/(e^{x^{(k)}} + 1)^2} \\ &= x^{(k)} - \frac{(e^{2x^{(k)}} - 1)}{2e^{x^{(k)}}} \\ &= x^{(k)} - \sinh x^{(k)}. \end{aligned}$$

b. By symmetry, we need $x^{(1)} = -x^{(0)}$ for cycling. Therefore, $x^{(0)}$ must satisfy

$$-x^{(0)} = x^{(0)} - \sinh x^{(0)}.$$

The algorithm cycles if $x^{(0)} = c$, where $c > 0$ is the solution to $2c = \sinh c$.

c. The algorithm converges to 0 if and only if $|x^{(0)}| < c$, where c is from part b.

7.9

The quadratic function that matches the given data $x^{(k)}$, $x^{(k-1)}$, $x^{(k-2)}$, $f(x^{(k)})$, $f(x^{(k-1)})$, and $f(x^{(k-2)})$ can be computed by solving the following three linear equations for the parameters a, b, and c:

$$a(x^{(k-i)})^2 + bx^{(k-i)} + c = f(x^{(k-i)}), \quad i = 0, 1, 2.$$

Then, the algorithm is given by $x^{(k+1)} = -b/2a$ (so, in fact, we only need to find the *ratio* of a and b). With some elementary algebra (e.g., using Cramer's rule without needing to calculate the determinant in the denominator), the algorithm can be written as:

$$x^{(k+1)} = \frac{\sigma_{12} f(x^{(k)}) + \sigma_{20} f(x^{(k-1)}) + \sigma_{01} f(x^{(k-2)})}{2(\delta_{12} f(x^{(k)}) + \delta_{20} f(x^{(k-1)}) + \delta_{01} f(x^{(k-2)}))}$$

where $\sigma_{ij} = (x^{(k-i)})^2 - (x^{(k-j)})^2$ and $\delta_{ij} = x^{(k-i)} - x^{(k-j)}$.

7.10

a. A MATLAB routine for implementing the secant method is as follows.

```
function [x,v] = secant(g,xcurr,xnew,uncert);
%Matlab routine for finding root of g(x) using secant method
%
%  secant;
%  secant(g);
%  secant(g,xcurr,xnew);
%  secant(g,xcurr,xnew,uncert);
%
%  x=secant;
%  x=secant(g);
%  x=secant(g,xcurr,xnew);
%  x=secant(g,xcurr,xnew,uncert);
%
%  [x,v]=secant;
%  [x,v]=secant(g);
%  [x,v]=secant(g,xcurr,xnew);
%  [x,v]=secant(g,xcurr,xnew,uncert);
%
%The first variant finds the root of g(x) in the M-file g.m, with
%initial conditions 0 and 1, and uncertainty 10^(-5).
%The second variant finds the root of the function in the M-file specified
%by the string g, with initial conditions 0 and 1, and uncertainty 10^(-5).
%The third variant finds the root of the function in the M-file specified
%by the string g, with initial conditions specified by xcurr and xnew, and
%uncertainty 10^(-5).
%The fourth variant finds the root of the function in the M-file specified
%by the string g, with initial conditions specified by xcurr and xnew, and
%uncertainty specified by uncert.
%
%The next four variants returns the final value of the root as x.
%The last four variants returns the final value of the root as x, and
%the value of the function at the final value as v.

if nargin < 4
  uncert=10^(-5);
  if nargin < 3
    if nargin == 1
      xcurr=0;
      xnew=1;
    elseif nargin == 0
```

```
      g='g';
    else
      disp('Cannot have 2 arguments.');
      return;
    end
  end
end

g_curr=feval(g,xcurr);

while abs(xnew-xcurr)>xcurr*uncert,
  xold=xcurr;
  xcurr=xnew;
  g_old=g_curr;
  g_curr=feval(g,xcurr);
  xnew=(g_curr*xold-g_old*xcurr)/(g_curr-g_old);
end %while

%print out solution and value of g(x)
if nargout >= 1
  x=xnew;
  if nargout == 2
    v=feval(g,xnew);
  end
else
  final_point=xnew
  value=feval(g,xnew)
end %if
%-------------------------------------------------------------------
```

b. We get a solution of $x = 0.0039671$, with corresponding value $g(x) = -9.908 \times 10^{-8}$.

7.11

```
function alpha=linesearch_secant(grad,x,d)
%Line search using secant method

epsilon=10^(-4); %line search tolerance
max = 100; %maximum number of iterations
alpha_curr=0;
alpha=0.001;
dphi_zero=feval(grad,x)'*d;
dphi_curr=dphi_zero;

i=0;
while abs(dphi_curr)>epsilon*abs(dphi_zero),
  alpha_old=alpha_curr;
  alpha_curr=alpha;
  dphi_old=dphi_curr;
  dphi_curr=feval(grad,x+alpha_curr*d)'*d;
  alpha=(dphi_curr*alpha_old-dphi_old*alpha_curr)/(dphi_curr-dphi_old);
  i=i+1;
  if (i >= max) & (abs(dphi_curr)>epsilon*abs(dphi_zero)),
    disp('Line search terminating with number of iterations:');
    disp(i);
    break;
  end
end %while
%-------------------------------------------------------------------
```

7.12

a. We could carry out the bracketing using the one-dimensional function $\phi_0(\alpha) = f(\boldsymbol{x}^{(0)} + \alpha \boldsymbol{d}^{(0)})$, where $\boldsymbol{d}^{(0)}$ is the negative gradient at $\boldsymbol{x}^{(0)}$, as described in Section 7.8. The decision variable would be α. However, here we will directly represent the points in $\mathbb{R}^2$ (which is equivalent, though unnecessary in general).

The uncertainty interval is calculated by the following procedure:

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \boldsymbol{x}, \qquad \nabla f(\boldsymbol{x}) = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \boldsymbol{x}$$

Therefore,

$$\boldsymbol{d} = -\nabla f(\boldsymbol{x}^{(0)}) = -\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} = \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix}$$

As the problem requires, we use $\varepsilon = 0.075$.

First, we begin calculating $f(\boldsymbol{x}^{(0)})$ and $\boldsymbol{x}^{(1)}$:

$$f(\boldsymbol{x}^{(0)}) = f\left(\begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix}\right) = 0.5025,$$

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \varepsilon d = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} + 0.075 \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix} = \begin{bmatrix} 0.6987 \\ -0.2725 \end{bmatrix}$$

Then, we proceed as follows to find the uncertainty interval:

$$f(\boldsymbol{x}^{(1)}) = f\left(\begin{bmatrix} 0.6987 \\ -0.2725 \end{bmatrix}\right) = 0.3721$$

$$\boldsymbol{x}^{(2)} = \boldsymbol{x}^{(0)} + 2\varepsilon d = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} + 0.15 \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix} = \begin{bmatrix} 0.5975 \\ -0.2950 \end{bmatrix}$$

$$f(\boldsymbol{x}^{(2)}) = f\left(\begin{bmatrix} 0.5975 \\ -0.2950 \end{bmatrix}\right) = 0.2678$$

$$\boldsymbol{x}^{(3)} = \boldsymbol{x}^{(0)} + 4\varepsilon d = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} + 0.3 \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix} = \begin{bmatrix} 0.3950 \\ -0.3400 \end{bmatrix}$$

$$f(\boldsymbol{x}^{(3)}) = f\left(\begin{bmatrix} 0.3950 \\ -0.3400 \end{bmatrix}\right) = 0.1373$$

$$\boldsymbol{x}^{(4)} = \boldsymbol{x}^{(0)} + 8\varepsilon d = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} + 0.6 \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix} = \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}$$

$$f(\boldsymbol{x}^{(4)}) = f\left(\begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right) = 0.1893$$

Between $f(\boldsymbol{x}^{(3)})$ and $f(\boldsymbol{x}^{(4)})$ the function increases, which means that the minimizer must occur on the interval $[\boldsymbol{x}^{(2)}, \boldsymbol{x}^{(4)}] = \left[\begin{bmatrix} 0.5975 \\ -0.2950 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right]$, with $\boldsymbol{d} = \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix}$.

MATLAB code to solve the problem is listed next.

```
% Coded by David Schvartzman
```

```
% In our case we have:
Q = [2 1; 1 2];
x0 = [0.8; -.25];
e = 0.075;
f = zeros(1,10);
X = zeros(2,10);
x1 = x0;
d = -Q*x1;
f(1) = 0.5*(x1')*Q*x1;
for i=2:10
  X(:,i) = x1+e*d;
  f(i) = 0.5*X(:,i)'*Q*X(:,i);
  e = 2*e;
  if(f(i) > f(i-1))
    break;
  end
end
% The interval is defined by:
a = X(:,i-2);
b = X(:,i);
str = sprintf('The minimizer is located in: [a, b], where a = [%.4f; %.4f]...
 and b = [%.4f; %.4f]', a(1,1), a(2,1), b(1,1), b(2,1));
disp(str);
```

b. First, we determine the number of necessary iterations:
The initial uncertainty interval width is 0.6223. This width will be $(0.6223)(0.618)^N$ after N stages. We choose N so that

$$(0.618)^N \leq \frac{0.01}{0.6223} \qquad \Rightarrow N = 9$$

We show the first iteration of the algorithm; the rest are analogous and shown in the following table. From part a, we know that $[a_0, b_0] = [x^{(2)}, x^{(4)}]$, then:

$$[a_0, b_0] = \left[\begin{bmatrix} 0.5975 \\ -0.2950 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right], \text{ with } f(a_0) = 0.2678,\ f(b_0) = 0.1893.$$

$a_1 = a_0 + \rho(b_0 - a_0) = [0.3655\ -0.3466]^T$
$b_1 = a_0 + (1-\rho)(b_0 - a_0) = [0.2220\ -0.3784]^T$
$f(a_1) = 0.1270$
$f(b_1) = 0.1085$

We can see $f(a_1) > f(b_1)$, hence the uncertainty interval is reduced to:

$$[a_1, b_0] = \left[\begin{bmatrix} 0.3655 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right]$$

So, calculating the norm of $(b_0 - a_1)$, we see that the uncertainty region width is now 0.38461.

Iteration	a_k	b_k	$f(a_k)$	$f(b_k)$	New Uncertainty Interval
1	$\begin{bmatrix} 0.3655 \\ -0.3466 \end{bmatrix}$	$\begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}$	0.1270	0.1085	$\left[\begin{bmatrix} .3655 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} 0.0100 \\ -0.4300 \end{bmatrix}\right]$
2	$\begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}$	$\begin{bmatrix} 0.1334 \\ -0.3981 \end{bmatrix}$	0.1085	0.1232	$\left[\begin{bmatrix} .3655 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} .1334 \\ -0.3981 \end{bmatrix}\right]$
3	$\begin{bmatrix} 0.2768 \\ -0.3663 \end{bmatrix}$	$\begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}$	0.1094	0.1085	$\left[\begin{bmatrix} .2768 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} .1334 \\ -0.3981 \end{bmatrix}\right]$
4	$\begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}$	$\begin{bmatrix} 0.1882 \\ -0.3860 \end{bmatrix}$	0.1085	0.1117	$\left[\begin{bmatrix} .2768 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} .1882 \\ -0.3860 \end{bmatrix}\right]$
5	$\begin{bmatrix} 0.2430 \\ -0.3738 \end{bmatrix}$	$\begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}$	0.1079	0.1085	$\left[\begin{bmatrix} .2768 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} .2220 \\ -0.3784 \end{bmatrix}\right]$
6	$\begin{bmatrix} 0.2559 \\ -0.3709 \end{bmatrix}$	$\begin{bmatrix} 0.2430 \\ -0.3738 \end{bmatrix}$	0.1081	0.1079	$\left[\begin{bmatrix} 0.2559 \\ -0.3709 \end{bmatrix}, \begin{bmatrix} 0.2220 \\ -0.3784 \end{bmatrix}\right]$
7	$\begin{bmatrix} 0.2430 \\ -0.3738 \end{bmatrix}$	$\begin{bmatrix} 0.2350 \\ -0.3756 \end{bmatrix}$	0.1079	0.1080	$\left[\begin{bmatrix} 0.2559 \\ -0.3709 \end{bmatrix}, \begin{bmatrix} 0.2350 \\ -0.3756 \end{bmatrix}\right]$
8	$\begin{bmatrix} 0.2479 \\ -0.3727 \end{bmatrix}$	$\begin{bmatrix} 0.2430 \\ -0.3738 \end{bmatrix}$	0.1080	0.1079	$\left[\begin{bmatrix} 0.2479 \\ -0.3727 \end{bmatrix}, \begin{bmatrix} 0.2350 \\ -0.3756 \end{bmatrix}\right]$
9	$\begin{bmatrix} 0.2430 \\ -0.3738 \end{bmatrix}$	$\begin{bmatrix} 0.2399 \\ -0.3745 \end{bmatrix}$	0.1079	0.1079	$\left[\begin{bmatrix} 0.2479 \\ -0.3727 \end{bmatrix}, \begin{bmatrix} 0.2399 \\ -0.3745 \end{bmatrix}\right]$

We can now see that the minimizer is located within $\left[\begin{bmatrix} 0.2479 \\ -0.3727 \end{bmatrix}, \begin{bmatrix} 0.2399 \\ -0.3745 \end{bmatrix}\right]$, and its uncertainty interval width is 0.00819.

Matlab code used to perform calculations is listed next

```
% Coded by David Schvartzman
% To succesfully run this program, run
% the previous script to obatin a and b.
e = 0.01;
Q = [2 1; 1 2];
ro = 0.5*(3-sqrt(5));
% First we determine the number of necessary iterations.
d = norm(a-b);N = ceil( (log(e/d))/(log(1-ro)));
fa = 0.5*a'*Q*a;
fb = 0.5*b'*Q*b;
str1 = sprintf('Initial values:');
str2 = sprintf('a0 = [%.4f,%.4f].', a(1), a(2));
str3 = sprintf('b0 = [%.4f,%.4f].', b(1), b(2));
str4 = sprintf('f(a0) = %.4f.', fa);
str5 = sprintf('f(b0) = %.4f.', fb);
```

```
strn = sprintf('\n');
disp(strn);
disp(str1);disp(str2);
disp(str3);disp(str4);
disp(str5);
s = a + ro*(b-a);
t = a + (1-ro)*(b-a);
fs = 0.5*s'*Q*s;
ft = 0.5*t'*Q*t;
for i=1:N
  str1 = sprintf('Iteration number: %d', i);
  str2 = sprintf('a%d = [%.4f,%.4f].', i, s(1), s(2));
  str3 = sprintf('b%d = [%.4f,%.4f].', i, t(1), t(2));
  str4 = sprintf('f(a%d) = %.4f.', i, fs);
  str5 = sprintf('f(b%d) = %.4f.', i, ft);
  if (ft>fs)
     b = t;
     %fb = ft;
     t = s;
     ft = fs;
     s = a + ro*(b-a);
     fs = 0.5*s'*Q*s;
  else
     a = s;
     %fa = fs;
     s = t;
     fs = ft;
     t = a + (1-ro)*(b-a);
     ft = 0.5*t'*Q*t;
  end
  str6 = sprintf('New uncertainty interval: a%d = [%.4f,%.4f], ...
  b%d = [%.4f,%.4f].', i, a(1), a(2), i, b(1), b(2));
  disp(strn);
  disp(str1)
  disp(str2)
  disp(str3)
  disp(str4)
  disp(str5)
  disp(str6)
end
% The interval where the minimizer is boxed in is given by:
an = a;
bn = b;
% We can return (an+bn)/2 as the minimizer.
min = (an+bn)/2;disp(strn);
str = sprintf('The minimizer x* is: [%.4f; %.4f]', min(1,1), min(2,1));
disp(str);
```

c. We need to determine the number of necessary iterations:
The initial uncertainty interval width is 0.6223. This width will be $(0.6223)\frac{1+2\varepsilon}{F_{(N+1)}}$, where F_k is the k-th element of the Fibonacci sequence. We choose N so that

$$\frac{1+2\varepsilon}{F_{N+1}} \leq \frac{0.01}{0.6223} = 0.0161 \qquad \Rightarrow F_{N+1} \geq \frac{1+2\varepsilon}{0.0161}$$

For $\varepsilon = 0.05$, we require $F_{(N+1)} > 68.32$, thus $F_{(10)} = 89$ is enough, and we have $N = 10 - 1$, 9 iterations.

We show the first iteration of the algorithm; the rest are analogous and shown in the following table. From part a, we know that $[a_0, b_0] = [x^{(2)}, x^{(4)}]$, then:

$$[a_0, b_0] = \left[\begin{bmatrix} 0.5975 \\ -0.2950 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right], \text{ with } f(a_0) = 0.2678,\ f(b_0) = 0.1893.$$

Recall that in the Fibonacci method, $\rho_1 = 1 - \frac{F_N}{F_{N+1}} = 1 - \frac{55}{89} = 0.3820\,.$

$a_1 = a_0 + \rho_1(b_0 - a_0) = [0.3654\ -0.3466]^T$
$b_1 = a_0 + (1 - \rho_1)(b_0 - a_0) = [0.2221\ -0.3784]^T$
$f(a_1) = 0.1270$
$f(b_1) = 0.1085$

We can see $f(a_1) > f(b_1)$, hence the uncertainty interval is reduced to:

$$[a_1, b_0] = \left[\begin{bmatrix} 0.3654 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right]$$

So, calculating the norm of $(b_0 - a_1)$, we see that the uncertainty region width is now 0.38458.

k	ρ_k	a_k	b_k	$f(a_k)$	$f(b_k)$	New Uncertainty Interval
1	0.3820	$\begin{bmatrix} 0.3654 \\ -0.3466 \end{bmatrix}$	$\begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}$	0.1270	0.1085	$\left[\begin{bmatrix} 0.3654 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} -0.0100 \\ -0.4300 \end{bmatrix}\right]$
2	0.3818	$\begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}$	$\begin{bmatrix} 0.1333 \\ -0.3981 \end{bmatrix}$	0.1085	0.1232	$\left[\begin{bmatrix} 0.3654 \\ -0.3466 \end{bmatrix}, \begin{bmatrix} 0.1333 \\ -0.3981 \end{bmatrix}\right]$
3	0.3824	$\begin{bmatrix} 0.2767 \\ -0.3663 \end{bmatrix}$	$\begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}$	0.1094	0.1085	$\left[\begin{bmatrix} 0.2767 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} 0.1333 \\ -0.3981 \end{bmatrix}\right]$

k	ρ_k	a_k	b_k	$f(a_k)$	$f(b_k)$	New Uncertainty Interval
4	0.3810	$\begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}$	$\begin{bmatrix} 0.1879 \\ -0.3860 \end{bmatrix}$	0.1085	0.1118	$\left[\begin{bmatrix} 0.2767 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} 0.1879 \\ -0.3860 \end{bmatrix}\right]$
5	0.3846	$\begin{bmatrix} 0.2426 \\ -0.3739 \end{bmatrix}$	$\begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}$	0.1079	0.1085	$\left[\begin{bmatrix} 0.2767 \\ -0.3663 \end{bmatrix}, \begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}\right]$
6	0.3750	$\begin{bmatrix} 0.2562 \\ -0.3708 \end{bmatrix}$	$\begin{bmatrix} 0.2426 \\ -0.3739 \end{bmatrix}$	0.1082	0.1079	$\left[\begin{bmatrix} 0.2562 \\ -0.3708 \end{bmatrix}, \begin{bmatrix} 0.2221 \\ -0.3784 \end{bmatrix}\right]$
7	0.4000	$\begin{bmatrix} 0.2426 \\ -0.3739 \end{bmatrix}$	$\begin{bmatrix} 0.2357 \\ -0.3754 \end{bmatrix}$	0.1079	0.1080	$\left[\begin{bmatrix} 0.2562 \\ -0.3708 \end{bmatrix}, \begin{bmatrix} 0.2357 \\ -0.3754 \end{bmatrix}\right]$
8	0.3333	$\begin{bmatrix} 0.2494 \\ -0.3724 \end{bmatrix}$	$\begin{bmatrix} 0.2426 \\ -0.3739 \end{bmatrix}$	0.1080	0.1079	$\left[\begin{bmatrix} 0.2494 \\ -0.3724 \end{bmatrix}, \begin{bmatrix} 0.2357 \\ -0.3754 \end{bmatrix}\right]$
9	0.4500	$\begin{bmatrix} 0.2426 \\ -0.3739 \end{bmatrix}$	$\begin{bmatrix} 0.2419 \\ -0.3740 \end{bmatrix}$	0.1079	0.1079	$\left[\begin{bmatrix} 0.2494 \\ -0.3724 \end{bmatrix}, \begin{bmatrix} 0.2419 \\ -0.3740 \end{bmatrix}\right]$

We can now see that the minimizer is located within $\left[\begin{bmatrix} 0.2494 \\ -0.3724 \end{bmatrix}, \begin{bmatrix} 0.2419 \\ -0.3740 \end{bmatrix}\right]$, and its uncertainty interval width is 0.00769.

Matlab copde used to perform calculations is listed next.

```
% Coded by David Schvartzman
% To succesfully run this program, run the first of the above scripts
% to obtain a and b.
% We take
e = 0.01;
Q = [2 1; 1 2];
% First determine the number of necessary iterations.
d = norm(a-b);
F_N_1 =(2*0.05+1)/(e/d);
F = zeros(1,20);
F(1) = 0;
F(2) = 1;
for i=1:20
  F(i+1) = F(i)+F(i+1);
  if(F(i+2)>= F_N_1)
    break;
  end
end
N = i-1;
ro = zeros(1, N+1);
for i = 1:N
  ro(i) = 1 - F(N+3-i)/F(N+4-i);
end
ro(N) = ro(N) - 0.05;
fa = 0.5*a'*Q*a;
fb = 0.5*b'*Q*b;
```

```
str1 = sprintf('Initial values:');
str2 = sprintf('a0 = [%.4f,%.4f].', a(1), a(2));
str3 = sprintf('b0 = [%.4f,%.4f].', b(1), b(2));
str4 = sprintf('f(a0) = %.4f.', fa);
str5 = sprintf('f(b0) = %.4f.', fb);
strn = sprintf('\n');
disp(strn);
disp(str1);
disp(str2);
disp(str3);
disp(str4);
disp(str5);
s = a + ro(1)*(b-a);
t = a + (1-ro(1))*(b-a);
fs = 0.5*s'*Q*s;
ft = 0.5*t'*Q*t;
for i=1:N
  str1 = sprintf('Iteration number: %d', i);
  str2 = sprintf('a%d = [%.4f,%.4f].', i, s(1), s(2));
  str3 = sprintf('b%d = [%.4f,%.4f].', i, t(1), t(2));
  str4 = sprintf('f(a%d) = %.4f.', i, fs);
  str5 = sprintf('f(b%d) = %.4f.', i, ft);
  if (ft>fs)
    b = t;
    t = s;
    ft = fs;
    s = a + ro(i+1)*(b-a);
    fs = 0.5*s'*Q*s;
  else
    a = s;
    s = t;
    fs = ft;
    t = a + (1-ro(i+1))*(b-a);
    ft = 0.5*t'*Q*t;
  end
  str6 = sprintf('New uncertainty interval: a%d = [%.4f,%.4f],...
         b%d = [%.4f,%.4f].', i, a(1), a(2), i, b(1), b(2));
  str7 = sprintf('Uncertainty interval width: %.5f', norm(a-b));
  disp(strn);
  disp(str1)
  disp(str2)
  disp(str3)
  disp(str4)
  disp(str5)
  disp(str6)
  disp(str7)
end
% The minimizer is boxed in the interval:
an = a;
bn = b;
% We can return (an+bn)/2 as the minimizer.
min = (an+bn)/2;
disp(strn);
str = sprintf('The minimizer x* is: [%.4f; %.4f]', min(1,1), min(2,1));
```

```
disp(str);
```

8. Gradient Methods

8.1

The function f is a quadratic and so we can represent it in standard form as

$$f = \frac{1}{2}\boldsymbol{x}^\top \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \boldsymbol{x} - \boldsymbol{x}^\top \begin{bmatrix} -1 \\ -1/2 \end{bmatrix} + 3 = \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^\top \boldsymbol{b} + c.$$

The first iteration is

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} - \alpha_0 \nabla f\left(\boldsymbol{x}^{(0)}\right).$$

To find $\boldsymbol{x}^{(1)}$, we need to compute $\nabla f\left(\boldsymbol{x}^{(0)}\right) = \boldsymbol{g}^{(0)}$. We have

$$\boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} = \begin{bmatrix} 1 & 1/2 \end{bmatrix}^\top.$$

The step size, α_0, can be computed as

$$\alpha_0 = \frac{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{Q}\boldsymbol{g}^{(0)}} = \frac{5}{6}.$$

Hence,

$$\boldsymbol{x}^{(1)} = -\alpha_0 \boldsymbol{g}^{(0)} = -\frac{5}{6}\begin{bmatrix} 1 \\ 1/2 \end{bmatrix} = -\begin{bmatrix} 5/6 \\ 5/12 \end{bmatrix}.$$

The second iteration is

$$\boldsymbol{x}^{(2)} = \boldsymbol{x}^{(1)} - \alpha_1 \nabla f\left(\boldsymbol{x}^{(1)}\right),$$

where

$$\nabla f\left(\boldsymbol{x}^{(1)}\right) = \boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} - \boldsymbol{b} = \begin{bmatrix} 1/6 \\ -1/3 \end{bmatrix},$$

and

$$\alpha_1 = \frac{\boldsymbol{g}^{(1)\top}\boldsymbol{g}^{(1)}}{\boldsymbol{g}^{(1)\top}\boldsymbol{Q}\boldsymbol{g}^{(1)}} = \frac{5}{9}.$$

Hence,

$$\boldsymbol{x}^{(2)} = \boldsymbol{x}^{(1)} - \alpha_1 \boldsymbol{g}^{(1)} = \begin{bmatrix} -5/6 \\ -5/12 \end{bmatrix} - \frac{5}{9}\begin{bmatrix} 1/6 \\ -1/3 \end{bmatrix} = \begin{bmatrix} -\frac{25}{27} \\ -\frac{25}{108} \end{bmatrix}.$$

The optimal solution is $\boldsymbol{x}^* = [-1, -1/4]^\top$ obtaind by solving the equation $\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{b}$.

8.2

Let s be the order of convergence of $\{\boldsymbol{x}^{(k)}\}$. Suppose there exists $c > 0$ such that for all k sufficiently large,

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\| \geq c\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p.$$

Hence, for all k sufficiently large,

$$\begin{aligned} \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^s} &= \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p} \frac{1}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{s-p}} \\ &\geq \frac{c}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{s-p}}. \end{aligned}$$

Taking limits yields

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^s} \geq \frac{c}{\lim_{k\to\infty} \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{s-p}}.$$

Since by definition s is the order of convergence,

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^s} < \infty.$$

Combining the above two inequalities, we get

$$\frac{c}{\lim_{k\to\infty} \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{s-p}} < \infty.$$

Therefore, since $\lim_{k\to\infty} \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\| = 0$, we conclude that $s \leq p$, i.e., the order of convergence is at most p.

8.3

We use contradiction. Suppose $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^*$ and

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p} > 0$$

for some $p < 1$. We may assume that $\boldsymbol{x}^{(k)} \neq \boldsymbol{x}^*$ for an infinite number of k (for otherwise, by convention, the ratio above is eventually 0). Fix $\varepsilon > 0$. Then, there exists K_1 such that for all $k \geq K_1$,

$$\frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p} > \varepsilon.$$

Dividing both sides by $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{1-p}$, we obtain

$$\frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|} > \frac{\varepsilon}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{1-p}}.$$

Because $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^*$ and $p < 1$, we have $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{1-p} \to 0$. Hence, there exists K_2 such that for all $k \geq K_2$, $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^{1-p} < \varepsilon$. Combining this inequality with the previous one yields

$$\frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|} > 1$$

for all $k \geq \max(K_1, K_2)$; i.e.,

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\| > \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|,$$

which contradicts the assumption that $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^*$.

8.4

a. The sequence converges to 0, because the exponent -2^{k^2} grows unboundedly negative as $k \to \infty$.

b. The order of convergence of $\{x^{(k)}\}$ is ∞. To see this, we first write, for $p \geq 1$,

$$\begin{aligned}
\frac{|x^{(k+1)}|}{|x^{(k)}|^p} &= \frac{2^{-2^{(k+1)^2}}}{2^{-2^{k^2}p}} \\
&= 2^{-2^{k^2+2k+1}+p2^{k^2}} \\
&= 2^{-2^{k^2}(2^{2k+1}-p)}.
\end{aligned}$$

But notice that the exponent $-2^{k^2}(2^{2k+1} - p)$ grows unboundedly negative as $k \to \infty$, regardless of the value of p. Therefore, for any p,

$$\lim_{k\to\infty} \frac{|x^{(k+1)}|}{|x^{(k)}|^p} = 0,$$

which means that the order of convergence is ∞.

8.5

a. We have

$$\begin{aligned} x^{(k)} &= ax^{(k-1)} \\ &= a \cdot ax^{(k-2)} \\ &= a^2 x^{(k-2)} \\ &\vdots \\ &= a^k x^{(0)}. \end{aligned}$$

Because $0 < a < 1$, we have $a^k \to 0$, and hence $x^{(k)} \to 0$.

b. Similarly, we have

$$\begin{aligned} y^{(k)} &= (y^{(k-1)})^b \\ &= ((y^{(k-2)})^b)^b \\ &= (y^{(k-2)})^{b^2} \\ &\vdots \\ &= (y^{(0)})^{b^k}. \end{aligned}$$

Because $|y^{(0)}| < 1$ and $b > 1$, we have $b^k \to \infty$ and hence $y^{(k)} \to 0$.

c. The order of convergence of $\{x^{(k)}\}$ is 1 because

$$\lim_{k\to\infty} \frac{|x^{(k+1)}|}{|x^{(k)}|} = \lim_{k\to\infty} a = a,$$

and $0 < a < \infty$.

The order of convergence of $\{y^{(k)}\}$ is b because

$$\lim_{k\to\infty} \frac{|y^{(k+1)}|}{|y^{(k)}|^b} = \lim_{k\to\infty} 1 = 1,$$

and $0 < 1 < \infty$.

d. Suppose $|x^{(k)}| \le c|x^{(0)}|$. Using part a, we have $a^k \le c$, which implies that $k \ge \log(c)/\log(a)$. So the smallest number of iterations k such that $|x^{(k)}| \le c|x^{(0)}|$ is $\lceil \log(c)/\log(a) \rceil$ (the smallest integer not smaller than $\log(c)/\log(a)$).

e. Suppose $|y^{(k)}| \le c|y^{(0)}|$. Using part b, we have $|y^{(0)}|^{b^k} \le c|y^{(0)}|$. Taking logs (twice) and rearranging, we have

$$k \ge \frac{1}{\log(b)} \log\left(1 + \frac{\log(c)}{\log(|y^{(0)}|)}\right).$$

Denote the right-hand side by z. So the smallest number of iterations k such that $|y^{(k)}| \le c|y^{(0)}|$ is $\lceil z \rceil$.

f. Comparing the answer in part e with that of part d, we can see that as $c \to 0$, the answer in part d is $\Omega(\log(c))$, whereas the answer in part e is $O(\log\log(c))$. Hence, in the regime where c is very small, the number of iterations in part d (linear convergence) is (at least) exponentially larger than that in part e (superlinear convergence).

8.6

We have $u_{k+1} = (1-\rho)u_k$, and $u_k \to 0$. Therefore,

$$\lim_{k\to\infty} \frac{|u_{k+1}|}{|u_k|} = 1 - \rho > 0$$

and thus the order of convergence is 1.

8.7

a. The value of x^* (in terms of a, b, and c) that minimizes f is $x^* = b/a$.

b. We have $f'(x) = ax - b$. Therefore, the recursive equation for the DDS algorithm is

$$x^{(k+1)} = x^{(k)} - \alpha(ax^{(k)} - b) = (1 - \alpha a)x^{(k)} + \alpha b.$$

c. Let $\tilde{x} = \lim_{k\to\infty} x^{(k)}$. Taking limits of both sides of $x^{(k+1)} = x^{(k)} - \alpha(ax^{(k)} - b)$ (from part b), we get

$$\tilde{x} = \tilde{x} - \alpha(a\tilde{x} - b).$$

Hence, we get $\tilde{x} = b/a = x^*$.

d. To find the order of convergence, we compute

$$\begin{aligned}\frac{|x^{(k+1)} - b/a|}{|x^{(k)} - b/a|^p} &= \frac{|(1-\alpha a)x^{(k)} + \alpha b - b/a|}{|x^{(k)} - b/a|^p} \\ &= \frac{|(1-\alpha a)x^{(k)} - (1-\alpha a)b/a|}{|x^{(k)} - b/a|^p} \\ &= |1 - \alpha a||x^{(k)} - b/a|^{1-p}.\end{aligned}$$

Let $z^{(k)} = |1 - \alpha a||x^{(k)} - b/a|^{1-p}$. Note that $z^{(k)}$ converges to a finite nonzero number if and only if $p = 1$ (if $p < 1$, then $z^{(k)} \to 0$, and if $p > 1$, then $z^{(k)} \to \infty$). Therefore, the order of convergence of $\{x^{(k)}\}$ is 1,

e. Let $y^{(k)} = |x^{(k)} - b/a|$. From part d, after some manipulation we obtain

$$y^{(k+1)} = |1 - \alpha a|y^{(k)} = |1 - \alpha a|^{k+1}y^{(0)}.$$

The sequence $\{x^{(k)}\}$ converges (to b/a) if and only if $y^{(k)} \to 0$. This holds if and only if $|1 - \alpha a| < 1$, which is equivalent to $0 < \alpha < 2/a$.

8.8

We rewrite f as $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{x}$, where

$$\boldsymbol{Q} = \begin{bmatrix} 6 & 4 \\ 4 & 6 \end{bmatrix}$$

The characteristic polynomial of $\boldsymbol{Q}$ is $\lambda^2 - 12\lambda + 20$. Hence, the eigenvalues of $\boldsymbol{Q}$ are 2 and 10. Therefore, the largest range of values of α for which the algorithm is globally convergent is $0 < \alpha < 2/10$.

8.9

a. We can write $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}$, where $\boldsymbol{b} = [-4, -1]^\top$ and

$$\boldsymbol{Q} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}$$

is positive definite. Hence, the solution is

$$\boldsymbol{Q}^{-1}\boldsymbol{b} = \frac{1}{5}\begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix}\begin{bmatrix} -4 \\ -1 \end{bmatrix} = \begin{bmatrix} -2 \\ 1 \end{bmatrix}.$$

b. By part a, the algorithm is a fixed-step-size gradient algorithm for a problem with gradient $\boldsymbol{h}$. The eigenvalues of $\boldsymbol{Q}$ are 1 and 5. Hence, the largest range of values of α such that the algorithm is globally convergent to the solution is $0 < \alpha < 2/5$.

c. The eigenvectors of $\boldsymbol{Q}$ corresponding to eigenvalue 5 has the form $c[1, 1]^\top$, where $c \in \mathbb{R}$. Hence, to violate the descent property, we pick

$$\boldsymbol{x}^{(0)} = \boldsymbol{Q}^{-1}\boldsymbol{b} + c\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -3 \\ 0 \end{bmatrix}$$

where we choose $c = -1$ so that $\boldsymbol{x}^{(0)}$ has the specified form.

8.10

a. We have

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \begin{bmatrix} 3 & 1+a \\ 1+a & 3 \end{bmatrix} \boldsymbol{x} - \boldsymbol{x}^\top \begin{bmatrix} 1 \\ 1 \end{bmatrix} + b.$$

b. The unique global minimizer exists if and only if the Hessian is positive definite, which holds if and only if $(1+a)^2 < 9$ (by Sylvester's criterion). Hence, the largest set of values of a and b such that the global minimizer of f exists is given by $-4 < a < 2$ and $b \in \mathbb{R}$ (unrestricted).

The minimizer is given by

$$\boldsymbol{x}^* = \frac{1}{9-(1+a)^2} \begin{bmatrix} 3 & -(1+a) \\ -(1+a) & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{3-(1+a)}{9-(1+a)^2} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{4+a} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

c. The algorithm is a gradient algorithm with fixed step size $\alpha = 2/5$. The eigenvalues of the Hessian are (after some calculations) $4+a$ and $2-a$. For global convergence, we need $2/5 < 2/\lambda_{max}$, or $\lambda_{max} < 5$, where $\lambda_{max} = \max(4+a, 2-a)$. From this we deduce that $-3 < a < 1$. Hence, the largest set of values of a and b such that the algorithm is globally convergent is given by $-3 < a < 1$ and $b \in \mathbb{R}$ (unrestricted).

8.11

a. We have

$$\begin{aligned} f(x^{(k+1)}) &= (x^{(k+1)} - c)^2/2 \\ &= (x^{(k)} - \alpha_k(x^{(k)} - c) - c)^2/2 \\ &= (1-\alpha_k)^2(x^{(k)} - c)^2/2 \\ &= (1-\alpha_k)^2 f(x^{(k)}). \end{aligned}$$

b. We have $x^{(k)} \to c$ if and only if $f(x^{(k)}) \to 0$. Hence, the algorithm is globally convergent if and only if $f(x^{(k)}) \to 0$ for any x_0. From part a, we deduce that $f(x^{(k)}) \to 0$ for any x_0 if and only if $\prod_{k=0}^{\infty}(1-\alpha_k)^2 = 0$. Because $0 < \alpha < 1$, this condition is equivalent to $\prod_{k=0}^{\infty}(1-\alpha_k) = 0$, which holds if and only if

$$\sum_{k=0}^{\infty} \alpha_k = \infty.$$

8.12

The only local minimizer of f is $x^* = 1/\sqrt{3}$. Indeed, we have $f'(x^*) = 0$ and $f''(x^*) = 2\sqrt{3}$. To find the largest range of values of α such that the algorithm is locally convergent, we use a "linearization" argument: The algorithm is locally convergent if and only if the "linearized" algorithm $x^{(k+1)} = x^{(k)} - \alpha f''(x^*)(x^{(k)} - x^*)$ is globally convergent. But the linearized algorithm is just a fixed step size algorithm applied to a quadratic with second derivative $f''(x^*)$. Therefore, the largest range of values of α such that the algorithm is locally convergent is $0 < \alpha < 2/f''(x^*) = 1/\sqrt{3}$.

8.13

We use the formula from Lemma 8.1:

$$f(x^{(k+1)}) = (1-\gamma_k) f(x^{(k)})$$

(we have $V = f$ in this case). Using the expression for γ_k, we get, assuming $x^{(k)} \neq 1$,

$$\gamma_k = \alpha 4 \cdot 2^{-k}(1 - \alpha 2^{-k}).$$

Hence, $\gamma_k > 0$, which means that $f(x^{(k+1)}) < f(x^{(k)})$ if $x^{(k)} \neq 1$ for $k \geq 0$. This implies that the algorithm has the descent property (for $k \geq 0$).

We also note that

$$\sum_{k=0}^{\infty} \gamma_k = \alpha 4 \left(\sum_{k=0}^{\infty} 2^{-k} - \sum_{k=0}^{\infty} \alpha 4^{-k} \right) = \alpha 4 \left(2 - \frac{4\alpha}{3} \right) < \inf ty.$$

Since $\gamma_k > 0$ for all $k \geq 0$, we can apply the theorem given in class to deduce that the algorithm is not globally convergent.

8.14

We have

$$|x^{(k+1)} - x^*| = |x^{(k)} - x^* - f'(x^{(k)})/f''(x^*)|.$$

By Taylor's Theorem,

$$f'(x^{(k)}) = f'(x^*) + f''(x^*)(x^{(k)} - x^*) + O(|x^{(k)} - x^*|^2).$$

Since $f'(x^*) = 0$ by the FONC, we get

$$x^{(k)} - x^* - f'(x^{(k)})/f''(x^*) = O(|x^{(k)} - x^*|^2).$$

Combining the above with the first equation, we get

$$|x^{(k+1)} - x^*| = O(|x^{(k)} - x^*|^2),$$

which implies that the order of convergence is at least 2.

8.15

a. The objective function is a quadratic that can be written as

$$f(x) = (\boldsymbol{a}x - \boldsymbol{b})^\top (\boldsymbol{a}x - \boldsymbol{b}) = \|\boldsymbol{a}\|^2 x^2 - 2\boldsymbol{a}^\top \boldsymbol{b}x + \|\boldsymbol{b}\|^2.$$

Hence, the minimizer is $x^* = \boldsymbol{a}^\top \boldsymbol{b}/\|\boldsymbol{a}\|^2$.

b. Note that $f''(x) = 2\|\boldsymbol{a}\|^2$. Thus, by the result for fixed step size gradient algorithms, the required largest range for α is $(0, 1/\|\boldsymbol{a}\|^2)$.

8.16

a. We have

$$\begin{aligned} f(x) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 &= (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) \\ &= (\boldsymbol{x}^\top \boldsymbol{A}^\top - \boldsymbol{b}^\top)(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) \\ &= \boldsymbol{x}^\top (\boldsymbol{A}^\top \boldsymbol{A})\boldsymbol{x} - 2(\boldsymbol{A}^\top \boldsymbol{b})^\top \boldsymbol{x} + \boldsymbol{b}^\top \boldsymbol{b} \end{aligned}$$

which is a quadratic function. The gradient is given by $\nabla f(\boldsymbol{x}) = 2(\boldsymbol{A}^\top \boldsymbol{A})\boldsymbol{x} - 2(\boldsymbol{A}^\top \boldsymbol{b})$ and the Hessian is given by $\boldsymbol{F}(\boldsymbol{x}) = 2(\boldsymbol{A}^\top \boldsymbol{A})$.

b. The fixed step size gradient algorithm for solving the above optimization problem is given by

$$\begin{aligned} \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \alpha(2(\boldsymbol{A}^\top \boldsymbol{A})\boldsymbol{x}^{(k)} - 2\boldsymbol{A}^\top \boldsymbol{b}) \\ &= \boldsymbol{x}^{(k)} - 2\alpha \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{x}^{(k)} - \boldsymbol{b}). \end{aligned}$$

c. The largest range of values for α such that the algorithm in part b converges to the solution of the problem is given by

$$0 < \alpha < \frac{2}{\lambda_{max}(2\boldsymbol{A}^\top \boldsymbol{A})} = \frac{1}{4}.$$

8.17

a. We use contraposition. Suppose an eigenvalue of $\boldsymbol{A}$ is negative: $\boldsymbol{A}\boldsymbol{v} = \lambda \boldsymbol{v}$, where $\lambda < 0$ and $\boldsymbol{v}$ is a corresponding eigenvector. Choose $\boldsymbol{x}^{(0)} = \boldsymbol{v} + \boldsymbol{x}^*$. Then,

$$\boldsymbol{x}^{(1)} = \boldsymbol{v} + \boldsymbol{x}^* - \alpha(\boldsymbol{A}\boldsymbol{v} + \boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{b}) = \boldsymbol{v} + \boldsymbol{x}^* - \alpha\lambda \boldsymbol{v},$$

and hence

$$\boldsymbol{x}^{(1)} - \boldsymbol{x}^* = (1 - \alpha\lambda)(\boldsymbol{x}^{(0)} - \boldsymbol{x}^*).$$

Since $1 - \alpha\lambda > 1$, we conclude that the algorithm is not globally monotone.

b. Note that the algorithm is identical to a fixed step size gradient algorithm applied to a quadratic with Hessian $\boldsymbol{A}$. The eigenvalues of $\boldsymbol{A}$ are 1 and 5. Therefore, the largest range of values of α for which the algorithm is globally convergent is $0 < \alpha < 2/5$.

8.18

The steepest descent algorithm applied to the quadratic function f has the form

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)} = \boldsymbol{x}^{(k)} - \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{g}^{(k)}}{\boldsymbol{g}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k)}}\boldsymbol{g}^{(k)}.$$

$\Rightarrow$: If $\boldsymbol{x}^{(1)} = \boldsymbol{Q}^{-1}\boldsymbol{b}$, then

$$\boldsymbol{Q}^{-1}\boldsymbol{b} = \boldsymbol{x}^{(0)} - \alpha_0 \boldsymbol{g}^{(0)}.$$

Rearranging the above yields

$$\boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} = \alpha_0 \boldsymbol{Q}\boldsymbol{g}^{(0)}.$$

Since $\boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} \neq \boldsymbol{0}$, we have

$$\boldsymbol{Q}\boldsymbol{g}^{(0)} = \frac{1}{\alpha_0}\boldsymbol{g}^{(0)}$$

which means that $\boldsymbol{g}^{(0)}$ is an eigenvector of $\boldsymbol{Q}$ (with corresponding eigenvalue $1/\alpha_0$).

$\Leftarrow$: By assumption, $\boldsymbol{Q}\boldsymbol{g}^{(0)} = \lambda \boldsymbol{g}^{(0)}$, where $\lambda \in \mathbb{R}$. We want to show that $\boldsymbol{Q}\boldsymbol{x}^{(1)} = \boldsymbol{b}$. We have

$$\begin{aligned}
\boldsymbol{Q}\boldsymbol{x}^{(1)} &= \boldsymbol{Q}\left(\boldsymbol{x}^{(0)} - \frac{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{Q}\boldsymbol{g}^{(0)}}\boldsymbol{g}^{(0)}\right) \\
&= \boldsymbol{Q}\boldsymbol{x}^{(0)} - \frac{1}{\lambda}\frac{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}\boldsymbol{Q}\boldsymbol{g}^{(0)} \\
&= \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{g}^{(0)} \\
&= \boldsymbol{b}.
\end{aligned}$$

8.19

a. Possible. Pick f such that $\lambda_{\max} \geq 2\lambda_{\min}$ and $\boldsymbol{x}^{(0)}$ such that $\boldsymbol{g}^{(0)}$ is an eigenvector of $\boldsymbol{Q}$ with eigenvalue $\lambda_{\min}$. Then,

$$\alpha_0 = \frac{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{Q}\boldsymbol{g}^{(0)}} = \frac{1}{\lambda_{\min}} \geq \frac{2}{\lambda_{\max}}.$$

b. Not possible. Indeed, using Rayleigh's inequality,

$$\alpha_0 = \frac{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{Q}\boldsymbol{g}^{(0)}} \leq \frac{1}{\lambda_{\min}}.$$

8.20

a. We rewrite f as

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{x} - 22,$$

where

$$\boldsymbol{Q} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} -3 \\ 1 \end{bmatrix}.$$

The eigenvalues of $\boldsymbol{Q}$ are 1 and 5. Therefore, the range of values of the step size for which the algorithm converges to the minimizer is $0 < \alpha < 2/5$.

b. An eigenvector of $\boldsymbol{Q}$ corresponding to the eigenvalue 5 is $\boldsymbol{v} = [1, 1]^\top/5$. We have $\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{b} = [-11, 9]^\top/5$. Hence, an initial condition that results in the algorithm diverging is

$$\boldsymbol{x}^{(0)} = \boldsymbol{x}^* + \boldsymbol{v} = \begin{bmatrix} -2 \\ 2 \end{bmatrix}.$$

8.21

In both cases, we compute the Hessian $\boldsymbol{Q}$ of f, and find its largest eigenvalue $\lambda_{\max}$. Then the range we seek is $0 < \alpha < 2/\lambda_{\max}$.

a. In this case,

$$\boldsymbol{Q} = \begin{bmatrix} 6 & 4 \\ 4 & 6 \end{bmatrix},$$

with eigenvalues 2 and 10. Hence, the answer is $0 < \alpha < 1/5$.

b. In this case, again we have

$$\boldsymbol{Q} = \begin{bmatrix} 6 & 4 \\ 4 & 6 \end{bmatrix},$$

with eigenvalues 2 and 10. Hence, the answer is $0 < \alpha < 1/5$.

8.22

For the given algorithm we have

$$\gamma_k = \beta(2-\beta)\left(\frac{(\boldsymbol{g}^{(k)\top}\boldsymbol{g}^{(k)})^2}{(\boldsymbol{g}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k)})(\boldsymbol{g}^{(k)\top}\boldsymbol{Q}^{-1}\boldsymbol{g}^{(k)})}\right)$$

If $0 < \beta < 2$, then $\beta(2-\beta) > 0$, and by Lemma 8.2,

$$\gamma_k \geq \beta(2-\beta)\left(\frac{\lambda_{min}(\boldsymbol{Q})}{\lambda_{max}(\boldsymbol{Q})}\right) > 0$$

which implies that $\sum_{k=0}^{\infty}\gamma_k = \infty$. Hence, by Theorem 8.1, $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^*$ for any $\boldsymbol{x}^{(0)}$.

If $\beta \leq 0$ or $\beta \geq 2$, then $\beta(2-\beta) \leq 0$, and by Lemma 8.2,

$$\gamma_k \leq \beta(2-\beta)\left(\frac{\lambda_{max}(\boldsymbol{Q})}{\lambda_{min}(\boldsymbol{Q})}\right) < 0.$$

By Lemma 8.1, $V(\boldsymbol{x}^{(k)}) \geq V(\boldsymbol{x}^{(0)})$. Hence, if $\boldsymbol{x}^{(0)} \neq \boldsymbol{x}^*$, then $\{V(\boldsymbol{x}^{(k)}\}$ does not converge to 0, and consequently $\boldsymbol{x}^{(k)}$ does not converge to $\boldsymbol{x}^*$.

8.23

By Lemma 8.1, $V(\boldsymbol{x}^{(k+1)}) = (1-\gamma_k)V(\boldsymbol{x}^{(k)})$ for all k. Note that the algorithm has a descent property if and only if $V(\boldsymbol{x}^{(k+1)}) < V(\boldsymbol{x}^{(k)})$ whenever $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$. Clearly, whenever $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$, $V(\boldsymbol{x}^{(k+1)}) < V(\boldsymbol{x}^{(k)})$ if and only if $1 - \gamma_k < 1$. The desired result follows immediately.

8.24

We have

$$\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)} = \alpha_k \boldsymbol{d}^{(k)}$$

and hence

$$\langle \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}, \nabla f(\boldsymbol{x}^{(k+1)})\rangle = \alpha_k \langle \boldsymbol{d}^{(k)}, \nabla f(\boldsymbol{x}^{(k+1)})\rangle.$$

Now, let $\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)})$. Since α_k minimizes ϕ_k, then by the FONC, $\phi_k'(\alpha_k) = 0$. By the chain rule, $\phi_k'(\alpha) = \boldsymbol{d}^{(k)\top}\nabla f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)})$. Hence,

$$0 = \phi_k'(\alpha_k) = \boldsymbol{d}^{(k)\top}\nabla f(\boldsymbol{x}^{(k)} + \alpha_k\boldsymbol{d}^{(k)}) = \langle \boldsymbol{d}^{(k)}, \nabla f(\boldsymbol{x}^{(k+1)})\rangle,$$

and so

$$\langle \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}, \nabla f(\boldsymbol{x}^{(k+1)}) \rangle = 0.$$

8.25

A simple MATLAB routine for implementing the steepest descent method is as follows.

```
function [x,N]=steep_desc(grad,xnew,options);
%    STEEP_DESC('grad',x0);
%    STEEP_DESC('grad',x0,OPTIONS);
%
%    x = STEEP_DESC('grad',x0);
%    x = STEEP_DESC('grad',x0,OPTIONS);
%
%    [x,N] = STEEP_DESC('grad',x0);
%    [x,N] = STEEP_DESC('grad',x0,OPTIONS);
%
%The first variant finds the minimizer of a function whose gradient
%is described in grad (usually an M-file: grad.m), using a gradient
%descent algorithm with initial point x0. The line search used in the
%secant method.
%The second variant allows a vector of optional parameters to
%defined. OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results, (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required of the gradient.
%OPTIONS(14) is the maximum number of iterations.
%For more information type HELP FOPTIONS.
%
%The next two variants returns the value of the final point.
%The last two variants returns a vector of the final point and the
%number of iterations.

if nargin ~= 3
  options = [];
  if nargin ~= 2
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=1000*length(xnew);
  end
else
  options(14)=1000*length(xnew);
end

clc;
format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_g = options(3);
max_iter=options(14);

for k = 1:max_iter,
```

```
  xcurr=xnew;
  g_curr=feval(grad,xcurr);
  if norm(g_curr) <= epsilon_g
    disp('Terminating: Norm of gradient less than');
    disp(epsilon_g);
    k=k-1;
      break;
    end %if

  alpha=linesearch_secant(grad,xcurr,-g_curr);

  xnew = xcurr-alpha*g_curr;

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('Gradient = ');
    disp(g_curr'); %print gradient
    disp('New point =');
    disp(xnew'); %print new point
  end %if

  if norm(xnew-xcurr) <= epsilon_x*norm(xcurr)
    disp('Terminating: Norm of difference between iterates less than');
    disp(epsilon_x);
    break;
  end %if

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xnew');
  disp('Number of iterations =');
  disp(k);
end %if
%-----------------------------------------------------------------
```

To apply the above MATLAB routine to the function in Example 8.1, we need the following M-file to specify the gradient.

```
function y=g(x)

y=[4*(x(1)-4).^3; 2*(x(2)-3); 16*(x(3)+5).^3];
```

We applied the algorithm as follows:

```
>> options(2) = 10^(-6);
>> options(3) = 10^(-6);
```

```
>> steep_desc('g',[-4;5;1],options)

Terminating: Norm of gradient less than
   1.0000e-06
Final point =
   4.0022e+00   3.0000e+00  -4.9962e+00
Number of iterations =
    25
```

As we can see above, we obtained the final point $[4.002, 3.000, -4.996]^\top$ after 25 iterations. The value of the objective function at the final point is 7.2×10^{-10}.

8.26

The algorithm terminated after 9127 iterations. The final point was $[0.99992, 0.99983]^\top$.

9. Newton's Method

9.1

a. We have $f'(x) = 4(x - x_0)^3$ and $f''(x) = 12(x - x_0)^2$. Hence, Newton's method is represented as

$$x^{(k+1)} = x^{(k)} - \frac{x^{(k)} - x_0}{3},$$

which upon rewriting becomes

$$x^{(k+1)} - x_0 = \frac{2}{3}\left(x^{(k)} - x_0\right)$$

b. From part a, $y^{(k)} = |x^{(k)} - x_0| = (2/3)|x^{(k-1)} - x_0| = (2/3)y^{(k-1)}$.

c. From part b, we see that $y^{(k)} = (2/3)^k y^{(0)}$ and therefore $y^{(k)} \to 0$. Hence $x^{(k)} \to x_0$ for any $x^{(0)}$.

d. From part b, we have

$$\lim_{k\to\infty} \frac{|x^{(k+1)} - x_0|}{|x^{(k)} - x_0|} = \lim_{k\to\infty} \frac{2}{3} = \frac{2}{3} > 0$$

and hence the order of convergence is 1.

e. The theorem assumes that $f''(x^*) \neq 0$. However, in this problem, $x^* = x_0$, and $f''(x^*) = 0$.

9.2

a. We have

$$|x^{(k+1)} - x^*| = |x^{(k)} - x^* - \alpha_k f'(x^{(k)})|.$$

By Taylor's theorem applied to f',

$$f'(x^{(k)}) = f'(x^*) + f''(x^*)(x^{(k)} - x^*) + o(|x^{(k)} - x^*|).$$

Since $f'(x^*) = 0$ by the FONC, we get

$$\begin{aligned} x^{(k)} - x^* - \alpha_k f'(x^{(k)}) &= (1 - \alpha_k f''(x^*))(x^{(k)} - x^*) + \alpha_k o(|x^{(k)} - x^*|) \\ &= o(|x^{(k)} - x^*|) + \alpha_k o(|x^{(k)} - x^*|) \\ &= (1 + \alpha_k) o(|x^{(k)} - x^*|). \end{aligned}$$

Because $\{\alpha_k\}$ converges, it is bounded, and so $(1 + \alpha_k)o(|x^{(k)} - x^*|) = o(|x^{(k)} - x^*|)$. Combining the above with the first equation, we get

$$|x^{(k+1)} - x^*| = o(|x^{(k)} - x^*|),$$

which implies that the order of convergence is superlinear.

b. In the secant algorithm, if $x^{(k)} \to x^*$, then $(f'(x^{(k)}) - f'(x^{(k-1)}))/(x^{(k)} - x^{(k-1)}) \to f''(x^*)$. Since the secant algorithm has the form $x^{(k+1)} = x^{(k)} - \alpha_k f'(x^{(k)})$ with $\alpha_k = (x^{(k)} - x^{(k-1)})/(f'(x^{(k)}) - f'(x^{(k-1)}))$, we

deduce that $\alpha_k \to 1/f''(x^*)$. Hence, if we apply the secant algorithm to a function $f \in \mathcal{C}^2$, and it converges to a local minimizer x^* such that $f''(x^*) \neq 0$, then the order of convergence is superlinear.

9.3

a. We compute $f'(x) = 4x^{1/3}/3$ and $f''(x) = 4x^{-2/3}/9$. Therefore Newton's algorithm for this problem takes the form

$$x^{(k+1)} = x^{(k)} - \frac{4(x^{(k)})^{1/3}/3}{4(x^{(k)})^{-2/3}/9} = -2x^{(k)}.$$

b. From part a, we have $x^{(k)} = 2^k x^{(0)}$. Therefore, as long as $x^{(0)} \neq 0$, the sequence $\{x^{(k)}\}$ does not converge to 0.

9.4

a. Clearly $f(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$. We have

$$\begin{aligned} f(\boldsymbol{x}) = 0 &\Leftrightarrow x_2 - x_1^2 = 0 \quad \text{and} \quad 1 - x_1 = 0 \\ &\Leftrightarrow \boldsymbol{x} = [1,1]^\top. \end{aligned}$$

Hence, $f(\boldsymbol{x}) > f([1,1]^\top)$ for all $\boldsymbol{x} \neq [1,1]^\top$, and therefore $[1,1]^\top$ is the unique global minimizer.

b. We compute

$$\begin{aligned} \nabla f(\boldsymbol{x}) &= \begin{bmatrix} 400x_1^3 - 400x_1x_2 + 2x_1 - 2 \\ 200(x_2 - x_1^2) \end{bmatrix} \\ \boldsymbol{F}(\boldsymbol{x}) &= \begin{bmatrix} 1200x_1^2 - 400x_2 + 2 & -400x_1 \\ -400x_1 & 200 \end{bmatrix}. \end{aligned}$$

To apply Newton's method we use the inverse of the Hessian, which is

$$\boldsymbol{F}(\boldsymbol{x})^{-1} = \frac{1}{80000(x_1^2 - x_2) + 400} \begin{bmatrix} 200 & 400x_1 \\ 400x_1 & 1200x_1^2 - 400x_2 + 2 \end{bmatrix}.$$

Applying two iterations of Newton's method, we have $\boldsymbol{x}^{(1)} = [1,0]^\top$, $\boldsymbol{x}^{(2)} = [1,1]^\top$. Therefore, in this particular case, the method converges in two steps! We emphasize, however, that this fortuitous situation is by no means typical, and is highly dependent on the initial condition.

c. Applying the gradient algorithm $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \nabla f(\boldsymbol{x}^{(k)})$ with a fixed step size of $\alpha_k = 0.05$, we obtain $\boldsymbol{x}^{(1)} = [0.1, 0]^\top$, $\boldsymbol{x}^{(2)} = [0.17, 0.1]^\top$.

9.5

If $\boldsymbol{x}^{(0)} = \boldsymbol{x}^*$, we are done. So, assume $\boldsymbol{x}^{(0)} \neq \boldsymbol{x}^*$. Since the standard Newton's method reaches the point $\boldsymbol{x}^*$ in one step, we have

$$\begin{aligned} f(\boldsymbol{x}^*) &= f(\boldsymbol{x}^{(0)} + \boldsymbol{Q}^{-1}\boldsymbol{g}^{(0)}) \\ &= \min f(\boldsymbol{x}) \\ &\leq f(\boldsymbol{x}^{(0)} + \alpha \boldsymbol{Q}^{-1}\boldsymbol{g}^{(0)}) \end{aligned}$$

for any $\alpha \geq 0$. Hence,

$$\alpha_0 = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(0)} + \alpha \boldsymbol{Q}^{-1}\boldsymbol{g}^{(0)}) = 1.$$

Hence, in this case, the modified Newton's algorithm is equivalent to the standard Newton's algorithm, and thus $\boldsymbol{x}^{(1)} = \boldsymbol{x}^*$.

10. Conjugate Direction Methods

10.1

We proceed by induction to show that for $k = 0, \ldots, n-1$, the set $\{\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k)}\}$ is $\boldsymbol{Q}$-conjugate. We assume that $\boldsymbol{d}^{(i)} \neq \boldsymbol{0}$, $i = 1, \ldots, k$, so that $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} \neq 0$ and the algorithm is well defined.

For $k = 0$, the statement trivially holds. So, assume that the statement is true for $k < n-1$, i.e., $\{\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k)}\}$ is $\boldsymbol{Q}$-conjugate. We now show that $\{\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k+1)}\}$ is $\boldsymbol{Q}$-conjugate. For this, we need only to show that for each $j = 0, \ldots, k$, we have $\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = 0$. To this end,

$$\begin{aligned}
\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} &= \left(\boldsymbol{p}^{(k+1)\top} - \sum_{i=0}^{k} \frac{\boldsymbol{p}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)}}{\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)}}\boldsymbol{d}^{(i)\top}\right)\boldsymbol{Q}\boldsymbol{d}^{(j)} \\
&= \boldsymbol{p}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} - \sum_{i=0}^{k} \frac{\boldsymbol{p}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)}}{\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)}}\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)}.
\end{aligned}$$

By the induction hypothesis, $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = 0$ for $i \neq j$. Therefore

$$\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = \boldsymbol{p}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} - \frac{\boldsymbol{p}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)}}{\boldsymbol{d}^{(j)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)}}\boldsymbol{d}^{(j)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = 0.$$

In the above, we have assumed that the vectors $\boldsymbol{d}^{(k)}$ are nonzero (so that $\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)} \neq 0$ and the algorithm is well defined). To prove that this assumption holds, we use induction to show that $\boldsymbol{d}^{(k)}$ is a (nonzero) linear combination of $\boldsymbol{p}^{(0)}, \ldots, \boldsymbol{p}^{(k)}$ (which immediately implies that $\boldsymbol{d}^{(k)}$ is nonzero because of the linear independence of $\boldsymbol{p}^{(0)}, \ldots, \boldsymbol{p}^{(k)}$).

For $k = 0$, we have $\boldsymbol{d}^{(0)} = \boldsymbol{p}^{(0)}$ by definition. Assume that the result holds for $k < n-1$; i.e., $\boldsymbol{d}^{(k)} = \sum_{j=0}^{k} \alpha_j^{(k)}\boldsymbol{p}^{(j)}$, where the coefficients $\alpha_j^{(k)}$ are not all zero. Consider $\boldsymbol{d}^{(k+1)}$:

$$\begin{aligned}
\boldsymbol{d}^{(k+1)} &= \boldsymbol{p}^{(k+1)} - \sum_{i=0}^{k} \beta_i \boldsymbol{d}^{(i)} \\
&= \boldsymbol{p}^{(k+1)} - \sum_{i=0}^{k} \beta_i \sum_{j=0}^{i} \alpha_j^{(k)}\boldsymbol{p}^{(j)} \\
&= \boldsymbol{p}^{(k+1)} - \sum_{j=0}^{k} \sum_{i=j}^{k} \beta_i \alpha_j^{(k)}\boldsymbol{p}^{(j)}.
\end{aligned}$$

So, clearly $\boldsymbol{d}^{(k+1)}$ is a nonzero linear combination of $\boldsymbol{p}^{(0)}, \ldots, \boldsymbol{p}^{(k+1)}$.

10.2

Let $k \in \{0, \ldots, n-1\}$ and $\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)})$. By the chain rule, we have

$$\phi'(\alpha_k) = \nabla f(\boldsymbol{x}^{(k)} + \alpha_k\boldsymbol{d}^{(k)})^\top \boldsymbol{d}^{(k)} = \boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(k)}.$$

Since $\boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(k)} = 0$, we have $\phi'(\alpha_k) = 0$. Note that

$$\phi_k(\alpha) = \frac{1}{2}\left(\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}\right)\alpha^2 + \boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(k)}\alpha + \text{constant}.$$

As ϕ is a quadratic function of α (with positive coefficient in the quadratic term), we conclude that $\alpha_k = \arg\min_\alpha f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)})$.

Note that since $\boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(k)} \neq 0$ is the coefficient of the linear term in ϕ_k, we have $\alpha_k \neq 0$. For $i \in \{0, \ldots, k-1\}$, we have

$$\begin{aligned}
\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} &= \frac{1}{\alpha_k}(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})^\top\boldsymbol{Q}\boldsymbol{d}^{(i)} \\
&= \frac{1}{\alpha_k}(\boldsymbol{g}^{(k+1)} - \boldsymbol{g}^{(k)})^\top\boldsymbol{d}^{(i)} \\
&= \frac{1}{\alpha_k}\left(\boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(i)} - \boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(i)}\right) \\
&= 0
\end{aligned}$$

by assumption, which completes the proof.

10.3

From the conjugate gradient algorithm we have

$$\boldsymbol{d}^{(k)} = -\boldsymbol{g}^{(k)} + \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k-1)}}{\boldsymbol{d}^{(k-1)\top}\boldsymbol{Q}\boldsymbol{d}^{(k-1)}}\boldsymbol{d}^{(k-1)}.$$

Premultiplying the above by $\boldsymbol{d}^{(k)\top}\boldsymbol{Q}$ and using the fact that $\boldsymbol{d}^{(k)}$ and $\boldsymbol{d}^{(k-1)}$ are $\boldsymbol{Q}$-conjugate, yields

$$\begin{aligned}\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)} &= -\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k)} + \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k-1)}}{\boldsymbol{d}^{(k-1)\top}\boldsymbol{Q}\boldsymbol{d}^{(k-1)}}\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k-1)} \\ &= -\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k)}.\end{aligned}$$

10.4

a. Since $\boldsymbol{Q}$ is symmetric, then there exists a set of vectors $\{\boldsymbol{d}^{(1)}, \ldots, \boldsymbol{d}^{(n)}\}$ such that $\boldsymbol{Q}\boldsymbol{d}^{(i)} = \lambda_i \boldsymbol{d}^{(i)}$, $i = 1, \ldots, n$, and $\boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(j)} = 0$, $j \neq i$, where the λ_i are (real) eigenvalues of $\boldsymbol{Q}$. Therefore, if $i \neq j$, we have $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = \boldsymbol{d}^{(i)\top}(\lambda_j \boldsymbol{d}^{(j)}) = \lambda_j(\boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(j)}) = 0$. Hence the set $\{\boldsymbol{d}^{(1)}, \ldots, \boldsymbol{d}^{(n)}\}$ is $\boldsymbol{Q}$-conjugate.

b. Define $\lambda_i = (\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)})/(\boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(i)})$. Let

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{d}^{(1)\top} \\ \vdots \\ \boldsymbol{d}^{(n)\top} \end{bmatrix}.$$

Since $\boldsymbol{Q}$ is positive definite and the set $\{\boldsymbol{d}^{(1)}, \ldots, \boldsymbol{d}^{(n)}\}$ is $\boldsymbol{Q}$-conjugate, then by Lemma 10.1, the set is also linearly independent. Hence, $\boldsymbol{D}$ is nonsingular. By $\boldsymbol{Q}$-conjugacy, we have that for all $i \neq j$, $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = 0$. By assumption, we have $\boldsymbol{d}^{(i)\top}\lambda_j \boldsymbol{d}^{(j)} = \lambda_j \boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(j)} = 0$. Hence, $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(j)} = \lambda_j \boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(j)}$. Moreover, for each $i = 1, \ldots, n$, we have $\boldsymbol{d}^{(i)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} = \boldsymbol{d}^{(i)\top}\lambda_i \boldsymbol{d}^{(i)} = \lambda_i \boldsymbol{d}^{(i)\top}\boldsymbol{d}^{(i)}$. We can write the above conditions in matrix form:

$$\boldsymbol{D}\boldsymbol{Q}\boldsymbol{d}^{(i)} = \boldsymbol{D}(\lambda_i \boldsymbol{d}^{(i)}).$$

Since $\boldsymbol{D}$ is nonsingular, then we have

$$\boldsymbol{Q}\boldsymbol{d}^{(i)} = \lambda_i \boldsymbol{d}^{(i)},$$

which completes the proof.

10.5

We have

$$\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k+1)} = \gamma_k \boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k+1)} + \boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}.$$

Hence, in order to have $\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k+1)} = 0$, we need

$$\gamma_k = -\frac{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{g}^{(k+1)}}.$$

10.6

We use induction. For $k = 0$, we have

$$\boldsymbol{d}^{(0)} = a_0 \boldsymbol{g}^{(0)} = -a_0 \boldsymbol{b} \in \mathcal{V}_1.$$

Moreover, $\boldsymbol{x}^{(0)} = \boldsymbol{0} \in \mathcal{V}_0$. Hence, the proposition is true at $k = 0$. Assume it is true at k. To show that it is also true at $k + 1$, note first that

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}.$$

Because $\boldsymbol{x}^{(k)} \in \mathcal{V}_k \subset \mathcal{V}_{k+1}$ and $\boldsymbol{d}^{(k)} \in \mathcal{V}_{k+1}$ by the induction hypothesis, we deduce that $\boldsymbol{x}^{(k+1)} \in \mathcal{V}_{k+1}$. Moreover,

$$\begin{aligned} \boldsymbol{d}^{(k+1)} &= a_k \boldsymbol{g}^{(k+1)} + b_k \boldsymbol{d}^{(k)} \\ &= a_k(\boldsymbol{Q}\boldsymbol{x}^{(k+1)} - \boldsymbol{b}) + b_k \boldsymbol{d}^{(k)}. \end{aligned}$$

But because $\boldsymbol{x}^{(k+1)} \in \mathcal{V}_{k+1}$, $\boldsymbol{Q}\boldsymbol{x}^{(k+1)} - \boldsymbol{b} \in \mathcal{V}_{k+2}$. Moreover, $\boldsymbol{d}^{(k)} \in \mathcal{V}_{k+1} \subset \mathcal{V}_{k+2}$. Hence, $\boldsymbol{d}^{(k+1)} \in \mathcal{V}_{k+2}$. This completes the induction proof.

b. The conjugate gradient algorithm is an instance of the algorithm given in the question. By the "expanding subspace" theorem, we can say that in the conjugate gradient algorithm (with $\boldsymbol{x}^{(0)} = \boldsymbol{0}$), at each k, $\boldsymbol{x}^{(k)}$ is the global minimizer of f on the Krylov subspace $\mathcal{V}_k$. Note that for all $k \geq n$, $\mathcal{V}_{k+1} = \mathcal{V}_k$, because of the Cayley-Hamilton theorem, which allows us to express $\boldsymbol{Q}^n$ as a linear combination of $\boldsymbol{I}, \boldsymbol{Q}, \ldots, \boldsymbol{Q}^{n-1}$.

10.7

Expanding $\phi(\boldsymbol{a})$ yields

$$\begin{aligned} \phi(\boldsymbol{a}) &= \frac{1}{2}(\boldsymbol{x}_0 + \boldsymbol{D}\boldsymbol{a})^\top \boldsymbol{Q}(\boldsymbol{x}_0 + \boldsymbol{D}\boldsymbol{a}) - (\boldsymbol{x}_0 + \boldsymbol{D}\boldsymbol{a})^\top \boldsymbol{b} \\ &= \frac{1}{2}\boldsymbol{a}^\top \left(\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{D}\right)\boldsymbol{a} + \boldsymbol{a}^\top \left(\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{x}_0 - \boldsymbol{D}^\top \boldsymbol{b}\right) + \left(\frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{Q}\boldsymbol{x}_0 - \boldsymbol{x}_0^\top \boldsymbol{b}\right). \end{aligned}$$

Clearly ϕ is a quadratic function on $\mathbb{R}^r$. It remains to show that the matrix in the quadratic term, $\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{D}$, is positive definite. Since $\boldsymbol{Q} > 0$, for any $\boldsymbol{a} \in \mathbb{R}^r$, we have

$$\boldsymbol{a}^\top \left(\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{D}\right)\boldsymbol{a} = (\boldsymbol{D}\boldsymbol{a})^\top \boldsymbol{Q}(\boldsymbol{D}\boldsymbol{a}) \geq 0$$

and

$$\boldsymbol{a}^\top \left(\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{D}\right)\boldsymbol{a} = (\boldsymbol{D}\boldsymbol{a})^\top \boldsymbol{Q}(\boldsymbol{D}\boldsymbol{a}) = 0$$

if and only if $\boldsymbol{D}\boldsymbol{a} = \boldsymbol{0}$. Since $\operatorname{rank} \boldsymbol{D} = r$, $\boldsymbol{D}\boldsymbol{a} = \boldsymbol{0}$ if and only if $\boldsymbol{a} = \boldsymbol{0}$. Hence, the matrix $\boldsymbol{D}^\top \boldsymbol{Q}\boldsymbol{D}$ is positive definite.

10.8

a. Let $0 \leq k \leq n-1$ and $0 \leq i \leq k$. Then,

$$\begin{aligned} \boldsymbol{g}^{(k+1)T}\boldsymbol{g}^{(i)} &= \boldsymbol{g}^{(k+1)T}(\beta_{i-1}\boldsymbol{d}^{(i-1)} - \boldsymbol{d}^{(i)}) \\ &= \beta_{i-1}\boldsymbol{g}^{(k+1)T}\boldsymbol{d}^{(i-1)} - \boldsymbol{g}^{(k+1)T}\boldsymbol{d}^{(i)} \\ &= 0 \end{aligned}$$

by Lemma 10.2.

b. Let $0 \leq k \leq n-1$. and $0 \leq i \leq k-1$. Then,

$$\begin{aligned} &\boldsymbol{g}^{(k+1)T}\boldsymbol{Q}\boldsymbol{g}^{(i)} \\ &= (\beta_k \boldsymbol{d}^{(k)} - \boldsymbol{d}^{(k+1)})^T \boldsymbol{Q}(\beta_{i-1}\boldsymbol{d}^{(i-1)} - \boldsymbol{d}^{(i)}) \\ &= \beta_k\beta_{i-1}\boldsymbol{d}^{(k)T}\boldsymbol{Q}\boldsymbol{d}^{(i-1)} - \beta_k \boldsymbol{d}^{(k)T}\boldsymbol{Q}\boldsymbol{d}^{(i)} - \beta_{i-1}\boldsymbol{d}^{(k+1)T}\boldsymbol{Q}\boldsymbol{d}^{(i-1)} + \boldsymbol{d}^{(k+1)T}\boldsymbol{Q}\boldsymbol{d}^{(i)} \\ &= 0 \end{aligned}$$

by $\boldsymbol{Q}$-conjugacy of $\boldsymbol{d}^{(k+1)}$, $\boldsymbol{d}^{(k)}$, $\boldsymbol{d}^{(i)}$ and $\boldsymbol{d}^{(i-1)}$ (note that the iteration indices here are all distinct).

10.9

We represent f as

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix} \boldsymbol{x} - \boldsymbol{x}^\top \begin{bmatrix} 0 \\ 1 \end{bmatrix} - 7.$$

The conjugate gradient algorithm is based on the following formulas:

$$\begin{aligned}\boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}, \quad \alpha_k = -\frac{\boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}} \\ \boldsymbol{d}^{(k+1)} &= -\boldsymbol{g}^{(k+1)} + \beta_k \boldsymbol{d}^{(k)}, \quad \beta_k = \frac{\boldsymbol{g}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}.\end{aligned}$$

We have,

$$\boldsymbol{d}^{(0)} = \boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} = -\boldsymbol{b} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}.$$

We then proceed to compute

$$\alpha_0 = -\frac{\boldsymbol{g}^{(0)\top}\boldsymbol{d}^{(0)}}{\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)}} = -\frac{\begin{bmatrix} 0 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ -1 \end{bmatrix}}{\begin{bmatrix} 0 & -1 \end{bmatrix}\begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} 0 \\ -1 \end{bmatrix}} = -\frac{1}{2}.$$

Hence,

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0 \boldsymbol{d}^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{2}\begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2 \end{bmatrix}.$$

We next proceed by evaluating the gradient of the objective function at $\boldsymbol{x}^{(1)}$,

$$\boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} - \boldsymbol{b} = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} 0 \\ 1/2 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix}.$$

Because the gradient is nonzero, we can proceed with the next step where we compute

$$\beta_0 = \frac{\boldsymbol{g}^{(1)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)}}{\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)}} = \frac{\begin{bmatrix} -3/2 & 0 \end{bmatrix}\begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} 0 \\ -1 \end{bmatrix}}{\begin{bmatrix} 0 & -1 \end{bmatrix}\begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} 0 \\ -1 \end{bmatrix}} = -\frac{9}{4}.$$

Hence, the direction $\boldsymbol{d}^{(1)}$ is

$$\boldsymbol{d}^{(1)} = -\boldsymbol{g}^{(1)} + \beta_0 \boldsymbol{d}^{(0)} = \begin{bmatrix} 3/2 \\ 0 \end{bmatrix} - \frac{9}{4}\begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 3/2 \\ 9/4 \end{bmatrix}.$$

It is easy to verify that the directions $\boldsymbol{d}^{(0)}$ and $\boldsymbol{d}^{(1)}$ are $\boldsymbol{Q}$-conjugate. Indeed,

$$\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(1)} = \begin{bmatrix} 0 & -1 \end{bmatrix}\begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} 3/2 \\ 9/4 \end{bmatrix} = 0.$$

10.10

a. We have $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}^\top\boldsymbol{x}$ where

$$\boldsymbol{Q} = \begin{bmatrix} 5 & 2 \\ 2 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}.$$

b. Since f is a quadratic function on $\mathbb{R}^2$, we need to perform only two iterations. For the first iteration we compute

$$\begin{aligned} \boldsymbol{d}^{(0)} &= -\boldsymbol{g}^{(0)} = [3,1]^\top \\ \alpha_0 &= \frac{5}{29} \\ \boldsymbol{x}^{(1)} &= [0.51724, 0.17241]^\top \\ \boldsymbol{g}^{(1)} &= [-0.06897, 0.20690]^\top. \end{aligned}$$

For the second iteration we compute

$$\begin{aligned} \beta_0 &= 0.0047534 \\ \boldsymbol{d}^{(1)} &= [0.08324, -0.20214]^\top \\ \alpha_1 &= 5.7952 \\ \boldsymbol{x}^{(2)} &= [1.000, -1.000]^\top. \end{aligned}$$

c. The minimizer is given by $\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{b} = [1,-1]^\top$, which agrees with part b.

10.11

A MATLAB routine for the conjugate gradient algorithm with options for different formulas of β_k is:

```
function [x,N]=conj_grad(grad,xnew,options);
%    CONJ_GRAD('grad',x0);
%    CONJ_GRAD('grad',x0,OPTIONS);
%
%    x = CONJ_GRAD('grad',x0);
%    x = CONJ_GRAD('grad',x0,OPTIONS);
%
%    [x,N] = CONJ_GRAD('grad',x0);
%    [x,N] = CONJ_GRAD('grad',x0,OPTIONS);
%
%The first variant finds the minimizer of a function whose gradient
%is described in grad (usually an M-file: grad.m), using initial point
%x0.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results, (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required of the gradient.
%OPTIONS(5) specifies the formula for beta:
%   0=Powell;
%   1=Fletcher-Reeves;
%   2=Polak-Ribiere;
%   3=Hestenes-Stiefel.
%OPTIONS(14) is the maximum number of iterations.
%For more information type HELP FOPTIONS.
%
%The next two variants return the value of the final point.
%The last two variants return a vector of the final point and the
%number of iterations.

if nargin ~= 3
  options = [];
  if nargin ~= 2
    disp('Wrong number of arguments.');
    return;
  end
```

```
end

numvars = length(xnew);
if length(options) >= 14
  if options(14)==0
    options(14)=1000*numvars;
  end
else
  options(14)=1000*numvars;
end

clc;
format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_g = options(3);
max_iter=options(14);

g_curr=feval(grad,xnew);
if norm(g_curr) <= epsilon_g
  disp('Terminating: Norm of initial gradient less than');
  disp(epsilon_g);
  return;
end %if

d=-g_curr;
reset_cnt = 0;
for k = 1:max_iter,

  xcurr=xnew;
  alpha=linesearch_secant(grad,xcurr,d);
  %alpha=-(d'*g_curr)/(d'*Q*d);
  xnew = xcurr+alpha*d;

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('Gradient = ');
    disp(g_curr'); %print gradient
    disp('New point =');
    disp(xnew'); %print new point
  end %if

  if norm(xnew-xcurr) <= epsilon_x*norm(xcurr)
    disp('Terminating: Norm of difference between iterates less than');
    disp(epsilon_x);
    break;
  end %if

  g_old=g_curr;
  g_curr=feval(grad,xnew);

  if norm(g_curr) <= epsilon_g
    disp('Terminating: Norm of gradient less than');
```

```
    disp(epsilon_g);
    break;
  end %if

  reset_cnt = reset_cnt+1;
  if reset_cnt == 3*numvars
    d=-g_curr;
    reset_cnt = 0;
  else
    if options(5)==0 %Powell
      beta = max(0,(g_curr'*(g_curr-g_old))/(g_old'*g_old));
    elseif options(5)==1 %Fletcher-Reeves
      beta = (g_curr'*g_curr)/(g_old'*g_old);
    elseif options(5)==2 %Polak-Ribiere
      beta = (g_curr'*(g_curr-g_old))/(g_old'*g_old);
    else %Hestenes-Stiefel
      beta = (g_curr'*(g_curr-g_old))/(d'*(g_curr-g_old));
    end %if
    d=-g_curr+beta*d;
  end

  if print,
    disp('New beta =');
    disp(beta);
    disp('New d =');
    disp(d);
  end

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xnew');
  disp('Number of iterations =');
  disp(k);
end %if
%-------------------------------------------------------------------
```

We created the following M-file, `g.m`, for the gradient of Rosenbrock's function:

```
function y=g(x)
y=[-400*(x(2)-x(1).^2)*x(1)-2*(1-x(1)), 200*(x(2)-x(1).^2)]';
```

We tested the above routine as follows:

```
>> options(2)=10^(-7);
>> options(3)=10^(-7);
>> options(14)=100;
>> options(5)=0;
>> conj_grad('g',[-2;2],options);

Terminating: Norm of difference between iterates less than
```

```
   1.0000e-07
Final_Point =
   1.0000e+00   1.0000e+00
Number_of_iteration =
     8
>> options(5)=1;
>> conj_grad('g',[-2;2],options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final_Point =
   1.0000e+00   1.0000e+00
Number_of_iteration =
     10
>> options(5)=2;
>> conj_grad('g',[-2;2],options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final_Point =
   1.0000e+00   1.0000e+00
Number_of_iteration =
     8
>> options(5)=3;
>> conj_grad('g',[-2;2],options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final_Point =
   1.0000e+00   1.0000e+00
Number_of_iteration =
     8
```

The reader is cautioned *not* to draw any conclusions about the superiority or inferiority of any of the formulas for β_k based only on the above single numerical experiment.

11. Quasi-Newton Methods

11.1

a. Let

$$\phi(\alpha) = f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}).$$

Then, using the chain rule, we obtain

$$\phi'(\alpha) = \boldsymbol{d}^{(k)\top} \nabla f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}).$$

Hence

$$\phi'(0) = \boldsymbol{d}^{(k)\top} \boldsymbol{g}^{(k)}.$$

Since ϕ' is continuous, then, if $\boldsymbol{d}^{(k)\top} \boldsymbol{g}^{(k)} < 0$, there exists $\bar{\alpha} > 0$ such that for all $\alpha \in (0, \bar{\alpha}]$, $\phi(\alpha) < \phi(0)$, i.e., $f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}) < f(\boldsymbol{x}^{(k)})$.

b. By part a, $\phi(\alpha) < \phi(0)$ for all $\alpha \in (0, \bar{\alpha}]$. Hence,

$$\alpha_k = \arg\min_{\alpha \geq 0} \phi(\alpha) \neq 0$$

which implies that $\alpha_k > 0$.

c. Now,

$$\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k+1)} = \boldsymbol{d}^{(k)\top}\nabla f(\boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}) = \phi_k'(\alpha_k).$$

Since $\alpha_k = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}) > 0$, we have $\phi_k'(\alpha_k) = 0$. Hence, $\boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(k)} = 0$.

d.

i. We have $\boldsymbol{d}^{(k)} = -\boldsymbol{g}^{(k)}$. Hence, $\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} = -\|\boldsymbol{g}^{(k)}\|^2$. If $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$, then $\|\boldsymbol{g}^{(k)}\|^2 > 0$, and hence $\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} < 0$.

ii. We have $\boldsymbol{d}^{(k)} = -\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{g}^{(k)}$. Since $\boldsymbol{F}(\boldsymbol{x}^{(k)}) > 0$, we also have $\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1} > 0$. Therefore, $\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} = -\boldsymbol{g}^{(k)\top}\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{g}^{(k)} < 0$ if $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$.

iii. We have

$$\boldsymbol{d}^{(k)} = -\boldsymbol{g}^{(k)} + \beta_{k-1}\boldsymbol{d}^{(k-1)}.$$

Hence,

$$\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} = -\|\boldsymbol{g}^{(k)}\|^2 + \beta_{k-1}\boldsymbol{d}^{(k-1)\top}\boldsymbol{g}^{(k)}.$$

By part c, $\boldsymbol{d}^{(k-1)\top}\boldsymbol{g}^{(k)} = 0$. Hence, if $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$, then $\|\boldsymbol{g}^{(k)}\|^2 > 0$, and

$$\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} = -\|\boldsymbol{g}^{(k)}\|^2 < 0.$$

iv. We have $\boldsymbol{d}^{(k)} = -\boldsymbol{H}_k\boldsymbol{g}^{(k)}$. Therefore, if $\boldsymbol{H}_k > 0$ and $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$, then $\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)} = -\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\boldsymbol{g}^{(k)} < 0$.

e. Using the equation $\nabla f(\boldsymbol{x}) = \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}$, we get

$$\begin{aligned}
\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k+1)} &= \boldsymbol{d}^{(k)\top}(\boldsymbol{Q}\boldsymbol{x}^{(k+1)} - \boldsymbol{b}) \\
&= \boldsymbol{d}^{(k)\top}(\boldsymbol{Q}(\boldsymbol{x}^{(k)} + \alpha_k\boldsymbol{d}^{(k)}) - \boldsymbol{b}) \\
&= \alpha_k\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)} + \boldsymbol{d}^{(k)\top}(\boldsymbol{Q}\boldsymbol{x}^{(k)} - \boldsymbol{b}) \\
&= \alpha_k\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)} + \boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)}.
\end{aligned}$$

By part c, $\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k+1)} = 0$, which implies

$$\alpha_k = -\frac{\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}.$$

11.2

Yes, because:

1. The search direction is of the form $\boldsymbol{d}^{(k)} = \boldsymbol{H}_k\nabla f(\boldsymbol{x}^{(k)})$ for matrix $\boldsymbol{H}_k = \boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}$;
2. The matrix $\boldsymbol{H}_k = \boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}$ is symmetric for $f \in \mathcal{C}^2$;
3. If f is quadratic, then the quasi-Newton condition is satisfied: $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}$, $0 \leq i \leq k$. To see this, note that if the Hessian is $\boldsymbol{Q}$, then $\boldsymbol{Q}\Delta\boldsymbol{x}^{(i)} = \Delta\boldsymbol{g}^{(i)}$. Multiplying both sides by $\boldsymbol{H}_k = \boldsymbol{Q}^{-1}$, we obtain the desired result.

11.3

a. We have

$$f\left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right) = \frac{1}{2}\left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right)^\top \boldsymbol{Q}\left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right) - \left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right)^\top \boldsymbol{b} + c.$$

Using the chain rule, we obtain

$$\frac{d}{d\alpha}f\left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right) = \left(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)}\right)^\top \boldsymbol{Q}\boldsymbol{d}^{(k)} - \boldsymbol{d}^{(k)\top}\boldsymbol{b}.$$

Equating the above to zero and solving for α gives

$$\left(\boldsymbol{x}^{(k)\top}\boldsymbol{Q} - \boldsymbol{b}^\top\right)\boldsymbol{d}^{(k)} = -\alpha \boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}.$$

Taking into account that $\boldsymbol{g}^{(k)\top} = \boldsymbol{x}^{(k)\top}\boldsymbol{Q} - \boldsymbol{b}^\top$ and that $\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)} > 0$ for $\boldsymbol{g}^{(k)} \neq \boldsymbol{0}$, we obtain

$$\alpha_k = -\frac{\boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}} = \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\boldsymbol{g}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}.$$

b. The matrix $\boldsymbol{Q}$ is symmetric and positive definite; hence $\alpha_k > 0$ if $\boldsymbol{H}_k = \boldsymbol{H}_k^\top > 0$.

11.4

a. The appropriate choice is $\boldsymbol{H} = \boldsymbol{F}(\boldsymbol{x}^*)^{-1}$. To show this, we can apply the same argument as in the proof of the theorem on the convergence of Newton's method. (We won't repeat it here.)

b. Yes (provided we incorporate the usual step size). Indeed, if we apply the algorithm with the choice of $\boldsymbol{H}$ in part a, then when applied to a quadratic with Hessian $\boldsymbol{Q}$, the algorithm uses $\boldsymbol{H} = \boldsymbol{Q}^{-1}$, which definitely satisfies the quasi-Newton condition. In fact, the algorithm then behaves just like Newton's algorithm.

11.5

Our objective is to minimize the quadratic

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^\top\boldsymbol{b} + c.$$

We first compute the gradient ∇f and evaluate it at $\boldsymbol{x}^{(0)}$,

$$\nabla f\left(\boldsymbol{x}^{(0)}\right) = \boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

It is a non-zero vector, so we proceed with the first iteration. Let $\boldsymbol{H}_0 = \boldsymbol{I}_2$. Then,

$$\boldsymbol{d}^{(0)} = -\boldsymbol{H}_0\boldsymbol{g}^{(0)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

The step size α_0 is

$$\alpha_0 = -\frac{\boldsymbol{g}^{(0)\top}\boldsymbol{d}^{(0)}}{\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)}} = -\frac{\begin{bmatrix} -1 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix}}{\begin{bmatrix} 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix}} = \frac{2}{3}.$$

Hence,

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0\boldsymbol{d}^{(0)} = \begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix}.$$

We evaluate the gradient ∇f and evaluate it at $\boldsymbol{x}^{(1)}$ to obtain

$$\nabla f\left(\boldsymbol{x}^{(1)}\right) = \boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} - \boldsymbol{b} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1/3 \\ -1/3 \end{bmatrix}.$$

It is a non-zero vector, so we proceed with the second iteration. We compute $\boldsymbol{H}_1$, where

$$\boldsymbol{H}_1 = \boldsymbol{H}_0 + \frac{\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)^\top}{\Delta\boldsymbol{g}^{(0)\top}\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)}.$$

To find $\boldsymbol{H}_1$ we need to compute,

$$\Delta\boldsymbol{x}^{(0)} = \boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)} = \begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix} \quad \text{and} \quad \Delta\boldsymbol{g}^{(0)} = \boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)} = \begin{bmatrix} 2/3 \\ -4/3 \end{bmatrix}.$$

Using the above, we determine,

$$\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)} = \begin{bmatrix} 0 \\ 2/3 \end{bmatrix} \quad \text{and} \quad \Delta\boldsymbol{g}^{(0)\top}\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right) = -\frac{8}{9}.$$

Then, we obtain

$$\begin{aligned} \boldsymbol{H}_1 &= \boldsymbol{H}_0 + \frac{\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)^\top}{\Delta\boldsymbol{g}^{(0)\top}\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{\begin{bmatrix} 0 & 0 \\ 0 & 4/9 \end{bmatrix}}{-8/9} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1/2 \end{bmatrix} \end{aligned}$$

and

$$\boldsymbol{d}^{(1)} = -\boldsymbol{H}_1\boldsymbol{g}^{(1)} = \begin{bmatrix} 1/3 \\ 1/6 \end{bmatrix}.$$

We next compute

$$\alpha_1 = -\frac{\boldsymbol{g}^{(1)\top}\boldsymbol{d}^{(1)}}{\boldsymbol{d}^{(1)\top}\boldsymbol{Q}\boldsymbol{d}^{(1)}} = 1.$$

Therefore,

$$\boldsymbol{x}^{(2)} = \boldsymbol{x}^* = \boldsymbol{x}^{(1)} + \alpha_1\boldsymbol{d}^{(1)} = \begin{bmatrix} 1 \\ -1/2 \end{bmatrix}.$$

Note that $\boldsymbol{g}^{(2)} = \boldsymbol{Q}\boldsymbol{x}^{(2)} - \boldsymbol{b} = \boldsymbol{0}$ as expected.

11.6

We are guaranteed that the step size satisfies $\alpha_k > 0$ if the search direction is in the descent direction, i.e., the search direction $\boldsymbol{d}^{(k)} = -\boldsymbol{M}_k\nabla f(\boldsymbol{x}^{(k)})$ has strictly positive inner product with $-\nabla f(\boldsymbol{x}^{(k)})$ (see Exercise 11.1). Thus, the condition on $\boldsymbol{M}_k$ that guarantees $\alpha_k > 0$ is $\nabla f(\boldsymbol{x}^{(k)})^\top\boldsymbol{M}_k\nabla f(\boldsymbol{x}^{(k)}) > 0$, which corresponds to $1 + a > 0$, or $a > -1$. (Note that if $a \leq -1$, the search direction is not in the descent direction, and thus we cannot guarantee that $\alpha_k > 0$.)

11.7

Let $\boldsymbol{x} \in \mathbb{R}^n$. Then

$$\begin{aligned} \boldsymbol{x}^\top\boldsymbol{H}_{k+1}\boldsymbol{x} &= \boldsymbol{x}^\top\boldsymbol{H}_k\boldsymbol{x} + \boldsymbol{x}^\top\left(\frac{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})^\top}{\Delta\boldsymbol{g}^{(k)\top}(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})}\right)\boldsymbol{x} \\ &= \boldsymbol{x}^\top\boldsymbol{H}_k\boldsymbol{x} + \frac{(\boldsymbol{x}^\top(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}))^2}{\Delta\boldsymbol{g}^{(k)\top}(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})}. \end{aligned}$$

Note that since $\boldsymbol{H}_k > 0$, we have $\boldsymbol{x}^\top\boldsymbol{H}_k\boldsymbol{x} > 0$. Hence, if $\Delta\boldsymbol{g}^{(k)\top}(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}) > 0$, then $\boldsymbol{x}^\top\boldsymbol{H}_{k+1}\boldsymbol{x} > 0$.

11.8

The complement of the Rank One update equation is

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \frac{(\Delta\boldsymbol{g}^{(k)} - \boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)})(\Delta\boldsymbol{g}^{(k)} - \boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)})^\top}{\Delta\boldsymbol{x}^{(k)\top}(\Delta\boldsymbol{g}^{(k)} - \boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)})}.$$

Using the matrix inverse formula, we get

$$\begin{aligned}
&\boldsymbol{B}_{k+1}^{-1} \\
&\quad = \boldsymbol{B}_k^{-1} \\
&\quad - \frac{\boldsymbol{B}_k^{-1}(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{B}_k \Delta \boldsymbol{x}^{(k)})(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{B}_k \Delta \boldsymbol{x}^{(k)})\boldsymbol{B}_k^{-1}}{\Delta \boldsymbol{x}^{(k)\top}(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{B}_k \Delta \boldsymbol{x}^{(k)}) + (\Delta \boldsymbol{g}^{(k)} - \boldsymbol{B}_k \Delta \boldsymbol{x}^{(k)})^\top \boldsymbol{B}_k^{-1}(\Delta \boldsymbol{g}^{(k)} - \boldsymbol{B}_k \Delta \boldsymbol{x}^{(k)})} \\
&\quad = \boldsymbol{B}_k^{-1} + \frac{(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{B}_k^{-1}\Delta \boldsymbol{g}^{(k)})(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{B}_k^{-1}\Delta \boldsymbol{g}^{(k)})^\top}{\Delta \boldsymbol{g}^{(k)\top}(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{B}_k^{-1}\Delta \boldsymbol{g}^{(k)})}.
\end{aligned}$$

Substituting $\boldsymbol{H}_k$ for $\boldsymbol{B}_k^{-1}$, we get a formula identical to the Rank One update equation. This should not be surprising, since there is only one update equation involving a rank one correction that satisfies the quasi-Newton condition.

11.9

We first compute the gradient ∇f and evaluate it at $\boldsymbol{x}^{(0)}$,

$$\nabla f\left(\boldsymbol{x}^{(0)}\right) = \boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

It is a nonzero vector, so we proceed with the first iteration. Let $\boldsymbol{H}_0 = \boldsymbol{I}_2$. Then,

$$\boldsymbol{d}^{(0)} = -\boldsymbol{H}_0 \boldsymbol{g}^{(0)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

The step size α_0 is

$$\alpha_0 = -\frac{\boldsymbol{g}^{(0)\top}\boldsymbol{d}^{(0)}}{\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)}} = -\frac{\begin{bmatrix} -1 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix}}{\begin{bmatrix} 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix}} = \frac{2}{3}.$$

Hence,

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0 \boldsymbol{d}^{(0)} = \begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix}.$$

We evaluate the gradient ∇f and evaluate it at $\boldsymbol{x}^{(1)}$ to obtain

$$\nabla f\left(\boldsymbol{x}^{(1)}\right) = \boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} - \boldsymbol{b} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1/3 \\ -1/3 \end{bmatrix}.$$

It is a nonzero vector, so we proceed with the second iteration. We compute $\boldsymbol{H}_1$, where

$$\boldsymbol{H}_1 = \boldsymbol{H}_0 + \frac{\left(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}\right)\left(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}\right)^\top}{\Delta \boldsymbol{g}^{(0)\top}\left(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}\right)}.$$

To find $\boldsymbol{H}_1$ we need

$$\Delta \boldsymbol{x}^{(0)} = \boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)} = \begin{bmatrix} 2/3 \\ -2/3 \end{bmatrix} \quad \text{and} \quad \Delta \boldsymbol{g}^{(0)} = \boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)} = \begin{bmatrix} 2/3 \\ -4/3 \end{bmatrix}.$$

Using the above, we determine,

$$\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)} = \begin{bmatrix} 0 \\ 2/3 \end{bmatrix} \quad \text{and} \quad \Delta \boldsymbol{g}^{(0)\top}\left(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}\right) = -\frac{8}{9}.$$

Then, we obtain

$$\begin{aligned}\boldsymbol{H}_1 &= \boldsymbol{H}_0 + \frac{\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)^\top}{\Delta\boldsymbol{g}^{(0)\top}\left(\Delta\boldsymbol{x}^{(0)} - \boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}\right)} \\ &= \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + \frac{\begin{bmatrix}0 & 0\\ 0 & 4/9\end{bmatrix}}{-8/9} \\ &= \begin{bmatrix}1 & 0\\ 0 & 1/2\end{bmatrix}\end{aligned}$$

and

$$\boldsymbol{d}^{(1)} = -\boldsymbol{H}_1\boldsymbol{g}^{(1)} = \begin{bmatrix}1/3\\ 1/6\end{bmatrix}.$$

Note that $\boldsymbol{d}^{(0)\top}\boldsymbol{Q}\boldsymbol{d}^{(1)} = 0$, that is, $\boldsymbol{d}^{(0)}$ and $\boldsymbol{d}^{(1)}$ are $\boldsymbol{Q}$-conjugate.

11.10

The calculations are similar until we get to the second step:

$$\begin{aligned}H_1 &= \begin{bmatrix}1/2 & -1/2\\ -1/2 & 1/2\end{bmatrix} \\ \boldsymbol{d}^{(0)} &= \mathbf{0}.\end{aligned}$$

So the algorithm gets stuck at this point, which illustrates that it doesn't work.

11.11

a. Since f is quadratic, and $\alpha_k = \arg\min_{\alpha\geq 0} f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{d}^{(k)})$, then

$$\alpha_k = -\frac{\boldsymbol{g}^{(k)\top}\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}}.$$

b. Now, $\boldsymbol{d}^{(k)} = -\boldsymbol{H}_k\boldsymbol{g}^{(k)}$, where $\boldsymbol{H}_k = \boldsymbol{H}_k^\top > 0$. Substituting this into the formula for α_k in part a, yields

$$\alpha_k = \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\boldsymbol{g}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}} > 0.$$

11.12

(Our solution to this problem is based on a solution that was furnished to us by Michael Mera, a student in ECE 580 at Purdue in Spring 2005.) To proceed, we recall the formula of Lemma 11.1,

$$(\boldsymbol{A} + \boldsymbol{u}\boldsymbol{v}^\top)^{-1} = \boldsymbol{A}^{-1} - \frac{(\boldsymbol{A}^{-1}\boldsymbol{u})(\boldsymbol{v}^\top\boldsymbol{A}^{-1})}{1 + \boldsymbol{v}^\top\boldsymbol{A}^{-1}\boldsymbol{u}}$$

for $1 + \boldsymbol{v}^\top\boldsymbol{A}^{-1}\boldsymbol{u} \neq 0$. Recall the definitions from the hint,

$$\boldsymbol{A}_0 = \boldsymbol{B}_k,\ \ \boldsymbol{u}_0 = \frac{\Delta\boldsymbol{g}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}},\ \ \boldsymbol{v}_0^\top = \Delta\boldsymbol{g}^{(k)\top},$$

and

$$\boldsymbol{A}_1 = \boldsymbol{B}_k + \frac{\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}} = \boldsymbol{A}_0 + \boldsymbol{u}_0\boldsymbol{v}_0^\top,\ \ \boldsymbol{u}_1 = -\frac{\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}},$$

and

$$\boldsymbol{v}_1^\top = \Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k.$$

Using the above notation, we represent $\boldsymbol{B}_{k+1}$ as

$$\begin{aligned}\boldsymbol{B}_{k+1} &= \boldsymbol{A}_0 + \boldsymbol{u}_0\boldsymbol{v}_0^\top + \boldsymbol{u}_1\boldsymbol{v}_1^\top \\ &= \boldsymbol{A}_1 + \boldsymbol{u}_1\boldsymbol{v}_1^\top.\end{aligned}$$

Applying to the above Lemma 11.1 gives

$$\begin{aligned}\boldsymbol{H}_{k+1}^{BFGS} &= (\boldsymbol{B}_{k+1})^{-1} \\ &= (\boldsymbol{A}_1 + \boldsymbol{u}_1\boldsymbol{v}_1^\top)^{-1} \\ &= \boldsymbol{A}_1^{-1} - \frac{\boldsymbol{A}_1^{-1}\boldsymbol{u}_1\boldsymbol{v}_1^\top\boldsymbol{A}_1^{-1}}{1+\boldsymbol{v}_1^\top\boldsymbol{A}_1^{-1}\boldsymbol{u}_1}.\end{aligned}$$

Substituting into the above the expression for $\boldsymbol{A}_1^{-1}$ yields

$$\boldsymbol{H}_{k+1}^{BFGS} = \boldsymbol{A}_0^{-1} - \frac{\boldsymbol{A}_0^{-1}\boldsymbol{u}_0\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}}{1+\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}\boldsymbol{u}_0} - \frac{\left(\boldsymbol{A}_0^{-1} - \frac{\boldsymbol{A}_0^{-1}\boldsymbol{u}_0\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}}{1+\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}\boldsymbol{u}_0}\right)\boldsymbol{u}_1\boldsymbol{v}_1^\top\left(\boldsymbol{A}_0^{-1} - \frac{\boldsymbol{A}_0^{-1}\boldsymbol{u}_0\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}}{1+\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}\boldsymbol{u}_0}\right)}{1+\boldsymbol{v}_1^\top\left(\boldsymbol{A}_0^{-1} - \frac{\boldsymbol{A}_0^{-1}\boldsymbol{u}_0\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}}{1+\boldsymbol{v}_0^\top\boldsymbol{A}_0^{-1}\boldsymbol{u}_0}\right)\boldsymbol{u}_1}.$$

Note that $\boldsymbol{A}_0 = \boldsymbol{B}_k$. Hence, $\boldsymbol{A}_0^{-1} = \boldsymbol{B}_k^{-1} = \boldsymbol{H}_k$. Using this and the notation introduced at the beginning of the solution, we obtain

$$\begin{aligned}\boldsymbol{H}_{k+1}^{BFGS} &= \boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)} + \Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}} \\ &- \frac{\left(\boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)\left(\frac{-\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k}{\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}\right)}{1+\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\left(\boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)\left(\frac{-\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}\right)} \\ &\times\left(\boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)} + \Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right).\end{aligned}$$

We next perform some multiplications taking into account that $\boldsymbol{H}_k = \boldsymbol{B}_k^{-1}$ and hence

$$\boldsymbol{H}_k\boldsymbol{B}_k = \boldsymbol{B}_k\boldsymbol{H}_k = \boldsymbol{I}_n.$$

We obtain

$$\begin{aligned}\boldsymbol{H}_{k+1}^{BFGS} &= \boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)} + \Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}} \\ &- \frac{\left(1 - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)(-\Delta\boldsymbol{x}^{(k)}\Delta\boldsymbol{x}^{(k)\top})\left(1 - \frac{\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)}{\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)} + \Delta\boldsymbol{x}^{(k)\top}\left(\boldsymbol{B}_k - \frac{\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)(-\Delta\boldsymbol{x}^{(k)})}.\end{aligned}$$

We proceed with our manipulations. We first perform multiplications by $\Delta\boldsymbol{x}^{(k)}$ and $\Delta\boldsymbol{x}^{(k)\top}$ to obtain

$$\begin{aligned}\boldsymbol{H}_{k+1}^{BFGS} &= \boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)} + \Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}} \\ &- \frac{\left(\frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}} - \Delta\boldsymbol{x}^{(k)}\right)\left(\Delta\boldsymbol{x}^{(k)\top} - \frac{\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}\right)}{\Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)} - \Delta\boldsymbol{x}^{(k)\top}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)} + \frac{\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}+\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}}.\end{aligned}$$

Cancelling the terms in the denominator of the last term above and performing further multiplications gives

$$\begin{aligned} H_{k+1}^{BFGS} = \boldsymbol{H}_k &- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}} \\ &+ \frac{\frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right) \left(\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}\right) \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}}}{\left(\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}\right) \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right)} \\ &+ \frac{\Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{x}^{(k)\top} \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}\right)}{\left(\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}\right) \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right)} \\ &- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right) \Delta \boldsymbol{x}^{(k)\top} + \Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\left(\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}\right) \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right)}. \end{aligned}$$

Further simplification of the third and the fifth terms on the right hand-side of the above equation gives

$$\begin{aligned} H_{k+1}^{BFGS} = \boldsymbol{H}_k &- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}} \\ &+ \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}} \\ &+ \frac{\Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{x}^{(k)\top} \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)} + \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}\right)}{\left(\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}\right) \left(\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}\right)} \\ &- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{x}^{(k)\top} + \Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}}. \end{aligned}$$

Note that the second and the third terms cancel out each other. We then represent the fourth term in alternative manner to obtain

$$\begin{aligned} \boldsymbol{H}_{k+1}^{BFGS} = \boldsymbol{H}_k &+ \frac{\Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{x}^{(k)\top}}{\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}} \left(1 + \frac{\Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}}{\Delta \boldsymbol{g}^{(k)\top} \Delta \boldsymbol{x}^{(k)}}\right) \\ &- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{x}^{(k)\top} + \Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{g}^{(k)\top} \boldsymbol{H}_k}{\Delta \boldsymbol{x}^{(k)\top} \Delta \boldsymbol{g}^{(k)}}, \end{aligned}$$

which is the desired BFGS update formula.

11.13

The first step for both algorithms is clearly the same, since in either case we have

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} - \alpha_0 \boldsymbol{g}^{(0)}.$$

For the second step,

$$\begin{aligned} \boldsymbol{d}^{(1)} &= -\boldsymbol{H}_1 \boldsymbol{g}^{(1)} \\ &= -\left(\boldsymbol{I}_n + \left(1 + \frac{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{g}^{(0)}}{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{x}^{(0)}}\right) \frac{\Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{x}^{(0)\top}}{\Delta \boldsymbol{x}^{(0)\top} \Delta \boldsymbol{g}^{(0)}} \right. \\ &\qquad \left. - \frac{\Delta \boldsymbol{g}^{(0)} \Delta \boldsymbol{x}^{(0)\top} + (\Delta \boldsymbol{g}^{(0)} \Delta \boldsymbol{x}^{(0)\top})^\top}{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{x}^{(0)}}\right) \boldsymbol{g}^{(1)} \\ &= -\boldsymbol{g}^{(1)} - \left(1 + \frac{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{g}^{(0)}}{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{x}^{(0)}}\right) \frac{\Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{x}^{(0)\top} \boldsymbol{g}^{(1)}}{\Delta \boldsymbol{x}^{(0)\top} \Delta \boldsymbol{g}^{(0)}} \\ &\qquad + \frac{\Delta \boldsymbol{g}^{(0)} \Delta \boldsymbol{x}^{(0)\top} \boldsymbol{g}^{(1)} + \Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{g}^{(0)\top} \boldsymbol{g}^{(1)}}{\Delta \boldsymbol{g}^{(0)\top} \Delta \boldsymbol{x}^{(0)}}. \end{aligned}$$

Since the line search is exact, we have

$$\Delta \boldsymbol{x}^{(0)\top} \boldsymbol{g}^{(1)} = \alpha_0 \boldsymbol{d}^{(0)\top} \boldsymbol{g}^{(1)} = 0.$$

Hence,

$$\begin{aligned}\boldsymbol{d}^{(1)} &= -\boldsymbol{g}^{(1)} + \left(\frac{\Delta\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(1)}}{\Delta\boldsymbol{g}^{(0)\top}\Delta\boldsymbol{x}^{(0)}}\right)\Delta\boldsymbol{x}^{(0)} \\ &= -\boldsymbol{g}^{(1)} + \left(\frac{\boldsymbol{g}^{(1)\top}\Delta\boldsymbol{g}^{(0)}}{\Delta\boldsymbol{g}^{(0)\top}\boldsymbol{d}^{(0)}}\right)\boldsymbol{d}^{(0)} \\ &= -\boldsymbol{g}^{(1)} + \beta_0\boldsymbol{d}^{(0)}\end{aligned}$$

where

$$\beta_0 = \frac{\boldsymbol{g}^{(1)\top}\Delta\boldsymbol{g}^{(0)}}{\boldsymbol{d}^{(0)\top}\Delta\boldsymbol{g}^{(0)}} = \frac{\boldsymbol{g}^{(1)\top}(\boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)})}{\boldsymbol{d}^{(0)\top}(\boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)})}$$

is the Hestenes-Stiefel update formula for β_0. Since $\boldsymbol{d}^{(0)} = -\boldsymbol{g}^{(0)}$, and $\boldsymbol{g}^{(1)\top}\boldsymbol{g}^{(0)} = 0$, we have

$$\beta_0 = \frac{\boldsymbol{g}^{(1)\top}(\boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)})}{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}},$$

which is the Polak-Ribiere formula. Applying $\boldsymbol{g}^{(1)\top}\boldsymbol{g}^{(0)} = 0$ again, we get

$$\beta_0 = \frac{\boldsymbol{g}^{(1)\top}\boldsymbol{g}^{(1)}}{\boldsymbol{g}^{(0)\top}\boldsymbol{g}^{(0)}},$$

which is the Fletcher-Reeves formula.

11.14

a. Suppose the three conditions hold whenever applied to a quadratic. We need to show that when applied to a quadratic, for $k = 0, \ldots, n-1$ and $i = 0, \ldots, k$, $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}$. For $i = k$, we have

$$\begin{aligned}\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(k)} &= \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)} + \boldsymbol{U}_k\Delta\boldsymbol{g}^{(k)} \quad \text{by condition 1} \\ &= \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)} + \Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)} \quad \text{by condition 2} \\ &= \Delta\boldsymbol{x}^{(k)},\end{aligned}$$

as required. For the rest of the proof ($i = 0, \ldots, k-1$), we use induction on k.

For $k = 0$, there is nothing to prove (covered by the $i = k$ case). So suppose the result holds for $k-1$. To show the result for k, first fix $i \in \{0, \ldots, k-1\}$. We have

$$\begin{aligned}\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} &= \boldsymbol{H}_k\Delta\boldsymbol{g}^{(i)} + \boldsymbol{U}_k\Delta\boldsymbol{g}^{(i)} \\ &= \Delta\boldsymbol{x}^{(i)} + \boldsymbol{U}_k\Delta\boldsymbol{g}^{(i)} \quad \text{by the induction hypothesis} \\ &= \Delta\boldsymbol{x}^{(i)} + \boldsymbol{a}^{(k)}\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(i)} + \boldsymbol{b}^{(k)}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(i)} \quad \text{by condition 3.}\end{aligned}$$

So it suffices to show that the second and third terms are both 0. For the second term,

$$\begin{aligned}\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(i)} &= \Delta\boldsymbol{x}^{(k)\top}\boldsymbol{Q}\Delta\boldsymbol{x}^{(i)} \\ &= \alpha_k\alpha_i\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} \\ &= 0\end{aligned}$$

because of the induction hypothesis, which implies $\boldsymbol{Q}$-conjugacy (where $\boldsymbol{Q}$ is the Hessian of the given quadratic). Similarly, for the third term,

$$\begin{aligned}\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(i)} &= \Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(i)} \quad \text{by the induction hypothesis} \\ &= \Delta\boldsymbol{x}^{(k)\top}\boldsymbol{Q}\Delta\boldsymbol{x}^{(i)} \\ &= \alpha_k\alpha_i\boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} \\ &= 0,\end{aligned}$$

again because of the induction hypothesis, which implies $\boldsymbol{Q}$-conjugacy. This completes the proof.

b. All three algorithms satisfy the conditions in part a. Condition 1 holds, as described in class. Condition 2 is straightforward to check for all three algorithms. For the rank-one and DFP algorithms, this is shown in the book. For BFGS, some simple matrix algebra establishes that it holds. Condition 3 holds by appropriate definition of the vectors $\boldsymbol{a}^{(k)}$ and $\boldsymbol{b}^{(k)}$. In particular, for the rank-one algorithm,

$$\boldsymbol{a}^{(k)} = \frac{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})}{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})^\top\Delta\boldsymbol{g}^{(k)}}, \qquad \boldsymbol{b}^{(k)} = -\frac{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})}{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})^\top\Delta\boldsymbol{g}^{(k)}}.$$

For the DFP algorithm,

$$\boldsymbol{a}^{(k)} = \frac{\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(k)}}, \qquad \boldsymbol{b}^{(k)} = -\frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}.$$

Finally, for the BFGS algorithm,

$$\boldsymbol{a}^{(k)} = \left(1 + \frac{\Delta\boldsymbol{g}^{(k)\top}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}}\right)\frac{\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{x}^{(k)\top}\Delta\boldsymbol{g}^{(k)}} - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}}, \qquad \boldsymbol{b}^{(k)} = \frac{\Delta\boldsymbol{x}^{(k)}}{\Delta\boldsymbol{g}^{(k)\top}\Delta\boldsymbol{x}^{(k)}}.$$

11.15

a. Suppose we apply the algorithm to a quadratic. Then, by the quasi-Newton property of DFP, we have $\boldsymbol{H}_{k+1}^{DFP}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}$, $0 \le i \le k$. The same holds for BFGS. Thus, for the given $\boldsymbol{H}_k$, we have for $0 \le i \le k$,

$$\begin{aligned}\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} &= \phi\boldsymbol{H}_{k+1}^{DFP}\Delta\boldsymbol{g}^{(i)} + (1-\phi)\boldsymbol{H}_{k+1}^{BFGS}\Delta\boldsymbol{g}^{(i)} \\ &= \phi\Delta\boldsymbol{x}^{(i)} + (1-\phi)\Delta\boldsymbol{x}^{(i)} \\ &= \Delta\boldsymbol{x}^{(i)},\end{aligned}$$

which shows that the above algorithm is a quasi-Newton algorithm (and hence also a conjugate direction algorithm).

b. By Theorem 11.4 and the discussion on BFGS, we have $\boldsymbol{H}_k^{DFP} > 0$ and $\boldsymbol{H}_k^{BFGS} > 0$. Hence, for any $\boldsymbol{x} \ne \boldsymbol{0}$,

$$\boldsymbol{x}^\top\boldsymbol{H}_k\boldsymbol{x} = \phi\boldsymbol{x}^\top\boldsymbol{H}_k^{DFP}\boldsymbol{x} + (1-\phi)\boldsymbol{x}^\top\boldsymbol{H}_k^{BFGS}\boldsymbol{x} > 0$$

since ϕ and $1-\phi$ are nonnegative. Hence, $\boldsymbol{H}_k > 0$, from which we conclude that the algorithm has the descent property if α_k is computed by line search (by Proposition 11.1).

11.16

To show the result, we will prove the following precise statement: In the quadratic case (with Hessian $\boldsymbol{Q}$), suppose that $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \rho_i\Delta\boldsymbol{x}^{(i)}$, $0 \le i \le k$, $k \le n-1$. If $\alpha_i \ne 0$, $0 \le i \le k$, then $\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k+1)}$ are $\boldsymbol{Q}$-conjugate.

We proceed by induction. We begin with the $k = 0$ case: that $\boldsymbol{d}^{(0)}$ and $\boldsymbol{d}^{(1)}$ are $\boldsymbol{Q}$-conjugate. Because $\alpha_0 \ne 0$, we can write $\boldsymbol{d}^{(0)} = \Delta\boldsymbol{x}^{(0)}/\alpha_0$. Hence,

$$\begin{aligned}\boldsymbol{d}^{(1)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)} &= -\boldsymbol{g}^{(1)\top}\boldsymbol{H}_1\boldsymbol{Q}\boldsymbol{d}^{(0)} \\ &= -\boldsymbol{g}^{(1)\top}\boldsymbol{H}_1\frac{\boldsymbol{Q}\Delta\boldsymbol{x}^{(0)}}{\alpha_0} \\ &= -\boldsymbol{g}^{(1)\top}\frac{\boldsymbol{H}_1\Delta\boldsymbol{g}^{(0)}}{\alpha_0} \\ &= -\boldsymbol{g}^{(1)\top}\frac{\rho_0\Delta\boldsymbol{x}^{(0)}}{\alpha_0} \\ &= -\rho_0\boldsymbol{g}^{(1)\top}\boldsymbol{d}^{(0)}.\end{aligned}$$

But $\boldsymbol{g}^{(1)\top}\boldsymbol{d}^{(0)} = 0$ as a consequence of $\alpha_0 > 0$ being the minimizer of $\phi(\alpha) = f(\boldsymbol{x}^{(0)} + \alpha\boldsymbol{d}^{(0)})$. Hence, $\boldsymbol{d}^{(1)\top}\boldsymbol{Q}\boldsymbol{d}^{(0)} = 0$.

Assume that the result is true for $k-1$ (where $k < n-1$). We now prove the result for k, that is, that $\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k+1)}$ are $\boldsymbol{Q}$-conjugate. It suffices to show that $\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} = 0$, $0 \le i \le k$. Given i, $0 \le i \le k$, using the same algebraic steps as in the $k = 0$ case, and using the assumption that $\alpha_i \neq 0$, we obtain

$$\begin{aligned}\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} &= -\boldsymbol{g}^{(k+1)\top}\boldsymbol{H}_{k+1}\boldsymbol{Q}\boldsymbol{d}^{(i)} \\ &\vdots \\ &= -\rho_i \boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(i)}.\end{aligned}$$

Because $\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k)}$ are $\boldsymbol{Q}$-conjugate by assumption, we conclude from the expanding subspace lemma (Lemma 10.2) that $\boldsymbol{g}^{(k+1)\top}\boldsymbol{d}^{(i)} = 0$. Hence, $\boldsymbol{d}^{(k+1)\top}\boldsymbol{Q}\boldsymbol{d}^{(i)} = 0$, which completes the proof.

11.17

A MATLAB routine for the quasi-Newton algorithm with options for different formulas of $\boldsymbol{H}_k$ is:

```
function [x,N]=quasi_newton(grad,xnew,H,options);
%    QUASI_NEWTON('grad',x0,H0);
%    QUASI_NEWTON('grad',x0,H0,OPTIONS);
%
%    x = QUASI_NEWTON('grad',x0,H0);
%    x = QUASI_NEWTON('grad',x0,H0,OPTIONS);
%
%    [x,N] = QUASI_NEWTON('grad',x0,H0);
%    [x,N] = QUASI_NEWTON('grad',x0,H0,OPTIONS);
%
%The first variant finds the minimizer of a function whose gradient
%is described in grad (usually an M-file: grad.m), using initial point
%x0 and initial inverse Hessian approximation H0.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results, (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required of the gradient.
%OPTIONS(5) specifies the formula for the inverse Hessian update:
%   0=Rank One;
%   1=DFP;
%   2=BFGS;
%OPTIONS(14) is the maximum number of iterations.
%For more information type HELP FOPTIONS.
%
%The next two variants return the value of the final point.
%The last two variants return a vector of the final point and the
%number of iterations.

if nargin ~= 4
  options = [];
  if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

numvars = length(xnew);
if length(options) >= 14
  if options(14)==0
    options(14)=1000*numvars;
  end
else
```

```
  options(14)=1000*numvars;
end

clc;
format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_g = options(3);
max_iter=options(14);

reset_cnt = 0;
g_curr=feval(grad,xnew);
if norm(g_curr) <= epsilon_g
  disp('Terminating: Norm of initial gradient less than');
  disp(epsilon_g);
  return;
end %if

d=-H*g_curr;
for k = 1:max_iter,

  xcurr=xnew;
  alpha=linesearch_secant(grad,xcurr,d);
  xnew = xcurr+alpha*d;

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('Gradient = ');
    disp(g_curr'); %print gradient
    disp('New point =');
    disp(xnew'); %print new point
  end %if

  if norm(xnew-xcurr) <= epsilon_x*norm(xcurr)
    disp('Terminating: Norm of difference between iterates less than');
    disp(epsilon_x);
    break;
  end %if

  g_old=g_curr;
  g_curr=feval(grad,xnew);

  if norm(g_curr) <= epsilon_g
    disp('Terminating: Norm of gradient less than');
    disp(epsilon_g);
    break;
  end %if

  p=alpha*d;
  q=g_curr-g_old;

  reset_cnt = reset_cnt+1;
  if reset_cnt == 3*numvars
```

```
    d=-g_curr;
    reset_cnt = 0;
  else
    if options(5)==0 %Rank One
      q'*(p-H*q)
      H = H+(p-H*q)*(p-H*q)'/(q'*(p-H*q));
    elseif options(5)==1 %DFP
      H = H+p*p'/(p'*q)-(H*q)*(H*q)'/(q'*H*q);
    else %BFGS
      H = H+(1+q'*H*q/(q'*p))*p*p'/(p'*q)-(H*q*p'+(H*q*p')')/(q'*p);
    end %if
    d=-H*g_curr;
  end

  if print,
    disp('New H =');
    disp(H);
    disp('New d =');
    disp(d);
  end

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xnew');
  disp('Number of iterations =');
  disp(k);
end %if
%-------------------------------------------------------------------
```

We created the following M-file, `g.m`, for the gradient of Rosenbrock's function:

```
function y=g(x)
y=[-400*(x(2)-x(1).^2)*x(1)-2*(1-x(1)), 200*(x(2)-x(1).^2)]';
```

We tested the above routine as follows:

```
>> options(2)=10^(-7);
>> options(3)=10^(-7);
>> options(14)=100;
>> x0=[-2;2];
>> H0=eye(2);
>> options(5)=0;
>> quasi_newton('g',x0,H0,options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final point =
   1.0000e+00   1.0000e+00
Number of iterations =
     8
```

```
>> options(5)=1;
>> quasi_newton('g',x0,H0,options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final point =
   1.0000e+00   1.0000e+00
Number of iterations =
     8
>> options(5)=2;
>> quasi_newton('g',x0,H0,options);

Terminating: Norm of difference between iterates less than
   1.0000e-07
Final point =
   1.0000e+00   1.0000e+00
Number of iterations =
     8
```

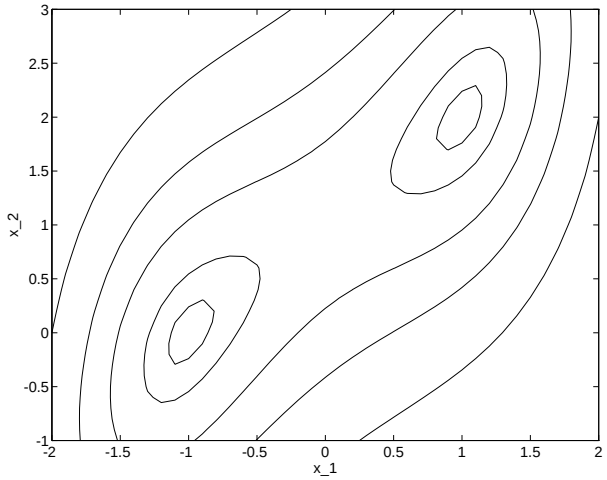
The reader is again cautioned *not* to draw any conclusions about the superiority or inferiority of any of the formulas for $\boldsymbol{H}_k$ based only on the above single numerical experiment.

11.18

a. The plot of the level sets of f were obtained using the following MATLAB commands:

```
>> [X,Y]=meshdom(-2:0.1:2, -1:0.1:3);
>> Z=X.^4/4+Y.^2/2 -X.*Y +X-Y;
>> V=[-0.72, -0.6, -0.2, 0.5, 2];
>> contour(X,Y,Z,V)
```

The plot is depicted below:



b. With the initial condition $[0,0]^\top$, the algorithm converges to $[-1,0]^\top$, while with the initial condition $[1.5,1]^\top$, the algorithm converges to $[1,2]^\top$. These two points are the two strict local minimizers of f (as can be checked using the SOSC). The algorithm apparently converges to the minimizer "closer" to the initial point.

12. Solving $\boldsymbol{Ax} = \boldsymbol{b}$

12.1

Write the least squares cost in the usual notation $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|^2$ where

$$\boldsymbol{A}=\begin{bmatrix}3\\5\\6\end{bmatrix},\qquad \boldsymbol{x}=\begin{bmatrix}m\end{bmatrix},\qquad \boldsymbol{b}=\begin{bmatrix}1\\2\\3\end{bmatrix}.$$

The least squares estimate of the mass is

$$m^*=(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}=\frac{31}{70}.$$

12.2

Write the least squares cost in the usual notation $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|^2$ where

$$\boldsymbol{A}=\begin{bmatrix}1&1\\1&2\\1&4\end{bmatrix},\qquad \boldsymbol{x}=\begin{bmatrix}a\\b\end{bmatrix},\qquad \boldsymbol{b}=\begin{bmatrix}3\\4\\5\end{bmatrix}.$$

The least squares estimate for $[a,b]^\top$ is

$$\begin{aligned}\begin{bmatrix}a^*\\b^*\end{bmatrix}&=(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}\\&=\begin{bmatrix}3&7\\7&21\end{bmatrix}^{-1}\begin{bmatrix}12\\31\end{bmatrix}\\&=\frac{1}{14}\begin{bmatrix}21&-7\\-7&3\end{bmatrix}^{-1}\begin{bmatrix}12\\31\end{bmatrix}\\&=\frac{1}{14}\begin{bmatrix}35\\9\end{bmatrix}\\&=\begin{bmatrix}5/2\\9/14\end{bmatrix}.\end{aligned}$$

12.3

a. We form

$$\boldsymbol{A}=\begin{bmatrix}1^2/2\\2^2/2\\3^2/2\end{bmatrix},\qquad \boldsymbol{b}=\begin{bmatrix}5.00\\19.5\\44.0\end{bmatrix}.$$

The least squares estimate of g is then given by

$$g=(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}=9.776.$$

b. We start with $\boldsymbol{P}_0=0.040816$, and $\boldsymbol{x}^{(0)}=9.776$. We have $\boldsymbol{a}_1=4^2/2=8$, and $\boldsymbol{b}^{(1)}=78.5$. Using the RLS formula, we get $\boldsymbol{x}^{(1)}=9.802$, which is our updated estimate of g.

12.4

Let $\boldsymbol{x}=[x_1,x_2,\ldots,x_n]^\top$ and $\boldsymbol{y}=[y_1,y_2,\ldots,y_n]^\top$. This least-squares estimation problem can be expressed as

$$\text{minimize}\,\|\alpha\boldsymbol{x}-\boldsymbol{y}\|^2,$$

with α as the decision variable. Assuming that $\boldsymbol{x}\neq\boldsymbol{0}$, the solution is unique and is given by

$$\alpha^*=(\boldsymbol{x}^\top\boldsymbol{x})^{-1}\boldsymbol{x}^\top\boldsymbol{y}=\frac{\boldsymbol{x}^\top\boldsymbol{y}}{\boldsymbol{x}^\top\boldsymbol{x}}=\frac{\sum_{i=1}^n x_iy_i}{\sum_{i=1}^n x_i^2}.$$

12.5

The least squares estimate of R is the least squares solution to

$$\begin{aligned} 1 \cdot R &= V_1 \\ &\vdots \\ 1 \cdot R &= V_n. \end{aligned}$$

Therefore, the least squares solution is

$$R^* = \left([1, \ldots, 1]^\top \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \right)^{-1} [1, \ldots, 1]^\top \begin{bmatrix} V_1 \\ \vdots \\ V_n \end{bmatrix} = \frac{V_1 + \cdots + V_n}{n}.$$

12.6

We represent the data in the table and the decision variables a and b using the usual least squares matrix notation:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 3 & 2 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 6 \\ 4 \\ 5 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} a \\ b \end{bmatrix}.$$

The least squares estimate is given by

$$\boldsymbol{x}^* = \begin{bmatrix} a^* \\ b^* \end{bmatrix} = (\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{b} = \begin{bmatrix} 11 & 9 \\ 9 & 9 \end{bmatrix}^{-1} \begin{bmatrix} 25 \\ 26 \end{bmatrix} = \frac{1}{18} \begin{bmatrix} 9 & -9 \\ -9 & 11 \end{bmatrix} \begin{bmatrix} 25 \\ 26 \end{bmatrix} = \begin{bmatrix} -1/2 \\ 61/18 \end{bmatrix}.$$

12.7

The problem can be formulated as a least-squares problem with

$$\boldsymbol{A} = \begin{bmatrix} 0.3 & 0.1 \\ 0.4 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 5 \\ 3 \\ 4 \end{bmatrix},$$

where the decision variable is $\boldsymbol{x} = [x_1, x_2]$, and x_1 and x_2 are the amounts of A and B, respectively. After some algebra, we obtain the solution:

$$\boldsymbol{x}^* = (\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{b} = \frac{1}{(0.34)(0.54) - (0.32)^2} \begin{bmatrix} 0.54 & -0.32 \\ -0.32 & 0.34 \end{bmatrix} \begin{bmatrix} 3.9 \\ 3.9 \end{bmatrix}.$$

Since we are only interest in the ratio of the first component of $\boldsymbol{x}^*$ to the second component, we need only explicitly compute:

$$\text{Ratio} = \frac{0.54 - 0.32}{-0.32 + 0.34} = \frac{0.22}{0.02} = 11.$$

12.8

For each k, we can write

$$\begin{aligned} y_k &= a y_{k-1} + b u_k + v_k \\ &= a^2 y_{k-2} + ab u_{k-1} + a v_{k-1} + b u_k + v_k \\ &\vdots \\ &= a^{k-1} b u_1 + a^{k-2} b u_2 + \cdots + b u_k + a^{k-1} v_1 + a^{k-2} v_2 + \cdots + v_k \end{aligned}$$

Write $\boldsymbol{u} = [u_1, \ldots, u_n]^\top$, $\boldsymbol{v} = [v_1, \ldots, v_n]^\top$, and $\boldsymbol{y} = [y_1, \ldots, y_n]^\top$. Then, $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{u} + \boldsymbol{D}\boldsymbol{v}$, where

$$\boldsymbol{C} = \begin{bmatrix} b & 0 & \cdots & 0 \\ ab & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a^{n-1}b & \cdots & ab & b \end{bmatrix}, \qquad \boldsymbol{D} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ a & 1 & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ a^{n-1} & a^{n-2} & \cdots & 1 \end{bmatrix}.$$

Write $\boldsymbol{b} = \boldsymbol{D}^{-1}\boldsymbol{y}$ and $\boldsymbol{A} = \boldsymbol{D}^{-1}\boldsymbol{C}$ so that $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{u} + \boldsymbol{v}$. Therefore, the linear least-squares estimate of $\boldsymbol{u}$ given $\boldsymbol{y}$ is

$$\boldsymbol{u}^* = (\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{A}^\top \boldsymbol{b} = (\boldsymbol{C}^\top \boldsymbol{D}^{-\top}\boldsymbol{D}^{-1}\boldsymbol{C})^{-1}\boldsymbol{C}^\top \boldsymbol{D}^{-\top}\boldsymbol{D}^{-1}\boldsymbol{y}.$$

But $\boldsymbol{C} = b\boldsymbol{D}$. Hence,

$$\boldsymbol{u}^* = \frac{1}{b}\boldsymbol{D}^{-1}\boldsymbol{y} = \frac{1}{b}\begin{bmatrix} 1 & 0 & \cdots & 0 \\ -a & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & -a & 1 \end{bmatrix}\boldsymbol{y}.$$

(Notice that $\boldsymbol{D}^{-1}$ has the simple form shown above.)

An alternative solution is first to define $\boldsymbol{z} = [z_1, \ldots, z_n]^\top$ by $z_k = y_k - ay_{k-1}$. Then, we have $\boldsymbol{z} = b\boldsymbol{u} + \boldsymbol{v}$. Therefore, the linear least-squares estimate of $\boldsymbol{u}$ given $\boldsymbol{y}$ (or, equivalently, $\boldsymbol{z}$) is

$$\boldsymbol{u}^* = \frac{1}{b}\boldsymbol{z} = \frac{1}{b}\begin{bmatrix} 1 & 0 & \cdots & 0 \\ -a & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & -a & 1 \end{bmatrix}\boldsymbol{y}.$$

12.9

Define

$$\boldsymbol{X} = \begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_p & 1 \end{bmatrix}, \qquad \boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix}.$$

Since the x_i are not all equal, we have $\operatorname{rank} \boldsymbol{X} = 2$. The objective function can be written as

$$f(a, b) = \left\| \boldsymbol{X}\begin{bmatrix} a \\ b \end{bmatrix} - \boldsymbol{y} \right\|^2.$$

Therefore, by Theorem 12.1 there exists a unique minimizer $[a^*, b^*]^\top$ given by

$$\begin{aligned} \begin{bmatrix} a^* \\ b^* \end{bmatrix} &= (\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{y} \\ &= \begin{bmatrix} \sum_{i=1}^p x_i^2 & \sum_{i=1}^p x_i \\ \sum_{i=1}^p x_i & p \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^p x_i y_i \\ \sum_{i=1}^p y_i \end{bmatrix} \\ &= \begin{bmatrix} \overline{X^2} & \overline{X} \\ \overline{X} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \overline{XY} \\ \overline{Y} \end{bmatrix} \\ &= \frac{1}{\overline{X^2} - (\overline{X})^2} \begin{bmatrix} 1 & -\overline{X} \\ -\overline{X} & \overline{X^2} \end{bmatrix} \begin{bmatrix} \overline{XY} \\ \overline{Y} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\overline{XY} - (\overline{X})(\overline{Y})}{\overline{X^2} - (\overline{X})^2} \\ \frac{(\overline{X^2})(\overline{Y}) - (\overline{X})(\overline{XY})}{\overline{X^2} - (\overline{X})^2} \end{bmatrix}. \end{aligned}$$

As we can see, the solution does not depend on $\overline{Y^2}$.

12.10

a. We wish to find ω and θ such that

$$\begin{aligned} \sin(\omega t_1 + \theta) &= y_1 \\ &\vdots \\ \sin(\omega t_p + \theta) &= y_p. \end{aligned}$$

Taking arcsin, we get the following system of linear equations:

$$\begin{aligned} \omega t_1 + \theta &= \arcsin y_1 \\ &\vdots \\ \omega t_p + \theta &= \arcsin y_p. \end{aligned}$$

b. We may write the system of linear equations in part a as $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where

$$\boldsymbol{A} = \begin{bmatrix} t_1 & 1 \\ \vdots & \vdots \\ t_p & 1 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} \omega \\ \theta \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} \arcsin y_1 \\ \vdots \\ \arcsin y_p \end{bmatrix}.$$

Since the t_i are not all equal, the first column of $\boldsymbol{A}$ is not a scalar multiple of the second column. Therefore, $\operatorname{rank} \boldsymbol{A} = 2$. Hence, the least squares solution is

$$\begin{aligned} \boldsymbol{x} &= (\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{b} \\ &= \begin{bmatrix} \sum_{i=1}^p t_i^2 & \sum_{i=1}^p t_i \\ \sum_{i=1}^p t_i & p \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^p t_i \arcsin y_i \\ \sum_{i=1}^p \arcsin y_i \end{bmatrix} \\ &= \begin{bmatrix} \overline{T^2} & \overline{T} \\ \overline{T} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \overline{TY} \\ \overline{Y} \end{bmatrix} \\ &= \frac{1}{\overline{T^2} - (\overline{T})^2} \begin{bmatrix} 1 & -\overline{T} \\ -\overline{T} & \overline{T^2} \end{bmatrix} \begin{bmatrix} \overline{TY} \\ \overline{Y} \end{bmatrix} \\ &= \frac{1}{\overline{T^2} - (\overline{T})^2} \begin{bmatrix} \overline{TY} - (\overline{T})(\overline{Y}) \\ -(\overline{T})(\overline{TY}) + (\overline{T^2})(\overline{Y}) \end{bmatrix}. \end{aligned}$$

12.11

The given line can be expressed as the range of the matrix $\boldsymbol{A} = [1, m]^\top$. Let $\boldsymbol{b} = [x_0, y_0]^\top$ be the given point. Therefore, the problem is a linear least squares problem of minimizing $\|\boldsymbol{A}x - \boldsymbol{b}\|^2$. The solution is given by

$$x^* = (\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{b} = \frac{x_0 + m y_0}{1 + m^2}.$$

Therefore, the point on the straight line that is closest to the given point $[x_0, y_0]$ is given by $[x^*, mx^*]^\top$.

12.12

a. Write

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{x}_1^\top & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_p^\top & 1 \end{bmatrix} \in \mathbb{R}^{p \times (n+1)}, \qquad \boldsymbol{z} = \begin{bmatrix} \boldsymbol{a} \\ c \end{bmatrix} \in \mathbb{R}^{n+1}, \qquad \boldsymbol{b} = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} \in \mathbb{R}^p.$$

The objective function can then be written as $\|\boldsymbol{A}\boldsymbol{z} - \boldsymbol{b}\|^2$.

b. Let $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p]^\top \in \mathbb{R}^{p\times n}$, and $\boldsymbol{e} = [1, \ldots, 1]^\top \in \mathbb{R}^p$. Then we may write $\boldsymbol{A} = [\boldsymbol{X}\ \boldsymbol{e}]$. The solution to the problem is $(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}$. But

$$\boldsymbol{A}^\top\boldsymbol{A} = \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{X}^\top\boldsymbol{e} \\ \boldsymbol{e}^\top\boldsymbol{X} & p \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{0} \\ \boldsymbol{0}^\top & p \end{bmatrix}$$

since $\boldsymbol{X}^\top\boldsymbol{e} = \boldsymbol{x}_1 + \cdots + \boldsymbol{x}_p = \boldsymbol{0}$ by assumption. Also,

$$\boldsymbol{A}^\top\boldsymbol{y} = \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{y} \\ \boldsymbol{e}^\top\boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{e}^\top\boldsymbol{y} \end{bmatrix}$$

since $\boldsymbol{X}^\top\boldsymbol{y} = y_1\boldsymbol{x}_1 + \cdots + y_p\boldsymbol{x}_p = \boldsymbol{0}$ by assumption. Therefore, the solution is given by

$$\boldsymbol{z}^* = (\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b} = \begin{bmatrix} (\boldsymbol{X}^\top\boldsymbol{X})^{-1} & \boldsymbol{0} \\ \boldsymbol{0}^\top & 1/p \end{bmatrix} \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{e}^\top\boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \frac{1}{p}\boldsymbol{e}^\top\boldsymbol{y} \end{bmatrix}.$$

The affine function of best fit is the constant function $f(\boldsymbol{x}) = c$, were

$$c = \frac{1}{p}\sum_{i=1}^{p} y_i.$$

12.13

a. Using the least squares formula, we have

$$\hat{\theta}_n = \left([u_1, \ldots, u_n] \begin{bmatrix} u_1 \\ \cdots \\ u_n \end{bmatrix} \right)^{-1} [u_1, \ldots, u_n] \begin{bmatrix} y_1 \\ \cdots \\ y_n \end{bmatrix} = \frac{\sum_{k=1}^n u_k y_k}{\sum_{k=1}^n u_k^2}.$$

b. Given $u_k = 1$ for all k, we have

$$\hat{\theta}_n = \frac{1}{n}\sum_{k=1}^{n} y_k = \frac{1}{n}\sum_{k=1}^{n}(\theta + e_k) = \theta + \frac{1}{n}\sum_{k=1}^{n} e_k.$$

Hence, $\hat{\theta}_n \to \theta$ if and only if $\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^n e_k = 0$.

12.14

We pose the problem as a least squares problem: minimize $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ where $\boldsymbol{x} = [a, b]^\top$, and

$$\boldsymbol{A} = \begin{bmatrix} x_0 & 1 \\ x_1 & 1 \\ x_2 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

We have

$$\boldsymbol{A}^\top\boldsymbol{A} = \begin{bmatrix} \sum_{i=0}^2 x_i^2 & \sum_{i=0}^2 x_i \\ \sum_{i=0}^2 x_i & 3 \end{bmatrix}, \qquad \boldsymbol{A}^\top\boldsymbol{b} = \begin{bmatrix} \sum_{i=0}^2 x_i x_{i+1} \\ \sum_{i=0}^2 x_{i+1} \end{bmatrix}.$$

Therefore, the least squares solution is

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^2 x_i^2 & \sum_{i=0}^2 x_i \\ \sum_{i=0}^2 x_i & 3 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=0}^2 x_i x_{i+1} \\ \sum_{i=0}^2 x_{i+1} \end{bmatrix} = \begin{bmatrix} 5 & 3 \\ 3 & 3 \end{bmatrix}^{-1} \begin{bmatrix} 18 \\ 11 \end{bmatrix} = \begin{bmatrix} 7/2 \\ 1/6 \end{bmatrix}.$$

12.15

We pose the problem as a least squares problem: minimize $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ where $\boldsymbol{x} = [a, b]^\top$, and

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ h_1 & 0 \\ \vdots & \vdots \\ h_{n-1} & 0 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix}$$

(note that $h_0 = 0$). We have

$$\boldsymbol{A}^\top \boldsymbol{A} = \begin{bmatrix} \sum_{i=1}^{n-1} h_i^2 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \boldsymbol{A}^\top \boldsymbol{b} = \begin{bmatrix} \sum_{i=1}^{n-1} h_i h_{i+1} \\ h_1 \end{bmatrix}.$$

The matrix $\boldsymbol{A}^\top \boldsymbol{A}$ is nonsingular because we assume that at least one h_k is nonzero. Therefore, the least squares solution is

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n-1} h_i^2 & 0 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^{n-1} h_i h_{i+1} \\ h_1 \end{bmatrix} = \begin{bmatrix} (\sum_{i=1}^{n-1} h_i h_{i+1})/(\sum_{i=1}^{n-1} h_i^2) \\ h_1 \end{bmatrix}.$$

12.16

We pose the problem as a least squares problem: minimize $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ where $\boldsymbol{x} = [a, b]^\top$, and

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ s_1 & 1 \\ \vdots & \vdots \\ s_{n-1} & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix}$$

(where we use $s_0 = 0$). We have

$$\boldsymbol{A}^\top \boldsymbol{A} = \begin{bmatrix} \sum_{i=1}^{n-1} s_i^2 & \sum_{i=1}^{n-1} s_i \\ \sum_{i=1}^{n-1} s_i & n \end{bmatrix}, \qquad \boldsymbol{A}^\top \boldsymbol{b} = \begin{bmatrix} \sum_{i=1}^{n-1} s_i s_{i+1} \\ \sum_{i=1}^{n} s_i \end{bmatrix}.$$

The matrix $\boldsymbol{A}^\top \boldsymbol{A}$ is nonsingular because we assume that at least one s_k is nonzero. Therefore, the least squares solution is

$$\begin{aligned} \begin{bmatrix} a \\ b \end{bmatrix} &= \begin{bmatrix} \sum_{i=1}^{n-1} s_i^2 & \sum_{i=1}^{n-1} s_i \\ \sum_{i=1}^{n-1} s_i & n \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^{n-1} s_i s_{i+1} \\ \sum_{i=1}^{n} s_i \end{bmatrix} \\ &= \frac{1}{n \sum_{i=1}^{n-1} s_i^2 - \left(\sum_{i=1}^{n-1} s_i\right)^2} \begin{bmatrix} n \sum_{i=1}^{n-1} s_i s_{i+1} - \sum_{i=1}^{n-1} s_i \sum_{i=1}^{n} s_i \\ -\sum_{i=1}^{n-1} s_i \sum_{i=1}^{n-1} s_i s_{i+1} + \sum_{i=1}^{n-1} s_i^2 \sum_{i=1}^{n} s_i \end{bmatrix}. \end{aligned}$$

12.17

This least-squares estimation problem can be expressed as

$$\text{minimize } \|a\boldsymbol{x} - \boldsymbol{y}\|^2.$$

If $\boldsymbol{x} = \boldsymbol{0}$, then the problem has an infinite number of solutions: any a solves the problem. Assuming that $\boldsymbol{x} \neq \boldsymbol{0}$, the solution is unique and is given by

$$a^* = (\boldsymbol{x}^\top \boldsymbol{x})^{-1} \boldsymbol{x}^\top \boldsymbol{y} = \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\boldsymbol{x}^\top \boldsymbol{x}}.$$

12.18

The solution to this problem is the same as the solution to:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{b}\|^2 \\ \text{subject to} & \boldsymbol{x} \in \mathcal{R}(\boldsymbol{A}). \end{array}$$

Substituting $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{y}$, we see that this is simply a linear least squares problem with decision variable $\boldsymbol{y}$. The solution to the least squares problem is $\boldsymbol{y}^* = (\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}$, which implies that the solution to the given problem is $\boldsymbol{x}^* = \boldsymbol{A}(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}$.

12.19

We solve the problem using two different methods. The first method would be to use the Lagrange multiplier technique to solve the equivalent problem,

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{x}-\boldsymbol{x}_0\|^2 \\ \text{subject to} & h(\boldsymbol{x}) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}\boldsymbol{x} - 1 = 0, \end{array}$$

The lagrangian for the above problem has the form,

$$l(\boldsymbol{x},\lambda) = x_1^2 + (x_2+3)^2 + x_3^2 + \lambda(x_1+x_2+x_3-1).$$

Applying the FONC gives

$$\nabla_{\boldsymbol{x}} l = \begin{bmatrix} 2x_1+\lambda \\ 2x_2+6+\lambda \\ 2x_3+\lambda \end{bmatrix} \quad \text{and} \quad x_1+x_2+x_3-1=0.$$

Solving the above yields

$$\boldsymbol{x}^* = \begin{bmatrix} \frac{4}{3} \\ -\frac{5}{3} \\ \frac{4}{3} \end{bmatrix}.$$

The second approach is to use the well-known solution to the minimum norm problem. We first derive a general solution formula for the problem,

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{x}-\boldsymbol{x}_0\| \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \end{array}$$

where $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $m \leq n$, and $\operatorname{rank}\boldsymbol{A} = m$. To proceed, we first transform the above problem from the $\boldsymbol{x}$ coordinates into the $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{x}_0$ coordinates to obtain,

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{z}\| \\ \text{subject to} & \boldsymbol{A}\boldsymbol{z} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0. \end{array}$$

The solution to the above problem has the form,

$$\begin{aligned} \boldsymbol{z}^* &= \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}_0) \\ &= \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{b} - \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{A}\boldsymbol{x}_0. \end{aligned}$$

Therefore, the solution to the original problem is

$$\begin{aligned} \boldsymbol{x}^* &= \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}_0) + \boldsymbol{x}_0 \\ &= \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{b} - \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{x}_0 \\ &= \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{b} + \left(\boldsymbol{I}_n - \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{A}\right)\boldsymbol{x}_0. \end{aligned}$$

We substitute into the above formula the given numerical data to obtain

$$\begin{aligned}\boldsymbol{x}^* &= \begin{bmatrix}\frac{1}{3}\\ \frac{1}{3}\\ \frac{1}{3}\end{bmatrix} + \begin{bmatrix}\frac{2}{3} & -\frac{1}{3} & -\frac{1}{3}\\ -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3}\\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3}\end{bmatrix}\begin{bmatrix}0\\ -3\\ 0\end{bmatrix}\\ &= \begin{bmatrix}\frac{4}{3}\\ -\frac{5}{3}\\ \frac{4}{3}\end{bmatrix}.\end{aligned}$$

12.20

For each $\boldsymbol{x} \in \mathbb{R}^n$, let $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{x}_0$. Then, the original problem is equivalent to

$$\begin{aligned}&\text{minimize} && \|\boldsymbol{y}\|\\ &\text{subject to} && \boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0,\end{aligned}$$

in the sense that $\boldsymbol{y}^*$ is a solution to the above problem if and only if $\boldsymbol{x}^* = \boldsymbol{y}^* + \boldsymbol{x}_0$ is a solution to the original problem. By Theorem 12.2, the above problem has a unique solution given by

$$\boldsymbol{y}^* = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0) = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}_0.$$

Therefore, the solution to the original problem is

$$\boldsymbol{x}^* = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{x}_0 = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} + (\boldsymbol{I}_n - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A})\boldsymbol{x}_0.$$

Note that

$$\begin{aligned}\|\boldsymbol{x}^* - \boldsymbol{x}_0\| &= \boldsymbol{y}^*\\ &= \|\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0)\|\\ &= \|\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}_0\|.\end{aligned}$$

12.21

The objective function of the given problem can be written as

$$f(\boldsymbol{x}) = \|\boldsymbol{B}\boldsymbol{x} - \boldsymbol{c}\|^2,$$

where

$$\boldsymbol{B} = \begin{bmatrix}\boldsymbol{A}\\ \vdots\\ \boldsymbol{A}\end{bmatrix}, \qquad \boldsymbol{c} = \begin{bmatrix}\boldsymbol{b}_1\\ \vdots\\ \boldsymbol{b}_p\end{bmatrix}.$$

The solution is therefore

$$\boldsymbol{x}^* = (\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top\boldsymbol{c} = (p\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top(\boldsymbol{b}_1 + \cdots + \boldsymbol{b}_p) = \frac{1}{p}\sum_{i=1}^{p}(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b}_i = \frac{1}{p}\sum_{i=1}^{p}\boldsymbol{x}_i^*$$

Alternatively: Write

$$\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}_i\|^2 = \boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{b}_i + \|\boldsymbol{b}_i\|^2$$

Therefore, the given objective function can be written as

$$p\boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{x}^\top\boldsymbol{A}^\top(\boldsymbol{b}_1 + \cdots + \boldsymbol{b}_p) + \|\boldsymbol{b}_1\|^2 + \cdots + \|\boldsymbol{b}_i\|^2.$$

The solution is therefore

$$\boldsymbol{x}^* = (p\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top(\boldsymbol{b}_1 + \cdots + \boldsymbol{b}_p) = \frac{1}{p}\sum_{i=1}^{p}\boldsymbol{x}_i^*$$

Note that the original problem can be written as the least squares problem

$$\text{minimize } \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2,$$

where

$$\boldsymbol{b} = \frac{\boldsymbol{b}_1 + \cdots + \boldsymbol{b}_p}{p}.$$

12.22

Write

$$\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}_i\|^2 = \boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{b}_i + \|\boldsymbol{b}_i\|^2$$

Therefore, the given objective function can be written as

$$(\alpha_1 + \cdots + \alpha_p)\boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{x}^\top \boldsymbol{A}^\top (\alpha_1 \boldsymbol{b}_1 + \cdots + \alpha_p \boldsymbol{b}_p) + \alpha_1 \|\boldsymbol{b}_1\|^2 + \cdots + \alpha_p \|\boldsymbol{b}_i\|^2.$$

The solution is therefore (by inspection)

$$\boldsymbol{x}^* = ((\alpha_1 + \cdots + \alpha_p)\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top (\alpha_1 \boldsymbol{b}_1 + \cdots + \alpha_p \boldsymbol{b}_p) = \frac{1}{\alpha_1 + \cdots + \alpha_p} \sum_{i=1}^{p} \alpha_i \boldsymbol{x}_i^* = \sum_{i=1}^{p} \beta_i \boldsymbol{x}_i^*,$$

where $\beta_i = \alpha_i/(\alpha_1 + \cdots + \alpha_p)$.

Note that the original problem can be written as the least squares problem

$$\text{minimize } \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2,$$

where

$$\boldsymbol{b} = \frac{\alpha_1 \boldsymbol{b}_1 + \cdots + \alpha_p \boldsymbol{b}_p}{\alpha_1 + \cdots + \alpha_p}.$$

12.23

Let $\boldsymbol{x}^* = \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}$. Suppose $\boldsymbol{y}^*$ is a point in $\mathcal{R}(\boldsymbol{A}^\top)$ that satisfies $\boldsymbol{A}\boldsymbol{y}^* = \boldsymbol{b}$. Then, there exists $\boldsymbol{z}^* \in \mathbb{R}^m$ such that $\boldsymbol{y}^* = \boldsymbol{A}^\top \boldsymbol{z}^*$. Then, subtracting the equation $\boldsymbol{A}(\boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}) = \boldsymbol{b}$ from the equation $\boldsymbol{A}(\boldsymbol{A}^\top \boldsymbol{z}^*) = \boldsymbol{b}$, we get

$$(\boldsymbol{A}\boldsymbol{A}^\top)(\boldsymbol{z}^* - (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}) = \boldsymbol{0}.$$

Since $\operatorname{rank} \boldsymbol{A} = m$, $\boldsymbol{A}\boldsymbol{A}^\top$ is nonsingular. Therefore, $\boldsymbol{z}^* - (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} = \boldsymbol{0}$, which implies that

$$\boldsymbol{y}^* = \boldsymbol{A}^\top \boldsymbol{z}^* = \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b} = \boldsymbol{x}^*.$$

Hence, $\boldsymbol{x}^* = \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}$ is the only vector in $\mathcal{R}(\boldsymbol{A}^\top)$ that satisfies $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}$.

12.24

a. We have

$$\boldsymbol{x}^{(0)} = (\boldsymbol{A}_0^\top \boldsymbol{A}_0)^{-1} \boldsymbol{A}_0^\top \boldsymbol{b}^{(0)} = \boldsymbol{G}_0^{-1} \boldsymbol{A}_0^\top \boldsymbol{b}^{(0)}.$$

Similarly,

$$\boldsymbol{x}^{(1)} = (\boldsymbol{A}_1^\top \boldsymbol{A}_1)^{-1} \boldsymbol{A}_1^\top \boldsymbol{b}^{(1)} = \boldsymbol{G}_1^{-1} \boldsymbol{A}_1^\top \boldsymbol{b}^{(1)}.$$

b. Now,

$$\begin{aligned} \boldsymbol{G}_0 &= \begin{bmatrix} \boldsymbol{A}_1^\top & \boldsymbol{a}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{a}_1^\top \end{bmatrix} \\ &= \boldsymbol{A}_1^\top \boldsymbol{A}_1 + \boldsymbol{a}_1 \boldsymbol{a}_1^\top \\ &= \boldsymbol{G}_1 + \boldsymbol{a}_1 \boldsymbol{a}_1^\top. \end{aligned}$$

Hence,

$$\boldsymbol{G}_1 = \boldsymbol{G}_0 - \boldsymbol{a}_1 \boldsymbol{a}_1^\top.$$

c. Using the Sherman-Morrison formula,

$$\begin{aligned}\boldsymbol{P}_1 &= \boldsymbol{G}_1^{-1} \\ &= (\boldsymbol{G}_0 - \boldsymbol{a}_1\boldsymbol{a}_1^\top)^{-1} \\ &= \boldsymbol{G}_0^{-1} - \frac{\boldsymbol{G}_0^{-1}(-\boldsymbol{a}_1)\boldsymbol{a}_1^\top\boldsymbol{G}_0^{-1}}{1+(-\boldsymbol{a}_1)^\top\boldsymbol{G}_0^{-1}\boldsymbol{a}_1} \\ &= \boldsymbol{P}_0 + \frac{\boldsymbol{P}_0\boldsymbol{a}_1\boldsymbol{a}_1^\top\boldsymbol{P}_0}{1-\boldsymbol{a}_1^\top\boldsymbol{P}_0\boldsymbol{a}_1}.\end{aligned}$$

d. We have

$$\begin{aligned}\boldsymbol{A}_0^\top\boldsymbol{b}^{(0)} &= \boldsymbol{G}_0\boldsymbol{G}_0^{-1}\boldsymbol{A}_0^\top\boldsymbol{b}^{(0)} \\ &= \boldsymbol{G}_0\boldsymbol{x}^{(0)} \\ &= (\boldsymbol{G}_1 + \boldsymbol{a}_1\boldsymbol{a}_1^\top)\boldsymbol{x}^{(0)} \\ &= \boldsymbol{G}_1\boldsymbol{x}^{(0)} + \boldsymbol{a}_1\boldsymbol{a}_1^\top\boldsymbol{x}^{(0)}.\end{aligned}$$

e. Finally,

$$\begin{aligned}\boldsymbol{x}^{(1)} &= \boldsymbol{G}_1^{-1}\boldsymbol{A}_1^\top\boldsymbol{b}^{(1)} \\ &= \boldsymbol{G}_1^{-1}(\boldsymbol{A}_1^\top\boldsymbol{b}^{(1)} + \boldsymbol{a}_1 b_1 - \boldsymbol{a}_1 b_1) \\ &= \boldsymbol{G}_1^{-1}\left(\boldsymbol{A}_0^\top\boldsymbol{b}^{(0)} - \boldsymbol{a}_1 b_1\right) \\ &= \boldsymbol{G}_1^{-1}\left(\boldsymbol{G}_1\boldsymbol{x}^{(0)} + \boldsymbol{a}_1\boldsymbol{a}_1^\top\boldsymbol{x}^{(0)} - \boldsymbol{a}_1 b_1\right) \\ &= \boldsymbol{x}^{(0)} - \boldsymbol{G}_1^{-1}\boldsymbol{a}_1\left(b_1 - \boldsymbol{a}_1^\top\boldsymbol{x}^{(0)}\right) \\ &= \boldsymbol{x}^{(0)} - \boldsymbol{P}_1\boldsymbol{a}_1\left(b_1 - \boldsymbol{a}_1^\top\boldsymbol{x}^{(0)}\right).\end{aligned}$$

The general RLS algorithm for removals of rows is:

$$\begin{aligned}\boldsymbol{P}^{(k+1)} &= \boldsymbol{P}_k + \frac{\boldsymbol{P}_k\boldsymbol{a}_{k+1}\boldsymbol{a}_{k+1}^\top\boldsymbol{P}_k}{1-\boldsymbol{a}_{k+1}^\top\boldsymbol{P}_k\boldsymbol{a}_{k+1}} \\ \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \boldsymbol{P}_{k+1}\boldsymbol{a}_{k+1}\left(b_{k+1} - \boldsymbol{a}_{k+1}^\top\boldsymbol{x}^{(k)}\right).\end{aligned}$$

12.25

Using the notation of the proof of Theorem 12.3, we can write

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \mu(b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top\boldsymbol{x}^{(k)})\frac{\boldsymbol{a}_{R(k)+1}}{\|\boldsymbol{a}_{R(k)+1}\|^2}.$$

Hence,

$$\boldsymbol{x}^{(k)} = \sum_{i=0}^{k-1}\left(\frac{\mu(2-\mu)}{\|\boldsymbol{a}_{R(k)+1}\|^2}(b_{R(i)+1} - \boldsymbol{a}_{R(i)+1}^\top\boldsymbol{x}^{(i)})\right)\boldsymbol{a}_{R(i)+1}$$

which means that $\boldsymbol{x}^{(k)}$ is in $\text{span}[\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m] = \mathcal{R}(\boldsymbol{A}^\top)$.

12.26

a. We claim that $\boldsymbol{x}^*$ minimizes $\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|$ subject to $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$ if and only if $\boldsymbol{y}^* = \boldsymbol{x}^* - \boldsymbol{x}^{(0)}$ minimizes $\|\boldsymbol{y}\|$ subject to $\{\boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$.

To prove sufficiency, suppose $\boldsymbol{y}^*$ minimizes $\|\boldsymbol{y}\|$ subject to $\{\boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$. Let $\boldsymbol{x}^* = \boldsymbol{y}^* + \boldsymbol{x}^{(0)}$. Consider any point $\boldsymbol{x}_1 \in \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$. Now,

$$\boldsymbol{A}(\boldsymbol{x}_1 - \boldsymbol{x}^{(0)}) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}.$$

Hence, by definition of $\boldsymbol{y}^*$,

$$\|\boldsymbol{x}_1 - \boldsymbol{x}^{(0)}\| \geq \|\boldsymbol{y}^*\| = \|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|.$$

Therefore $\boldsymbol{x}^*$ minimizes $\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|$ subject to $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$.

To prove necessity, suppose $\boldsymbol{x}^*$ minimizes $\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|$ subject to $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$. Let $\boldsymbol{y}^* = \boldsymbol{x}^* - \boldsymbol{x}^{(0)}$. Consider any point $\boldsymbol{y}_1 \in \{\boldsymbol{y} : \boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$. Now,

$$\boldsymbol{A}(\boldsymbol{y}_1 + \boldsymbol{x}^{(0)}) = \boldsymbol{b}.$$

Hence, by definition of $\boldsymbol{x}^*$,

$$\|\boldsymbol{y}_1\| = \|(\boldsymbol{y}_1 + \boldsymbol{x}^{(0)}) - \boldsymbol{x}^{(0)}\| \geq \|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\| = \|\boldsymbol{y}^*\|.$$

Therefore, $\boldsymbol{y}^*$ minimizes $\|\boldsymbol{y}\|$ subject to $\{\boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$.

By Theorem 12.2, there exists a unique vector $\boldsymbol{y}^*$ minimizing $\|\boldsymbol{y}\|$ subject to $\{\boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$. Hence, by the above claim, there exists a unique $\boldsymbol{x}^*$ minimizing $\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|$ subject to $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$

b. Using the notation of the proof of Theorem 12.3, Kaczmarz's algorithm is given by

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \mu(b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(k)})\boldsymbol{a}_{R(k)+1}.$$

Subtract $\boldsymbol{x}^{(0)}$ from each side to give

$$(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(0)}) = (\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(0)}) + \mu((b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(0)}) - \boldsymbol{a}_{R(k)+1}^\top (\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(0)}))\boldsymbol{a}_{R(k)+1}.$$

Writing $\boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}^{(0)}$, we get

$$\boldsymbol{y}^{(k+1)} = \boldsymbol{y}^{(k)} + \mu((b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(0)}) - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{y}^{(k)})\boldsymbol{a}_{R(k)+1}.$$

Note that $\boldsymbol{y}^{(0)} = \boldsymbol{0}$. By Theorem 12.3, the sequence $\{\boldsymbol{y}^{(k)}\}$ converges to the unique point $\boldsymbol{y}^*$ that minimizes $\|\boldsymbol{y}\|$ subject to $\{\boldsymbol{A}\boldsymbol{y} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)}\}$. Hence $\{\boldsymbol{x}^{(k)}\}$ converges to $\boldsymbol{y}^* + \boldsymbol{x}^{(0)}$. From the proof of part a, $\boldsymbol{x}^* = \boldsymbol{y}^* + \boldsymbol{x}^{(0)}$ minimizes $\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|$ subject to $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$. This completes the proof.

12.27

Following the proof of Theorem 12.3, assuming $\|\boldsymbol{a}\| = 1$ without loss of generality, we arrive at

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|^2 = \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2 - \mu(2-\mu)(\boldsymbol{a}^\top(\boldsymbol{x}^{(k)} - \boldsymbol{x}^*))^2.$$

Since $\boldsymbol{x}^{(k)}, \boldsymbol{x}^* \in \mathcal{R}(\boldsymbol{A}) = \mathcal{R}([\boldsymbol{a}^\top])$ by Exercise 12.25, we have $\boldsymbol{x}^{(k)} - \boldsymbol{x}^* \in \mathcal{R}(\boldsymbol{A})$. Hence, by the Cauchy-Schwarz inequality,

$$(\boldsymbol{a}^\top(\boldsymbol{x}^{(k)} - \boldsymbol{x}^*))^2 = \|\boldsymbol{a}\|^2\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2 = \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2,$$

since $\|\boldsymbol{a}\| = 1$ by assumption. Thus, we obtain

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|^2 = (1 - \mu(2-\mu))\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2 = \gamma^2\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2$$

where $\gamma = \sqrt{1-\mu(2-\mu)}$. It is easy to check that $0 \leq 1 - \mu(2-\mu) < 1$ for all $\mu \in (0,2)$. Hence, $0 \leq \gamma < 1$.

12.28

In Kaczmarz's algorithm with $\mu = 1$, we may write

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + (b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(k)})\frac{\boldsymbol{a}_{R(k)+1}}{\|\boldsymbol{a}_{R(k)+1}\|^2}.$$

Subtracting $\boldsymbol{x}^*$ and premultiplying both sides by $\boldsymbol{a}_{R(k)+1}^\top$ yields

$$\begin{aligned}
\boldsymbol{a}_{R(k)+1}^\top(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*) &= \boldsymbol{a}_{R(k)+1}^\top\left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^* + (b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(k)})\frac{\boldsymbol{a}_{R(k)+1}}{\|\boldsymbol{a}_{R(k)+1}\|^2}\right) \\
&= \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(k)} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^* + (b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^{(k)}) \\
&= b_{R(k)+1} - \boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^* \\
&= 0.
\end{aligned}$$

Substituting $\boldsymbol{a}_{R(k)+1}^\top \boldsymbol{x}^* = b_{R(k)+1}$ yields the desired result.

12.29

We will prove this by contradiction. Suppose $\boldsymbol{C}\boldsymbol{x}^*$ is not the minimizer of $\|\boldsymbol{B}\boldsymbol{y} - \boldsymbol{b}\|^2$ over $\mathbb{R}^r$. Let $\hat{\boldsymbol{y}}$ be the minimizer of $\|\boldsymbol{B}\boldsymbol{y} - \boldsymbol{b}\|^2$ over $\mathbb{R}^r$. Then, $\|\boldsymbol{B}\hat{\boldsymbol{y}} - \boldsymbol{b}\|^2 < \|\boldsymbol{B}\boldsymbol{C}\boldsymbol{x}^* - \boldsymbol{b}\|^2 = \|\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{b}\|^2$. Since $\boldsymbol{C}$ is of full rank, there exists $\hat{\boldsymbol{x}} \in \mathbb{R}^n$ such that $\hat{\boldsymbol{y}} = \boldsymbol{C}\hat{\boldsymbol{x}}$. Therefore,

$$\|\boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{b}\|^2 = \|\boldsymbol{B}\boldsymbol{C}\hat{\boldsymbol{x}} - \boldsymbol{b}\|^2 = \|\boldsymbol{B}\hat{\boldsymbol{y}} - \boldsymbol{b}\|^2 < \|\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{b}\|^2$$

which contradicts the assumption that $\boldsymbol{x}^*$ is a minimizer of $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ over $\mathbb{R}^n$.

12.30

a. Let $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{C}$ be a full rank factorization of $\boldsymbol{A}$. Now, we have $\boldsymbol{A}^\dagger = \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger$, where $\boldsymbol{B}^\dagger = (\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top$ and $\boldsymbol{C}^\dagger = \boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1}$. On the other hand $(\boldsymbol{A}^\top)^\dagger = (\boldsymbol{C}^\top\boldsymbol{B}^\top)^\dagger$. Since $\boldsymbol{A}^\top = \boldsymbol{C}^\top\boldsymbol{B}^\top$ is a full rank factorization of $\boldsymbol{A}^\top$, we have $(\boldsymbol{A}^\top)^\dagger = (\boldsymbol{C}^\top\boldsymbol{B}^\top)^\dagger = (\boldsymbol{B}^\top)^\dagger(\boldsymbol{C}^\top)^\dagger$. Therefore, to show that $(\boldsymbol{A}^\top)^\dagger = (\boldsymbol{A}^\dagger)^\top$, it is enough to show that

$$\begin{aligned}(\boldsymbol{B}^\top)^\dagger &= (\boldsymbol{B}^\dagger)^\top \\ (\boldsymbol{C}^\top)^\dagger &= (\boldsymbol{C}^\dagger)^\top.\end{aligned}$$

To this end, note that $(\boldsymbol{B}^\top)^\dagger = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}$, and $(\boldsymbol{C}^\top)^\dagger = (\boldsymbol{C}\boldsymbol{C}^\top)^{-1}\boldsymbol{C}$. On the other hand, $(\boldsymbol{B}^\dagger)^\top = ((\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top)^\top = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}$, and $(\boldsymbol{C}^\dagger)^\top = (\boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1})^\top = (\boldsymbol{C}\boldsymbol{C}^\top)^{-1}\boldsymbol{C}$, which completes the proof.

b. Note that $\boldsymbol{A}^\dagger = \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger$, which is a full rank factorization of $\boldsymbol{A}^\dagger$. Therefore, $(\boldsymbol{A}^\dagger)^\dagger = (\boldsymbol{B}^\dagger)^\dagger(\boldsymbol{C}^\dagger)^\dagger$. Hence, to show that $(\boldsymbol{A}^\dagger)^\dagger = \boldsymbol{A}$, it is enough to show that

$$\begin{aligned}(\boldsymbol{B}^\dagger)^\dagger &= \boldsymbol{B} \\ (\boldsymbol{C}^\dagger)^\dagger &= \boldsymbol{C}.\end{aligned}$$

To this end, note that $(\boldsymbol{B}^\dagger)^\dagger = ((\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top)^\dagger = \boldsymbol{B}$ since $\boldsymbol{B}^\dagger$ is a full rank matrix. Similarly, $(\boldsymbol{C}^\dagger)^\dagger = (\boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1})^\dagger = \boldsymbol{C}$ since $\boldsymbol{C}^\dagger$ is a full rank matrix. This completes the proof.

12.31

$\Rightarrow$: We prove properties 1–4 in turn.

1. This is immediate.

2. Let $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{C}$ be a full rank factorization of $\boldsymbol{A}$. We have $\boldsymbol{A}^\dagger = \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger$, where $\boldsymbol{B}^\dagger = (\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top$ and $\boldsymbol{C}^\dagger = \boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1}$. Note that $\boldsymbol{B}^\dagger\boldsymbol{B} = \boldsymbol{I}$ and $\boldsymbol{C}\boldsymbol{C}^\dagger = \boldsymbol{I}$. Now,

$$\begin{aligned}\boldsymbol{A}^\dagger\boldsymbol{A}\boldsymbol{A}^\dagger &= \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger\boldsymbol{B}\boldsymbol{C}\boldsymbol{C}^\dagger\boldsymbol{B}^\dagger \\ &= \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger \\ &= \boldsymbol{A}^\dagger.\end{aligned}$$

3. We have

$$\begin{aligned}(\boldsymbol{A}\boldsymbol{A}^\dagger)^\top &= (\boldsymbol{B}\boldsymbol{C}\boldsymbol{C}^\dagger\boldsymbol{B}^\dagger)^\top \\ &= (\boldsymbol{B}\boldsymbol{B}^\dagger)^\top \\ &= (\boldsymbol{B}^\dagger)^\top\boldsymbol{B}^\top \\ &= ((\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top)^\top\boldsymbol{B}^\top \\ &= \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top \\ &= \boldsymbol{B}\boldsymbol{B}^\dagger \\ &= \boldsymbol{B}\boldsymbol{C}\boldsymbol{C}^\dagger\boldsymbol{B}^\dagger \\ &= \boldsymbol{A}\boldsymbol{A}^\dagger.\end{aligned}$$

4. We have

$$\begin{aligned}(\boldsymbol{A}^\dagger\boldsymbol{A})^\top &= (\boldsymbol{C}^\dagger\boldsymbol{B}^\dagger\boldsymbol{B}\boldsymbol{C})^\top \\ &= (\boldsymbol{C}^\dagger\boldsymbol{C})^\top \\ &= \boldsymbol{C}^\top(\boldsymbol{C}^\dagger)^\top \\ &= \boldsymbol{C}^\top(\boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1})^\top \\ &= \boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{C}^\top)^{-1}\boldsymbol{C} \\ &= \boldsymbol{C}^\dagger\boldsymbol{C} \\ &= \boldsymbol{C}^\dagger\boldsymbol{B}^\dagger\boldsymbol{B}\boldsymbol{C} \\ &= \boldsymbol{A}^\dagger\boldsymbol{A}.\end{aligned}$$

$\Leftarrow$: By property 1, we immediately have $\boldsymbol{A}\boldsymbol{A}^\dagger\boldsymbol{A} = \boldsymbol{A}$. Therefore, it remains to show that there exist matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ such that $\boldsymbol{A}^\dagger = \boldsymbol{U}\boldsymbol{A}^\top$ and $\boldsymbol{A}^\dagger = \boldsymbol{A}^\top\boldsymbol{V}$.

For this, we note from property 2 that $\boldsymbol{A}^\dagger = \boldsymbol{A}^\dagger\boldsymbol{A}\boldsymbol{A}^\dagger$. But from property 3, $\boldsymbol{A}\boldsymbol{A}^\dagger = (\boldsymbol{A}\boldsymbol{A}^\dagger)^\top = (\boldsymbol{A}^\dagger)^\top\boldsymbol{A}^\top$. Hence, $\boldsymbol{A}^\dagger = \boldsymbol{A}^\dagger(\boldsymbol{A}^\dagger)^\top\boldsymbol{A}^\top$. Setting $\boldsymbol{U} = \boldsymbol{A}^\dagger(\boldsymbol{A}^\dagger)^\top$, we get that $\boldsymbol{A}^\dagger = \boldsymbol{U}\boldsymbol{A}^\top$.

Similarly, we note from property 4 that $\boldsymbol{A}^\dagger\boldsymbol{A} = (\boldsymbol{A}^\dagger\boldsymbol{A})T = \boldsymbol{A}^\top(\boldsymbol{A}^\dagger)^\top$. Substituting this back into property 2 yields $\boldsymbol{A}^\dagger = \boldsymbol{A}^\dagger\boldsymbol{A}\boldsymbol{A}^\dagger = \boldsymbol{A}^\top(\boldsymbol{A}^\dagger)^\top\boldsymbol{A}^\dagger$. Setting $\boldsymbol{V} = (\boldsymbol{A}^\dagger)^\top\boldsymbol{A}^\dagger$ yields $\boldsymbol{A}^\dagger = \boldsymbol{A}^\top\boldsymbol{V}$. This completes the proof.

12.32

(Taken from [23, p. 24]) Let

$$\boldsymbol{A}_1 = \begin{bmatrix}0 & 0 & 0\\ 0 & 1 & 1\\ 0 & 1 & 0\end{bmatrix}, \qquad \boldsymbol{A}_2 = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}.$$

We compute

$$\boldsymbol{A}_1^\dagger = \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 1 & -1\end{bmatrix}, \qquad \boldsymbol{A}_2^\dagger = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix} = \boldsymbol{A}_2.$$

We have

$$\boldsymbol{A}_1\boldsymbol{A}_2 = \begin{bmatrix}0 & 0 & 0\\ 0 & 1 & 0\\ 0 & 1 & 0\end{bmatrix} = \begin{bmatrix}0\\ 1\\ 1\end{bmatrix}\begin{bmatrix}0 & 1 & 0\end{bmatrix}$$

which is a full rank factorization. Therefore,

$$(\boldsymbol{A}_1\boldsymbol{A}_2)^\dagger = \begin{bmatrix}0 & 0 & 0\\ 0 & 1/2 & 1/2\\ 0 & 0 & 0\end{bmatrix}.$$

But

$$\boldsymbol{A}_2^\dagger\boldsymbol{A}_1^\dagger = \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{bmatrix}.$$

Hence, $(\boldsymbol{A}_1\boldsymbol{A}_2)^\dagger \neq \boldsymbol{A}_2^\dagger\boldsymbol{A}_1^\dagger$.

13. Unconstrained Optimization and Feedforward Neural Networks

13.1

a. The gradient of f is given by

$$\nabla f(\boldsymbol{w}) = -\boldsymbol{X}_d(\boldsymbol{y}_d - \boldsymbol{X}_d^\top\boldsymbol{w}).$$

b. The Conjugate Gradient algorithm applied to our training problem is:

1. Set $k := 0$; select the initial point $\boldsymbol{w}^{(0)}$.
2. $\boldsymbol{g}^{(0)} = -\boldsymbol{X}_d(\boldsymbol{y}_d - \boldsymbol{X}_d^\top \boldsymbol{w}^{(0)})$. If $\boldsymbol{g}^{(0)} = \boldsymbol{0}$, stop, else set $\boldsymbol{d}^{(0)} = -\boldsymbol{g}^{(0)}$.
3. $\alpha_k = -\frac{\boldsymbol{d}^{(k)\top}\boldsymbol{g}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{X}_d\boldsymbol{X}_d^\top\boldsymbol{d}^{(k)}}$
4. $\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} + \alpha_k\boldsymbol{d}^{(k)}$
5. $\boldsymbol{g}^{(k+1)} = \boldsymbol{X}_d(\boldsymbol{y}_d - \boldsymbol{X}_d^\top \boldsymbol{w}^{(k+1)})$. If $\boldsymbol{g}^{(k+1)} = \boldsymbol{0}$, stop.
6. $\beta_k = \frac{\boldsymbol{g}^{(k+1)\top}\boldsymbol{X}_d\boldsymbol{X}_d^\top\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top}\boldsymbol{X}_d\boldsymbol{X}_d^\top\boldsymbol{d}^{(k)}}$
7. $\boldsymbol{d}^{(k+1)} = -\boldsymbol{g}^{(k+1)} + \beta_k\boldsymbol{d}^{(k)}$
8. Set $k := k + 1$; go to 3.

c. We form the matrix $\boldsymbol{X}_d$ as

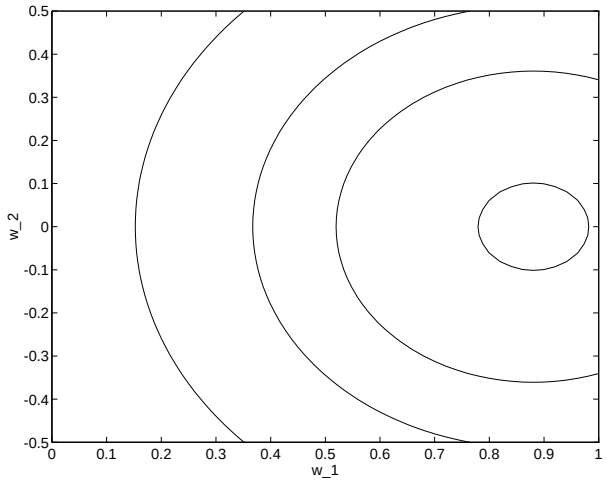
$$\boldsymbol{X}_d = \begin{bmatrix} -0.5 & -0.5 & -0.5 & 0 & 0 & 0 & 0.5 & 0.5 & 0.5 \\ -0.5 & 0 & 0.5 & -0.5 & 0 & 0.5 & -0.5 & 0 & 0.5 \end{bmatrix}$$

and the vector $\boldsymbol{y}_d$ as

$$\boldsymbol{y}_d = [-0.42074, -0.47943, -0.42074, 0, 0, 0, 0.42074, 0.47943, 0.42074]^\top.$$

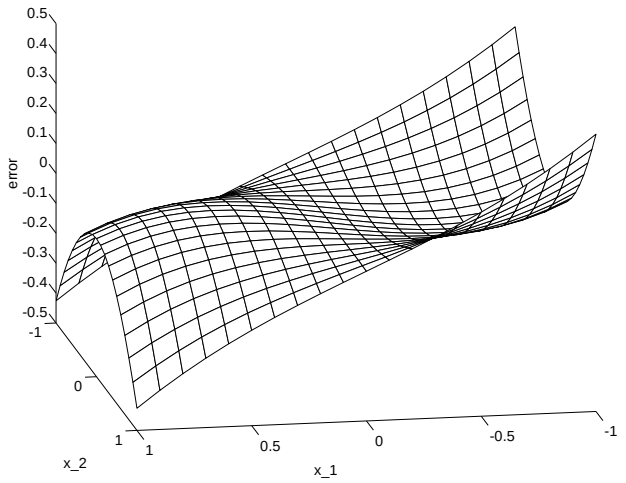
Running the Conjugate Gradient algorithm, we get a solution of $\boldsymbol{w}^* = [0.8806, 0.000]^\top$.

d. The level sets are shown in the figure below.



The solution in part c agrees with the level sets.

e. The plot of the error function is depicted below.



13.2

a. The expression we seek is

$$e_{k+1} = (1-\mu)e_k.$$

To derive the above, we write

$$\begin{aligned} e_{k+1} - e_k &= y_d - \boldsymbol{x}_d^\top \boldsymbol{w}^{(k+1)} - \left(y_d - \boldsymbol{x}_d^\top \boldsymbol{w}^{(k)}\right) \\ &= -\boldsymbol{x}_d^\top \left(\boldsymbol{w}^{(k+1)} - \boldsymbol{w}^{(k)}\right). \end{aligned}$$

Substituting for $\boldsymbol{w}^{(k+1)} - \boldsymbol{w}^{(k)}$ from the Widrow-Hoff algorithm yields

$$e_{k+1} - e_k = -\mu \boldsymbol{x}_d^\top \frac{e_k \boldsymbol{x}_d}{\boldsymbol{x}_d^\top \boldsymbol{x}_d} = -\mu e_k.$$

Hence, $e_{k+1} = (1-\mu)e_k$.

b. For $e_k \to 0$, it is necessary and sufficient that $|1-\mu| < 1$, which is equivalent to $0 < \mu < 2$.

13.3

a. The error satisfies

$$\boldsymbol{e}^{(k+1)} = (\boldsymbol{I}_p - \boldsymbol{\mu})\boldsymbol{e}^{(k)}.$$

To derive the above expression, we write

$$\begin{aligned} \boldsymbol{e}^{(k+1)} - \boldsymbol{e}^{(k)} &= \boldsymbol{y}_d - \boldsymbol{X}_d^\top \boldsymbol{w}^{(k+1)} - \left(\boldsymbol{y}_d - \boldsymbol{X}_d^\top \boldsymbol{w}^{(k)}\right) \\ &= -\boldsymbol{X}_d^\top \left(\boldsymbol{w}^{(k+1)} - \boldsymbol{w}^{(k)}\right). \end{aligned}$$

Substituting for $\boldsymbol{w}^{(k+1)} - \boldsymbol{w}^{(k)}$ from the algorithm yields

$$\boldsymbol{e}^{(k+1)} - \boldsymbol{e}^{(k)} = -\boldsymbol{X}_d^\top \boldsymbol{X}_d (\boldsymbol{X}_d^\top \boldsymbol{X}_d)^{-1} \boldsymbol{\mu} \boldsymbol{e}^{(k)} = -\boldsymbol{\mu} \boldsymbol{e}^{(k)}.$$

Hence, $\boldsymbol{e}^{(k+1)} = (\boldsymbol{I}_p - \boldsymbol{\mu})\boldsymbol{e}^{(k)}$.

b. From part a, we see that $\boldsymbol{e}^{(k)} = (\boldsymbol{I}_p - \boldsymbol{\mu})^k \boldsymbol{e}^{(0)}$. Hence, by Lemma 5.1, a necessary and sufficient condition for $\boldsymbol{e}^{(k)} \to \boldsymbol{0}$ for any $\boldsymbol{e}^{(0)}$ is that all the eigenvalues of $\boldsymbol{I}_p - \boldsymbol{\mu}$ must be located in the open unit circle. From Exercise 3.6, it follows that the above condition holds if and only if $|1 - \lambda_i(\boldsymbol{\mu})| < 1$ for each eigenvalue $\lambda_i(\boldsymbol{\mu})$ of $\boldsymbol{\mu}$. This is true if and only if $0 < |\lambda_i(\boldsymbol{\mu})| < 2$ for each eigenvalue $\lambda_i(\boldsymbol{\mu})$ of $\boldsymbol{\mu}$.

13.4

We modified the MATLAB routine of Exercise 8.25, by fixing the step size at a value $\eta = 100$. We need the following M-file for the gradient:

```
function y=Dfbp(w);

wh11=w(1);
wh21=w(2);
wh12=w(3);
wh22=w(4);
wo11=w(5);
wo12=w(6);

xd1=0; xd2=1; yd=1;

v1=wh11*xd1+wh12*xd2;
v2=wh21*xd1+wh22*xd2;
z1=sigmoid(v1);
z2=sigmoid(v2);
y1=sigmoid(wo11*z1+wo12*z2)
d1=(yd-y1)*y1*(1-y1);

y(1)=-d1*wo11*z1*(1-z1)*xd1;
y(2)=-d1*wo12*z2*(1-z2)*xd1;
y(3)=-d1*wo11*z1*(1-z1)*xd2;
y(4)=-d1*wo12*z2*(1-z2)*xd2;
y(5)=-d1*z1;
y(6)=-d1*z2;

y=y';
```

After 20 iterations of the backpropagation algorithm, we get the following weights:

$$\begin{aligned} w_{11}^{o(20)} &= 2.883 \\ w_{12}^{o(20)} &= 3.194 \\ w_{11}^{h(20)} &= 0.1000 \\ w_{12}^{h(20)} &= 0.8179 \\ w_{21}^{h(20)} &= 0.3000 \\ w_{22}^{h(20)} &= 1.106. \end{aligned}$$

The corresponding output of the network is $y_1^{(20)} = 0.9879$.

13.5

We used the following MATLAB routine:

```
function [x,N]=backprop(grad,xnew,options);
%    BACKPROP('grad',x0);
%    BACKPROP('grad',x0,OPTIONS);
%
%    x = BACKPROP('grad',x0);
%    x = BACKPROP('grad',x0,OPTIONS);
%
%    [x,N] = BACKPROP('grad',x0);
%    [x,N] = BACKPROP('grad',x0,OPTIONS);
%
%The first variant trains a net whose gradient
%is described in grad (usually an M-file: grad.m), using a backprop
%algorithm with initial point x0.
%The second variant allows a vector of optional parameters to
%defined. OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results, (default is no display: 0).
```

```
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required of the gradient.
%OPTIONS(14) is the maximum number of iterations.
%For more information type HELP FOPTIONS.
%
%The next two variants returns the value of the final point.
%The last two variants returns a vector of the final point and the
%number of iterations.

if nargin ~= 3
  options = [];
  if nargin ~= 2
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=1000*length(xnew);
  end
else
  options(14)=1000*length(xnew);
end

clc;
format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_g = options(3);
max_iter=options(14);

for k = 1:max_iter,

  xcurr=xnew;

  g_curr=feval(grad,xcurr);

  if norm(g_curr) <= epsilon_g
    disp('Terminating: Norm of gradient less than');
    disp(epsilon_g);
    k=k-1;
    break;
  end %if

  alpha=10.0;

  xnew = xcurr-alpha*g_curr;

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('Gradient = ');
    disp(g_curr'); %print gradient
```

```
    disp('New point =');
    disp(xnew'); %print new point
  end %if

  if norm(xnew-xcurr) <= epsilon_x*norm(xcurr)
    disp('Terminating: Norm of difference between iterates less than');
    disp(epsilon_x);
    break;
  end %if

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xnew');
  disp('Number of iterations =');
  disp(k);
end %if
%-------------------------------------------------------------------
```

To apply the above routine, we need the following M-file for the gradient.

```
function y=grad(w,xd,yd);

wh11=w(1);
wh21=w(2);
wh12=w(3);
wh22=w(4);
wo11=w(5);
wo12=w(6);
t1=w(7);
t2=w(8);
t3=w(9);

xd1=xd(1); xd2=xd(2);

v1=wh11*xd1+wh12*xd2-t1;
v2=wh21*xd1+wh22*xd2-t2;
z1=sigmoid(v1);
z2=sigmoid(v2);
y1=sigmoid(wo11*z1+wo12*z2-t3);
d1=(yd-y1)*y1*(1-y1);

y(1)=-d1*wo11*z1*(1-z1)*xd1;
y(2)=-d1*wo12*z2*(1-z2)*xd1;
y(3)=-d1*wo11*z1*(1-z1)*xd2;
y(4)=-d1*wo12*z2*(1-z2)*xd2;
y(5)=-d1*z1;
y(6)=-d1*z2;
y(7)=d1*wo11*z1*(1-z1);
y(8)=d1*wo12*z2*(1-z2);
y(9)=d1;
```

```
y=y';
```

We applied our MATLAB routine as follows.

```
>> options(2)=10^(-7);
>> options(3)=10^(-7);
>> options(14)=10000;
>> w0=[0.1,0.3,0.3,0.4,0.4,0.6,0.1,0.1,-0.1]';
>> [wstar,N]=backprop('grad',w0,options)

Terminating with maximum number of iterations
wstar =
  -7.7771e+00
  -5.5932e+00
  -8.4027e+00
  -5.6384e+00
  -1.1010e+01
   1.0918e+01
  -3.2773e+00
  -8.3565e+00
   5.2606e+00
N =
       10000
```

As we can see from the above, the results coincide with Example 13.3. The table of the outputs of the trained network corresponding to the training input data is shown in Table 13.2.

14. Global Search Algorithms

14.1

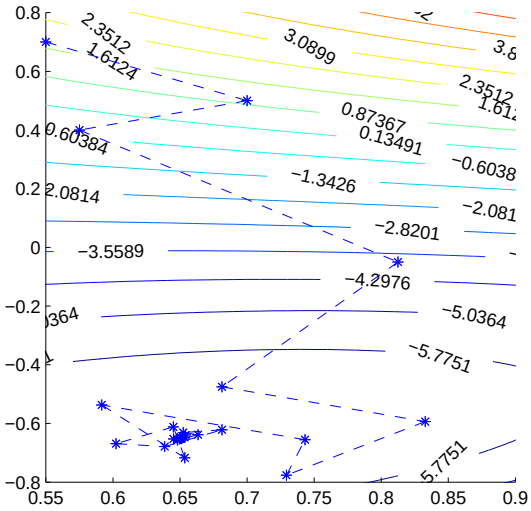
The MATLAB program is as follows.

```
function [ output_args ] = nm_simplex( input_args )
%Nelder-Mead simplex method
%Based on the program by the Spring 2007 ECE580 student, Hengzhou Ding

disp ('We minimize a function using the Nelder-Mead method.')
disp ('There are two initial conditions.')
disp ('You can enter your own starting point.')
disp ('-------------------------------------------')

% disp('Select one of the starting points')
% disp ('[0.55;0.7] or [-0.9;-0.5]')
% x0=input('')
disp ('                    ')
clear
close all;

disp('Select one of the starting points, or enter your own point')
disp('[0.55;0.7] or [-0.9;-0.5]')
disp('(Copy one of the above points and paste it at the prompt)')
x0=input('')

hold on
axis square
%Plot the contours of the objective function
[X1,X2]=meshgrid(-1:0.01:1);
```

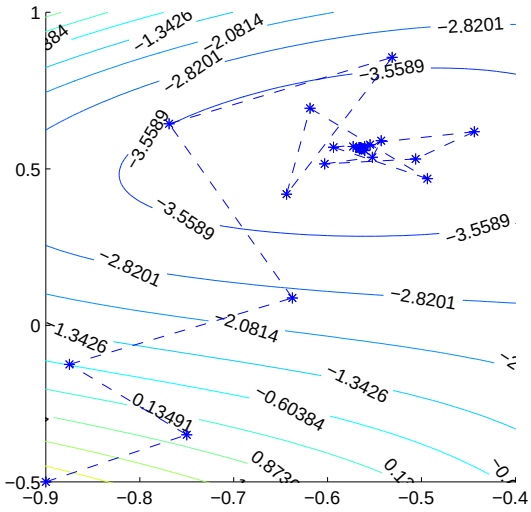
```
Y=(X2-X1).^4+12.*X1.*X2-X1+X2-3;
[C,h] = contour(X1,X2,Y,20);
clabel(C,h);

% Initialize all parameters
lambda=0.1;
rho=1;
chi=2;
gamma=1/2;
sigma=1/2;
e1=[1 0]';
e2=[0 1]';
%x0=[0.55 0.7]';
%x0=[-0.9 -0.5]';

% Plot initial point and initialize the simplex
plot(x0(1),x0(2),'--*');
x(:,3)=x0;
x(:,1)=x0+lambda*e1;
x(:,2)=x0+lambda*e2;

while 1
% Check the size of simplex for stopping criterion
simpsize=norm(x(:,1)-x(:,2))+norm(x(:,2)-x(:,3))+norm(x(:,3)-x(:,1));
if(simpsize<1e-6)
break;
end
lastpt=x(:,3);
% Sort the simplex
x=sort_points(x,3);
% Reflection
centro=1/2*(x(:,1)+x(:,2));
xr=centro+rho*(centro-x(:,3));
% Accept condition
if(obj_fun(xr)>=obj_fun(x(:,1)) && obj_fun(xr)<obj_fun(x(:,2)))
x(:,3)=xr;
% Expand condition
elseif(obj_fun(xr)<obj_fun(x(:,1)))
xe=centro+rho*chi*(centro-x(:,3));
if(obj_fun(xe)<obj_fun(xr))
x(:,3)=xe;
else
x(:,3)=xr;
end

% Outside contraction or shrink
elseif(obj_fun(xr)>=obj_fun(x(:,2)) &&
obj_fun(xr)<obj_fun(x(:,3)))
xc=centro+gamma*rho*(centro-x(:,3));
if(obj_fun(xc)<obj_fun(x(:,3)))
x(:,3)=xc;
else
x=shrink(x,sigma);
end
% Inside contraction or shrink
else
xcc=centro-gamma*(centro-x(:,3));
if(obj_fun(xcc)<obj_fun(x(:,3)))
x(:,3)=xcc;
```

```
else
x=shrink(x,sigma);
end
end
% Plot the new point and connect
plot([lastpt(1),x(1,3)],[lastpt(2),x(2,3)],'--*');
end
% Output the final simplex (minimizer)
x(:,1)

% obj_fun
function y = obj_fun(x)
y=(x(1)-x(2))4+12*x(1)*x(2)-x(1)+x(2)-3;

% sort_points
function y = sort_points(x,N)
for i=1:(N-1)
for j=1:(N-i)
if(obj_fun(x(:,j))>obj_fun(x(:,j+1)))
tmp=x(:,j);
x(:,j)=x(:,j+1);
x(:,j+1)=tmp;
end
end
end
y=x;

% shrink
function y = shrink(x,sigma)
x(:,2)=x(:,1)+sigma*(x(:,2)-x(:,1));
x(:,3)=x(:,1)+sigma*(x(:,3)-x(:,1));
y=x;
```

When we run the MATLAB code above with initial condition $[0.55, 0.7]^\top$, we obtain the following plot:



When we run the MATLAB code above with initial condition $[-0.9, -0.5]^\top$, we obtain the following plot:



Note that this function has two local minimizers. The algorithm terminates at these two minimizers with the two different initial conditions. This behavior depends on the value of lambda, which determines the initial simplex. It is possible to reach both minimizers starting from the same initial point by using different values of lambda. In the solution above, the initial simplex is "small"—lambda is just 0.1.

14.2

A MATLAB routine for a naive random search algorithm is given by the M-file `rs_demo` shown below:

```
function [x,N]=random_search(funcname,xnew,options);
%Naive random search demo
% [x,N]=random_search(funcname,xnew,options);
% print = options(1);
% alpha = options(18);

if nargin ~= 3
  options = [];
  if nargin ~= 2
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=1000*length(xnew);
  end
else
  options(14)=1000*length(xnew);
end
if length(options) < 18
  options(18)=1.0; %optional step size
end

format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
```

```
epsilon_g = options(3);
max_iter=options(14);

alpha0 = options(18);

if funcname == 'f_r',
  ros_cnt
elseif funcname == 'f_p',
  pks_cnt;
end %if

if length(xnew) == 2
  plot(xnew(1),xnew(2),'o')
  text(xnew(1),xnew(2),'Start Point')
  xlower = [-2;-1];
  xupper = [2;3];
end

f_0=feval(funcname,xnew);
xbestcurr = xnew;
xbestold = xnew;
f_best=feval(funcname,xnew);
f_best=10^(sign(f_best))*f_best;

for k = 1:max_iter,

  xcurr=xbestcurr;
  f_curr=feval(funcname,xcurr);

  alpha = alpha0;
  xnew = xcurr + alpha*(2*rand(length(xcurr),1)-1);

  for i=1:length(xnew),
    xnew(i) = max(xnew(i),xlower(i));
    xnew(i) = min(xnew(i),xupper(i));
  end %for

  f_new=feval(funcname,xnew);

  if f_new < f_best,
    xbestold = xbestcurr;
    xbestcurr = xnew;
    f_best = f_new;
  end

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('New point =');
    disp(xnew'); %print new point
    disp('Function value =');
    disp(f_new); %print func value at new point
  end %if

  if norm(xnew-xbestold) <= epsilon_x*norm(xbestold)
    disp('Terminating: Norm of difference between iterates less than');
    disp(epsilon_x);
```

```
    break;
  end %if

  pltpts(xbestcurr,xbestold);

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xbestcurr');
  disp('Number of iterations =');
  disp(k);
end %if
```

A MATLAB routine for a simulated annealing algorithm is given by the M-file `sa_demo` shown below:

```
function [x,N]=simulated_annealing(funcname,xnew,options);
%Simulated annealing demo
% random_search(funcname,xnew,options);
% print = options(1);
% gamma = options(15);
% alpha = options(18);

if nargin ~= 3
  options = [];
  if nargin ~= 2
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=1000*length(xnew);
  end
else
  options(14)=1000*length(xnew);
end

if length(options) < 15
  options(15)=5.0; %
end
if options(15)==0
  options(15)=5.0; %
end

if length(options) < 18
  options(18)=0.5; %optional step size
end

format compact;
```

```
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_g = options(3);
max_iter=options(14);

alpha = options(18);
gamma = options(15);
k0=2;

if funcname == 'f_r',
  ros_cnt
elseif funcname == 'f_p',
  pks_cnt;
end %if

if length(xnew) == 2
  plot(xnew(1),xnew(2),'o')
  text(xnew(1),xnew(2),'Start Point')
  xlower = [-2;-1];
  xupper = [2;3];
end

f_0=feval(funcname,xnew);
xbestcurr = xnew;
xbestold = xnew;
xcurr = xnew;
f_best=feval(funcname,xnew);
f_best=10^(sign(f_best))*f_best;

for k = 1:max_iter,

  f_curr=feval(funcname,xcurr);

  xnew = xcurr + alpha*(2*rand(length(xcurr),1)-1);

  for i=1:length(xnew),
    xnew(i) = max(xnew(i),xlower(i));
    xnew(i) = min(xnew(i),xupper(i));
  end %for

  f_new=feval(funcname,xnew);

  if f_new < f_curr,
    xcurr = xnew;
    f_curr = f_new;
  else
    cointoss = rand(1);
    Temp = gamma/log(k+k0);
    Prob = exp(-(f_new-f_curr)/Temp);
    if cointoss < Prob,
      xcurr = xnew;
      f_curr = f_new;
    end
  end

  if f_new < f_best,
```

```
    xbestold = xbestcurr;
    xbestcurr = xnew;
    f_best = f_new;
  end

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha =');
    disp(alpha);  %print alpha
    disp('New point =');
    disp(xnew'); %print new point
    disp('Function value =');
    disp(f_new); %print func value at new point
  end %if

  if norm(xnew-xbestold) <= epsilon_x*norm(xbestold)
    disp('Terminating: Norm of difference between iterates less
than');
    disp(epsilon_x);
    break;
  end %if

  pltpts(xbestcurr,xbestold);

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xbestcurr');
  disp('Objective function value =');
  disp(f_best);
  disp('Number of iterations =');
  disp(k);
end %if
```

To use the above routines, we also need the following M-files:
pltpts.m:

```
function out=pltpts(xnew,xcurr)

plot([xcurr(1),xnew(1)],[xcurr(2),xnew(2)],'r-',xnew(1),xnew(2),'o','Erasemode',
'none');
drawnow; % Draws current graph now
%pause(1)

out = [];
```

f_p.m:

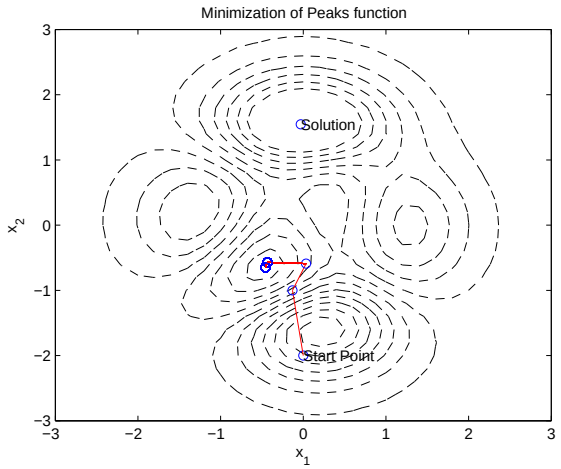
```
function y=f_p(x);
```

```
y=3*(1-x(1)).^2.*exp(-(x(1).^2)-(x(2)+1).^2) -
10.*(x(1)/5-x(1).^3-x(2).^5).*exp
(-x(1).^2-x(2).^2) - exp(-(x(1)+1).^2-x(2).^2)/3;
y=-y;
```

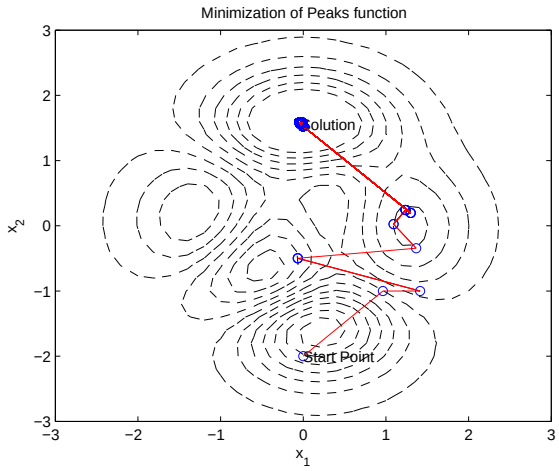
pks_cnt.m:

```
echo off
X = [-3:0.2:3]';
Y = [-3:0.2:3]';
[x,y]=meshgrid(X',Y') ;
func = 3*(1-x).^2.*exp(-x.^2-(y+1).^2) -
10.*(x/5-x.^3-y.^5).*exp(-x.^2-y.^2) -
exp(-(x+1).^2-y.^2)/3;
func = -func;
clf
levels = exp(-5:10);
levels = [-5:0.9:10];
contour(X,Y,func,levels,'k--')
xlabel('x_1')
ylabel('x_2')
title('Minimization of Peaks function')
drawnow;
hold on
plot(-0.0303,1.5455,'o')
text(-0.0303,1.5455,'Solution')
```

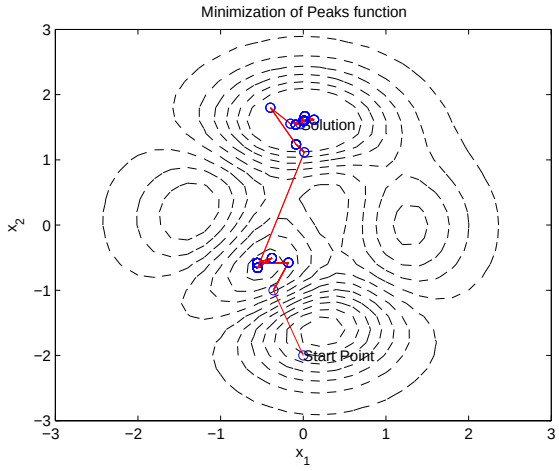
To run the naive random search algorithms, we first pick a value of $\alpha = 0.5$, which involves setting `options(18)=0.5`. We then use the command `rs_demo('f_p',[0;-2],options)`. The resulting plot of the algorithm trajectory is given below. As we can see, the algorithm is stuck at a local minimizer. (By running the algorithm several times, the reader can verify that this non-convergent behavior is typical.)



Next, we try $\alpha = 1.5$, which involves setting `options(18)=1.5`. We then use the command `rs_demo('f_p',[0;-2],options)` again, to obtain the plot shown below. This time, the algorithm reaches the global minimizer.



Finally, we again set $\alpha = 0.5$, using `options(18)=0.5`. We then run the simulated annealing code using `sa_demo('f_p',[0;-2],options)`. The algorithm can be seen to converge to the global minimizer, as plotted below.



14.3

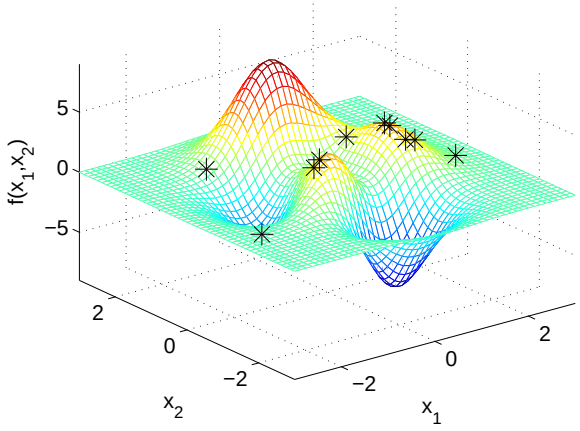
A MATLAB routine for a particle swarm algorithm is:

```
% A particle swarm optimizer
% to find the minimum/maximum of the MATLABs' peaks function
%    D---# of inputs to the function (dimension of problem)
clear
%Parameters
ps=10;
D=2;
ps_lb=-3;
ps_ub=3;
vel_lb=-1;
vel_ub=1;
iteration_n = 50;
range = [-3, 3; -3, 3]; % Range of the input variables
% Plot contours of peaks function
[x, y, z] = peaks;
%pcolor(x,y,z); shading interp; hold on;
```

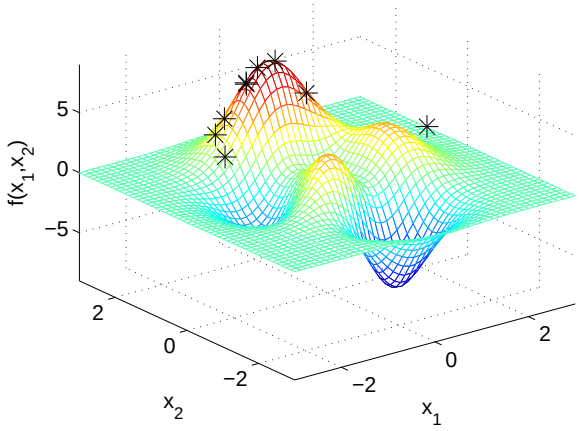
```
   %contour(x, y, z, 20, 'r');
   mesh(x,y,z)
%   hold off;
   %colormap(gray);
   set(gca,'Fontsize',14)
   axis([-3 3 -3 3 -9 9])
%axis square;
    xlabel('x_1','Fontsize',14);
    ylabel('x_2','Fontsize',14);
    zlabel('f(x_1,x_2)','Fontsize',14);
        hold on

upper = zeros(iteration_n, 1);
average = zeros(iteration_n, 1);
lower = zeros(iteration_n, 1);

% initialize population of particles and their velocities at time
% zero,
% format of pos= (particle#, dimension)
% construct random population positions bounded by VR
% need to bound positions

ps_pos=ps_lb + (ps_ub-ps_lb).*rand(ps,D);

% need to bound velocities between -mv,mv

ps_vel=vel_lb + (vel_ub-vel_lb).*rand(ps,D);

% initial pbest positions

p_best = ps_pos;

% returns column of cost values (1 for each particle)

f1='3*(1-ps_pos(i,1))^2*exp(-ps_pos(i,1)^2-(ps_pos(i,2)+1)^2)';
f2='-10*(ps_pos(i,1)/5-ps_pos(i,1)^3-ps_pos(i,2)^5)*exp(-ps_pos(i,1)^2-ps_pos(i,2)^2)';
f3='-(1/3)*exp(-(ps_pos(i,1)+1)^2-ps_pos(i,2)^2)';

p_best_fit=zeros(ps,1);
for i=1:ps
     g1(i)=3*(1-ps_pos(i,1))^2*exp(-ps_pos(i,1)^2-(ps_pos(i,2)+1)^2);
     g2(i)=-10*(ps_pos(i,1)/5-ps_pos(i,1)^3-ps_pos(i,2)^5)*exp(-ps_pos(i,1)^2-ps_pos(i,2)^2);
     g3(i)=-(1/3)*exp(-(ps_pos(i,1)+1)^2-ps_pos(i,2)^2);
     p_best_fit(i)=g1(i)+g2(i)+g3(i);
end
p_best_fit;
hand_p3=plot3(ps_pos(:,1),ps_pos(:,2),p_best_fit','*k','markersize',15,'erase','xor');

% initial g_best

[g_best_val,g_best_idx] = max(p_best_fit);
%[g_best_val,g_best_idx] = min(p_best_fit); this is to minimize
g_best=ps_pos(g_best_idx,:);

% get new velocities, positions (this is the heart of the PSO
% algorithm)
for k=1:iteration_n
      for count=1:ps
   ps_vel(count,:) = 0.729*ps_vel(count,:)...              % prev vel
```

```
        +1.494*rand*(p_best(count,:)-ps_pos(count,:))...   % independent
        +1.494*rand*(g_best-ps_pos(count,:));              % social
      end
   ps_vel;
% update new position
   ps_pos = ps_pos + ps_vel;

%update p_best
for i=1:ps
   g1(i)=3*(1-ps_pos(i,1))^2*exp(-ps_pos(i,1)^2-(ps_pos(i,2)+1)^2);
   g2(i)=-10*(ps_pos(i,1)/5-ps_pos(i,1)^3-ps_pos(i,2)^5)*exp(-ps_pos(i,1)^2-ps_pos(i,2)^2);
   g3(i)=-(1/3)*exp(-(ps_pos(i,1)+1)^2-ps_pos(i,2)^2);
   ps_current_fit(i)=g1(i)+g2(i)+g3(i);

   if ps_current_fit(i)>p_best_fit(i)
       p_best_fit(i)=ps_current_fit(i);
       p_best(i,:)=ps_pos(i,:);
   end
end
p_best_fit;
%update g_best
[g_best_val,g_best_idx] = max(p_best_fit);
g_best=ps_pos(g_best_idx,:);

% Fill objective function vectors
  upper(k) = max(p_best_fit);
  average(k) = mean(p_best_fit);
lower(k) = min(p_best_fit);

set(hand_p3,'xdata',ps_pos(:,1),'ydata',ps_pos(:,2),'zdata',ps_current_fit');
drawnow
pause
end
g_best
g_best_val
    figure;
x = 1:iteration_n;
plot(x, upper, 'o', x, average, 'x', x, lower, '*');
hold on;
plot(x, [upper average lower]);
hold off;
legend('Best', 'Average', 'Poorest');
xlabel('Iterations'); ylabel('Objective function value');
```

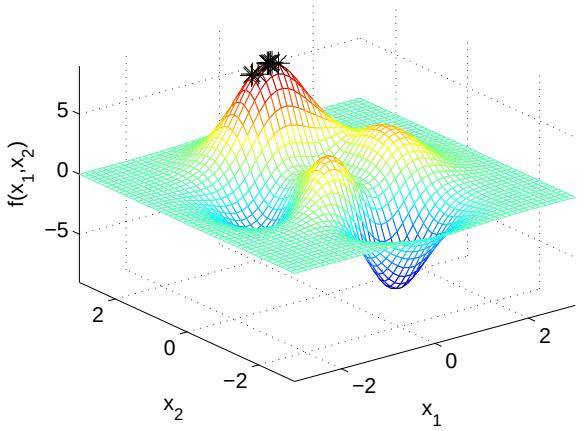
When we run the MATLAB code above, we obtain a plot of the initial set of particles, as shown below.



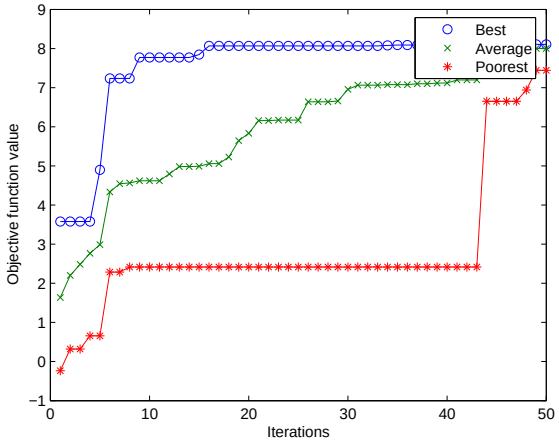
Then, after 30 iterations, we obtain:



Finally, after 50 iterations, we obtain:



A plot of the objective function values (best, average, and poorest) is shown below.



14.4

a. Expanding the right hand side of the second expression gives the desired result.

b. Applying the algorithm, we get a binary representation of 11111001011, i.e.,

$$1995 = 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^3 + 2^1 + 2^0.$$

c. Applying the algorithm, we get a binary representation of 0.1011101, i.e.,

$$0.7265625 = 2^{-1} + 2^{-3} + 2^{-4} + 2^{-5} + 2^{-7}.$$

d. We have $19 = 2^4 + 2^1 + 2^0$, i.e., the binary representation for 19 is 10011. For the fractional part, we need at least 7 bits to keep at least the same accuracy. We have $0.95 = 2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-7} + \cdots$, i.e., the binary representation is $0.1111001\cdots$. Therefore, the binary representation of 19.95 with at least the same degree of accuracy is 10011.1111001.

14.5

It suffices to prove the result for the case where only one symbol is swapped, since the general case is obtained by repeating the argument. We have two scenarios. First, suppose the symbol swapped is at a position corresponding to a don't care symbol in H. Clearly, after the swap, both chromosomes will still be in H. Second, suppose the symbol swapped is at a position corresponding to a fixed symbol in H. Since both chromosomes are in H, their symbols at that position must be identical. Hence, the swap does not change the chromosomes. This completes the proof.

14.6

Consider a given chromosome in $M(k) \bigcap H$. The probability that it is chosen for crossover is q_c. If neither of its offsprings is in H, then at least one of the crossover points must be between the corresponding first and last fixed symbols of H. The probability of this is $1 - (1 - \delta(H)/(L-1))^2$. To see this, note that the probability that each crossover point is not between the corresponding first and last fixed symbols is $1 - \delta(H)/(L-1)$, and thus the probability that both crossover points are not between the corresponding first and last fixed symbols of H is $(1 - \delta(H)/(L-1))^2$. Hence, the probability that the given chromosome is chosen for crossover and neither of its offsprings is in H is bounded above by

$$q_c \left(1 - \left(1 - \frac{\delta(H)}{L-1}\right)^2\right).$$

14.7

As for two-point crossover, the n-point crossover operation is a composition of n one-point crossover operations (i.e., n one-point crossover operations in succession). The required result for this case is as follows.

Lemma:

Given a chromosome in $M(k) \bigcap H$, the probability that it is chosen for crossover and neither of its offsprings is in H is bounded above by

$$q_c \left(1 - \left(1 - \frac{\delta(H)}{L-1}\right)^n\right).$$

□

For the proof, proceed as in the solution of Exercise 14.6, replacing 2 by n.

14.8

```
function M=roulette_wheel(fitness);
%function M=roulette_wheel(fitness)
%fitness = vector of fitness values of chromosomes in population
%M = vector of indices indicating which chromosome in the
%    given population should appear in the mating pool

fitness = fitness - min(fitness); % to keep the fitness positive
if sum(fitness) == 0,
  disp('Population has identical chromosomes -- STOP');
  break;
else
  fitness = fitness/sum(fitness);
end
cum_fitness = cumsum(fitness);
for i = 1:length(fitness),
  tmp = find(cum_fitness-rand>0);
  M(i) = tmp(1);
end
```

14.9

```
% parent1, parent2 = two binary parent chromosomes (row vectors)

L = length(parent1);
crossover_pt = ceil(rand*(L-1));
offspring1 = [parent1(1:crossover_pt) parent2(crossover_pt+1:L)];
offspring2 = [parent2(1:crossover_pt) parent1(crossover_pt+1:L)];
```

14.10

```
% mating_pool = matrix of 0-1 elements; each row represents a chromosome
% p_m = probability of mutation
N = size(mating_pool,1);
L = size(mating_pool,2);
mutation_points = rand(N,L) < p_m;
new_population = xor(mating_pool,mutation_points);
```

14.11

A MATLAB routine for a genetic algorithm with binary encoding is:

```
function [winner,bestfitness] = ga(L,N,fit_func,options)
% function winner = GA(L,N,fit_func)
% Function call: GA(L,N,'f')
% L = length of chromosomes
% N = population size (must be an even number)
% f = name of fitness value function
%
%Options:
%print = options(1);
%selection = options(5);
%max_iter=options(14);
%p_c = options(18);
%p_m = p_c/100;
%
%Selection:
% options(5) = 0 for roulette wheel, 1 for tournament

clf;

if nargin ~= 4
  options = [];
  if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=3*N;
  end
else
  options(14)=3*N;
end
if length(options) < 18
  options(18)=0.75; %optional crossover rate
end

%format compact;
%format short e;

options = foptions(options);
print = options(1);
selection = options(5);
max_iter=options(14);
p_c = options(18);
p_m = p_c/100;

P = rand(N,L)>0.5;
bestvaluesofar = 0;

%Initial evaluation
for i = 1:N,
  fitness(i) = feval(fit_func,P(i,:));
end
[bestvalue,best] = max(fitness);
if bestvalue > bestvaluesofar,
  bestsofar = P(best,:);
  bestvaluesofar = bestvalue;
```

```
end

for k = 1:max_iter,

  %Selection
  fitness = fitness - min(fitness); % to keep the fitness positive
  if sum(fitness) == 0,
    disp('Population has identical chromosomes -- STOP');
    disp('Number of iterations:');
    disp(k);
    for i = k:max_iter,
      upper(i)=upper(i-1);
      average(i)=average(i-1);
      lower(i)=lower(i-1);
    end
    break;
  else
    fitness = fitness/sum(fitness);
  end
  if selection == 0,
    %roulette-wheel
    cum_fitness = cumsum(fitness);
    for i = 1:N,
      tmp = find(cum_fitness-rand>0);
      m(i) = tmp(1);
    end
  else
    %tournament
    for i = 1:N,
      fighter1=ceil(rand*N);
      fighter2=ceil(rand*N);
      if fitness(fighter1)>fitness(fighter2),
        m(i) = fighter1;
      else
        m(i) = fighter2;
      end
    end
  end
  M = zeros(N,L);
  for i = 1:N,
    M(i,:) = P(m(i),:);
  end

  %Crossover
  Mnew = M;
  for i = 1:N/2
    ind1 = ceil(rand*N);
    ind2 = ceil(rand*N);
    parent1 = M(ind1,:);
    parent2 = M(ind2,:);
    if rand < p_c
      crossover_pt = ceil(rand*(L-1));
      offspring1 = [parent1(1:crossover_pt) parent2(crossover_pt+1:L)];
      offspring2 = [parent2(1:crossover_pt) parent1(crossover_pt+1:L)];
      Mnew(ind1,:) = offspring1;
      Mnew(ind2,:) = offspring2;
    end
  end
```

```
  %Mutation
  mutation_points = rand(N,L) < p_m;
  P = xor(Mnew,mutation_points);

  %Evaluation
  for i = 1:N,
    fitness(i) = feval(fit_func,P(i,:));
  end
  [bestvalue,best] = max(fitness);
  if bestvalue > bestvaluesofar,
    bestsofar = P(best,:);
    bestvaluesofar = bestvalue;
  end

upper(k) = bestvalue;
average(k) = mean(fitness);
lower(k) = min(fitness);

end %for

if k == max_iter,
  disp('Algorithm terminated after maximum number of iterations:');
  disp(max_iter);
end

winner = bestsofar;
bestfitness = bestvaluesofar;

if print,
  iter = [1:max_iter]';
  plot(iter,upper,'o:',iter,average,'x-',iter,lower,'*--');
  legend('Best', 'Average', 'Worst');
  xlabel('Generations','Fontsize',14);
  ylabel('Objective Function Value','Fontsize',14);
  set(gca,'Fontsize',14);
  hold off;
end
```

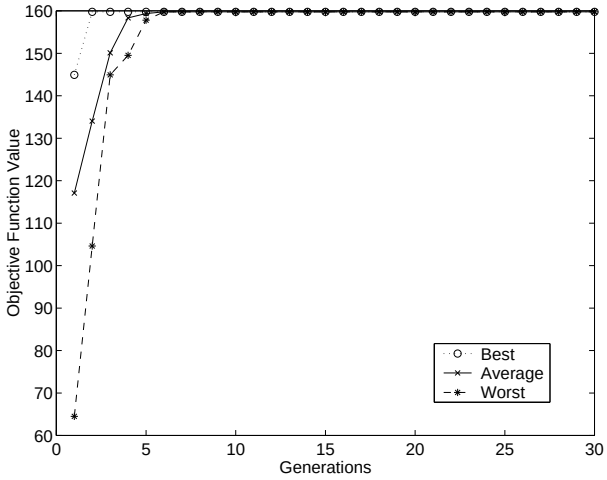
a. To run the routine, we create the following M-files.

```
function dec = bin2dec(bin,range);
%function dec = bin2dec(bin,range);
%Function to convert from binary (bin) to decimal (dec) in a given range

index = polyval(bin,2);
dec = index*((range(2)-range(1))/(2^length(bin)-1)) + range(1);

function y=f_manymax(x);

y=-15*(sin(2*x))^2-(x-2)^2+160;

function y=fit_func1(binchrom);
%1-D fitness function

f='f_manymax';
range=[-10,10];
x=bin2dec(binchrom,range);
y=feval(f,x);
```

We use the following script to run the algorithm:

```
clear;
options(1)=1;
[x,y]=ga(8,10,'fit_func1',options);
f='f_manymax';
range=[-10,10];
disp('GA Solution:');
disp(bin2dec(x,range));
disp('Objective function value:');
disp(y);
```

Running the above algorithm, we obtain a solution of $x^* = 1.6078$, and an objective function value of 159.7640. The figure below shows a plot of the best, average, and worst solution from each generation of the population.



b. To run the routine, we create the following M-files (we also use the routine `bin2dec` from part a.

```
function y=f_peaks(x);

y=3*(1-x(1)).^2.*exp(-(x(1).^2)-(x(2)+1).^2) -
10.*(x(1)/5-x(1).^3-x(2).^5).*exp
(-x(1).^2-x(2).^2) - exp(-(x(1)+1).^2-x(2).^2)/3;
```

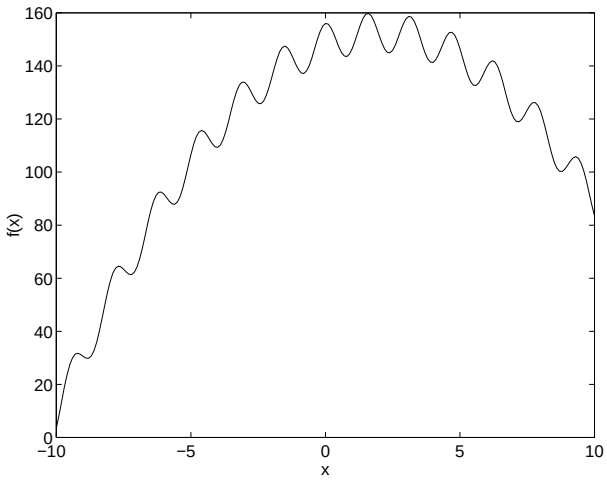
```
function y=fit_func2(binchrom);
%2-D fitness function

f='f_peaks';
xrange=[-3,3];
yrange=[-3,3];
L=length(binchrom);
x1=bin2dec(binchrom(1:L/2),xrange);
x2=bin2dec(binchrom(L/2+1:L),yrange);
y=feval(f,[x1,x2]);
```

We use the following script to run the algorithm:

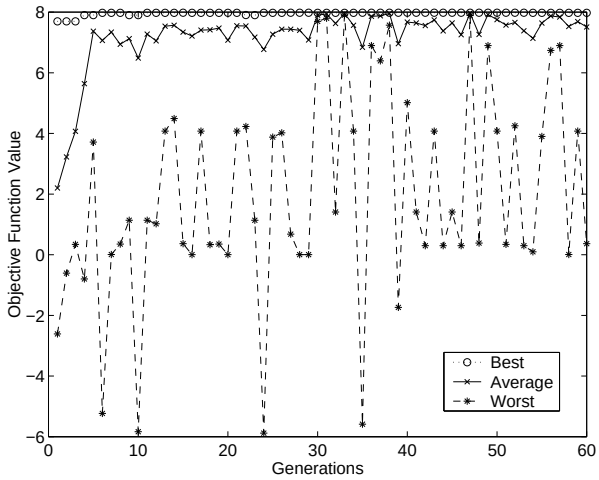
```
clear;
options(1)=1;
[x,y]=ga(16,20,'fit_func2',options);
f='f_peaks';
xrange=[-3,3];
```

```
yrange=[-3,3];
L=length(x);
x1=bin2dec(x(1:L/2),xrange);
x2=bin2dec(x(L/2+1:L),yrange);
disp('GA Solution:');
disp([x1,x2]);
disp('Objective function value:');
disp(y);
```

A plot of the objective function is shown below.



Running the above algorithm, we obtain a solution of $[-0.0353, 1.4941]^\top$, and an $\boldsymbol{x}^* = [-0.0588, 1.5412]^\top$, and an objective function value of 7.9815. (Compare this solution with that of Example 14.3.) The figure below shows a plot of the best, average, and worst solution from each generation of the population.



14.12

A MATLAB routine for a real-number genetic algorithm:

```
function [winner,bestfitness] = gar(Domain,N,fit_func,options)
% function winner = GAR(Domain,N,fit_func)
% Function call: GAR(Domain,N,'f')
% Domain = search space; e.g., [-2,2;-3,3] for the space [-2,2]x[-3,3]
```

```
%  (number of rows of Domain = dimension of search space)
% N = population size (must be an even number)
% f = name of fitness value function
%
%Options:
%print = options(1);
%selection = options(5);
%max_iter=options(14);
%p_c = options(18);
%p_m = p_c/100;
%
%Selection:
% options(5) = 0 for roulette wheel, 1 for tournament

clf;

if nargin ~= 4
  options = [];
  if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

if length(options) >= 14
  if options(14)==0
    options(14)=3*N;
  end
else
  options(14)=3*N;
end
if length(options) < 18
  options(18)=0.75; %optional crossover rate
end

%format compact;
%format short e;

options = foptions(options);
print = options(1);
selection = options(5);
max_iter=options(14);
p_c = options(18);
p_m = p_c/100;
n = size(Domain,1);
lowb = Domain(:,1)';
upb = Domain(:,2)';
bestvaluesofar = 0;
for i = 1:N,
  P(i,:) = lowb + rand(1,n).*(upb-lowb);
  %Initial evaluation
  fitness(i) = feval(fit_func,P(i,:));
end
[bestvalue,best] = max(fitness);
if bestvalue > bestvaluesofar,
  bestsofar = P(best,:);
  bestvaluesofar = bestvalue;
end
```

```
for k = 1:max_iter,

  %Selection
  fitness = fitness - min(fitness); % to keep the fitness positive
  if sum(fitness) == 0,
    disp('Population has identical chromosomes -- STOP');
    disp('Number of iterations:');
    disp(k);
    for i = k:max_iter,
      upper(i)=upper(i-1);
      average(i)=average(i-1);
      lower(i)=lower(i-1);
    end
    break;
  else
    fitness = fitness/sum(fitness);
  end
  if selection == 0,
    %roulette-wheel
    cum_fitness = cumsum(fitness);
    for i = 1:N,
      tmp = find(cum_fitness-rand>0);
      m(i) = tmp(1);
    end
  else
    %tournament
    for i = 1:N,
      fighter1=ceil(rand*N);
      fighter2=ceil(rand*N);
      if fitness(fighter1)>fitness(fighter2),
        m(i) = fighter1;
      else
        m(i) = fighter2;
      end
    end
  end
  M = zeros(N,n);
  for i = 1:N,
    M(i,:) = P(m(i),:);
  end

  %Crossover
  Mnew = M;
  for i = 1:N/2
    ind1 = ceil(rand*N);
    ind2 = ceil(rand*N);
    parent1 = M(ind1,:);
    parent2 = M(ind2,:);
    if rand < p_c
  a = rand;
      offspring1 = a*parent1+(1-a)*parent2+(rand(1,n)-0.5).*(upb-lowb)/10;
      offspring2 = a*parent2+(1-a)*parent1+(rand(1,n)-0.5).*(upb-lowb)/10;
      %do projection
      for j = 1:n,
        if offspring1(j)<lowb(j),
          offspring1(j)=lowb(j);
        elseif offspring1(j)>upb(j),
          offspring1(j)=upb(j);
        end
```

```
        if offspring2(j)<lowb(j),
          offspring2(j)=lowb(j);
        elseif offspring2(j)>upb(j),
          offspring2(j)=upb(j);
        end
      end
      Mnew(ind1,:) = offspring1;
      Mnew(ind2,:) = offspring2;
    end
  end

  %Mutation
  for i = 1:N,
if rand < p_m,
      a = rand;
  Mnew(i,:) = a*Mnew(i,:) + (1-a)*(lowb + rand(1,n).*(upb-lowb));
    end
  end

  P = Mnew;

  %Evaluation
  for i = 1:N,
    fitness(i) = feval(fit_func,P(i,:));
  end
  [bestvalue,best] = max(fitness);
  if bestvalue > bestvaluesofar,
    bestsofar = P(best,:);
    bestvaluesofar = bestvalue;
  end

upper(k) = bestvalue;
average(k) = mean(fitness);
lower(k) = min(fitness);

end %for

if k == max_iter,
  disp('Algorithm terminated after maximum number of iterations:');
  disp(max_iter);
end

winner = bestsofar;
bestfitness = bestvaluesofar;

if print,
  iter = [1:max_iter]';
  plot(iter,upper,'o:',iter,average,'x-',iter,lower,'*--');
  legend('Best', 'Average', 'Worst');
  xlabel('Generations','Fontsize',14);
  ylabel('Objective Function Value','Fontsize',14);
  set(gca,'Fontsize',14);
  hold off;
end
```

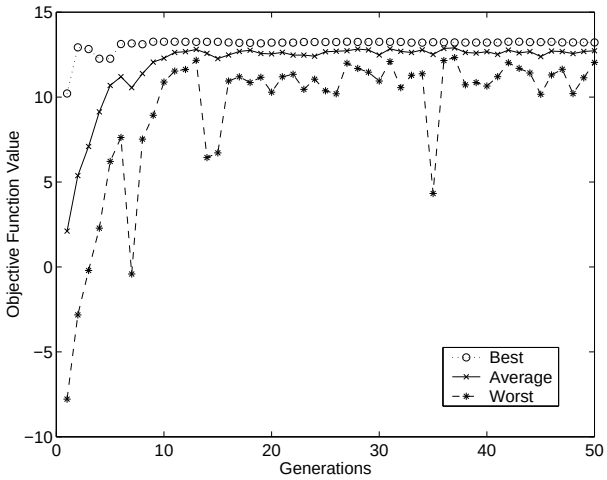
To run the routine, we create the following M-file for the given function.

```
function y=f_wave(x);

y=x(1)*sin(x(1)) + x(2)*sin(5*x(2));
```

We use the following script to run the algorithm:

```
options(1)=1;
options(14)=50;
[x,y]=gar([0,10;4,6],20,'f_wave',options);
disp('GA Solution:');
disp(x);
disp('Objective function value:');
disp(y);
```

Running the above algorithm, we obtain a solution of $\boldsymbol{x}^* = [7.9711, 5.3462]^\top$, and an objective function value of 13.2607. The figure below shows a plot of the best, average, and worst solution from each generation of the population.



Using the MATLAB function `fminunc` (from the Optimization Toolbox), we found the optimal point to be $[7.9787, 5.3482]^\top$, with objective function value 13.2612. We can see that this solution agrees with the solution obtained using our real-number genetic algorithm.

15. Introduction to Linear Programming

15.1

$$\begin{aligned} \text{minimize } & -2x_1 - x_2 \\ \text{subject to } & x_1 + x_3 = 2 \\ & x_1 + x_2 + x_4 = 3 \\ & x_1 + 2x_2 + x_5 = 5 \\ & x_1, \ldots, x_5 \geq 0 \end{aligned}$$

15.2

We have

$$x_2 = ax_1 + bu_1 = a^2 x_0 + abu_0 + bu_1 = a^2 + [ab, b]\boldsymbol{u}$$

where $\boldsymbol{u} = [u_0, u_1]^\top$ is the decision variable. We can write the constraint as $u_i \leq 1$ and $u_i \geq -1$. Hence, the problem is:

$$\begin{aligned} \text{minimize} \quad & a^2 + [ab, b]\boldsymbol{u} \\ \text{subject to} \quad & -1 \leq u_i \leq 1, \ i = 1, 2. \end{aligned}$$

Since a^2 is a constant, we can remove it from the objective function without changing the solution. Introducing slack variables v_1, v_2, v_3, v_4, we obtain the standard form problem

$$\begin{aligned} \text{minimize} \quad & [ab, b]\boldsymbol{u} \\ \text{subject to} \quad & u_0 + v_1 = 1 \\ & -u_0 + v_2 = 1 \\ & u_1 + v_3 = 1 \\ & -u_1 + v_4 = 1. \end{aligned}$$

15.3

Let $x_i^+, x_i^- \geq 0$ be such that $|x_i| = x_i^+ + x_i^-$, $x_i = x_i^+ - x_i^-$. Substituting into the original problem, we have

$$\begin{aligned} \text{minimize} \quad & c_1(x_1^+ + x_1^-) + c_2(x_2^+ + x_2^-) + \cdots + c_n(x_2^+ + x_2^-) \\ \text{subject to} \quad & \boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) = \boldsymbol{b} \\ & \boldsymbol{x}^+, \boldsymbol{x}^- \geq \boldsymbol{0}, \end{aligned}$$

where $\boldsymbol{x}^+ = [x_1^+, \ldots, x_n^+]^\top$ and $\boldsymbol{x}^- = [x_1^-, \ldots, x_n^-]^\top$. Rewriting, we get

$$\begin{aligned} \text{minimize} \quad & [\boldsymbol{c}^\top, \boldsymbol{c}^\top]\boldsymbol{z} \\ \text{subject to} \quad & [\boldsymbol{A}, -\boldsymbol{A}]\boldsymbol{z} = \boldsymbol{b} \\ & \boldsymbol{z} \geq \boldsymbol{0}, \end{aligned}$$

which is an equivalent linear programming problem in standard form.

Note that although the variables x_i^+ and x_i^- in the solution are required to satisfy $x_i^+ x_i^- = 0$, we do not need to explicitly include this in the constraint because any optimal solution to the above transformed problem automatically satisfies the condition $x_i^+ x_i^- = 0$. To see this, suppose we have an optimal solution with both $x_i^+ > 0$ and $x_i^- > 0$. In this case, note that $c_i > 0$ (for otherwise we can add any arbitrary constant to both x_i^+ and x_i^- and still satisfy feasibility, but decrease the objective function value). Then, by subtracting $\min(x_i^+, x_i^-)$ from x_i^+ and x_i^-, we have a new feasible point with lower objective function value, contradicting the optimality assumption. [See also M. A. Dahleh and I. J. Diaz-Bobillo, *Control of Uncertain Systems: A Linear Programming Approach*, Prentice Hall, 1995, pp. 189–190.]

15.4

Not every linear programming problem in standard form has a nonempty feasible set. Example:

$$\begin{aligned} \text{minimize} \quad & x_1 \\ \text{subject to} \quad & -x_1 = 1 \\ & x_1 \geq 0. \end{aligned}$$

Not every linear programming problem in standard form (even assuming a nonempty feasible set) has an optimal solution. Example:

$$\begin{aligned} \text{minimize} \quad & -x_1 \\ \text{subject to} \quad & x_2 = 1 \\ & x_1, x_2 \geq 0. \end{aligned}$$

15.5

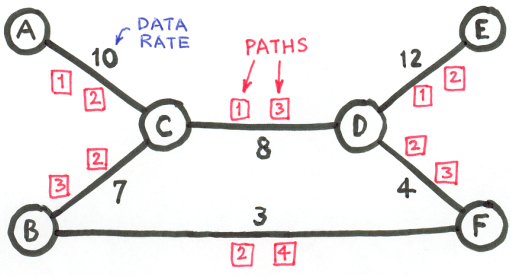
Let x_1 and x_2 represent the number of units to be shipped from A to C and to D, respectively, and x_3 and x_4 represent the number of units to be shipped from B to C and to D, respectively. Then, the given problem can be formulated as the following linear program:

$$\begin{aligned} \text{minimize} \quad & x_1 + 2x_2 + 3x_3 + 4x_4 \\ \text{subject to} \quad & x_1 + x_3 = 50 \\ & x_2 + x_4 = 60 \\ & x_1 + x_2 \leq 70 \\ & x_3 + x_4 \leq 80 \\ & x_1, x_2, x_3, x_4 \geq 0. \end{aligned}$$

Introducing slack variables x_5 and x_6, we have the standard form problem

$$\begin{aligned} \text{minimize} \quad & x_1 + 2x_2 + 3x_3 + 4x_4 \\ \text{subject to} \quad & x_1 + x_3 = 50 \\ & x_2 + x_4 = 60 \\ & x_1 + x_2 + x_5 = 70 \\ & x_3 + x_4 + x_6 = 80 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0. \end{aligned}$$

15.6



We can see that there are two paths from A to E ($ACDE$ and $ACBFDE$), and two paths from B to F ($BCDF$ and BF). Let x_1 and x_2 be the data rates for the two paths from A to E, respectively, and x_3 and x_4 the data rates for the two paths from B to F, respectively. The total revenue is then $2(x_1 + x_2) + 3(x_3 + x_4)$. For each link, we have a data rate constraint on the sum of all x_is passing through that link. For example,

for link BC, there are two paths passing through it, with total data rate $x_2 + x_3$. Hence, the constraint for link BC is $x_2 + x_3 \leq 7$. Hence, the optimization problem is the following linear programming problem:

$$\begin{array}{rl} \text{maximize} & 2(x_1 + x_2) + 3(x_3 + x_4) \\ \text{subject to} & x_1 + x_2 \leq 10 \\ & x_1 + x_2 \leq 12 \\ & x_1 + x_3 \leq 8 \\ & x_2 + x_3 \leq 7 \\ & x_2 + x_3 \leq 4 \\ & x_2 + x_4 \leq 3 \\ & x_1, \ldots, x_4 \geq 0. \end{array}$$

Converting this to standard form:

$$\begin{array}{rl} \text{minimize} & -2(x_1 + x_2) - 3(x_3 + x_4) \\ \text{subject to} & x_1 + x_2 + x_5 = 10 \\ & x_1 + x_2 + x_6 = 12 \\ & x_1 + x_3 + x_7 = 8 \\ & x_2 + x_3 + x_8 = 7 \\ & x_2 + x_3 + x_9 = 4 \\ & x_2 + x_4 + x_{10} = 3 \\ & x_1, \ldots, x_{10} \geq 0. \end{array}$$

15.7

Let $x_i \geq 0$, $i = 1, \ldots, 4$, be the weight in pounds of item i to be used. Then, the total weight is $x_1+x_2+x_3+x_4$. To satisfy the percentage content of fiber, fat, and sugar, and the total weight of 1000, we need

$$\begin{array}{rcl} 3x_1 + 8x_2 + 16x_3 + 4x_4 & = & 10(x_1 + x_2 + x_3 + x_4) \\ 6x_1 + 46x_2 + 9x_3 + 9x_4 & = & 2(x_1 + x_2 + x_3 + x_4) \\ 20x_1 + 5x_2 + 4x_3 + 0x_4 & = & 5(x_1 + x_2 + x_3 + x_4) \\ x_1 + x_2 + x_3 + x_4 & = & 1000 \end{array}$$

The total cost is $2x_1 + 4x_2 + x_3 + 2x_4$. Therefore, the problem is:

$$\begin{array}{rrcl} \text{minimize} & 2x_1 + 4x_2 + x_3 + 2x_4 & & \\ \text{subject to} & -7x_1 - 2x_2 + 6x_3 - 6x_4 & = & 0 \\ & 4x_1 + 44x_2 + 7x_3 + 7x_4 & = & 0 \\ & 15x_1 - x_3 - 5x_4 & = & 0 \\ & x_1 + x_2 + x_3 + x_4 & = & 1000 \\ & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

Alternatively, we could have simply replaced $x_1 + x_2 + x_3 + x_4$ in the first three equality constraints above by 1000, to obtain:

$$\begin{array}{rcl} 3x_1 + 8x_2 + 16x_3 + 4x_4 & = & 10000 \\ 6x_1 + 46x_2 + 9x_3 + 9x_4 & = & 2000 \\ 20x_1 + 5x_2 + 4x_3 + 0x_4 & = & 5000 \\ x_1 + x_2 + x_3 + x_4 & = & 1000. \end{array}$$

Note that the only vector satisfying the above linear equations is $[179, -175, 573, 422]^\top$, which is not feasible. Therefore, the constraint does not have any any feasible points, which means that the problem does not have a solution.

15.8

The objective function is $p_1 + \cdots + p_n$. The constraint for the ith location is: $g_{i,1}p_1 + \cdots + g_{i,n}p_n \geq P$. Hence, the the optimization problem is:

$$\begin{array}{ll} \text{minimize} & p_1 + \cdots + p_n \\ \text{subject to} & g_{i,1}p_1 + \cdots + g_{i,n}p_n \geq P,\ i = 1, \ldots, m \\ & p_1, \ldots, p_n \geq 0. \end{array}$$

By defining the notation $\boldsymbol{G} = [g_{i,j}]$ $(m \times n)$, $\boldsymbol{e}_n = [1, \ldots, 1]^\top$ (with n components), and $\boldsymbol{p} = [p_1, \ldots, p_n]^\top$, we can rewrite the problem as

$$\begin{array}{ll} \text{minimize} & \boldsymbol{e}_n^\top \boldsymbol{p} \\ \text{subject to} & \boldsymbol{G}\boldsymbol{p} \geq P\boldsymbol{e}_m \\ & \boldsymbol{p} \geq \boldsymbol{0}. \end{array}$$

15.9

It is easy to check (using MATLAB, for example) that the matrix

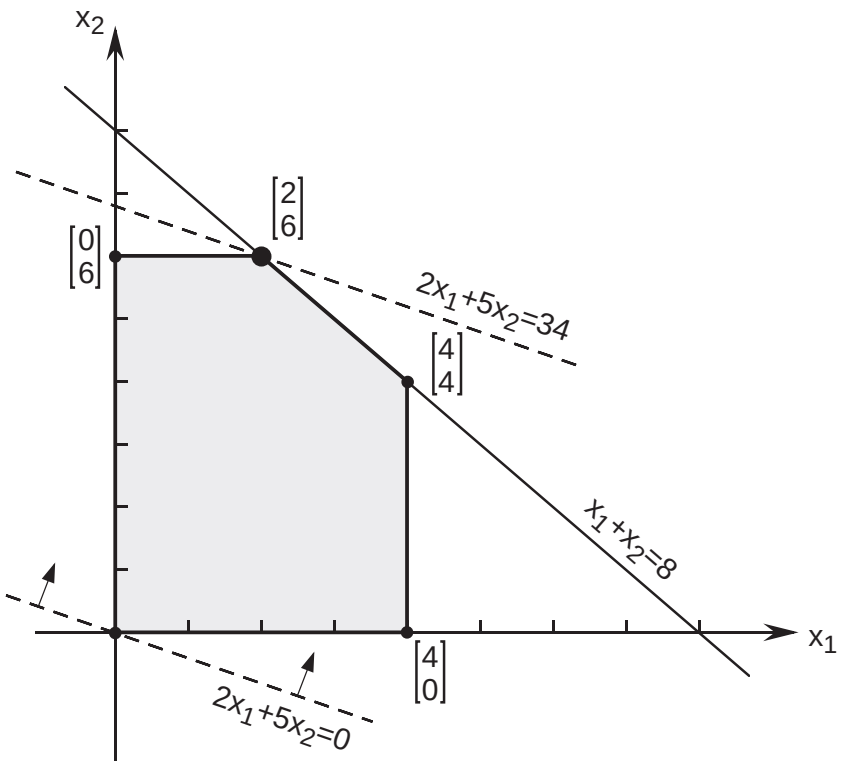
$$\boldsymbol{A} = \begin{bmatrix} 2 & -1 & 2 & -1 & 3 \\ 1 & 2 & 3 & 1 & 0 \\ 1 & 0 & -2 & 0 & -5 \end{bmatrix}$$

is of full rank (i.e., rank $\boldsymbol{A} = 3$). Therefore, the system has basic solutions. To find the basic solutions, we first select bases. Each basis consists of three linearly independent columns of $\boldsymbol{A}$. These columns correspond to basic variables of the basic solution. The remaining variables are nonbasic and are set to 0. The matrix $\boldsymbol{A}$ has 5 columns; therefore, we have $\binom{5}{3} = 10$ possible candidate basic solutions (corresponding to the 10 combinations of 3 columns out of 5). It turns out that all 10 combinations of 3 columns of $\boldsymbol{A}$ are linearly independent. Therefore, we have 10 basic solutions. These are tabulated as follows:

Columns	Basic Solutions
$1, 2, 3$	$[-4/17, -80/17, 83/17, 0, 0]^\top$
$1, 2, 4$	$[-10, 49, 0, -83, 0]^\top$
$1, 2, 5$	$[105/31, 25/31, 0, 0, 83/31]^\top$
$1, 3, 4$	$[-12/11, 0, 49/11, -80/11, 0]^\top$
$1, 3, 5$	$[100/35, 0, 25/35, 0, 80/35]^\top$
$1, 4, 5$	$[65/18, 0, 0, 25/18, 49/18]^\top$
$2, 3, 4$	$[0, -6, 5, 2, 0]^\top$
$2, 3, 5$	$[0, -100/23, 105/23, 0, 4/23]^\top$
$2, 4, 5$	$[0, 13, 0, -21, 2]^\top$
$3, 4, 5$	$[0, 0, 65/19, -100/19, 12/19]^\top$

15.10

In the figure below, the shaded region corresponds to the feasible set. We then translate the line $2x_1 + 5x_2 = 0$ across the shaded region until the line just touches the region at one point, and the line is as far as possible from the origin. The point of contact is the solution to the problem. In this case, the solution is $[2, 6]^\top$, and the corresponding cost is 34.



15.11

We use the following MATLAB commands:

```
>> f=[0,-10,0,-6,-20];
>> A=[1,-1,-1,0,0; 0,0,1,-1,-1];
>> b=[0;0];
>> vlb=zeros(5,1);
>> vub=[4;3;3;2;2];
>> x0=zeros(5,1);
>> neqcstr=2;
>> x=linprog(f,A,b,vlb,vub,x0,neqcstr)

x =

    4.0000
    2.0000
    2.0000
    0.0000
    2.0000
```

The solution is $[4, 2, 2, 0, 2]^\top$.

16. The Simplex Method

16.1

a. Performing a sequence of elementary row operations, we obtain

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & -1 & 3 & 2 \\ 2 & -1 & 3 & 0 & 1 \\ 3 & 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 & -1 & 3 & 2 \\ 0 & -5 & 5 & -6 & -3 \\ 0 & -5 & 5 & -6 & -3 \\ 0 & 0 & 4 & -2 & -1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 & -1 & 3 & 2 \\ 0 & -5 & 5 & -6 & -3 \\ 0 & 0 & 4 & -2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \boldsymbol{B}.$$

Because elementary row operations do not change the rank of a matrix, $\operatorname{rank} \boldsymbol{A} = \operatorname{rank} \boldsymbol{B}$. Therefore $\operatorname{rank} \boldsymbol{A} = 3$.

b. Performing a sequence of elementary row operations, we obtain

$$\boldsymbol{A} = \begin{bmatrix} 1 & \gamma & -1 & 2 \\ 2 & -1 & \gamma & 5 \\ 1 & 10 & -6 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 10 & -6 & 1 \\ 1 & \gamma & -1 & 2 \\ 2 & -1 & \gamma & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 10 & -6 & 1 \\ 0 & \gamma - 10 & 5 & 1 \\ 0 & -21 & \gamma + 12 & 3 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & 1 & 10 & -6 \\ 0 & 1 & \gamma - 10 & 5 \\ 0 & 3 & -21 & \gamma + 12 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 10 & -6 \\ 0 & 1 & \gamma - 10 & 5 \\ 0 & 0 & -3(\gamma - 3) & \gamma - 3 \end{bmatrix} = \boldsymbol{B}$$

Because elementary row operations do not change the rank of a matrix, $\operatorname{rank} \boldsymbol{A} = \operatorname{rank} \boldsymbol{B}$. Therefore $\operatorname{rank} \boldsymbol{A} = 3$ if $\gamma \neq 3$ and $\operatorname{rank} \boldsymbol{A} = 2$ if $\gamma = 3$.

16.2

a.

$$\boldsymbol{A} = \begin{bmatrix} 3 & 1 & 0 & 1 \\ 6 & 2 & 1 & 1 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 4 \\ 5 \end{bmatrix}, \qquad \boldsymbol{c} = [2, -1, -1, 0]\,.$$

b. Pivoting the problem tableau about the elements $(1, 4)$ and $(2, 3)$, we obtain

$$\begin{array}{ccccc} 3 & 1 & 0 & 1 & 4 \\ 3 & 1 & 1 & 0 & 1 \\ 5 & 0 & 0 & 0 & 1 \end{array}$$

c. Basic feasible solution: $\boldsymbol{x} = [0, 0, 1, 4]^\top$, $\boldsymbol{c}^\top \boldsymbol{x} = -1$.

d. $[r_1, r_2, r_3, r_4] = [5, 0, 0, 0]$.

e. Since the reduced cost coefficients are all ≥ 0, the basic feasible solution in part c is optimal.

f. The original problem does indeed have a feasible solution, because the artificial problem has an optimal feasible solution with objective function value 0, as shown in the final phase I tableau.

g. Extract the submatrices corresponding to $\boldsymbol{A}$ and $\boldsymbol{b}$, append the last row $[\boldsymbol{c}^\top, 0]$, and pivot about the $(2, 1)$th element to obtain

$$\begin{array}{ccccc} 0 & 0 & -1 & 1 & 3 \\ 1 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & -5/3 & -5/3 & 0 & -2/3 \end{array}$$

16.3

The problem in standard form is:

$$\begin{array}{rl} \text{minimize} & -x_1 - x_2 - 3x_3 \\ \text{subject to} & x_1 + x_3 = 1 \\ & x_2 + x_3 = 2 \\ & x_1, x_2, x_3 \geq 0. \end{array}$$

We form the tableau for the problem:

$$\begin{array}{cccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ -1 & -1 & -3 & 0 \end{array}$$

Performing necessary row operations, we obtain a tableau in canonical form:

$$\begin{array}{cccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & -1 & 3 \end{array}$$

We pivot about the $(1,3)$th element to get:

$$\begin{array}{cccc} 1 & 0 & 1 & 1 \\ -1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 4 \end{array}$$

The reduced cost coefficients are all nonnegative. Hence, the current basic feasible solution is optimal: $[0,1,1]^\top$. The optimal cost is 4.

16.4

The problem in standard form is:

$$\begin{array}{rl} \text{minimize} & -2x_1 - x_2 \\ \text{subject to} & x_1 + x_3 = 5 \\ & x_2 + x_4 = 7 \\ & x_1 + x_2 + x_5 = 9 \\ & x_1, \ldots, x_5 \geq 0. \end{array}$$

We form the tableau for the problem:

$$\begin{array}{cccccc} 1 & 0 & 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 1 & 0 & 7 \\ 1 & 1 & 0 & 0 & 1 & 9 \\ -2 & -1 & 0 & 0 & 0 & 0 \end{array}$$

The above tableau is already in canonical form, and therefore we can proceed with the simplex procedure. We first pivot about the $(1,1)$th element, to get

$$\begin{array}{cccccc} 1 & 0 & 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 1 & 0 & 7 \\ 0 & 1 & -1 & 0 & 1 & 4 \\ 0 & -1 & 2 & 0 & 0 & 10 \end{array}$$

Next, we pivot about the $(3,2)$th element to get

$$\begin{array}{cccccc} 1 & 0 & 1 & 0 & 0 & 5 \\ 0 & 0 & 1 & 1 & -1 & 3 \\ 0 & 1 & -1 & 0 & 1 & 4 \\ 0 & 0 & 1 & 0 & 1 & 14 \end{array}$$

The reduced cost coefficients are all nonnegative. Hence, the optimal solution to the problem in standard form is $[5,4,0,3,0]^\top$. The corresponding optimal cost is -14.

16.5

a. Let $\boldsymbol{B} = [\boldsymbol{a}_2, \boldsymbol{a}_1]$ represent the first two columns of $\boldsymbol{A}$ ordered according to the basis corresponding to the given canonical tableau, and $\boldsymbol{D}$ the second two columns. Then,

$$\boldsymbol{B}^{-1}\boldsymbol{D} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix},$$

Hence,

$$\boldsymbol{B} = \boldsymbol{D}\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 3/2 & -1/2 \\ -2 & 1 \end{bmatrix}.$$

Hence,

$$\boldsymbol{A} = \begin{bmatrix} -1/2 & 3/2 & 0 & 1 \\ 1 & -2 & 1 & 0 \end{bmatrix}$$

An alternative approach is to realize that the canonical tableau is obtained from the problem tableau via elementary row operations. Therefore, we can obtain the entries of $\boldsymbol{A}$ from the 2×4 upper-left submatrix of the canonical tableau via elementary row operations also. Specifically, start with

$$\begin{bmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 3 & 4 \end{bmatrix}$$

and then do two pivoting operations, one about $(1, 4)$ and the other about $(2, 3)$.

b. The right-half of $\boldsymbol{c}$ is given by

$$\boldsymbol{c}_D^\top = \boldsymbol{r}_D^\top + \boldsymbol{c}_B^\top \boldsymbol{B}^{-1}\boldsymbol{D} = [-1, 1] + [7, 8]\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = [-1, 1] + [31, 46] = [30, 47].$$

So $\boldsymbol{c} = [8, 7, 30, 47]^\top$.

c. First we calculate $\boldsymbol{B}^{-1}\boldsymbol{b}$, giving us the basic variable values:

$$\boldsymbol{B}^{-1}\boldsymbol{b} = \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix}\begin{bmatrix} 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 16 \\ 38 \end{bmatrix}.$$

Hence, the BFS is $[38, 16, 0, 0]^\top$.

d. The first two entries are 16 and 38, respectively. The last component is $-\boldsymbol{c}_B^\top \boldsymbol{B}^{-1}\boldsymbol{b} = -(7(16) + 8(38)) = -416$. Hence, the last column is the vector $[16, 38, -416]^\top$.

16.6

The columns in the constraint matrix $\boldsymbol{A}$ corresponding to x_i^+ and x_i^- are linearly dependent. Hence they cannot both enter a basis at the same time. This means that only one variable, x_i^+ or x_i^-, can assume a nonnegative value; the nonbasic variable is necessarily zero.

16.7

a. From the given information, we have the 4×6 canonical tableau

$$\begin{bmatrix} 1 & * & 0 & 0 & 1/2 & 1 \\ 0 & * & 1 & 0 & 0 & 2 \\ 0 & * & 0 & 1 & 0 & 3 \\ 0 & 1 & 0 & 0 & -1 & -6 \end{bmatrix}$$

Explanations:

- The given vector $\boldsymbol{x}$ indicates that $\boldsymbol{A}$ is 3×5.

- In the above tableau, we assume that the basis is $[\boldsymbol{a}_1, \boldsymbol{a}_3, \boldsymbol{a}_4]$, in this order. Other permutations of orders will result in interchanging rows among the first three rows of the tableau.
- The fifth column represents the coordinates of $\boldsymbol{a}_5$ with respect to the basis $[\boldsymbol{a}_1, \boldsymbol{a}_3, \boldsymbol{a}_4]$. Because $[-2, 0, 0, 0, 4]^\top$ lies in the nullspace of $\boldsymbol{A}$, we deduce that $-2\boldsymbol{a}_1 + 4\boldsymbol{a}_5 = \boldsymbol{0}$, which can be rewritten as $\boldsymbol{a}_5 = (1/2)\boldsymbol{a}_1 + 0\boldsymbol{a}_3 + 0\boldsymbol{a}_4$, and hence the coordinate vector is $[1/2, 0, 0]^\top$.

b. Let $\boldsymbol{d}_0 = [-2, 0, 0, 0, 4]^\top$. Then, $\boldsymbol{A}\boldsymbol{d}_0 = \boldsymbol{0}$. Therefore, the vector $\boldsymbol{x}' = \boldsymbol{x} + \varepsilon\boldsymbol{d}_0$ also satisfies $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. Now, $\boldsymbol{x}' = \boldsymbol{x} + \varepsilon\boldsymbol{d}_0 = [1 - 2\varepsilon, 0, 2, 3, 4\varepsilon]^\top$. For $\boldsymbol{x}'$ to be feasible, we must have $\varepsilon \leq 1/2$. Moreover, the objective function value of $\boldsymbol{x}'$ is $\boldsymbol{c}^\top\boldsymbol{x}' = z_0 + r_5 x_5' = 6 - 4\varepsilon$, where z_0 is the objective function value of $\boldsymbol{x}$. So, if we pick any $\varepsilon \in (0, 1/2]$, then $\boldsymbol{x}'$ will be a feasible solution with objective function value strictly less than 6. For example, with $\varepsilon = 1/2$, $\boldsymbol{x}' = [0, 0, 2, 3, 2]^\top$ is such a point. (We could also have obtained this solution by pivoting about the element $(1, 5)$ in the tableau of part a.)

16.8

a. The BFS is $[6, 0, 7, 5, 0]^\top$, with objective function value -8.

b. $\boldsymbol{r} = [0, 4, 0, 0, -4]^\top$.

c. Yes, because the 5th column has all negative entries.

d. We pivot about the element $(3, 2)$. The new canonical tableau is:

$$\begin{bmatrix} 0 & 0 & -1/3 & 1 & 0 & 8/3 \\ 1 & 0 & -2/3 & 0 & 0 & 4/3 \\ 0 & 1 & 1/3 & 0 & -1 & 7/3 \\ 0 & 0 & -4/3 & 0 & 0 & -4/3 \end{bmatrix}$$

e. First note that based on the 5th column, the following point is feasible:

$$\boldsymbol{x} = \begin{bmatrix} 6 \\ 0 \\ 7 \\ 5 \\ 0 \end{bmatrix} + \epsilon \begin{bmatrix} 2 \\ 0 \\ 3 \\ 1 \\ 1 \end{bmatrix}.$$

Note that $x_5 = \epsilon$. Now, any solution of the form $\boldsymbol{x} = [*, 0, *, *, \epsilon]^\top$ has an objective function value given by

$$z = z_0 + r_5\epsilon$$

where $z_0 = -8$ and $r_5 = -4$ (from parts a and b). If $z = -100$, then $\epsilon = 23$. Hence, the following point has objective function value $z = -100$:

$$\boldsymbol{x} = \begin{bmatrix} 6 \\ 0 \\ 7 \\ 5 \\ 0 \end{bmatrix} + 23 \begin{bmatrix} 2 \\ 0 \\ 3 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 52 \\ 0 \\ 76 \\ 28 \\ 23 \end{bmatrix}.$$

f. The entries of the 2nd column of the given canonical tableau are the coordinates of $\boldsymbol{a}_2$ with respect to the basis $\{\boldsymbol{a}_4, \boldsymbol{a}_1, \boldsymbol{a}_3\}$. Therefore,

$$\boldsymbol{a}_2 = \boldsymbol{a}_4 + 2\boldsymbol{a}_1 + 3\boldsymbol{a}_3.$$

Therefore, the vector $[2, -1, 3, 1, 0]^\top$ lies in the nullspace of $\boldsymbol{A}$. Similarly, using the entries of the 5th column, we deduce that $[-2, 0, -3, -1, -1]^\top$ also lies in the nullspace of $\boldsymbol{A}$. These two vectors are linearly independent. Because $\boldsymbol{A}$ has rank 3, the dimension of the nullspace of $\boldsymbol{A}$ is 2. Hence, these two vectors form a basis for the nullspace of $\boldsymbol{A}$.

16.9

a. We can convert the problem to standard form by multiplying the objective function by -1 and introducing a surplus variable x_3. We obtain:

$$\begin{array}{rl} \text{minimize} & x_1 + 2x_2 \\ \text{subject to} & x_2 - x_3 = 1 \\ & x_1, x_2, x_3 \geq 0. \end{array}$$

Note that we do not need to deal with the absence of the constraint $x_2 \geq 0$ in the original problem, since $x_2 \geq 1$ implies that $x_2 \geq 0$ also. Had we used the rule of writing $x_2 = u - v$ with $u, v \geq 0$, we obtain the standard form problem:

$$\begin{array}{rl} \text{minimize} & x_1 + 2u - 2v \\ \text{subject to} & u - v - x_3 = 1 \\ & x_1, u, v, x_3 \geq 0. \end{array}$$

b. For phase I, we set up the artificial problem tableau as:

$$\begin{array}{ccccc} 0 & 1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{array}$$

Pivoting about element $(1, 4)$, we obtain the canonical tableau:

$$\begin{array}{ccccc} 0 & 1 & -1 & 1 & 1 \\ 0 & -1 & 1 & 0 & -1 \end{array}$$

Pivoting now about element $(1, 2)$, we obtain the next canonical tableau:

$$\begin{array}{ccccc} 0 & 1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{array}$$

Hence, phase I terminates, and we use x_2 as our initial basic variable for phase II.

For phase II, we set up the problem tableau as:

$$\begin{array}{cccc} 0 & 1 & -1 & 1 \\ 1 & 2 & 0 & 0 \end{array}$$

Pivoting about element $(1, 2)$, we obtain

$$\begin{array}{cccc} 0 & 1 & -1 & 1 \\ 1 & 0 & 2 & -2 \end{array}$$

Hence, the BFS $[0, 1, 0]^\top$ is optimal, with objective function value 2. Therefore, the optimal solution to the original problem is $[0, 1]^\top$ with objective function value -2.

16.10

a. $[1, 0]^\top$

b. $\begin{bmatrix} 1 & -1 & 1 \end{bmatrix}$

Note that the answer is not $\begin{bmatrix} 1 & -1 & 1 \\ 0 & -1 & 1 \end{bmatrix}$, which is the canonical tableau.

c. We choose $q = 2$ because the only negative RCC value is r_2. However, $y_{1,2} < 0$. Therefore, the simplex algorithm terminates with the condition that the problem is unbounded.

d. Any vector of the form $[x_1, x_1 - 1]^\top$, $x_1 \geq 1$, is feasible. Therefore the first component can take arbitrarily large (positive) values. Hence, the objective function, which is $-x_1$, can take arbitrarily negative values.

16.11

The problem in standard form is:

$$\begin{array}{rl} \text{minimize} & x_1 + x_2 \\ \text{subject to} & x_1 + 2x_2 - x_3 = 3 \\ & 2x_1 + x_2 - x_4 = 3 \\ & x_1, x_2, x_3, x_4 \geq 0. \end{array}$$

We will use x_1 and x_2 as initial basic variables. Therefore, Phase I is not needed, and we immediately proceed with Phase II. The tableau for the problem is:

	$\boldsymbol{a}_1$	$\boldsymbol{a}_2$	$\boldsymbol{a}_3$	$\boldsymbol{a}_4$	$\boldsymbol{b}$
	1	2	-1	0	3
	2	1	0	-1	3
$\boldsymbol{c}^\top$	1	1	0	0	0

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= \boldsymbol{c}_B^\top \boldsymbol{B}^{-1} = [1,1] \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}^{-1} = [1/3, 1/3], \\ \boldsymbol{r}_D^\top &= \boldsymbol{c}_D^\top - \boldsymbol{\lambda}^\top \boldsymbol{D} = [0,0] - [1/3, 1/3] \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = [1/3, 1/3] = [r_3, r_4]. \end{aligned}$$

The reduced cost coefficients are all nonnegative. Hence, the solution to the standard form problem is $[1,1,0,0]^\top$. Therefore, the solution to the original problem is $[1,1]^\top$, and the corresponding cost is 2.

16.12

a. The problem in standard form is:

$$\begin{array}{rl} \text{minimize} & 4x_1 + 3x_2 \\ \text{subject to} & 5x_1 + x_2 - x_3 = 11 \\ & 2x_1 + x_2 - x_4 = 8 \\ & x_1 + 2x_2 - x_5 = 7 \\ & x_1, \ldots, x_5 \geq 0. \end{array}$$

We do not have an apparent basic feasible solution. Therefore, we will need to use the two phase method.
Phase I: We introduce artificial variables x_6, x_7, x_8 and form the following tableau.

	$\boldsymbol{a}_1$	$\boldsymbol{a}_2$	$\boldsymbol{a}_3$	$\boldsymbol{a}_4$	$\boldsymbol{a}_5$	$\boldsymbol{a}_6$	$\boldsymbol{a}_7$	$\boldsymbol{a}_8$	$\boldsymbol{b}$
	5	1	-1	0	0	1	0	0	11
	2	1	0	-1	0	0	1	0	8
	1	2	0	0	-1	0	0	1	7
$\boldsymbol{c}^\top$	0	0	0	0	0	1	1	1	0

We then form the following revised tableau:

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_6	1	0	0	11
x_7	0	1	0	8
x_8	0	0	1	7

We compute:

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [1,1,1] \\ \boldsymbol{r}_D^\top &= [r_1, r_2, r_3, r_4, r_5] = [-8, -4, 1, 1, 1]. \end{aligned}$$

We form the augmented revised tableau by introducing $\boldsymbol{y}_1 = \boldsymbol{B}^{-1}\boldsymbol{a}_1 = \boldsymbol{a}_1$:

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$	$\boldsymbol{y}_1$
x_6	1	0	0	11	5
x_7	0	1	0	8	2
x_8	0	0	1	7	1

We now pivot about the first component of $\boldsymbol{y}_1$ to get

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_1	1/5	0	0	11/5
x_7	−2/5	1	0	18/5
x_8	−1/5	0	1	24/5

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [-3/5, 1, 1] \\ \boldsymbol{r}_D^\top &= [r_2, r_3, r_4, r_5, r_6] = [-12/5, -3/5, 1, 1, 8/5]. \end{aligned}$$

We bring $\boldsymbol{y}_2 = \boldsymbol{B}^{-1}\boldsymbol{a}_2$ into the basis to get

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$	$\boldsymbol{y}_2$
x_1	1/5	0	0	11/5	1/5
x_7	−2/5	1	0	18/5	3/5
x_8	−1/5	0	1	24/5	9/5

We pivot about the third component of $\boldsymbol{y}_2$ to get

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_1	2/9	0	−1/9	5/3
x_7	−1/3	1	−1/3	2
x_2	−1/9	0	5/9	8/3

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [-1/3, 1, -1/3] \\ \boldsymbol{r}_D^\top &= [r_3, r_4, r_5, r_6, r_8] = [-1/3, 1, -1/3, 4/3, 4/3]. \end{aligned}$$

We bring $\boldsymbol{y}_3 = \boldsymbol{B}^{-1}\boldsymbol{a}_3$ into the basis to get

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$	$\boldsymbol{y}_3$
x_1	2/9	0	−1/9	5/3	−2/9
x_7	−1/3	1	−1/3	2	1/3
x_2	−1/9	0	5/9	8/3	1/9

We pivot about the second component of $\boldsymbol{y}_3$ to obtain

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_1	0	2/3	−1/3	3
x_3	−1	3	−1	6
x_2	0	−1/3	2/3	2

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [0, 0, 0] \\ \boldsymbol{r}_D^\top &= [r_4, r_5, r_6, r_7, r_8] = [0, 0, 1, 1, 1] \geq \mathbf{0}^\top. \end{aligned}$$

Thus, Phase I is complete, and the initial basic feasible solution is $[3, 2, 6, 0, 0]^\top$.

Phase II

We form the tableau for the original problem:

	$\boldsymbol{a}_1$	$\boldsymbol{a}_2$	$\boldsymbol{a}_3$	$\boldsymbol{a}_4$	$\boldsymbol{a}_5$	$\boldsymbol{b}$
	5	1	-1	0	0	11
	2	1	0	-1	0	8
	1	2	0	0	-1	7
$\boldsymbol{c}^\top$	4	3	0	0	0	0

The initial revised tableau for Phase II is the final revised tableau for Phase I. We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [0, 5/3, 2/3] \\ \boldsymbol{r}_D^\top &= [r_4, r_5] = [5/3, 2/3] > \mathbf{0}^\top. \end{aligned}$$

Hence, the optimal solution to the original problem is $[3, 2]^\top$.

b. The problem in standard form is:

$$\begin{aligned} \text{minimize} \quad & -6x_1 - 4x_2 - 7x_3 - 5x_4 \\ \text{subject to} \quad & x_1 + 2x_2 + x_3 + 2x_4 + x_5 = 20 \\ & 6x_1 + 5x_2 + 3x_3 + 2x_4 + x_6 = 100 \\ & 3x_1 + 4x_2 + 9x_3 + 12x_4 + x_7 = 75 \\ & x_1, \ldots, x_7 \geq 0. \end{aligned}$$

We have an apparent basic feasible solution: $[0, 0, 0, 20, 100, 75]^\top$, corresponding to $\boldsymbol{B} = \boldsymbol{I}_3$. We form the revised tableau corresponding to this basic feasible solution:

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_5	1	0	0	20
x_6	0	1	0	100
x_7	0	0	1	75

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [0, 0, 0] \\ \boldsymbol{r}_D^\top &= [r_1, r_2, r_3, r_4] = [-6, -4, -7, -5]. \end{aligned}$$

We bring $\boldsymbol{y}_2 = \boldsymbol{B}^{-1}\boldsymbol{a}_3 = \boldsymbol{a}_3$ into the basis to obtain

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$	$\boldsymbol{y}_3$
x_5	1	0	0	20	1
x_6	0	1	0	100	3
x_7	0	0	1	75	9

We pivot about the third component of $\boldsymbol{y}_3$ to get

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_5	1	0	$-1/9$	$35/3$
x_6	0	1	$-1/3$	75
x_3	0	0	$1/9$	$25/3$

We compute

$$\begin{aligned} \boldsymbol{\lambda}^\top &= [0, 0, -7/9] \\ \boldsymbol{r}_D^\top &= [r_1, r_2, r_4, r_7] = [-11/3, -8/9, 13/3, 7/9]. \end{aligned}$$

We bring $\boldsymbol{y}_1 = \boldsymbol{B}^{-1}\boldsymbol{a}_1$ into the basis to obtain

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$	$\boldsymbol{y}_1$
x_5	1	0	$-1/9$	$35/3$	$2/3$
x_6	0	1	$-1/3$	75	5
x_3	0	0	$1/9$	$25/3$	$1/3$

We pivot about the second component of $\boldsymbol{y}_1$ to obtain

Variable	$\boldsymbol{B}^{-1}$			$\boldsymbol{y}_0$
x_5	1	$-2/15$	$-1/15$	$5/3$
x_1	0	$1/5$	$-1/15$	15
x_3	0	$-1/15$	$2/15$	$10/3$

We compute

$$\begin{aligned}\boldsymbol{\lambda}^\top &= [0, -11/15, -8/15] \\ \boldsymbol{r}_D^\top &= [r_2, r_4, r_6, r_7] = [27/15, 43/15, 11/15, 8/15] > \mathbf{0}^\top.\end{aligned}$$

The optimal solution to the original problem is therefore $[15, 0, 10/3, 0]^\top$.

16.13

a. By inspection of $\boldsymbol{r}^\top$, we conclude that the basic variables are x_1, x_3, x_4, and the basis matrix is

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Since $\boldsymbol{r}^\top \geq \mathbf{0}^\top$, the basic feasible solution corresponding to the basis $\boldsymbol{B}$ is optimal. This optimal basic feasible solution is $[8, 0, 9, 7]^\top$.

b. An optimal solution to the dual is given by

$$\boldsymbol{\lambda}^\top = \boldsymbol{c}_B^\top \boldsymbol{B}^{-1},$$

where $\boldsymbol{c}_B^\top = [6, 4, 5]$, and

$$\boldsymbol{B}^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

We obtain $\boldsymbol{\lambda}^\top = [5, 6, 4]$.

c. We have $\boldsymbol{r}_D^\top = \boldsymbol{c}_D^\top - \boldsymbol{\lambda}^\top \boldsymbol{D}$, where $\boldsymbol{r}_D^\top = [1]$, $\boldsymbol{c}_D^\top = [c_2]$, $\boldsymbol{\lambda}^\top = [5, 6, 4]$, and $\boldsymbol{D} = [2, 1, 3]^\top$. We get $1 = c_2 - 10 - 6 - 12$, which yields $c_2 = 29$.

16.14

a. There are two basic feasible solutions: $[1, 0]^\top$ and $[0, 2]^\top$.

b. The feasible set in $\mathbb{R}^2$ for this problem is the line segment joining the two basic feasible solutions $[1, 0]^\top$ and $[0, 2]^\top$. Therefore, if the problem has an optimal feasible solution that is not basic, then all points in the feasible set are optimal. For this, we need

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \alpha \begin{bmatrix} 2 \\ 1 \end{bmatrix},$$

where $\alpha \in \mathbb{R}$.

c. Since all basic feasible solutions are optimal, the relative cost coefficients are all zero.

16.15

a. $2 - \alpha < 0$, $\beta \leq 0$, $\gamma \leq 0$, and δ anything.

b. $2 - \alpha \geq 0$, $\delta = -7$, β and γ anything.

c. $2 - \alpha < 0$, $\gamma > 0$, either $\beta \leq 0$ or $5/\gamma \leq 4/\beta$, and δ anything.

16.16

a. The value of α must be 0, because the objective function value is 0 (lower right corner), and α is the value of an artificial variable.

The value of β must be 0, because it is the RCC value corresponding to a basic column.

The value of γ must be 2, because it must be a positive value. Otherwise, there is a feasible solution to the artificial problem with objective function value smaller than 0, which is impossible.

The value of δ must be 0, because we must be able to bring the fourth column into the basis without changing the objective function value.

b. The given linear programming problem does indeed have a feasible solution: $[0, 5, 6, 0]^\top$. We obtain this by noticing that the right-most column is a linear combination of the second and third columns, with coefficients 5 and 6.

16.17

First, we convert the inequality constraint $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ into standard form. To do this, we introduce a variable $\boldsymbol{w} \in \mathbb{R}^m$ of surplus variables to convert the inequality constraint into the following equivalent constraint:

$$[\boldsymbol{A}, -\boldsymbol{I}] \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{b}, \quad \boldsymbol{w} \geq \boldsymbol{0}.$$

Next, we introduce variables $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ to replace the free variable $\boldsymbol{x}$ by $\boldsymbol{u} - \boldsymbol{v}$. We then obtain the following equivalent constraint:

$$[\boldsymbol{A}, -\boldsymbol{A}, -\boldsymbol{I}] \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{b}, \quad \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \geq \boldsymbol{0}.$$

This form of the constraint is now in standard form. So we can now use Phase I of the simplex method to implement an algorithm to find a vectors $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ satisfying the above constraint, if such exist, or to declare that none exists. If such exist, we output $\boldsymbol{x} = \boldsymbol{u} - \boldsymbol{v}$; otherwise, we declare that no $\boldsymbol{x}$ exists such that $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$. By construction, this algorithm is guaranteed to behave in the way specified by the question.

16.18

a. We form the tableau for the problem:

$$\begin{array}{cccccccc} 1 & 0 & 0 & 1/4 & -8 & -1 & 9 & 0 \\ 0 & 1 & 0 & 1/2 & -12 & -1/2 & 3 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & -3/4 & 20 & -1/2 & 6 & 0 \end{array}$$

The above tableau is already in canonical form, and therefore we can proceed with the simplex procedure. We first pivot about the $(1, 4)$th element, to get

$$\begin{array}{cccccccc} 4 & 0 & 0 & 1 & -32 & -4 & 36 & 0 \\ -2 & 1 & 0 & 0 & 4 & 3/2 & -15 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 3 & 0 & 0 & 0 & -4 & -7/2 & 33 & 0 \end{array}$$

Pivoting about the $(2, 5)$th element, we get

$$\begin{array}{cccccccc} -12 & 8 & 0 & 1 & 0 & 8 & -84 & 0 \\ -1/2 & 1/4 & 0 & 0 & 1 & 3/8 & -15/4 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & -2 & 18 & 0 \end{array}$$

Pivoting about the $(1,6)$th element, we get

$$\begin{array}{cccccccc} -3/2 & 1 & 0 & 1/8 & 0 & 1 & -21/2 & 0 \\ 1/16 & -1/8 & 0 & -3/64 & 1 & 0 & 3/16 & 0 \\ 3/2 & -1 & 1 & -1/8 & 0 & 0 & 21/2 & 1 \\ -2 & 3 & 0 & 1/4 & 0 & 0 & -3 & 0 \end{array}$$

Pivoting about the $(2,7)$th element, we get

$$\begin{array}{cccccccc} 2 & -6 & 0 & -5/2 & 56 & 1 & 0 & 0 \\ 1/3 & -2/3 & 0 & -1/4 & 16/3 & 0 & 1 & 0 \\ -2 & 6 & 1 & 5/2 & -56 & 0 & 0 & 1 \\ -1 & 1 & 0 & -1/2 & 16 & 0 & 0 & 0 \end{array}$$

Pivoting about the $(1,1)$th element, we get

$$\begin{array}{cccccccc} 1 & -3 & 0 & -5/4 & 28 & 1/2 & 0 & 0 \\ 0 & 1/3 & 0 & 1/6 & -4 & -1/6 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & -2 & 0 & -7/4 & 44 & 1/2 & 0 & 0 \end{array}$$

Pivoting about the $(2,2)$th element, we get

$$\begin{array}{cccccccc} 1 & 0 & 0 & 1/4 & -8 & -1 & 9 & 0 \\ 0 & 1 & 0 & 1/2 & -12 & -1/2 & 3 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & -3/4 & 20 & -1/2 & 6 & 0 \end{array}$$

which is identical to the initial tableau. Therefore, cycling occurs.

b. We start with the initial tableau of part a, and pivot about the $(1,4)$th element to obtain

$$\begin{array}{cccccccc} 4 & 0 & 0 & 1 & -32 & -4 & 36 & 0 \\ -2 & 1 & 0 & 0 & 4 & 3/2 & -15 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 3 & 0 & 0 & 0 & -4 & -7/2 & 33 & 0 \end{array}$$

Pivoting about the $(2,5)$th element, we get

$$\begin{array}{cccccccc} -12 & 8 & 0 & 1 & 0 & 8 & -84 & 0 \\ -1/2 & 1/4 & 0 & 0 & 1 & 3/8 & -15/4 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & -2 & 18 & 0 \end{array}$$

Pivoting about the $(1,6)$th element, we get

$$\begin{array}{cccccccc} -3/2 & 1 & 0 & 1/8 & 0 & 1 & -21/2 & 0 \\ 1/16 & -1/8 & 0 & -3/64 & 1 & 0 & 3/16 & 0 \\ 3/2 & -1 & 1 & -1/8 & 0 & 0 & 21/2 & 1 \\ -2 & 3 & 0 & 1/4 & 0 & 0 & -3 & 0 \end{array}$$

Pivoting about the $(2,1)$th element, we get

$$\begin{array}{cccccccc} 0 & -2 & 0 & -1 & 24 & 1 & -6 & 0 \\ 1 & -2 & 0 & -3/4 & 16 & 0 & 3 & 0 \\ 0 & 2 & 1 & 1 & -24 & 0 & 6 & 1 \\ 0 & -1 & 0 & -5/4 & 32 & 0 & 3 & 0 \end{array}$$

Pivoting about the $(3,2)$th element, we get

$$\begin{array}{cccccccc} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1/4 & -8 & 0 & 9 & 1 \\ 0 & 1 & 1/2 & 1/2 & -12 & 0 & 3 & 1/2 \\ 0 & 0 & 1/2 & -3/4 & 20 & 0 & 6 & 1/2 \end{array}$$

Pivoting about the $(3,4)$th element, we get

$$\begin{array}{cccccccc} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & -1/2 & 3/4 & 0 & -2 & 0 & 15/2 & 3/4 \\ 0 & 2 & 1 & 1 & -24 & 0 & 6 & 1 \\ 0 & 3/2 & 5/4 & 0 & 2 & 0 & 21/2 & 5/4 \end{array}$$

The reduced cost coefficients are all nonnegative. Hence, the optimal solution to the problem is $[3/4, 0, 0, 1, 0, 1, 0]^\top$. The corresponding optimal cost is $-5/4$.

16.19

a. We have

$$\boldsymbol{A}\boldsymbol{d}^{(0)} = \boldsymbol{A}(\boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)})/\alpha_0 = (\boldsymbol{b} - \boldsymbol{b})/\alpha_0 = 0.$$

Hence, $\boldsymbol{d}^{(0)} \in \mathcal{N}(\boldsymbol{A})$.

b. From our discussion of moving from one BFS to an adjacent BFS, we deduce that

$$\boldsymbol{d}^{(0)} = \begin{bmatrix} -\boldsymbol{y}_q \\ \boldsymbol{e}_{q-m} \end{bmatrix}.$$

In other words, the first m components of $\boldsymbol{d}^{(0)}$ are $-y_{1q}, \ldots, -y_{mq}$, and all the other components are 0 except the qth component, which is 1.

16.20

The following is a MATLAB function that implements the simplex algorithm.

```
function [x,v]=simplex(c,A,b,v,options)
%               SIMPLEX(c,A,b,v);
%               SIMPLEX(c,A,b,v,options);
%
%               x = SIMPLEX(c,A,b,v);
%               x = SIMPLEX(c,A,b,v,options);
%
%               [x,v] = SIMPLEX(c,A,b,v);
%               [x,v] = SIMPLEX(c,A,b,v,options);
%
%SIMPLEX(c,A,b,v) solves the following linear program using the
%Simplex Method:
%   min c'x  subject to Ax=b, x>=0,
%where [A b] is in canonical form, and v is the vector of indices of
%basic columns. Specifically, the v(i)-th column of A is the i-th
%standard basis vector.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(5) specifies how the pivot element is selected;
%   0=choose the most negative relative cost coefficient;
%   1=use Bland's rule.
```

```
if nargin ~= 5
  options = [];
  if nargin ~= 4
    disp('Wrong number of arguments.');
    return;
  end
end

format compact;
%format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

cB=c(v(:));

r = c'-cB'*A; %row vector of relative cost coefficients

cost = -cB'*b;

tabl=[A b;r cost];

if print,
  disp(' ');
  disp('Initial tableau:');
  disp(tabl);
end %if

while ones(1,n)*(r' >= zeros(n,1)) ~= n
  if options(5) == 0;
    [r_q,q] = min(r);
  else
    %Bland's rule
    q=1;
    while r(q) >= 0
      q=q+1;
    end
  end %if

  min_ratio = inf;
  p=0;
  for i=1:m,
    if tabl(i,q)>0
      if tabl(i,n+1)/tabl(i,q) < min_ratio
        min_ratio = tabl(i,n+1)/tabl(i,q);
        p = i;
      end %if
    end %if
  end %for
  if p == 0
    disp('Problem unbounded');
    break;
  end %if

  tabl=pivot(tabl,p,q);
```

```
  if print,
    disp('Pivot point:');
    disp([p,q]);
    disp('New tableau:');
    disp(tabl);
  end %if

  v(p) = q;
  r = tabl(m+1,1:n);
end %while

x=zeros(n,1);
x(v(:))=tabl(1:m,n+1);
```

The above function makes use of the following function that implements pivoting:

```
function Mnew=pivot(M,p,q)
%Mnew=pivot(M,p,q)
%Returns the matrix Mnew resulting from pivoting about the
%(p,q)th element of the given matrix M.

for i=1:size(M,1),
  if i==p
    Mnew(p,:)=M(p,:)/M(p,q);
  else
    Mnew(i,:)=M(i,:)-M(p,:)*(M(i,q)/M(p,q));
  end %if
end %for
%-------------------------------------------------------------------
```

We now apply the simplex algorithm to the problem in Example 16.2, as follows:

```
>> A=[1 0 1 0 0; 0 1 0 1 0; 1 1 0 0 1];
>> b=[4;6;8];
>> c=[-2;-5;0;0;0];
>> v=[3;4;5];
>> options(1)=1;
>> [x,v]=simplex(c,A,b,v,options);

Initial Tableau:
      1     0     1     0     0     4
      0     1     0     1     0     6
      1     1     0     0     1     8
     -2    -5     0     0     0     0
Pivot point:
      2     2
New tableau:
      1     0     1     0     0     4
      0     1     0     1     0     6
      1     0     0    -1     1     2
     -2     0     0     5     0     30
Pivot point:
      3     1
New tableau:
      0     0     1     1    -1     2
      0     1     0     1     0     6
      1     0     0    -1     1     2
      0     0     0     3     2     34
>> disp(x');
      2     6     2     0     0
```

```
>> disp(v');
      3     2     1
```

As indicated above, the solution to the problem in standard form is $[2, 6, 2, 0, 0]^\top$, and the objective function value is -34. The optimal cost for the original maximization problem is 34.

16.21

The following is a MATLAB routine that implements the two-phase simplex method, using the MATLAB function from Exercise 16.20.

```
function [x,v]=tpsimplex(c,A,b,options)
%               TPSIMPLEX(c,A,b);
%               TPSIMPLEX(c,A,b,options);
%
%               x = TPSIMPLEX(c,A,b);
%               x = TPSIMPLEX(c,A,b,options);
%
%               [x,v] = TPSIMPLEX(c,A,b);
%               [x,v] = TPSIMPLEX(c,A,b,options);
%
%TPSIMPLEX(c,A,b) solves the following linear program using the
%two-phase simplex method:
%  min c'x  subject to Ax=b, x>=0.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(5) specifies how the pivot element is selected;
%  0=choose the most negative relative cost coefficient;
%  1=use Bland's rule.


if nargin ~= 4
  options = [];
    if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

clc;
format compact;
%format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

%Phase I
if print,
  disp(' ');
  disp('Phase I');
  disp('-------');
end

v=n*ones(m,1);
for i=1:m
  v(i)=v(i)+i;
```

```
end

[x,v]=simplex([zeros(n,1);ones(m,1)],[A eye(m)],b,v,options);

if all(v<=n),
  %Phase II
  if print
    disp(' ');
    disp('Phase II');
    disp('--------');
    disp('Basic columns:')
    disp(v')
  end

  %Convert [A b] into canonical augmented matrix
  Binv=inv(A(:,[v]));
  A=Binv*A;
  b=Binv*b;

  [x,v]=simplex(c,A,b,v,options);

  if print
    disp(' ');
    disp('Final solution:');
    disp(x');
  end
else
  %assumes nondegeneracy
  disp('Terminating: problem has no feasible solution.');
end
%-------------------------------------------------------------------
```

We now apply the above MATLAB routine to the problem in Example 16.5, as follows:

```
>> A=[1 1 1 0; 5 3 0 -1];
>> b=[4;8];
>> c=[-3;-5;0;0];
>> options(1)=1;
>> format rat;
>> tpsimplex(c,A,b,options);

Phase I
-------

Initial Tableau:
      1     1     1     0     1     0     4
      5     3     0    -1     0     1     8
     -6    -4    -1     1     0     0    -12
Pivot point:
      2     1
New tableau:
      0    2/5    1    1/5    1   -1/5  12/5
      1    3/5    0   -1/5    0    1/5   8/5
      0   -2/5   -1   -1/5    0    6/5 -12/5
Pivot point:
      1     3
New tableau:
      0    2/5    1    1/5    1   -1/5  12/5
      1    3/5    0   -1/5    0    1/5   8/5
      0     *     0     *     1     1     *
```

```
Pivot point:
      2     2
New tableau:
    -2/3    0     1    1/3    1   -1/3   4/3
     5/3    1     0   -1/3    0    1/3   8/3
      *     0     0     *     1     1     *
Pivot point:
      1     4
New tableau:
     -2     0     3     1     3    -1     4
      1     1     1     0     1     0     4
      *     0     *     0     1     1     *

Basic columns:
      4     2

Phase II
--------

Initial Tableau:
     -2     0     3     1     4
      1     1     1     0     4
      2     0     5     0    20

Final solution:
      0     4     0     4
```

16.22

The following is a MATLAB function that implements the revised simplex algorithm.

```
function [x,v,Binv]=revsimp(c,A,b,v,Binv,options)
%               REVSIMP(c,A,b,v,Binv);
%               REVSIMP(c,A,b,v,Binv,options);
%
%               x = REVSIMP(c,A,b,v,Binv);
%               x = REVSIMP(c,A,b,v,Binv,options);
%
%               [x,v,Binv] = REVSIMP(c,A,b,v,Binv);
%               [x,v,Binv] = REVSIMP(c,A,b,v,Binv,options);
%
%REVSIMP(c,A,b,v,Binv) solves the following linear program using the
%revised simplex method:
%   min c'x  subject to Ax=b, x>=0,
%where v is the vector of indices of basic columns, and Binv is the
%inverse of the basis matrix. Specifically, the v(i)-th column of
%A is the i-th column of the basis vector.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(5) specifies how the pivot element is selected;
%   0=choose the most negative relative cost coefficient;
%   1=use Bland's rule.

if nargin ~= 6
  options = [];
  if nargin ~= 5
    disp('Wrong number of arguments.');
    return;
  end
```

```
end

format compact;
%format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

cB=c(v(:));
y0 = Binv*b;
lambdaT=cB'*Binv;
r = c'-lambdaT*A; %row vector of relative cost coefficients

if print,
  disp(' ');
  disp('Initial revised tableau [v B^(-1) y0]:');
  disp([v Binv y0]);
  disp('Relative cost coefficients:');
  disp(r);
end %if

while ones(1,n)*(r' >= zeros(n,1)) ~= n
  if options(5) == 0;
    [r_q,q] = min(r);
  else
    %Bland's rule
    q=1;
    while r(q) >= 0
      q=q+1;
    end
  end %if

  yq = Binv*A(:,q);
  min_ratio = inf;
  p=0;
  for i=1:m,
    if yq(i)>0
      if y0(i)/yq(i) < min_ratio
        min_ratio = y0(i)/yq(i);
        p = i;
      end %if
    end %if
  end %for
  if p == 0
    disp('Problem unbounded');
    break;
  end %if

  if print,
    disp('Augmented revised tableau [v B^(-1) y0 yq]:')
    disp([v Binv y0 yq]);
    disp('(p,q):');
    disp([p,q]);
  end

  augrevtabl=pivot([Binv y0 yq],p,m+2);
```

```
  Binv=augrevtabl(:,1:m);
  y0=augrevtabl(:,m+1);

  v(p) = q;

  cB=c(v(:));
  lambdaT=cB'*Binv;
  r = c'-lambdaT*A; %row vector of relative cost coefficients

  if print,
    disp('New revised tableau [v B^(-1) y0]:');
    disp([v Binv y0]);
    disp('Relative cost coefficients:');
    disp(r);
  end %if

end %while

x=zeros(n,1);
x(v(:))=y0;
```

The function makes use of the pivoting function in Exercise 16.20.

We now apply the simplex algorithm to the problem in Example 16.2, as follows:

```
>> A=[1 0 1 0 0; 0 1 0 1 0; 1 1 0 0 1];
>> b=[4;6;8];
>> c=[-2;-5;0;0;0];
>> v=[3;4;5];
>> Binv=eye(3);
>> options(1)=1;
>> [x,v,Binv]=rev_simp(c,A,b,v,Binv,options);

Initial revised tableau [v B^(-1) y0]:
      3     1     0     0     4
      4     0     1     0     6
      5     0     0     1     8
Relative cost coefficients:
     -2    -5     0     0     0
Augmented revised tableau [v B^(-1) y0 yq]:
      3     1     0     0     4     0
      4     0     1     0     6     1
      5     0     0     1     8     1
(p,q):
      2     2
New revised tableau [v B^(-1) y0]:
      3     1     0     0     4
      2     0     1     0     6
      5     0    -1     1     2
Relative cost coefficients:
     -2     0     0     5     0
Augmented revised tableau [v B^(-1) y0 yq]:
      3     1     0     0     4     1
      2     0     1     0     6     0
      5     0    -1     1     2     1
(p,q):
      3     1
New revised tableau [v B^(-1) y0]:
      3     1     1    -1     2
      2     0     1     0     6
      1     0    -1     1     2
```

```
Relative cost coefficients:
      0     0     0     3     2
>> disp(x');
      2     6     2     0     0
>> disp(v');
      3     2     1
>> disp(Binv);
      1     1    -1
      0     1     0
      0    -1     1
```

16.23

The following is a MATLAB routine that implements the two-phase revised simplex method, using the MATLAB function from Exercise 16.22.

```
function [x,v]=tprevsimp(c,A,b,options)
%               TPREVSIMP(c,A,b);
%               TPREVSIMP(c,A,b,options);
%
%               x = TPREVSIMP(c,A,b);
%               x = TPREVSIMP(c,A,b,options);
%
%               [x,v] = TPREVSIMP(c,A,b);
%               [x,v] = TPREVSIMP(c,A,b,options);
%
%TPREVSIMP(c,A,b) solves the following linear program using the
%two-phase revised simplex method:
%  min c'x  subject to Ax=b, x>=0.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(5) specifies how the pivot element is selected;
%  0=choose the most negative relative cost coefficient;
%  1=use Bland's rule.


if nargin ~= 4
  options = [];
    if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

clc;
format compact;
%format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

%Phase I
if print,
    disp(' ');
    disp('Phase I');
    disp('-------');
```

```
end

v=n*ones(m,1);
for i=1:m
  v(i)=v(i)+i;
end

[x,v,Binv]=rev_simp([zeros(n,1);ones(m,1)],[A eye(m)],b,v,eye(m),options);

%Phase II
if print
    disp(' ');
    disp('Phase II');
    disp('--------');
end

[x,v,Binv]=rev_simp(c,A,b,v,Binv,options);

if print
    disp(' ');
  disp('Final solution:');
  disp(x');
end
%-------------------------------------------------------------------
```

We now apply the above MATLAB routine to the problem in Example 16.5, as follows:

```
>> A=[4 2 -1 0; 1 4 0 -1];
>> b=[12;6];
>> c=[2;3;0;0];
>> options(1)=1;
>> format rat;
>> tprevsimp(c,A,b,options);

Phase I
-------

Initial revised tableau [v B^(-1) y0]:
      5     1     0    12
      6     0     1     6
Relative cost coefficients:
     -5    -6     1     1     0     0
Augmented revised tableau [v B^(-1) y0 yq]:
      5     1     0    12     2
      6     0     1     6     4
(p,q):
      2     2
New revised tableau [v B^(-1) y0]:
      5     1   -1/2    9
      2     0    1/4   3/2
Relative cost coefficients:
    -7/2    0     1   -1/2    0    3/2
Augmented revised tableau [v B^(-1) y0 yq]:
      5     1   -1/2    9    7/2
      2     0    1/4   3/2   1/4
(p,q):
      1     1
New revised tableau [v B^(-1) y0]:
      1    2/7  -1/7  18/7
      2   -1/14  2/7   6/7
```

```
Relative cost coefficients:
      0     0     0     0     1     1

Phase II
--------

Initial revised tableau [v B^(-1) y0]:
      1    2/7  -1/7  18/7
      2   -1/14  2/7   6/7
Relative cost coefficients:
      *     0    5/14  4/7

Final solution:
    18/7   6/7    0     0
```

17. Duality

17.1

Since $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ are feasible, we have $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$, and $\boldsymbol{\lambda}^\top \boldsymbol{A} \leq \boldsymbol{c}^\top$, $\boldsymbol{\lambda} \geq \boldsymbol{0}$. Postmultiplying both sides of $\boldsymbol{\lambda}^\top \boldsymbol{A} \leq \boldsymbol{c}^\top$ by $\boldsymbol{x} \geq \boldsymbol{0}$ yields

$$\boldsymbol{\lambda}^\top \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{c}^\top \boldsymbol{x}.$$

Since $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ and $\boldsymbol{\lambda}^\top \geq \boldsymbol{0}^\top$, we have $\boldsymbol{\lambda}^\top \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{\lambda}^\top \boldsymbol{b}$. Hence, $\boldsymbol{\lambda}^\top \boldsymbol{b} \leq \boldsymbol{c}^\top \boldsymbol{x}$.

17.2

The primal problem is:

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{e}_n^\top \boldsymbol{p} \\ \text{subject to} \quad & \boldsymbol{G}\boldsymbol{p} \geq P\boldsymbol{e}_m \\ & \boldsymbol{p} \geq \boldsymbol{0}, \end{aligned}$$

where $\boldsymbol{G} = [g_{i,j}]$, $\boldsymbol{e}_n = [1, \ldots, 1]^\top$ (with n components), and $\boldsymbol{p} = [p_1, \ldots, p_n]^\top$. The dual of the problem is (using symmetric duality):

$$\begin{aligned} \text{maximize} \quad & P\boldsymbol{\lambda}^\top \boldsymbol{e}_m \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top \boldsymbol{G} \leq \boldsymbol{e}_n^\top \\ & \boldsymbol{\lambda} \geq \boldsymbol{0}. \end{aligned}$$

17.3

a. We first transform the problem into standard form:

$$\begin{aligned} \text{minimize} \quad & -2x_1 - 3x_2 \\ \text{subject to} \quad & x_1 + 2x_2 + x_3 = 4 \\ & 2x_1 + x_2 + x_4 = 5 \\ & x_1, x_2, x_3, x_4 \geq 0. \end{aligned}$$

The initial tableau is:

$$\begin{array}{rrrrr} 1 & 2 & 1 & 0 & 4 \\ 2 & 1 & 0 & 1 & 5 \\ -2 & -3 & 0 & 0 & 0 \end{array}$$

We now pivot about the $(1,2)$th element to get:

$$\begin{array}{rrrrr} 1/2 & 1 & 1/2 & 0 & 2 \\ 3/2 & 0 & -1/2 & 1 & 3 \\ -1/2 & 0 & 3/2 & 0 & 6 \end{array}$$

Pivoting now about the $(2,1)$th element gives:

$$\begin{array}{ccccc} 0 & 1 & 2/3 & -1/3 & 1 \\ 1 & 0 & -1/3 & 2/3 & 2 \\ 0 & 0 & 4/3 & 1/3 & 7 \end{array}$$

Thus, the solution to the standard form problem is $x_1 = 2$, $x_2 = 1$, $x_3 = 0$, $x_4 = 0$. The solution to the original problem is $x_1 = 2$, $x_2 = 1$.

b. The dual to the standard form problem is

$$\begin{array}{rl} \text{maximize} & 4\lambda_1 + 5\lambda_2 \\ \text{subject to} & \lambda_1 + 2\lambda_2 \leq -2 \\ & 2\lambda_1 + \lambda_2 \leq -3 \\ & \lambda_1, \lambda_2 \leq 0. \end{array}$$

From the discussion before Example 17.6, it follows that the solution to the dual is $\boldsymbol{\lambda}^\top = \boldsymbol{c}_I^\top - \boldsymbol{r}_I^\top = [-4/3, -1/3]$.

17.4

The dual problem is

$$\begin{array}{rl} \text{maximize} & 11\lambda_1 + 8\lambda_2 + 7\lambda_3 \\ \text{subject to} & 5\lambda_1 + 2\lambda_2 + \lambda_3 \leq 4 \\ & \lambda_1 + \lambda_2 + 2\lambda_3 \leq 3 \\ & \lambda_1, \lambda_2, \lambda_3 \geq 0. \end{array}$$

Note that we may arrive at the above in one of two ways: by applying the asymmetric form of duality, or by applying the symmetric form of duality to the original problem in standard form. From the solution of Exercise 16.11a, we have that the solution to the dual is $\boldsymbol{\lambda}^\top = \boldsymbol{c}_B^\top \boldsymbol{B}^{-1} = [0, 5/3, 2/3]$ (using the proof of the duality theorem).

17.5

We represent the primal in the form

$$\begin{array}{rl} \text{minimize} & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0. \end{array}$$

The corresponding dual is

$$\begin{array}{rl} \text{maximize} & \boldsymbol{\lambda}^\top \boldsymbol{b} \\ \text{subject to} & \boldsymbol{\lambda}^\top \boldsymbol{A} \leq \boldsymbol{c}^\top, \end{array}$$

that is,

$$\begin{array}{rl} \text{maximize} & 2\lambda_1 + 7\lambda_2 + 3\lambda_3 \\ \text{subject to} & \begin{bmatrix} \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix} \begin{bmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \leq \begin{bmatrix} -1 & -2 & 0 & 0 & 0 \end{bmatrix}. \end{array}$$

The solution to the dual can be obtained using the formula, $\boldsymbol{\lambda}^{*\top} = \boldsymbol{c}_B^\top \boldsymbol{B}^{-1}$, where

$$\boldsymbol{c}_B^\top = \begin{bmatrix} -1 & -2 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{B} = \begin{bmatrix} -2 & 1 & 1 \\ -1 & 2 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

Note that because the last element in $\boldsymbol{c}_B^\top$ is zero, we do not need to calculate the last row of $\boldsymbol{B}^{-1}$ when computing $\boldsymbol{\lambda}^{*\top}$, that is, these elements are "don't care" elements that we denote using the asterisk. Hence,

$$\boldsymbol{\lambda}^{*\top} = \boldsymbol{c}_B^\top \boldsymbol{B}^{-1} = \begin{bmatrix} -1 & -2 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1/2 & 1/2 \\ * & * & * \end{bmatrix} = \begin{bmatrix} 0 & -1 & -2 \end{bmatrix}.$$

Note that

$$\boldsymbol{c}^\top \boldsymbol{x}^* = \boldsymbol{\lambda}^{*\top} \boldsymbol{b} = -13,$$

as expected.

17.6

a. Multiplying the objective function by -1, we see that the problem is of the form of the dual in the asymmetric form of duality. Therefore, the dual to the problem is of the form of the primal in the asymmetric form:

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{\lambda}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top \boldsymbol{A} = -\boldsymbol{c}^\top \\ & \boldsymbol{\lambda} \geq \boldsymbol{0} \end{aligned}$$

b. The given vector $\boldsymbol{y}$ is feasible in the dual. Since $\boldsymbol{b} = \boldsymbol{0}$, any feasible point in the dual is optimal. Thus, $\boldsymbol{y}$ is optimal in the dual, and the objective function value for $\boldsymbol{y}$ is 0. Therefore, by the Duality Theorem, the primal also has an optimal feasible solution, and the corresponding objective function value is 0. Since the vector $\boldsymbol{0}$ is feasible in the primal and has objective function value 0, the vector $\boldsymbol{0}$ is a solution to the primal.

17.7

We introduce two sets of nonnegative variables: $x_i^+ \geq 0$, $x_i^- \geq 0$, $i = 1, 2, \ldots, 3$. We can then represent the optimization problem in the form

$$\begin{aligned} \text{minimize} \quad & (x_1^+ + x_1^-) + (x_2^+ + x_2^-) + (x_3^+ + x_3^-) \\ \text{subject to} \quad & \left[\begin{array}{ccc|ccc} 1 & 1 & -1 & -1 & -1 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 \end{array}\right] \begin{bmatrix} x_1^+ \\ x_2^+ \\ x_3^+ \\ x_1^- \\ x_2^- \\ x_3^- \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \\ & x_i^+ \geq 0, x_i^- \geq 0. \end{aligned}$$

We form the initial tableau,

	x_1^+	x_2^+	x_3^+	x_1^-	x_2^-	x_3^-	$\boldsymbol{b}$
	1	1	−1	−1	−1	1	2
	0	−1	0	0	1	0	1
$\boldsymbol{c}^\top$	1	1	1	1	1	1	0

There is no apparent basic feasible solution. We add the second row to the first one to obtain,

	x_1^+	x_2^+	x_3^+	x_1^-	x_2^-	x_3^-	$\boldsymbol{b}$
	1	0	−1	−1	0	1	3
	0	−1	0	0	1	0	1
$\boldsymbol{c}^\top$	1	1	1	1	1	1	0

We next calculate the reduced cost coefficients,

	x_1^+	x_2^+	x_3^+	x_1^-	x_2^-	x_3^-	$\boldsymbol{b}$
	1	0	−1	−1	0	1	3
	0	−1	0	0	1	0	1
$\boldsymbol{c}^\top$	0	2	2	2	0	0	−4

We have zeros under the basic columns. The reduced cost coefficients are all nonnegative. The optimal solution is,

$$\boldsymbol{x}^* = \begin{bmatrix} 3 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^\top.$$

The optimal solution to the original problem is $\boldsymbol{x}^* = [3, -1, 0]^\top$.

The dual of the above linear program is

$$\begin{array}{ll} \text{maximize} & 2\lambda_1 + \lambda_2 \\ \text{subject to} & \begin{bmatrix} \lambda_1 & \lambda_2 \end{bmatrix} \left[\begin{array}{ccc|ccc} 1 & 1 & -1 & -1 & -1 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 \end{array}\right] \leq \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}. \end{array}$$

The optimal solution to the dual is

$$\begin{aligned} \boldsymbol{\lambda}^{*\top} &= \boldsymbol{c}_B^\top \boldsymbol{B}^{-1} \\ &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1 & 2 \end{bmatrix}. \end{aligned}$$

17.8

a. The dual (asymmetric form) is

$$\begin{array}{ll} \text{maximize} & \lambda \\ \text{subject to} & \lambda a_i \leq 1,\ i = 1, \ldots, n. \end{array}$$

We can write the constraint as

$$\lambda \leq \min\{1/a_i : i = 1, \ldots, n\} = 1/a_n.$$

Therefore, the solution to the dual problem is

$$\lambda = 1/a_n.$$

b. Duality Theorem: If the primal problem has an optimal solution, then so does the dual, and the optimal values of their respective objective functions are equal.

By the duality theorem, the primal has an optimal solution, and the optimal value of the objective function is $1/a_n$. The only feasible point in the primal with this objective function value is the basic feasible solution $[0, \ldots, 0, 1/a_n]^\top$.

c. Suppose we start at a nonoptimal initial basic feasible solution, $[0, \ldots, 1/a_i, \ldots, 0]^\top$, where $1 \leq i \leq n-1$. The relative cost coefficient for the qth column, $q \neq i$, is

$$r_q = 1 - \frac{a_q}{a_i}.$$

Since $a_n > a_j$ for any $j \neq n$, r_q is the most negative relative cost coefficient if and only if $q = n$.

17.9

a. By asymmetric duality, the dual is given by

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \lambda \geq c_i,\ i = 1, \ldots, n. \end{array}$$

b. The constraint in part a implies that λ is feasible if and only if $\lambda \geq c_4$. Hence, the solution is $\lambda^* = c_4$.

c. By the duality theorem, the optimal objective function value for the given problem is c_4. The only solution that achieves this value is $x_4^* = 1$ and $x_i^* = 0$ for all $i \neq 4$.

17.10

a. The dual is

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{\lambda}^\top \mathbf{0} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top \boldsymbol{A} \geq \boldsymbol{c}^\top \\ & \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned}$$

b. By the duality theorem, we conclude that the optimal value of the objective function is 0. The only vector satisfying $\boldsymbol{x} \geq \mathbf{0}$ that has an objective function value of 0 is $\boldsymbol{x} = \mathbf{0}$. Therefore, the solution is $\boldsymbol{x} = \mathbf{0}$.

c. The constraint set contains only the vector $\mathbf{0}$. Any other vector $\boldsymbol{x}$ satisfying $\boldsymbol{x} \geq \mathbf{0}$ has at least one positive component, and consequently has a positive objective function value. But this contradicts the fact that the optimal solution has an objective function value of 0.

17.11

a. The artificial problem is:

$$\begin{aligned} \text{minimize} \quad & [\mathbf{0}^\top, \boldsymbol{e}^\top]\boldsymbol{z} \\ \text{subject to} \quad & [\boldsymbol{A}, \boldsymbol{I}]\boldsymbol{z} = \boldsymbol{b} \\ & \boldsymbol{z} \geq \mathbf{0}, \end{aligned}$$

where $\boldsymbol{e} = [1, \ldots, 1]^\top$ and $\boldsymbol{z} = [\boldsymbol{x}^\top, \boldsymbol{y}^\top]^\top$.

b. The dual to the artificial problem is:

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{\lambda}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top \boldsymbol{A} \leq \mathbf{0}^\top \\ & \boldsymbol{\lambda}^\top \leq \boldsymbol{e}^\top. \end{aligned}$$

c. Suppose the given original linear programming problem has a feasible solution. By the FTLP, the original LP problem has a BFS. Then, by a theorem given in class, the artificial problem has an optimal feasible solution with $\boldsymbol{y} = \mathbf{0}$. Hence, by the Duality Theorem, the dual of the artificial problem also has an optimal feasible solution.

17.12

a. Possible. This situation arises if the primal is unbounded, which by the Weak Duality Lemma implies that the dual has no feasible solution.

b. Impossible, because the Duality Theorem requires that if the primal has an optimal feasible solution, then so does the dual.

c. Impossible, because the Duality Theorem requires that if the dual has an optimal feasible solution, then so does the primal. Also, the Weak Dual Lemma requires that if the primal is unbounded (i.e., has a feasible solution but no optimal feasible solution), then the dual must have no feasible solution.

17.13

To prove the result, we use Theorem 17.3 (Complementary Slackness). Since $\boldsymbol{\mu} \geq \mathbf{0}$, we have $\boldsymbol{A}^\top \boldsymbol{\lambda} = \boldsymbol{c} - \boldsymbol{\mu} \leq \boldsymbol{c}$. Hence, $\boldsymbol{\lambda}$ is a feasible solution to the dual. Now, $(\boldsymbol{c} - \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \boldsymbol{x} = \boldsymbol{\mu}^\top \boldsymbol{x} = 0$. Therefore, by Theorem 17.3, $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ are optimal for their respective problems.

17.14

To use the symmetric form of duality, we need to rewrite the problem as

$$\begin{aligned} \text{minimize} \quad & -\boldsymbol{c}^\top (\boldsymbol{u} - \boldsymbol{v}), \\ \text{subject to} \quad & -\boldsymbol{A}(\boldsymbol{u} - \boldsymbol{v}) \geq -\boldsymbol{b} \\ & \boldsymbol{u}, \boldsymbol{v} \geq \mathbf{0}, \end{aligned}$$

which we represent in the form

$$\begin{aligned} \text{minimize} \quad & [-\boldsymbol{c}^\top \ \boldsymbol{c}^\top] \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix}, \\ \text{subject to} \quad & [-\boldsymbol{A} \ \boldsymbol{A}] \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} \geq -\boldsymbol{b} \\ & \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} \geq \boldsymbol{0}. \end{aligned}$$

By the symmetric form of duality, the dual is:

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{\lambda}^\top(-\boldsymbol{b}) \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top[-\boldsymbol{A} \ \boldsymbol{A}] \leq [-\boldsymbol{c}^\top \ \boldsymbol{c}^\top] \\ & \boldsymbol{\lambda} \geq \boldsymbol{0}. \end{aligned}$$

Note that for the constraint involving $\boldsymbol{A}$, we have

$$\begin{aligned} \boldsymbol{\lambda}^\top[-\boldsymbol{A} \ \boldsymbol{A}] \leq [-\boldsymbol{c}^\top \ \boldsymbol{c}^\top] \quad & \Leftrightarrow \quad -\boldsymbol{\lambda}^\top\boldsymbol{A} \leq -\boldsymbol{c}^\top \text{ and } \boldsymbol{\lambda}^\top\boldsymbol{A} \leq \boldsymbol{c}^\top \\ & \Leftrightarrow \quad \boldsymbol{\lambda}^\top\boldsymbol{A} = \boldsymbol{c}^\top. \end{aligned}$$

Therefore, we can represent the dual as

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{\lambda}^\top\boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top\boldsymbol{A} = \boldsymbol{c}^\top \\ & \boldsymbol{\lambda} \geq \boldsymbol{0}. \end{aligned}$$

17.15

The corresponding dual can be written as:

$$\begin{aligned} \text{maximize} \quad & 3\lambda_1 + 3\lambda_2 \\ \text{subject to} \quad & \lambda_1 + 2\lambda_2 \leq 1 \\ & 2\lambda_1 + \lambda_2 \leq 1 \\ & \lambda_1, \lambda_2 \geq 0. \end{aligned}$$

To solve the dual, we refer back to the solution of Exercise 16.11. Using the idea of the proof of the duality theorem (Theorem 17.2), we obtain the solution to the dual as $\boldsymbol{\lambda}^\top = \boldsymbol{c}_B^\top\boldsymbol{B}^{-1} = [1/3, 1/3]$. The cost of the dual problem is 2, which verifies the duality theorem.

17.16

The dual to the above linear program (asymmetric form) is

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{\lambda}^\top\boldsymbol{0} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top\boldsymbol{O} \leq \boldsymbol{c}^\top. \end{aligned}$$

The above dual problem has a feasible solution if and only if $\boldsymbol{c} \geq \boldsymbol{0}$. Since any feasible solution to the dual is also optimal, the dual has an optimal solution if and only if $\boldsymbol{c} \geq \boldsymbol{0}$. Therefore, by the duality theorem, the primal problem has a solution if and only if $\boldsymbol{c} \geq \boldsymbol{0}$.

If the solution to the dual exists, then the optimal value of the objective function in the primal is equal to that of the dual, which is clearly 0. In this case, $\boldsymbol{0}$ is optimal, since $\boldsymbol{c}^\top\boldsymbol{0} = 0$.

17.17

Consider the primal problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{0}^\top\boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0}, \end{aligned}$$

and its corresponding dual

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{y}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{y}^\top \boldsymbol{A} \leq \boldsymbol{0} \\ & \boldsymbol{y} \geq \boldsymbol{0}. \end{aligned}$$

$\Rightarrow$: By assumption, there exists a feasible solution to the primal problem. Note that any feasible solution is also optimal, and has objective function value 0. Suppose $\boldsymbol{y}$ satisfies $\boldsymbol{A}^\top \boldsymbol{y} \leq \boldsymbol{0}$ and $\boldsymbol{y} \geq \boldsymbol{0}$. Then, $\boldsymbol{y}$ is a feasible solution to the dual. Therefore, by the Weak Duality Lemma, $\boldsymbol{b}^\top \boldsymbol{y} \leq 0$.

$\Leftarrow$: Note that the feasible region for the dual is nonempty, since $\boldsymbol{0}$ is a feasible point. Also, by assumption, $\boldsymbol{0}$ is an optimal solution, since any other feasible point $\boldsymbol{y}$ satisfies $\boldsymbol{b}^\top \boldsymbol{y} \leq \boldsymbol{b}^\top \boldsymbol{0} = 0$. Hence, by the duality theorem, the primal problem has an (optimal) feasible solution.

17.18

a. The dual is

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{y}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{y}^\top \boldsymbol{A} \leq \boldsymbol{0}^\top. \end{aligned}$$

b. The feasible set of the dual problem is always nonempty, because $\boldsymbol{0}$ is clearly guaranteed to be feasible.

c. Suppose $\boldsymbol{y}$ is feasible in the dual. Then, by assumption, $\boldsymbol{b}^\top \boldsymbol{y} \leq 0$. But the point $\boldsymbol{0}$ is feasible and has objective function value 0. Hence, $\boldsymbol{0}$ is optimal in the dual.

d. By parts b and c, the dual has an optimal feasible solution. Hence, by the duality theorem, the primal problem also has an optimal feasible solution.

e. By assumption, there exists a feasible solution to the primal problem. Note that any feasible solution in the primal has objective function value 0 (and hence so does the given solution). Suppose $\boldsymbol{y}$ satisfies $\boldsymbol{A}^\top \boldsymbol{y} \leq \boldsymbol{0}$. Then, $\boldsymbol{y}$ is a feasible solution to the dual. Therefore, by weak duality, $\boldsymbol{b}^\top \boldsymbol{y} \leq 0$.

17.19

Consider the primal problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{y}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{y}^\top \boldsymbol{A} = \boldsymbol{0} \\ & \boldsymbol{y} \geq \boldsymbol{0} \end{aligned}$$

and its corresponding dual

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{0}^\top \boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}. \end{aligned}$$

$\Rightarrow$: By assumption, there exists a feasible solution to the dual problem. Note that any feasible solution is also optimal, and has objective function value 0. Suppose $\boldsymbol{y}$ satisfies $\boldsymbol{A}^\top \boldsymbol{y} = \boldsymbol{0}$ and $\boldsymbol{y} \geq \boldsymbol{0}$. Then, $\boldsymbol{y}$ is a feasible solution to the primal. Therefore, by the Weak Duality Lemma, $\boldsymbol{b}^\top \boldsymbol{y} \geq 0$.

$\Leftarrow$: Note that the feasible region for the primal is nonempty, since $\boldsymbol{0}$ is a feasible point. Also, by assumption, $\boldsymbol{0}$ is an optimal solution, since any other feasible point $\boldsymbol{y}$ satisfies $\boldsymbol{b}^\top \boldsymbol{y} \geq \boldsymbol{b}^\top \boldsymbol{0} = 0$. Hence, by the duality theorem, the dual problem has an (optimal) feasible solution.

17.20

Let $\boldsymbol{e} = [1, \ldots, 1]^\top$. Consider the primal problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{0}^\top \boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} \leq -\boldsymbol{e} \end{aligned}$$

and its corresponding dual

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{e}^\top \boldsymbol{y} \\ \text{subject to} \quad & \boldsymbol{y}^\top \boldsymbol{A} = \boldsymbol{0} \\ & \boldsymbol{y} \geq \boldsymbol{0}. \end{aligned}$$

$\Rightarrow$: Suppose there exists $\boldsymbol{Ax} < \mathbf{0}$. Then, the vector $\boldsymbol{x}' = \boldsymbol{x}/\min\{|(\boldsymbol{Ax})_i|\}$ is a feasible solution to the primal problem. Note that any feasible solution is also optimal, and has objective function value 0. Suppose $\boldsymbol{y}$ satisfies $\boldsymbol{A}^\top \boldsymbol{y} = \mathbf{0}$, $\boldsymbol{y} \geq \mathbf{0}$. Then, $\boldsymbol{y}$ is a feasible solution to the dual. Therefore, by the Weak Duality Lemma, $\boldsymbol{e}^\top \boldsymbol{y} \leq 0$. Since $\boldsymbol{y} \geq \mathbf{0}$, we conclude that $\boldsymbol{y} = \mathbf{0}$.

$\Leftarrow$: Suppose $\mathbf{0}$ is the only feasible solution to the dual. Then, $\mathbf{0}$ is clearly also optimal. Hence, by the duality theorem, the primal problem has an (optimal) feasible solution $\boldsymbol{x}$. Since $\boldsymbol{Ax} \leq -\boldsymbol{e}$ and $-\boldsymbol{e} < \mathbf{0}$, we get $\boldsymbol{Ax} < \mathbf{0}$.

17.21

a. Rewrite the primal as

$$\begin{aligned} \text{minimize} \quad & (-\boldsymbol{e})^\top \boldsymbol{x} \\ \text{subject to} \quad & (\boldsymbol{P} - \boldsymbol{I})^\top \boldsymbol{x} = \mathbf{0} \\ & \boldsymbol{x} \geq \mathbf{0}. \end{aligned}$$

By asymmetric duality, the dual is

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{\lambda}^\top \mathbf{0} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top (\boldsymbol{P} - \boldsymbol{I})^\top \leq -\boldsymbol{e}^\top. \end{aligned}$$

b. To make the notation simpler, we rewrite the dual as:

$$\begin{aligned} \text{maximize} \quad & 0 \\ \text{subject to} \quad & (\boldsymbol{P} - \boldsymbol{I})\boldsymbol{y} \geq \boldsymbol{e}. \end{aligned}$$

Suppose the dual is feasible. Then, there exists a $\boldsymbol{y}$ such that $\boldsymbol{Py} \geq \boldsymbol{y} + \boldsymbol{e} > \boldsymbol{y}$. Let y_i be the largest element of $\boldsymbol{y}$, and $\boldsymbol{p}^{(i)\top}$ the ith row of $\boldsymbol{P}$. Then, $\boldsymbol{p}^{(i)\top}\boldsymbol{y} > y_i$. But, by definition of y_i, $\boldsymbol{y} \leq y_i \boldsymbol{e}$. Hence, $\boldsymbol{p}^{(i)\top}\boldsymbol{y} \leq y_i \boldsymbol{p}^{(i)\top}\boldsymbol{e} = y_i$, which contradicts the inequality $\boldsymbol{p}^{(i)\top}\boldsymbol{y} > y_i$. Hence, the dual is not feasible.

c. The primal is certainly feasible, because $\mathbf{0}$ is a feasible point. Therefore, by part b and strong duality, the primal must also be unbounded.

d. Because 0 is an achievable objective function value (it is the objective function value of $\mathbf{0}$), and the problem is unbounded, we deduce that 1 is also achievable. Hence, there exists a feasible $\boldsymbol{x}$ such that $\boldsymbol{x}^\top \boldsymbol{e} = 1$. This proves the desired result.

17.22

Write the LP problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{Ax} \geq \boldsymbol{b} \\ & \boldsymbol{x} \geq \mathbf{0} \end{aligned}$$

and the corresponding dual problem

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{\lambda}^\top \boldsymbol{b} \\ \text{subject to} \quad & \boldsymbol{\lambda}^\top \boldsymbol{A} \leq \boldsymbol{c}^\top \\ & \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned}$$

By a theorem on duality, if we can find feasible points $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ for the primal and dual, respectively, such that $\boldsymbol{c}^\top \boldsymbol{x} = \boldsymbol{\lambda}^\top \boldsymbol{b}$, then $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ are optimal for their respective problems. We can rewrite the previous set of relations as

$$\begin{bmatrix} -\boldsymbol{c}^\top & \boldsymbol{b}^\top \\ \boldsymbol{c}^\top & -\boldsymbol{b}^\top \\ \boldsymbol{A} & \mathbf{0} \\ \boldsymbol{I}_n & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{A}^\top \\ \mathbf{0} & \boldsymbol{I}_m \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ \boldsymbol{b} \\ \mathbf{0} \\ -\boldsymbol{c} \\ \mathbf{0} \end{bmatrix}.$$

Therefore, writing the above as $\hat{\boldsymbol{A}}\boldsymbol{y} \geq \hat{\boldsymbol{b}}$, where $\hat{\boldsymbol{A}} \in \mathbb{R}^{(2m+2n+2)\times(m+n)}$ and $\hat{\boldsymbol{b}} \in \mathbb{R}^{(2m+2n+2)}$, we have that the first n components of $\boldsymbol{\phi}((2m+2n+2),(m+n),\hat{\boldsymbol{A}},\hat{\boldsymbol{b}})$ is a solution to the given linear programming problem.

17.23

a. Consider the dual; $\boldsymbol{b}$ does not appear in the constraint (but it does appear in the dual objective function). Thus, provided the level sets of the dual objective function do not exactly align with one of the faces of the constraint set (polyhedron), the optimal dual vector will not change if we perturb $\boldsymbol{b}$ very slightly. Now, by the duality theorem, $z(\boldsymbol{b}) = \boldsymbol{\lambda}^\top \boldsymbol{b}$. Because $\boldsymbol{\lambda}$ is constant in a neighborhood of $\boldsymbol{b}$, we deduce that $\nabla z(\boldsymbol{b}) = \boldsymbol{\lambda}$.

b. By part a, we deduce that the optimal objective function value will change by $3\Delta b_1$.

17.24

a. Weak duality lemma: if $\boldsymbol{x}_0$ and $\boldsymbol{y}_0$ are feasible points in the primal and dual, respectively, then $f_1(\boldsymbol{x}_0) \geq f_2(\boldsymbol{y}_0)$.

Proof: Because $\boldsymbol{y}_0 \geq \boldsymbol{0}$ and $\boldsymbol{A}\boldsymbol{x}_0 - \boldsymbol{b} \leq \boldsymbol{0}$, we have $\boldsymbol{y}_0^\top(\boldsymbol{A}\boldsymbol{x}_0 - \boldsymbol{b}) \leq 0$. Therefore,

$$\begin{aligned} f_1(\boldsymbol{x}_0) &\geq f_1(\boldsymbol{x}_0) + \boldsymbol{y}_0^\top(\boldsymbol{A}\boldsymbol{x}_0 - \boldsymbol{b}) \\ &= \frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{x}_0 + \boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{x}_0 - \boldsymbol{y}_0^\top \boldsymbol{b}. \end{aligned}$$

Now, we know that

$$\frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{x}_0 + \boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{x}_0 \geq \frac{1}{2}\boldsymbol{x}^{*\top}\boldsymbol{x}^* + \boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{x}^*,$$

where $\boldsymbol{x}^* = -\boldsymbol{A}^\top \boldsymbol{y}_0$. Hence,

$$\frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{x}_0 + \boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{x}_0 \geq \frac{1}{2}\boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0 - \boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0 = -\frac{1}{2}\boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0.$$

Combining this with the above, we have

$$\begin{aligned} f_1(\boldsymbol{x}_0) &\geq -\frac{1}{2}\boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0 - \boldsymbol{y}_0^\top \boldsymbol{b} \\ &= f_2(\boldsymbol{x}_0). \end{aligned}$$

Alternatively, notice that

$$\begin{aligned} f_1(\boldsymbol{x}_0) - f_2(\boldsymbol{y}_0) &= \frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{x}_0 + \frac{1}{2}\boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0 + \boldsymbol{b}^\top \boldsymbol{y}_0 \\ &\geq \frac{1}{2}\boldsymbol{x}_0^\top \boldsymbol{x}_0 + \frac{1}{2}\boldsymbol{y}_0^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{y}_0 + \boldsymbol{x}_0^\top \boldsymbol{A}^\top \boldsymbol{y}_0 \\ &= \frac{1}{2}\|\boldsymbol{x}_0 + \boldsymbol{A}^\top \boldsymbol{y}_0\|^2 \\ &\geq 0. \end{aligned}$$

b. Suppose $f_1(\boldsymbol{x}_0) = f_2(\boldsymbol{y}_0)$ for feasible points $\boldsymbol{x}_0$ and $\boldsymbol{y}_0$. Let $\boldsymbol{x}$ be any feasible point in the primal. Then, by part a, $f_1(\boldsymbol{x}) \geq f_2(\boldsymbol{y}_0) = f_1(\boldsymbol{x}_0)$. Hence $\boldsymbol{x}_0$ is optimal in the primal.

Similarly, let $\boldsymbol{y}$ be any feasible point in the dual. Then, by part a, $f_2(\boldsymbol{y}) \leq f_1(\boldsymbol{x}_0) = f_2(\boldsymbol{y}_0)$. Hence $\boldsymbol{y}_0$ is optimal in the dual.

18. Non-Simplex Methods

18.1

The following is a MATLAB function that implements the affine scaling algorithm.

```
function [x,N] = affscale(c,A,b,u,options);
%               AFFSCALE(c,A,b,u);
%               AFFSCALE(c,A,b,u,options);
```

```
%
%               x = AFFSCALE(c,A,b,u);
%               x = AFFSCALE(c,A,b,u,options);
%
%               [x,N] = AFFSCALE(c,A,b,u);
%               [x,N] = AFFSCALE(c,A,b,u,options);
%
%AFFSCALE(c,A,b,u) solves the following linear program using the
%affine scaling Method:
%   min c'x  subject to Ax=b, x>=0,
%where u is a strictly feasible initial solution.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required cost value.
%OPTIONS(14) = max number of iterations.
%OPTIONS(18) = alpha.

if nargin ~= 5
  options = [];
  if nargin ~= 4
    disp('Wrong number of arguments.');
    return;
  end
end
xnew=u;

if length(options) >= 14
  if options(14)==0
    options(14)=1000*length(xnew);
  end
else
  options(14)=1000*length(xnew);
end

%if length(options) < 18
  options(18)=0.99; %optional step size
%end

%clc;
format compact;
format short e;

options = foptions(options);
print = options(1);
epsilon_x = options(2);
epsilon_f = options(3);
max_iter=options(14);
alpha=options(18);

n=length(c);
m=length(b);

for k = 1:max_iter,

  xcurr=xnew;
  D = diag(xcurr);
```

```
  Abar = A*D;
  Pbar = eye(n) - Abar'*inv(Abar*Abar')*Abar;
  d = -D*Pbar*D*c;
  if d ~= zeros(n,1),
    nonzd = find(d<0);
    r = min(-xcurr(nonzd)./d(nonzd));
  else
    disp('Terminating: d = 0');
    break;
  end
  xnew = xcurr+alpha*r*d;

  if print,
    disp('Iteration number k =')
    disp(k);  %print iteration index k
    disp('alpha_k =');
    disp(alpha*r);  %print alpha_k
    disp('New point =');
    disp(xnew'); %print new point
  end %if

  if norm(xnew-xcurr) <= epsilon_x*norm(xcurr)
    disp('Terminating: Relative difference between iterates <');
    disp(epsilon_x);
    break;
  end %if

  if abs(c'*(xnew-xcurr)) < epsilon_f*abs(c'*xcurr),
    disp('Terminating: Relative change in objective function < ');
    disp(epsilon_f);
    break;
  end %if

  if k == max_iter
    disp('Terminating with maximum number of iterations');
  end %if
end %for

if nargout >= 1
  x=xnew;
  if nargout == 2
    N=k;
  end
else
  disp('Final point =');
  disp(xnew');
  disp('Number of iterations =');
  disp(k);
end %if
%-------------------------------------------------------------------
```

We now apply the affine scaling algorithm to the problem in Example 16.2, as follows:

```
>> A=[1 0 1 0 0; 0 1 0 1 0; 1 1 0 0 1];
>> b=[4;6;8];
>> c=[-2;-5;0;0;0];
>> u=[2;3;2;3;3];
>> options(1)=0;
>> options(2)=10^(-7);
>> options(3)=10^(-7);
```

```
>> affscale(c,A,b,u,options);
Terminating: Relative difference between iterates <
   1.0000e-07
Final point =
   2.0000e+00   6.0000e+00   2.0000e+00   1.0837e-09   1.7257e-08
Number of iterations =
     8
```

The result obtained after 8 iterations as indicated above agrees with the solution in Example 16.2: $[2, 6, 2, 0, 0]^\top$.

18.2

The following is a MATLAB routine that implements the two-phase affine scaling method, using the MATLAB function from Exercise 18.1.

```
function [x,N]=tpaffscale(c,A,b,options)
% March 28, 2000
%
%               TPAFFSCALE(c,A,b);
%               TPAFFSCALE(c,A,b,options);
%
%               x = TPAFFSCALE(c,A,b);
%               x = TPAFFSCALE(c,A,b,options);
%
%               [x,N] = TPAFFSCALE(c,A,b);
%               [x,N] = TPAFFSCALE(c,A,b,options);
%
%TPAFFSCALE(c,A,b) solves the following linear program using the
%Two-Phase Affine Scaling Method:
%  min c'x  subject to Ax=b, x>=0.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required cost value.
%OPTIONS(14) = max number of iterations.
%OPTIONS(18) = alpha.

if nargin ~= 4
  options = [];
    if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

%clc;
format compact;
format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

%Phase I
if print,
  disp(' ');
```

```
  disp('Phase I');
  disp('-------');
end

u = rand(n,1);
v = b-A*u;
if v ~= zeros(m,1),
  u = affscale([zeros(1,n),1]',[A v],b,[u' 1]',options);
  u(n+1) = [];
end

if print
  disp(' ')
  disp('Initial condition for Phase II:')
  disp(u)
end

if u(n+1) < options(2),
  %Phase II
  u(n+1) = [];
  if print
    disp(' ');
    disp('Phase II');
    disp('--------');
    disp('Initial condition for Phase II:');
    disp(u);
  end
  [x,N]=affscale(c,A,b,u,options);
  if nargout == 0
    disp('Final point =');
    disp(x');
    disp('Number of iterations =');
    disp(N);
  end %if
else
  disp('Terminating: problem has no feasible solution.');
end
%-------------------------------------------------------------------
```

We now apply the above MATLAB routine to the problem in Example 16.5, as follows:

```
>> A=[1 1 1 0; 5 3 0 -1];
>> b=[4;8];
>> c=[-3;-5;0;0];
>> options(1)=0;
>> tpaffscale(c,A,b,options);
Terminating: Relative difference between iterates <
   1.0000e-07
Terminating: Relative difference between iterates <
   1.0000e-07
Final point =
   4.0934e-09   4.0000e+00   9.4280e-09   4.0000e+00
Number of iterations =
     7
```

The result obtained above agrees with the solution in Example 16.5: $[0, 4, 0, 4]^\top$.

18.3

The following is a MATLAB routine that implements the affine scaling method applied to LP problems of the form given in the question by converting the given problem in Karmarkar's artificial form and then using the MATLAB function from Exercise 18.1.

```
function [x,N]=karaffscale(c,A,b,options)
%
%               KARAFFSCALE(c,A,b);
%               KARAFFSCALE(c,A,b,options);
%
%               x = KARAFFSCALE(c,A,b);
%               x = KARAFFSCALE(c,A,b,options);
%
%               [x,N] = KARAFFSCALE(c,A,b);
%               [x,N] = KARAFFSCALE(c,A,b,options);
%
%KARAFFSCALE(c,A,b) solves the following linear program using the
%Affine Scaling Method:
%  min c'x  subject to Ax>=b, x>=0.
%We use Karmarkar's artificial problem to convert the above problem into
%a form usable by the affine scaling method.
%The second variant allows a vector of optional parameters to be
%defined:
%OPTIONS(1) controls how much display output is given; set
%to 1 for a tabular display of results (default is no display: 0).
%OPTIONS(2) is a measure of the precision required for the final point.
%OPTIONS(3) is a measure of the precision required cost value.
%OPTIONS(14) = max number of iterations.
%OPTIONS(18) = alpha.

if nargin ~= 4
  options = [];
    if nargin ~= 3
    disp('Wrong number of arguments.');
    return;
  end
end

%clc;
format compact;
format short e;

options = foptions(options);
print = options(1);

n=length(c);
m=length(b);

%Convert to Karmarkar's aftificial problem
x0 = ones(n,1);
l0 = ones(m,1);
u0 = ones(n,1);
v0 = ones(m,1);

AA = [
c' -b' zeros(1,n) zeros(1,m) (-c'*x0+b'*l0);
A zeros(m,m) zeros(m,n) -eye(m) (b-A*x0+v0);
zeros(n,n) A' eye(n) zeros(n,m) (c-A'*l0)
];
bb = [0; b; c];
cc = [zeros(2*m+2*n,1); 1];
y0 = [x0; l0; u0; v0; 1];

[y,N]=affscale(cc,AA,bb,y0,options);
```

```
if cc'*y <= options(3),
  x = y(1:n);
  if nargout == 0
    disp('Final point =');
    disp(x');
    disp('Final cost =');
    disp(c'*x);
    disp('Number of iterations =');
    disp(N);
  end %if
else
  disp('Terminating: problem has no optimal feasible solution.');
end
```

We now apply the above MATLAB routine to the problem in Example 15.15, as follows:

```
>> c=[-3;-5];
>> A=[-1 -5; -2 -1; -1 -1];
>> b=[-40;-20;-12];
>> options(2)=10^(-4);
>> karaffscale(c,A,b,options);
Terminating: Relative difference between iterates <
   1.0000e-04
Final point =
   5.1992e+00   6.5959e+00
Final cost =
  -4.8577e+01
Number of iterations =
     3
```

The solution from Example 15.15 is $[5, 7]^\top$. The accuracy of the result obtained above is disappointing. We believe that the inaccuracy here may be caused by our particularly simple numerical implementation of the affine scaling method. This illustrates the numerical issues that must be dealt with in any practically useful implementation of the affine scaling method.

18.4

a. Suppose $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{T}(\boldsymbol{y})$. Then, $T_i(\boldsymbol{x}) = T_i(\boldsymbol{y})$ for $i = 1, \ldots, n+1$. Note that for $i = 1, \ldots, n$, $T_i(\boldsymbol{x}) = (x_i/a_i)T_{n+1}(\boldsymbol{x})$ and $T_i(\boldsymbol{y}) = (y_i/a_i)T_{n+1}(\boldsymbol{y})$. Therefore,

$$T_i(\boldsymbol{x}) = (x_i/a_i)T_{n+1}(\boldsymbol{x}) = T_i(\boldsymbol{y}) = (y_i/a_i)T_{n+1}(\boldsymbol{y}) = (y_i/a_i)T_{n+1}(\boldsymbol{x}),$$

which implies that $x_i = y_i$, $i = 1, \ldots, n$. Hence $\boldsymbol{x} = \boldsymbol{y}$.
b. Let $\boldsymbol{y} \in \{\boldsymbol{x} \in \Delta : x_{n+1} > 0\}$. Hence $y_{n+1} > 0$. Define $\boldsymbol{x} = [x_1, \ldots, x_n]^\top$ by $x_i = a_i y_i / y_{n+1}$, $i = 1, \ldots, n$. Then, $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{y}$. To see this, note that

$$T_{n+1}(\boldsymbol{x}) = \frac{1}{y_1/y_{n+1} + \cdots + y_n/y_{n+1} + 1} = \frac{y_{n+1}}{y_1 + \cdots y_n + y_{n+1}} = y_{n+1}.$$

Also, for $i = 1, \ldots, n$,

$$T_i(\boldsymbol{x}) = (y_i/y_{n+1})T_{n+1}(\boldsymbol{x}) = y_i.$$

c. An immediate consequence of the solution to part b.

d. We have

$$T_{n+1}(\boldsymbol{a}) = \frac{1}{a_1/a_1 + \cdots + a_n/a_n + 1} = \frac{1}{n+1},$$

and, for $i = 1, \ldots, n$,

$$T_i(\boldsymbol{a}) = (a_i/a_i)T_{n+1}(\boldsymbol{a}) = \frac{1}{n+1}.$$

e. Since $\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{x})$, we have that for $i = 1, \ldots, n$, $y_i = (x_i/a_i)y_{n+1}$. Therefore, $x_i' = y_i a_i = x_i y_{n+1}$, which implies that $\boldsymbol{x}' = y_{n+1}\boldsymbol{x}$. Hence, $\boldsymbol{A}\boldsymbol{x}' = y_{n+1}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}y_{n+1}$.

18.5

Let $\boldsymbol{x} \in \mathbb{R}^n$, and $\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{x})$. Let $\boldsymbol{a}_i$ be the ith column of $\boldsymbol{A}$, $i = 1, \ldots, n$. As in the hint, let $\boldsymbol{A}'$ be given by

$$\boldsymbol{A}' = [a_1\boldsymbol{a}_1, \ldots, a_n\boldsymbol{a}_n, -\boldsymbol{b}]\,.$$

Then,

$$\begin{aligned}
\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} &\Leftrightarrow \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0} \\
&\Leftrightarrow [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n, -\boldsymbol{b}] \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ 1 \end{bmatrix} = \boldsymbol{0} \\
&\Leftrightarrow [a_1\boldsymbol{a}_1, \ldots, a_n\boldsymbol{a}_n, -\boldsymbol{b}] \begin{bmatrix} x_1/a_1 \\ \vdots \\ x_n/a_n \\ 1 \end{bmatrix} = \boldsymbol{0} \\
&\Leftrightarrow \boldsymbol{A}' \begin{bmatrix} (x_1/a_1)y_{n+1} \\ \vdots \\ (x_n/a_n)y_{n+1} \\ y_{n+1} \end{bmatrix} = \boldsymbol{0} \\
&\Leftrightarrow \boldsymbol{A}'\boldsymbol{y} = \boldsymbol{0}.
\end{aligned}$$

18.6

The result follows from Exercise 18.5 by setting $\boldsymbol{A} := \boldsymbol{c}^\top$ and $\boldsymbol{b} := 0$.

18.7

Consider the set $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{e}^\top\boldsymbol{x} = 1, \boldsymbol{x} \geq \boldsymbol{0}, x_1 = 0\}$, which can be written as $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}$, where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{e}^\top \\ \boldsymbol{e}_1^\top \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

with $\boldsymbol{e} = [1, \ldots, 1]^\top$, $\boldsymbol{e}_1 = [1, 0, \ldots, 0]^\top$. Let $\boldsymbol{a}_0 = \boldsymbol{e}/n$. By Exercise 12.20, the closest point on the set $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$ to the point $\boldsymbol{a}_0$ is

$$\boldsymbol{x}^* = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{a}_0) + \boldsymbol{a}_0 = \left[0, \frac{1}{n-1}, \ldots, \frac{1}{n-1}\right]^\top.$$

Since $\boldsymbol{x}^* \in \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\} \subset \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$, the point $\boldsymbol{x}^*$ is also the closest point on the set $\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}$ to the point $\boldsymbol{a}_0$.

Let $r = \|\boldsymbol{a}_0 - \boldsymbol{x}^*\|$. Then, the sphere of radius r is inscribed in Δ. Note that

$$r = \|\boldsymbol{a}_0 - \boldsymbol{x}^*\| = \frac{1}{\sqrt{n(n-1)}}.$$

Hence, the radius of the largest sphere inscribed in Δ is larger than or equal to $1/\sqrt{n(n-1)}$. It remains to show that this largest radius is less than or equal to $1/\sqrt{n(n-1)}$. To this end, we show that this largest radius is less than or equal to $1/\sqrt{n(n-1)} + \varepsilon$ for any $\varepsilon > 0$. For this, it suffices to show that the sphere of

radius $1/\sqrt{n(n-1)}+\varepsilon$ is not inscribed in Δ. To show this, consider the point

$$\begin{aligned}\boldsymbol{x} &= \boldsymbol{x}^* + \varepsilon\frac{\boldsymbol{x}^* - \boldsymbol{a}_0}{\|\boldsymbol{x}^* - \boldsymbol{a}_0\|} \\ &= \boldsymbol{x}^* + \varepsilon\sqrt{n(n-1)}(\boldsymbol{x}^* - \boldsymbol{a}_0) \\ &= \boldsymbol{x}^* + \varepsilon\left[-\sqrt{\frac{n-1}{n}}, \frac{1}{\sqrt{n(n-1)}}, \ldots, \frac{1}{\sqrt{n(n-1)}}\right]^\top.\end{aligned}$$

It is easy to verify that the point $\boldsymbol{x}$ above is on the sphere of radius $1/\sqrt{n(n-1)}+\varepsilon$. However, clearly the first component of $\boldsymbol{x}$ is negative. Therefore, the sphere of radius $1/\sqrt{n(n-1)}+\varepsilon$ is not inscribed in Δ. Our proof is thus completed.

18.8

We first consider the constraints. We claim that $\boldsymbol{x} \in \Delta \Leftrightarrow \bar{\boldsymbol{x}} \in \Delta$. To see this, note that if $\boldsymbol{x} \in \Delta$, then $\boldsymbol{e}^\top\boldsymbol{x} = 1$ and hence

$$\boldsymbol{e}^\top\bar{\boldsymbol{x}} = \boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x}/\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x} = 1,$$

which means that $\bar{\boldsymbol{x}} \in \Delta$. The same argument can be used for the converse. Next, we claim $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \Leftrightarrow \boldsymbol{A}\boldsymbol{D}\bar{\boldsymbol{x}} = \boldsymbol{0}$. To see this, write

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{x} = \boldsymbol{A}\boldsymbol{D}\bar{\boldsymbol{x}}(\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x}).$$

Since $\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x} > 0$, we have $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \Leftrightarrow \boldsymbol{A}\boldsymbol{D}\bar{\boldsymbol{x}} = \boldsymbol{0}$. Finally, we claim that if $\boldsymbol{x}^*$ is an optimal solution to the original problem, then $\bar{\boldsymbol{x}}^* = \boldsymbol{U}(\boldsymbol{x}^*)$ is an optimal solution to the transformed problem. To see this, recall that the problem in a Karmarkar's restricted problem, and hence by Assumption (B) we have $\boldsymbol{c}^\top\boldsymbol{x}^* = 0$. We now note that the minimum value of the objective function $\boldsymbol{c}^\top\boldsymbol{D}\bar{\boldsymbol{x}}$ in the transformed problem is zero. This is because $\boldsymbol{c}^\top\boldsymbol{D}\bar{\boldsymbol{x}} = \boldsymbol{c}^\top\boldsymbol{x}/\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x}$, and $\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x} > 0$. Finally, we observe that at the point, $\bar{\boldsymbol{x}}^* = \boldsymbol{U}(\boldsymbol{x}^*)$ the objective function value for the transformed problem is zero. Indeed,

$$\boldsymbol{c}^\top\boldsymbol{D}\bar{\boldsymbol{x}}^* = \boldsymbol{c}^\top\boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{x}^*/\boldsymbol{e}^\top\boldsymbol{D}^{-1}\boldsymbol{x}^* = 0.$$

Therefore, the two problems are equivalent.

18.9

Let $\boldsymbol{v} \in \mathbb{R}^{m+1}$ be such that $\boldsymbol{v}^\top\boldsymbol{B} = \boldsymbol{0}^\top$. We will show that

$$\boldsymbol{v}^\top\begin{bmatrix}\boldsymbol{A} \\ \boldsymbol{e}^\top\end{bmatrix} = \boldsymbol{0}^\top$$

and hence $\boldsymbol{v} = \boldsymbol{0}$ by virtue of the assumption that

$$\text{rank}\begin{bmatrix}\boldsymbol{A} \\ \boldsymbol{e}^\top\end{bmatrix} = m+1.$$

This in turn gives us the desired result.

To proceed, write $\boldsymbol{v}$ as

$$\boldsymbol{v} = \begin{bmatrix}\boldsymbol{u} \\ v_{m+1}\end{bmatrix}$$

where $\boldsymbol{u} \in \mathbb{R}^m$ constitute the first m components of $\boldsymbol{v}$. Then,

$$\boldsymbol{v}^\top\boldsymbol{B} = \boldsymbol{u}^\top\boldsymbol{A}\boldsymbol{D} + v_{m+1}\boldsymbol{e}^\top = \boldsymbol{0}^\top.$$

Postmultiplying the above by $\boldsymbol{e}$, and using the facts that $\boldsymbol{D}\boldsymbol{e} = \boldsymbol{x}_0$, $\boldsymbol{A}\boldsymbol{x}_0 = \boldsymbol{0}$, and $\boldsymbol{e}^\top\boldsymbol{e} = n$, we get

$$\boldsymbol{u}^\top\boldsymbol{A}\boldsymbol{x}_0 + v_{m+1}n = v_{m+1}n = 0,$$

which implies that $v_{m+1} = 0$. Hence, $\boldsymbol{u}^\top \boldsymbol{A}\boldsymbol{D} = \boldsymbol{0}^\top$, which after postmultiplying by $\boldsymbol{D}^{-1}$ gives $\boldsymbol{u}^\top \boldsymbol{A} = \boldsymbol{0}^\top$. Hence,

$$\boldsymbol{v}^\top \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{e}^\top \end{bmatrix} = \boldsymbol{0}^\top,$$

which implies that $\boldsymbol{v} = \boldsymbol{0}$. Hence, $\operatorname{rank} \boldsymbol{B} = m + 1$.

18.10

We proceed by induction. For $k = 0$, the result is true because $\boldsymbol{x}^{(0)} = \boldsymbol{a}_0$. Now suppose that $\boldsymbol{x}^{(k)}$ is a strictly interior point of Δ. We first show that $\bar{\boldsymbol{x}}^{(k+1)}$ is a strictly interior point. Now,

$$\bar{\boldsymbol{x}}^{(k+1)} = \boldsymbol{a}_0 - \alpha r \hat{\boldsymbol{c}}^{(k)}.$$

Then, since $\alpha \in (0, 1)$ and $\|\hat{\boldsymbol{c}}^{(k)}\| = 1$, we have

$$\|\bar{\boldsymbol{x}}^{(k+1)} - \boldsymbol{a}_0\| \leq |\alpha r| \|\hat{\boldsymbol{c}}^{(k)}\| < r.$$

Since r is the radius of the largest sphere inscribed in Δ, $\bar{\boldsymbol{x}}^{(k+1)}$ is a strictly interior point of Δ. To complete the proof, we write

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{U}_k^{-1}(\bar{\boldsymbol{x}}^{(k+1)}) = \frac{\boldsymbol{D}_k \bar{\boldsymbol{x}}^{(k+1)}}{\boldsymbol{e}^\top \boldsymbol{D}_k \bar{\boldsymbol{x}}^{(k+1)}}$$

We already know that $\boldsymbol{x}^{(k+1)} \in \Delta$. It therefore remains to show that it is strictly interior, i.e., $\boldsymbol{x}^{(k+1)} > \boldsymbol{0}$. To see this, note that $\boldsymbol{e}^\top \boldsymbol{D}_k \bar{\boldsymbol{x}}^{(k+1)} > 0$. Furthermore, we can write

$$\boldsymbol{D}_k \bar{\boldsymbol{x}}^{(k+1)} = \begin{bmatrix} \bar{x}_1^{(k+1)} x_1^{(k)} \\ \vdots \\ \bar{x}_n^{(k+1)} x_n^{(k)} \end{bmatrix}.$$

Since $\boldsymbol{x}^{(k)} = [x_1^{(k)}, \ldots, x_n^{(k)}]^\top > \boldsymbol{0}$ by the induction hypothesis, and $\bar{\boldsymbol{x}}^{(k+1)} = [\bar{x}_1^{(k+1)}, \ldots \bar{x}_n^{(k+1)}]^\top > \boldsymbol{0}$ by the above, $\boldsymbol{x}^{(k+1)} > \boldsymbol{0}$ and hence it is a strictly interior point of Δ.

19. Integer Linear Programming

19.1

The result follows from the simple observation that if $\boldsymbol{M}$ is a submatrix of $\boldsymbol{A}$, then any submatrix of $\boldsymbol{M}$ is also a submatrix of $\boldsymbol{A}$. Therefore, any property involving all submatrices of $\boldsymbol{A}$ also applies to all submatrices of $\boldsymbol{M}$.

19.2

The result follows from the simple observation that any submatrix of $\boldsymbol{A}^\top$ is the transpose of a submatrix of $\boldsymbol{A}$, and that the determinant of the transpose of a matrix equals the determinant of the original matrix.

19.3

The claim that $\boldsymbol{A}$ is totally unimodular if $[\boldsymbol{A}, \boldsymbol{I}]$ is totally unimodular follows from Exercise 19.1. To show the converse, suppose that $\boldsymbol{A}$ is totally unimodular. We will show that any $p \times p$ invertible submatrix of $[\boldsymbol{A}, \boldsymbol{I}]$, $p \leq \min(m, n)$, has determinant ± 1. We first note that any $p \times p$ invertible submatrix of $[\boldsymbol{A}, \boldsymbol{I}]$ that consists only of columns of $\boldsymbol{A}$ has determinant ± 1 because $\boldsymbol{A}$ is totally unimodular. Moreover, any $p \times p$ invertible submatrix of $\boldsymbol{I}$ has determinant 1.

Consider now a $p \times p$ invertible submatrix of $[\boldsymbol{A}, \boldsymbol{I}]$ composed of k columns of $\boldsymbol{A}$ and $p - k$ columns of $\boldsymbol{I}$. Without loss of generality, suppose that this submatrix is composed of the first p rows of $[\boldsymbol{A}, \boldsymbol{I}]$, the last k columns of $\boldsymbol{A}$, and the first $p - k$ columns of $\boldsymbol{I}$. (This choice of rows and columns is without loss of generality because we can exchange rows and columns to arrive at this form, and each exchange only changes the sign of the determinant.) We now proceed as in the proof of Proposition 19.1.

19.4

The result follows from these properties of determinants: (1) that exchanging columns only changes the sign

of the determinant; (2) the determinant of a block triangular matrix is the product of the determinants of the diagonal blocks; and (3) the determinant of the identity matrix is 1. See also Exercise 2.4.

19.5

The vectors $\boldsymbol{x}$ and $\boldsymbol{z}$ together satisfy

$$\boldsymbol{A}\boldsymbol{x} + \boldsymbol{z} = \boldsymbol{b},$$

which means that $\boldsymbol{z} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$. Because the right-hand side involves only integers, $\boldsymbol{z}$ is an integer vector.

19.6

The following MATLAB code generates the figures.

```
%-----------------------------------------------
% The vertices of the feasible set
x = [2/5 1; 2/5 -2/5]\[3; 1];
X = [0 0 x(1) 2.5];
Y = [0 3 x(2) 0];
fs=16; %Fontsize
% Draw the fesible set for x1 x2 \in R.
vi = convhull(X,Y);
plot(X,Y, 'o');
axis on; axis equal;
axis([-0.2 4.2 -0.2 3.2]);
hold on
fill (X(vi), Y(vi), 'b','facealpha', 0.2);
text(.1,.5,['\fontsize{48}\Omega'],'position', [1.5 1.25])
set(gca,'Fontsize',fs)
hold off
% The optimal solution has to be one of the extreme points
c = [-3 -4]';
% Draw the feasible set for the noninteger problem
figure
axis on; axis equal;
x = [-0.5:0.1:x(1)];
y1 = -x*0.4+3;
y2 = x-2.5;
fs=16; % Fontsize
plot(x,y1,'--b',x,y2,'--b','LineWidth',2);
axis([-0.2 4.2 -0.2 3.2]);
set(gca,'Fontsize',fs)
hold on
X = [zeros(1,4) ones(1,3) 2*ones(1,3) 3];
Y = [0:max(floor(Y)) 0:max(floor(Y)-1) 0:max(floor(Y)-1) 1];
plot(X,Y,'bls','LineWidth',2,...
                      'MarkerEdgeColor','k',...
                      'MarkerFaceColor','g',...
                      'MarkerSize',10)
% Plot of the cost function
xc = [-1:0.5:5];
yc = -0.75*xc+(14/4)*ones(1,length(xc));
yc0 = -0.75*xc+(17.5/4)*ones(1,length(xc));
fs=16; % Fontsize
plot(xc, yc, 'r', xc, yc0, 'o-k', 'LineWidth',2);
set(gca,'Fontsize',fs)
%text(.1,.5,['\fontsize{48}\Omega'],'position', [1.5 1.25])
[~,Xmin] = min(c'*[X; Y]);
str = sprintf('The maximizer is [%d, %d]'' and the maximum is %.4f]',...
        X(Xmin), Y(Xmin), -c'*[X(Xmin); Y(Xmin)]);
disp(str);
%-----------------------------------------------
```

19.7

It suffices to show the following claim: If we introduce the equation

$$x_i + \sum_{j=m+1}^{n} \lfloor y_{ij} \rfloor x_j + x_{n+1} = \lfloor y_{i0} \rfloor$$

into the original constraint, then the result holds. The reason this suffices is that the Gomory cut is obtained by subtracting this equation from an equation obtained by elementary row operations on $[\boldsymbol{A}, \boldsymbol{b}]$ (hence is equivalent to premultiplication by an invertible matrix).

To show the above claim, let x_{n+1} satisfy this constraint with an integer vector $\boldsymbol{x}$. Then,

$$x_i + \sum_{j=m+1}^{n} \lfloor y_{ij} \rfloor x_j + x_{n+1} = \lfloor y_{i0} \rfloor,$$

which implies that

$$x_{n+1} = \lfloor y_{i0} \rfloor - x_i - \sum_{j=m+1}^{n} \lfloor y_{ij} \rfloor x_j.$$

Because the right-hand side involves only integers, x_{n+1} is an integer.

19.8

If there is only one Gomory cut, then the result follows directly from Exercise 19.7. The general result follows by induction on the number of Gomory cuts, using Exercise 19.7 at each inductive step.

19.9

The result follows from Exercises 19.5 and 19.8.

19.10

The dual problem is:

$$\begin{aligned} \text{minimize} \quad & 3\lambda_1 + \lambda_2 \\ \text{subject to} \quad & \frac{2}{5}\lambda_1 + \frac{2}{5}\lambda_2 \geq 3 \\ & 1\lambda_1 - \frac{2}{5}\lambda_2 \geq 4 \\ & \lambda_1, \lambda_2 \geq 0 \\ & \lambda_1, \lambda_2 \in \mathbb{Z}. \end{aligned}$$

The problem is solved graphically using the same approach as in Example 19.5. We proceed by calculating the extreme points of the feasible set. We first assume that $\lambda_1, \lambda_2 \in \mathbb{R}$. The extreme points are calculated intersecting the given constraints, and they are:

$$\boldsymbol{\lambda}^{(1)} = [5, \frac{5}{2}]^\top, \quad \boldsymbol{\lambda}^{(2)} = [\frac{15}{2}, 0]^\top.$$

In Figure 24.10, we show the feasible set Ω for the case when $\lambda_1, \lambda_2 \in \mathbb{R}$.

Next we sketch the feasible set for the case when $\lambda_1, \lambda_2 \in \mathbb{Z}$ and solve the problem graphically. The graphical solution is depicted in Figure 24.11. We can see in Figure 24.11 that the optimal integer solution is

$$\boldsymbol{x}^* = [6, 2]^\top.$$

The following MATLAB code generates the figures.

```
%-------------------------------------------------------
% The vertices of the feasible set are:

x = [2/5 2/5;1 -2/5]\[3; 4];
X = [7.5 x(1) 20 20];
```

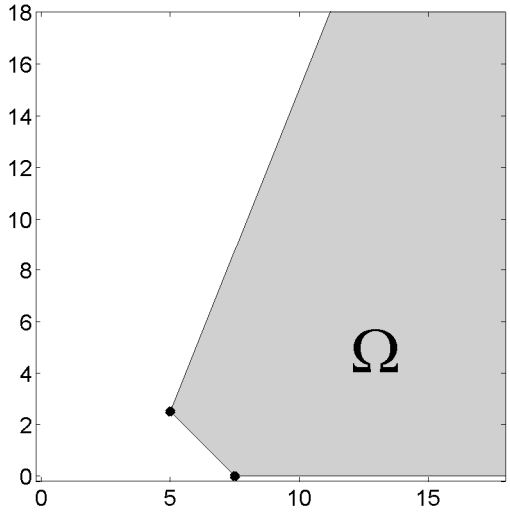


Figure 24.10 Feasible set Ω for the case when $\lambda_1, \lambda_2 \in \mathbb{R}$ in Example 19.5.

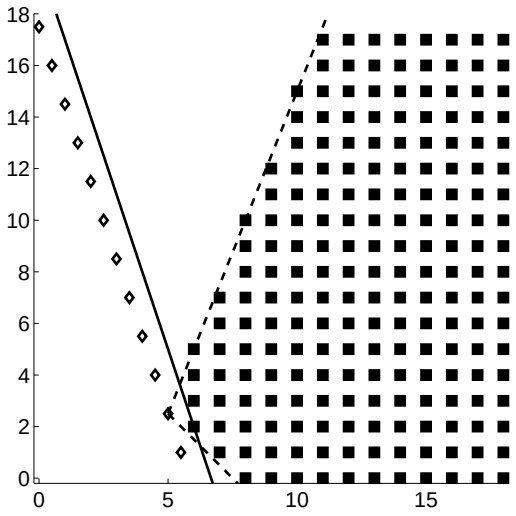


Figure 24.11 Real feasible set with $\lambda_1, \lambda_2 \in \mathbb{Z}$.

```
Y = [0 x(2) 0 40];
fs=16; % Fontsize
% Now we draw the set Omega, supposing x1 x2 \in R.
vi = convhull(X,Y);
plot(X(1:2),Y(1:2), 'rX', 'LineWidth',4);
axis on; axis equal;
axis([-.2 18 -0.2 18]);
set(gca,'Fontsize',fs)
%title('Feasible set supossing x1, x2 in R.','Fontsize',14,'Fontname','Avant-garde');
hold on
fill (X(vi), Y(vi), 'b','facealpha', 0.2);
text(.1,.5,['\fontsize{48}\Omega'],'position', [12 5])
hold off

% We now the optimal solution has to be one of the extreme points.
c = [3 1]';

%Now we draw the real feasible set for the problem.
figure
axis on; axis equal;
axis([-.2 18 -0.2 18]);
set(gca,'Fontsize',fs)
%title('Feasible set and cost function','Fontsize',14,'Fontname','Avant-garde');
hold on

X = [];
Y = [];

for i=1:18
     j=0;
     while ((j<=((i-4)*2.5)) && (j<18))
         if((j>=(7.5-i)))
            X = [X i];
            Y = [Y j];
         end
        j=j+1;
     end
end

plot(X,Y,'bls','LineWidth',1,...
                        'MarkerEdgeColor','k',...
                        'MarkerFaceColor','g',...
                        'MarkerSize',10)
x = [5:0.1:18];
y1 = (x-4)*2.5;
y2 = (7.5-x);

plot(x,y1,'--b',x,y2,'--b','LineWidth',2);
set(gca,'Fontsize',fs)
%text(.1,.5,['\fontsize{48}\Omega'],'position', [12 5])

%Plot of the cost function at level 17.5
xc = [-1:0.5:18];
yc = (35/2)*ones(1,length(xc))-3*xc;
plot(xc, yc, 'dk', 'LineWidth',2);

%Plot of the cost function
xc = [-1:0.5:18];
yc = (20)*ones(1,length(xc))-3*xc;
```

```
plot(xc, yc, 'r', 'LineWidth',2);

[~,Xmin] = min(c'*[X; Y]);

str = sprintf('The minimizer is [%d, %d]'' and the maximum is %.4f]',...
         X(Xmin), Y(Xmin), c'*[X(Xmin); Y(Xmin)]);
disp(str);
%-------------------------------------------------------
```

20. Problems with Equality Constraints

20.1

The feasible set consists of the points

$$\boldsymbol{x} = \begin{bmatrix} 2 \\ a \end{bmatrix}, \quad a \leq -1.$$

We next find the gradients:

$$\nabla h(\boldsymbol{x}) = \begin{bmatrix} 2(x_1 - 2) \\ 0 \end{bmatrix} \quad \text{and} \quad \nabla g(\boldsymbol{x}) = \begin{bmatrix} 0 \\ 3(x_2 + 1)^2 \end{bmatrix}.$$

All feasible points are not regular because at the above points the gradients of h and g are not linearly independent. There are no regular points of the constraints.

20.2

a. As usual, let f be the objective function, and $\boldsymbol{h}$ the constraint function. We form the Lagrangian $l(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^\top \boldsymbol{h}(\boldsymbol{x})$, and then find critical points by solving the following equations (Lagrange condition):

$$D_x l(\boldsymbol{x}, \boldsymbol{\lambda}) = \mathbf{0}^\top, \qquad D_\lambda l(\boldsymbol{x}, \boldsymbol{\lambda}) = \mathbf{0}^\top.$$

We obtain

$$\begin{bmatrix} 2 & 2 & 0 & 1 & 4 \\ 2 & 6 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 5 \\ 1 & 2 & 0 & 0 & 0 \\ 4 & 0 & 5 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -4 \\ -5 \\ -6 \\ 3 \\ 6 \end{bmatrix}.$$

The unique solution to the above system is

$$\boldsymbol{x}^* = [16/5, -1/10, -34/25]^\top, \qquad \boldsymbol{\lambda}^* = [-27/5, -6/5]^\top.$$

Note that $\boldsymbol{x}^*$ is a regular point. We now apply the SOSC. We compute

$$\boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \boldsymbol{F}(\boldsymbol{x}^*) + [\boldsymbol{\lambda}^* \boldsymbol{H}(\boldsymbol{x}^*)] = \begin{bmatrix} 2 & 2 & 0 \\ 2 & 6 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The tangent plane is

$$\begin{aligned} T(\boldsymbol{x}^*) &= \left\{ \boldsymbol{y} \in \mathbb{R}^3 : \begin{bmatrix} 1 & 2 & 0 \\ 4 & 0 & 5 \end{bmatrix} \boldsymbol{y} = \mathbf{0} \right\} \\ &= \{ a[-5/4, 5/8, 1]^\top : a \in \mathbb{R} \}. \end{aligned}$$

Let $\boldsymbol{y} = a[-5/4, 5/8, 1]^\top \in T(\boldsymbol{x}^*)$, $a \neq 0$. We have

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) \boldsymbol{y} = \frac{75}{32} a^2 > 0.$$

Therefore, $\boldsymbol{x}^*$ is a strict local minimizer.

b. The Lagrange condition for this problem is

$$\begin{aligned} 4 + 2\lambda x_1 &= 0 \\ 2x_2 + 2\lambda x_2 &= 0 \\ x_1^2 + x_2^2 - 9 &= 0. \end{aligned}$$

We have four points satisfying the Lagrange condition:

$$\begin{aligned}
\boldsymbol{x}^{(1)} &= [3,0]^\top, & \lambda^{(1)} &= -2/3\\
\boldsymbol{x}^{(2)} &= [-3,0]^\top, & \lambda^{(2)} &= 2/3\\
\boldsymbol{x}^{(3)} &= [2,\sqrt{5}]^\top, & \lambda^{(3)} &= -1\\
\boldsymbol{x}^{(4)} &= [2,-\sqrt{5}]^\top, & \lambda^{(4)} &= -1.
\end{aligned}$$

Note that all four points $\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(4)}$ are regular. We now apply the SOSC. We have

$$\begin{aligned}
\boldsymbol{L}(\boldsymbol{x},\lambda) &= \begin{bmatrix}0 & 0\\ 0 & 2\end{bmatrix} + \lambda\begin{bmatrix}2 & 0\\ 0 & 2\end{bmatrix},\\
T(\boldsymbol{x}) &= \{\boldsymbol{y} : [2x_1, 2x_2]\boldsymbol{y} = 0\}.
\end{aligned}$$

For the first point, we have

$$\begin{aligned}
\boldsymbol{L}(\boldsymbol{x}^{(1)},\lambda^{(1)}) &= \begin{bmatrix}-4/3 & 0\\ 0 & 2/3\end{bmatrix}\\
T(\boldsymbol{x}^{(1)}) &= \{a[0,1]^\top : a\in\mathbb{R}\}.
\end{aligned}$$

Let $\boldsymbol{y} = a[0,1]^\top \in T(\boldsymbol{x}^{(1)})$, $a\neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(1)},\lambda^{(1)})\boldsymbol{y} = \frac{2}{3}a^2 > 0.$$

Hence, $\boldsymbol{x}^{(1)}$ is a strict local minimizer.

For the second point, we have

$$\boldsymbol{L}(\boldsymbol{x}^{(2)},\lambda^{(2)}) = \begin{bmatrix}4/3 & 0\\ 0 & 10/3\end{bmatrix} > 0.$$

Hence, $\boldsymbol{x}^{(2)}$ is a strict local minimizer.

For the third point, we have

$$\begin{aligned}
\boldsymbol{L}(\boldsymbol{x}^{(3)},\lambda^{(3)}) &= \begin{bmatrix}-2 & 0\\ 0 & 0\end{bmatrix}\\
T(\boldsymbol{x}^{(3)}) &= \{a[-\sqrt{5},2]^\top : a\in\mathbb{R}\}.
\end{aligned}$$

Let $\boldsymbol{y} = a[-\sqrt{5},2]^\top \in T(\boldsymbol{x}^{(3)})$, $a\neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(3)},\lambda^{(3)})\boldsymbol{y} = -10a^2 < 0.$$

Hence, $\boldsymbol{x}^{(3)}$ is a strict local maximizer.

For the fourth point, we have

$$\begin{aligned}
\boldsymbol{L}(\boldsymbol{x}^{(4)},\lambda^{(4)}) &= \begin{bmatrix}-2 & 0\\ 0 & 0\end{bmatrix}\\
T(\boldsymbol{x}^{(4)}) &= \{a[\sqrt{5},2]^\top : a\in\mathbb{R}\}.
\end{aligned}$$

Let $\boldsymbol{y} = a[\sqrt{5},2]^\top \in T(\boldsymbol{x}^{(4)})$, $a\neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(4)},\lambda^{(4)})\boldsymbol{y} = -10a^2 < 0.$$

Hence, $\boldsymbol{x}^{(4)}$ is a strict local maximizer.

c. The Lagrange condition for this problem is

$$\begin{aligned} x_2 + 2\lambda x_1 &= 0 \\ x_1 + 8\lambda x_2 &= 0 \\ x_1^2 + 4x_2^2 - 1 &= 0. \end{aligned}$$

We have four points satisfying the Lagrange condition:

$$\begin{aligned} \boldsymbol{x}^{(1)} &= [1/\sqrt{2}, -1/(2\sqrt{2})]^\top, & \lambda^{(1)} &= 1/4 \\ \boldsymbol{x}^{(2)} &= [-1/\sqrt{2}, 1/(2\sqrt{2})]^\top, & \lambda^{(2)} &= 1/4 \\ \boldsymbol{x}^{(3)} &= [1/\sqrt{2}, 1/(2\sqrt{2})]^\top, & \lambda^{(3)} &= -1/4 \\ \boldsymbol{x}^{(4)} &= [-1/\sqrt{2}, -1/(2\sqrt{2})]^\top, & \lambda^{(4)} &= -1/4. \end{aligned}$$

Note that all four points $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(4)}$ are regular. We now apply the SOSC. We have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}, \lambda) &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \lambda \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}, \\ T(\boldsymbol{x}) &= \{\boldsymbol{y} : [2x_1, 8x_2]\boldsymbol{y} = 0\}. \end{aligned}$$

Note that

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}, -1/4) &= \begin{bmatrix} -1/2 & 1 \\ 1 & -2 \end{bmatrix} \\ \boldsymbol{L}(\boldsymbol{x}, 1/4) &= \begin{bmatrix} 1/2 & 1 \\ 1 & 2 \end{bmatrix}. \end{aligned}$$

After standard manipulations, we conclude that the first two points are strict local maximizers, while the last two points are strict local minimizers.

20.3

We form the lagrangian

$$l(\boldsymbol{x}, \boldsymbol{\lambda}) = (\boldsymbol{a}^\top \boldsymbol{x})(\boldsymbol{b}^\top \boldsymbol{x}) + \lambda_1(x_1 + x_2) + \lambda_2(x_2 + x_3).$$

The Lagrange conditions take the form,

$$\begin{aligned} \nabla_{\boldsymbol{x}} l &= (\boldsymbol{a}\boldsymbol{b}^\top + \boldsymbol{b}\boldsymbol{a}^\top)\boldsymbol{x} + \begin{bmatrix} \nabla_{\boldsymbol{x}} h_1(\boldsymbol{x}) & \nabla_{\boldsymbol{x}} h_2(\boldsymbol{x}) \end{bmatrix} \boldsymbol{\lambda} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \boldsymbol{\lambda} \\ &= \begin{bmatrix} x_2 + \lambda_1 \\ x_1 + x_3 + \lambda_1 + \lambda_2 \\ x_2 + \lambda_2 \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ \boldsymbol{h}(\boldsymbol{x}) &= \begin{bmatrix} x_1 + x_2 \\ x_2 + x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned}$$

It is easy to see that $\boldsymbol{x}^* = \boldsymbol{0}$ and $\boldsymbol{\lambda}^* = \boldsymbol{0}$ satisfy the Lagrange, FONC, conditions.

The Hessian of the lagrangian is

$$\boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \boldsymbol{a}\boldsymbol{b}^\top + \boldsymbol{b}\boldsymbol{a}^\top = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

and the tangent space

$$T(\boldsymbol{x}^*) = \left\{ \boldsymbol{y} : \boldsymbol{y} = a \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, a \in \mathbb{R} \right\}.$$

To verify if the critical point satisfies the SOSC, we evaluate

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)\boldsymbol{y} = -4a^2 < 0.$$

Thus the critical point is a strict local maximizer.

20.4

By the Lagrange condition, $\boldsymbol{x}^* = [x_1, x_2]^\top$ satisfies

$$\begin{aligned} x_1 + \lambda &= 0 \\ x_1 + 4 + 4\lambda &= 0. \end{aligned}$$

Eliminating λ we get

$$3x_1 - 4 = 0$$

which implies that $x_1 = 4/3$. Therefore, $\nabla f(\boldsymbol{x}^*) = [4/3, 16/3]^\top$.

20.5

a. The Lagrange condition for this problem is:

$$\begin{aligned} 2(\boldsymbol{x}^* - \boldsymbol{x}_0) + 2\lambda^* \boldsymbol{x}^* &= \boldsymbol{0} \\ \|\boldsymbol{x}^*\|^2 &= 9, \end{aligned}$$

where $\lambda^* \in \mathbb{R}$. Rewriting the first equation we get $(1 + \lambda^*)\boldsymbol{x}^* = \boldsymbol{x}_0$, which when combined with the second equation gives two values for $1 + \lambda^*$: $1 + \lambda_1^* = 2/3$ and $1 + \lambda_2^* = -2/3$. Hence there are two solutions to the Lagrange condition: $\boldsymbol{x}^{*(1)} = (3/2)[1, \sqrt{3}]$, and $\boldsymbol{x}^{*(2)} = -(3/2)[1, \sqrt{3}]$.

b. We have $\boldsymbol{L}(\boldsymbol{x}^{*(i)}, \lambda_i^*) = (1 + \lambda_i^*)\boldsymbol{I}$. To apply the SONC Theorem, we need to check regularity. This is easy, since the gradient of the constraint function at any point $\boldsymbol{x}$ is $2\boldsymbol{x}$, which is nonzero at both the points in part a.

For the second point, $1 + \lambda_2^* = -2/3$, which implies that the point is not a local minimizer because the SONC does not hold.

On the other hand, the first point satisfies the SOSC (since $1 + \lambda_1^* = 2/3$), which implies that it is a strict local minimizer.

20.6

a. Let x_1, x_2, and x_3 be the dimensions of the closed box. The problem is

$$\begin{aligned} \text{minimize} \quad & 2(x_1x_2 + x_2x_3 + x_3x_1) \\ \text{subject to} \quad & x_1x_2x_3 = V. \end{aligned}$$

We denote $f(\boldsymbol{x}) = 2(x_1x_2 + x_2x_3 + x_3x_1)$, and $h(\boldsymbol{x}) = x_1x_2x_3 - V$. We have $\nabla f(\boldsymbol{x}) = 2[x_2 + x_3, x_1 + x_3, x_1 + x_2]^\top$ and $\nabla h(\boldsymbol{x}) = [x_2x_3, x_1x_3, x_1x_2]^\top$. By the Lagrange condition, the dimensions of the box with minimum surface area satisfies

$$\begin{aligned} 2(b + c) + \lambda bc &= 0 \\ 2(a + c) + \lambda ac &= 0 \\ 2(a + b) + \lambda ab &= 0 \\ abc &= V, \end{aligned}$$

where $\lambda \in \mathbb{R}$.

b. Regularity of $\boldsymbol{x}^*$ means $\nabla h(\boldsymbol{x}^*) \neq 0$ (since there is only one scalar equality constraint in this case). Since $\boldsymbol{x}^* = [a, b, c]^\top$ is a feasible point, we must have $a, b, c \neq 0$ (for otherwise the volume will be 0). Hence, $\nabla h(\boldsymbol{x}^*) \neq 0$, which implies that $\boldsymbol{x}^*$ is regular.

c. Multiplying the first equation by a and the second equation by b, and then subtracting the first from the second, we obtain:

$$c(a - b) = 0.$$

Since $c \neq 0$ (see part b), we conclude that $a = b$. By a similar procedure on the second and third equations, we conclude that $b = c$. Hence, substituting into the fourth (constraint) equation, we obtain

$$a = b = c = V^{1/3},$$

with $\lambda = -4V^{-1/3}$.

d. The Hessian of the Lagrangian is given by

$$\boldsymbol{L}(\boldsymbol{x}^*, \lambda) = \begin{bmatrix} 0 & 2 + \lambda c & 2 + \lambda b \\ 2 + \lambda c & 0 & 2 + \lambda a \\ 2 + \lambda b & 2 + \lambda a & 0 \end{bmatrix} = \begin{bmatrix} 0 & -2 & -2 \\ -2 & 0 & -2 \\ -2 & -2 & 0 \end{bmatrix} = -2 \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

The matrix $\boldsymbol{L}(\boldsymbol{x}^*, \lambda)$ is not positive definite (there are several ways to check this: we could use Sylvester's criterion, or we could compute the eigenvalues of $\boldsymbol{L}(\boldsymbol{x}^*, \lambda)$, which are $2, 2, -4$). Therefore, we need to compute the tangent space $T(\boldsymbol{x}^*)$. Note that

$$Dh(\boldsymbol{x}^*) = \nabla h(\boldsymbol{x})^\top = [bc, ac, ab] = V^{2/3}[1, 1, 1].$$

Hence,

$$T(\boldsymbol{x}^*) = \{\boldsymbol{y} : Dh(\boldsymbol{x}^*)\boldsymbol{y} = 0\} = \{\boldsymbol{y} : [1, 1, 1]\boldsymbol{y} = 0\} = \{\boldsymbol{y} : y_3 = -(y_1 + y_2)\}.$$

Let $\boldsymbol{y} \in T(\boldsymbol{x}^*)$, $\boldsymbol{y} \neq \boldsymbol{0}$. Note that either $y_1 \neq 0$ or $y_2 \neq 0$. We have,

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda)\boldsymbol{y} = -2\boldsymbol{y}^\top \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \boldsymbol{y} = -4(y_1y_2 + y_1y_3 + y_2y_3).$$

Substituting $y_3 = -(y_1 + y_2)$, we obtain

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda)\boldsymbol{y} = -4(y_1y_2 - y_1(y_1 + y_2) - y_2(y_1 + y_2)) = 4(y_1^2 + y_1y_2 + y_2^2) = 4\boldsymbol{z}^\top \boldsymbol{Q}\boldsymbol{z}$$

where $\boldsymbol{z} = [y_1, y_2]^\top \neq \boldsymbol{0}$ and

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 1 \end{bmatrix} > 0.$$

Therefore, $\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda)\boldsymbol{y} > 0$, which shows that the SOSC is satisfied.

An alternative (simpler) calculation:

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda)\boldsymbol{y} = -2\boldsymbol{y}^\top \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \boldsymbol{y} = -2(y_1(y_2 + y_3) + y_2(y_1 + y_3) + y_3(y_1 + y_2)).$$

Substituting $y_1 = -(y_2 + y_3)$, $y_2 = -(y_1 + y_3)$, and $y_3 = -(y_1 + y_2)$ in the first, second, and third terms, respectively, we obtain

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda)\boldsymbol{y} = 2(y_1^2 + y_2^2 + y_2^2) > 0.$$

20.7

a. We first compute critical points by applying the Lagrange conditions. These are:

$$\begin{aligned} 2x_1 + 2\lambda x_1 &= 0 \\ 6x_1 + 2\lambda x_2 &= 0 \\ 1 + 2\lambda x_3 &= 0 \\ x_1^2 + x_2^2 + x_3^2 - 16 &= 0. \end{aligned}$$

There are six points satisfying the Lagrange condition:

$$\begin{aligned} \boldsymbol{x}^{(1)} &= [\sqrt{63}/2, 0, 1/2]^\top, & \lambda^{(1)} &= -1 \\ \boldsymbol{x}^{(2)} &= [-\sqrt{63}/2, 0, 1/2]^\top, & \lambda^{(2)} &= -1 \\ \boldsymbol{x}^{(3)} &= [0, 0, 4]^\top, & \lambda^{(3)} &= -1/8 \\ \boldsymbol{x}^{(4)} &= [0, 0, -4]^\top, & \lambda^{(4)} &= 1/8 \\ \boldsymbol{x}^{(5)} &= [0, \sqrt{575}/6, 1/6]^\top, & \lambda^{(5)} &= -3 \\ \boldsymbol{x}^{(6)} &= [0, -\sqrt{575}/6, 1/6]^\top, & \lambda^{(6)} &= -3. \end{aligned}$$

All the above points are regular. We now apply second order conditions to establish their nature. For this, we compute

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \boldsymbol{H}(\boldsymbol{x}) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix},$$

and

$$T(\boldsymbol{x}^*) = \{\boldsymbol{y} \in \mathbb{R}^3 : [2x_1, 2x_2, 2x_3]\boldsymbol{y} = 0\}.$$

For the first point, we have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(1)}, \lambda^{(1)}) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{bmatrix} \\ T(\boldsymbol{x}^{(1)}) &= \{[-a/\sqrt{63}, b, a]^\top : a, b \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = [-a/\sqrt{63}, b, a]^\top \in T(\boldsymbol{x}^{(1)})$, where a and b are not both zero. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(1)}, \lambda^{(1)})\boldsymbol{y} = 4b^2 - 2a^2 \begin{cases} > 0 & \text{if } |a| < b\sqrt{2} \\ = 0 & \text{if } |a| = b\sqrt{2} \\ < 0 & \text{if } |a| > b\sqrt{2} \end{cases}.$$

From the above, we see that $\boldsymbol{x}^{(1)}$ does not satisfy the SONC. Therefore, $\boldsymbol{x}^{(1)}$ cannot be an extremizer. Performing similar calculations for $\boldsymbol{x}^{(2)}$, we conclude that $\boldsymbol{x}^{(2)}$ cannot be an extremizer either.

For the third point, we have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(3)}, \lambda^{(3)}) &= \begin{bmatrix} 7/4 & 0 & 0 \\ 0 & 23/4 & 0 \\ 0 & 0 & -1/4 \end{bmatrix} \\ T(\boldsymbol{x}^{(3)}) &= \{[a, b, 0]^\top : a, b \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = [a, b, 0]^\top \in T(\boldsymbol{x}^{(3)})$, where a and b are not both zero. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(3)}, \lambda^{(3)})\boldsymbol{y} = \frac{7}{4}a^2 + \frac{23}{4}b^2 > 0.$$

Hence, $\boldsymbol{x}^{(3)}$ is a strict local minimizer. Performing similar calculations for the remaining points, we conclude that $\boldsymbol{x}^{(4)}$ is a strict local minimizer, and $\boldsymbol{x}^{(5)}$ and $\boldsymbol{x}^{(6)}$ are both strict local maximizers.

b. The Lagrange condition for the problem is:

$$\begin{aligned} 2x_1 + \lambda(6x_1 + 4x_2) &= 0 \\ 2x_2 + \lambda(4x_1 + 12x_2) &= 0 \\ 3x_2^2 + 4x_1x_2 + 6x_2^2 - 140 &= 0. \end{aligned}$$

We represent the first two equations as

$$\begin{bmatrix} 2+6\lambda & 4\lambda \\ 4\lambda & 2+12\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

From the constraint equation, we note that $\boldsymbol{x} = [0,0]^\top$ cannot satisfy the Lagrange condition. Therefore, the determinant of the above matrix must be zero. Solving for λ yields two possible values: $-1/7$ and $-1/2$. We then have four points satisfying the Lagrange condition:

$$\begin{aligned} \boldsymbol{x}^{(1)} &= [2,4]^\top, & \lambda^{(1)} &= -1/7 \\ \boldsymbol{x}^{(2)} &= [-2,-4]^\top, & \lambda^{(2)} &= -1/7 \\ \boldsymbol{x}^{(3)} &= [-2\sqrt{14}, \sqrt{14}]^\top, & \lambda^{(3)} &= -1/2 \\ \boldsymbol{x}^{(4)} &= [2\sqrt{14}, -\sqrt{14}]^\top, & \lambda^{(4)} &= -1/2. \end{aligned}$$

Applying the SOSC, we conclude that $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ are strict local minimizers, and $\boldsymbol{x}^{(3)}$ and $\boldsymbol{x}^{(4)}$ are strict local maximizers.

20.8

a. We can represent the problem as

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & h(\boldsymbol{x}) = \boldsymbol{0}, \end{aligned}$$

where $f(\boldsymbol{x}) = 2x_1 + 3x_2 - 4$, and $h(\boldsymbol{x}) = x_1x_2 - 6$. We have $Df(\boldsymbol{x}) = [2,3]$, and $Dh(\boldsymbol{x}) = [x_2, x_1]$. Note that $\boldsymbol{0}$ is not a feasible point. Therefore, any feasible point is regular. If $\boldsymbol{x}^*$ is a local extremizer, then by the Lagrange multiplier theorem, there exists $\lambda^* \in \mathbb{R}$ such that $Df(\boldsymbol{x}^*) + \lambda^* Dh(\boldsymbol{x}^*) = \boldsymbol{0}^\top$, or

$$\begin{aligned} 2 + \lambda^* x_2^* &= 0 \\ 3 + \lambda^* x_1^* &= 0. \end{aligned}$$

Solving, we get two possible extremizers: $\boldsymbol{x}^{(1)} = [3,2]^\top$, with corresponding Lagrange multiplier $\lambda^{(1)} = -1$, and $\boldsymbol{x}^{(2)} = -[3,2]^\top$, with corresponding Lagrange multiplier $\lambda^{(2)} = 1$.

b. We have $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{O}$, and

$$\boldsymbol{H}(\boldsymbol{x}) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

First, consider the point $\boldsymbol{x}^{(1)} = [3,2]^\top$, with corresponding Lagrange multiplier $\lambda^{(1)} = -1$. We have

$$\boldsymbol{L}(\boldsymbol{x}^{(1)}, \lambda^{(1)}) = -\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

and

$$T(\boldsymbol{x}^{(1)}) = \{\boldsymbol{y} : [2,3]\boldsymbol{y} = \boldsymbol{0}\} = \{\alpha[-3,2]^\top : \alpha \in \mathbb{R}\}.$$

Let $\boldsymbol{y} = \alpha[-3,2]^\top \in T(\boldsymbol{x}^{(1)})$, $\alpha \neq 0$. We have

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(1)}, \lambda^{(1)})\boldsymbol{y} = 12\alpha^2 > 0.$$

Therefore, by the SOSC, $\boldsymbol{x}^{(1)} = [3, 2]^\top$ is a strict local minimizer.

Next, consider the point $\boldsymbol{x}^{(2)} = -[3, 2]^\top$, with corresponding Lagrange multiplier $\lambda^{(2)} = 1$. We have

$$\boldsymbol{L}(\boldsymbol{x}^{(2)}, \lambda^{(2)}) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

and

$$T(\boldsymbol{x}^{(2)}) = \{\boldsymbol{y} : -[2, 3]\boldsymbol{y} = \boldsymbol{0}\} = \{\alpha[-3, 2]^\top : \alpha \in \mathbb{R}\} = T(\boldsymbol{x}^{(1)}).$$

Let $\boldsymbol{y} = \alpha[-3, 2]^\top \in T(\boldsymbol{x}^{(2)})$, $\alpha \neq 0$. We have

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(2)}, \lambda^{(2)})\boldsymbol{y} = -12\alpha^2 < 0.$$

Therefore, by the SOSC, $\boldsymbol{x}^{(2)} = -[3, 2]^\top$ is a strict local maximizer.

c. Note that $f(\boldsymbol{x}^{(1)}) = 8$, while $f(\boldsymbol{x}^{(2)}) = -16$. Therefore, $\boldsymbol{x}^{(1)}$, although a strict local minimizer, is not a global minimizer. Likewise, $\boldsymbol{x}^{(2)}$, although a strict local maximizer, is not a global maximizer.

20.9

We observe that $f(x_1, x_2)$ is a ratio of two quadratic functions, that is, we can represent $f(x_1, x_2)$ as

$$f(x_1, x_2) = \frac{\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}}.$$

Therefore, if a point $\boldsymbol{x}$ is a maximizer of $f(x_1, x_2)$ then so is any nonzero multiple of this point because

$$\frac{(t\boldsymbol{x})^\top \boldsymbol{Q}(t\boldsymbol{x})}{(t\boldsymbol{x})^\top \boldsymbol{P}(t\boldsymbol{x})} = \frac{t^2 \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}}{t^2 \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}} = \frac{\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}}.$$

Thus any nonzero multiple of a solution is also a solution. To proceed, represent the original problem in an equivalent form,

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} = 18x_1^2 - 8x_1x_2 + 12x_2^2 \\ \text{subject to} \quad & \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} = 2x_1^2 + 2x_2^2 = 1. \end{aligned}$$

Thus, we wish to maximize $f(x_1, x_2) = 18x_1^2 - 8x_1x_2 + 12x_2^2$ subject to the equality constraint, $h(x_1, x_2) = 1 - 2x_1^2 - 2x_2^2 = 0$. We apply the Lagrange's method to solve the problem. We form the Lagrangian function,

$$l(\boldsymbol{x}, \lambda) = f + \lambda h,$$

compute its gradient and find critical points. We have,

$$\begin{aligned} \nabla_{\boldsymbol{x}} l &= \nabla_{\boldsymbol{x}} \left(\boldsymbol{x}^\top \begin{bmatrix} 18 & -4 \\ -4 & 12 \end{bmatrix} \boldsymbol{x} + \lambda \left(1 - \boldsymbol{x}^\top \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \boldsymbol{x} \right) \right) \\ &= 2 \begin{bmatrix} 18 & -4 \\ -4 & 12 \end{bmatrix} \boldsymbol{x} - 2\lambda \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \boldsymbol{x} \\ &= \boldsymbol{0}. \end{aligned}$$

We represent the above in an equivalent form,

$$\left(\lambda \boldsymbol{I}_2 - \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 18 & -4 \\ -4 & 12 \end{bmatrix} \right) \boldsymbol{x} = \boldsymbol{0}.$$

That is, solving the problem is being reduced to solving an eigenvalue-eigenvector problem,

$$\left(\lambda \boldsymbol{I}_2 - \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix} \right) \boldsymbol{x} = \begin{bmatrix} \lambda - 9 & 2 \\ 2 & \lambda - 6 \end{bmatrix} \boldsymbol{x} = \boldsymbol{0}.$$

The characteristic polynomial is

$$\det \begin{bmatrix} \lambda - 9 & 2 \\ 2 & \lambda - 6 \end{bmatrix} = \lambda^2 - 15\lambda + 50 = (\lambda - 5)(\lambda - 10).$$

The eigenvalues are 5 and 10. Because we are interested in finding a maximizer, we conclude that the value of the maximized function is 10, while the corresponding maximizer corresponds to an appropriately scaled, to satisfy the constraint, eigenvector of this eigenvalue. An eigenvector can easily be found by taking any nonzero column of the adjoint matrix of

$$10\boldsymbol{I}_2 - \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}.$$

Performing simple manipulations gives

$$\operatorname{adj} \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} = \begin{bmatrix} 4 & -2 \\ -2 & 1 \end{bmatrix}.$$

Thus,

$$\sqrt{0.1} \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

is a maximizer for the equivalent problem. Any multiple of the above vector is a solution of the original maximization problem.

20.10

We use the technique of Example 20.8. First, we write the objective function in the form $\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}$, where

$$\boldsymbol{Q} = \boldsymbol{Q}^\top = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}.$$

The characteristic polynomial of $\boldsymbol{Q}$ is $\lambda^2 - 6\lambda + 5$, and the eigenvalues of $\boldsymbol{Q}$ are 1 and 5. The solutions to the problem are the unit length eigenvectors of $\boldsymbol{Q}$ corresponding to the eigenvalue 5, which are $\pm[1, 1]^\top/\sqrt{2}$.

20.11

Consider the problem

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{A}\boldsymbol{x}\|^2 \\ \text{subject to} & \|\boldsymbol{x}\|^2 = 1. \end{array}$$

The optimal objective function value of this problem is the smallest value that $\|\boldsymbol{y}\|^2$ can take. The above can be solved easily using Lagrange multipliers. The Lagrange conditions are

$$\begin{aligned} \boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A} - \lambda \boldsymbol{x}^\top &= \boldsymbol{0}^\top \\ 1 - \boldsymbol{x}^\top \boldsymbol{x} &= 0. \end{aligned}$$

The first equation can be rewritten as $\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{x}$, which implies that λ is an eigenvalue of $\boldsymbol{A}^\top \boldsymbol{A}$. Moreover, premultiplying by $\boldsymbol{x}^\top$ yields $\boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{x}^\top \boldsymbol{x} = \lambda$, which indicates that the Lagrange multiplier is equal to the optimal objective function value. Hence, the range of values that $\|\boldsymbol{y}\| = \|\boldsymbol{A}\boldsymbol{x}\|$ can take is 1 to $\sqrt{20}$.

20.12

Consider the following optimization problem (we need to use squared norms to make the functions differentiable):

$$\begin{array}{ll} \text{minimize} & -\|\boldsymbol{A}\boldsymbol{x}\|^2 \\ \text{subject to} & \|\boldsymbol{x}\|^2 = 1. \end{array}$$

As usual, write $f(\boldsymbol{x}) = -\|\boldsymbol{A}\boldsymbol{x}\|^2$ and $h(\boldsymbol{x}) = \|\boldsymbol{x}\|^2 - 1$. We have $\nabla f(\boldsymbol{x}) = -2\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x}$ and $\nabla h(\boldsymbol{x}) = 2\boldsymbol{x}$. Note that all feasible solutions are regular. Let $\boldsymbol{x}^*$ be an optimal solution. Note that the optimal value of the objective function is $f(\boldsymbol{x}^*) = -\|\boldsymbol{A}\|_2^2$. The Lagrange condition for the above problem is:

$$\begin{aligned} -2\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x}^* + \lambda^*(2\boldsymbol{x}^*) &= 0 \\ \|\boldsymbol{x}^*\|^2 &= 1. \end{aligned}$$

From the first equation, we see that

$$\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x}^* = \lambda^* \boldsymbol{x}^*,$$

which implies that λ^* is an eigenvalue of $\boldsymbol{A}^\top \boldsymbol{A}$, and $\boldsymbol{x}^*$ is the corresponding eigenvector. Premultiplying the above equation by $\boldsymbol{x}^{*\top}$ and combining the result with the constraint equation, we obtain

$$\lambda^* = \boldsymbol{x}^{*\top} \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x}^* = \|\boldsymbol{A}\boldsymbol{x}^*\|^2 = -f(\boldsymbol{x}^*) = \|\boldsymbol{A}\|_2^2.$$

Therefore, because $\boldsymbol{x}^*$ minimizes $f(\boldsymbol{x}^*)$, we deduce that λ^* must be the largest eigenvalue of $\boldsymbol{A}^\top \boldsymbol{A}$; i.e., $\lambda^* = \lambda_1$. Therefore,

$$\|\boldsymbol{A}\|_2 = \sqrt{\lambda_1}.$$

20.13

Let $h(\boldsymbol{x}) = 1 - \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} = 0$. Let $\boldsymbol{x}_0$ be such that $h(\boldsymbol{x}_0) = 0$. Then, $\boldsymbol{x}_0 \neq \boldsymbol{0}$. For $\boldsymbol{x}_0$ to be a regular point, we need to show that $\{\nabla h(\boldsymbol{x}_0)\}$ is a linearly independent set, i.e., $\nabla h(\boldsymbol{x}_0) \neq \boldsymbol{0}$. Now, $\nabla h(\boldsymbol{x}) = -2\boldsymbol{P}\boldsymbol{x}$. Since $\boldsymbol{P}$ is nonsingular, and $\boldsymbol{x}_0 \neq \boldsymbol{0}$, then $\nabla h(\boldsymbol{x}_0) = -2\boldsymbol{P}\boldsymbol{x}_0 \neq \boldsymbol{0}$.

20.14

Note that the point $[1, 1]^\top$ is a regular point. Applying the Lagrange multiplier theorem gives

$$\begin{aligned} a + 2\lambda^* &= 0 \\ b + 2\lambda^* &= 0. \end{aligned}$$

Hence, $a = b$.

20.15

a. Denote the solution by $[x_1^*, x_2^*]^\top$. The Lagrange condition for this problem has the form

$$\begin{aligned} x_2^* - 2 + 2\lambda^* x_1^* &= 0 \\ x_1^* - 2\lambda^* x_2^* &= 0 \\ (x_1^*)^2 - (x_2^*)^2 &= 0. \end{aligned}$$

From the first and third equations it follows that $x_1^*, x_2^* \neq 0$. Then, combining the first and second equations, we obtain

$$\lambda^* = \frac{2 - x_2^*}{2x_1^*} = \frac{x_1^*}{2x_2^*}$$

which implies that $2x_2^* - (x_2^*)^2 = (x_1^*)^2$. Hence, $x_2^* = 1$, and by the third Lagrange equation, $(x_1^*)^2 = 1$. Thus, the only two points satisfying the Lagrange condition are $[1, 1]^\top$ and $[-1, 1]^\top$. Note that both points are regular.

b. Consider the point $\boldsymbol{x}^* = [-1, 1]^\top$. The corresponding Lagrange multiplier is $\lambda^* = -1/2$. The Hessian of the Lagrangian is

$$\boldsymbol{L}(\boldsymbol{x}^*, \lambda^*) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} - \frac{1}{2}\begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}.$$

The tangent plane is given by

$$T(\boldsymbol{x}^*) = \{\boldsymbol{y} : [-2, -2]\boldsymbol{y} = 0\} = \{[a, -a]^\top : a \in \mathbb{R}\}.$$

Let $\boldsymbol{y} \in T(\boldsymbol{x}^*)$, $\boldsymbol{y} \neq \boldsymbol{0}$. Then, $\boldsymbol{y} = [a, -a]^\top$ for some $a \neq 0$. We have $\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda^*)\boldsymbol{y} = -2a^2 < 0$. Hence, SONC does not hold in this case, and therefore $\boldsymbol{x}^* = [-1, 1]^\top$ cannot be local minimizer. In fact, the point is a strict local maximizer.

c. Consider the point $\boldsymbol{x}^* = [1,1]^\top$. The corresponding Lagrange multiplier is $\lambda^* = 1/2$. The Hessian of the Lagrangian is

$$\boldsymbol{L}(\boldsymbol{x}^*, \lambda^*) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The tangent plane is given by

$$T(\boldsymbol{x}^*) = \{\boldsymbol{y} : [2,-2]\boldsymbol{y} = 0\} = \{[a,a]^\top : a \in \mathbb{R}\}.$$

Let $\boldsymbol{y} \in T(\boldsymbol{x}^*)$, $\boldsymbol{y} \neq \boldsymbol{0}$. Then, $\boldsymbol{y} = [a,a]$ for some $a \neq 0$. We have $\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \lambda^*)\boldsymbol{y} = 2a^2 > 0$. Hence, by the SOSC, the point $\boldsymbol{x}^* = [1,1]^\top$ is a strict local minimizer.

20.16

a. The point $\boldsymbol{x}^*$ is the solution to the optimization problem

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2 \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}. \end{aligned}$$

Since $\operatorname{rank} \boldsymbol{A} = m$, any feasible point is regular. By the Lagrange multiplier theorem, there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that

$$(\boldsymbol{x}^* - \boldsymbol{x}_0)^\top - \boldsymbol{\lambda}^{*\top}\boldsymbol{A} = \boldsymbol{0}^\top.$$

Postmultiplying both sides by $\boldsymbol{x}^*$ and using the fact that $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{0}$, we get

$$(\boldsymbol{x}^* - \boldsymbol{x}_0)^\top \boldsymbol{x}^* = 0.$$

b. From part a, we have

$$\boldsymbol{x}^* - \boldsymbol{x}_0 = \boldsymbol{A}^\top \boldsymbol{\lambda}^*.$$

Premultiplying both sides by $\boldsymbol{A}$ we get

$$-\boldsymbol{A}\boldsymbol{x}_0 = (\boldsymbol{A}\boldsymbol{A}^\top)\boldsymbol{\lambda}^*$$

from which we conclude that $\boldsymbol{\lambda}^* = -(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}_0$. Hence,

$$\boldsymbol{x}^* = \boldsymbol{x}_0 + \boldsymbol{A}^\top \boldsymbol{\lambda}^* = \boldsymbol{x}_0 - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}_0 = (\boldsymbol{I}_n - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A})\boldsymbol{x}_0.$$

20.17

a. The Lagrange condition is (omitting all superscript-$*$ for convenience):

$$\begin{aligned} (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top \boldsymbol{A} + \boldsymbol{\lambda}^\top \boldsymbol{C} &= \boldsymbol{0}^\top \\ \boldsymbol{C}\boldsymbol{x} &= \boldsymbol{d}. \end{aligned}$$

For simplicity, write $\boldsymbol{Q} = \boldsymbol{A}^\top \boldsymbol{A}$, which is positive definite. From the first equation, we have

$$\boldsymbol{x} = \boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} - \boldsymbol{Q}^{-1}\boldsymbol{C}^\top \boldsymbol{\lambda}.$$

Multiplying boths sides by $\boldsymbol{C}$ and using the second equation, we have

$$\boldsymbol{d} = \boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} - \boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^\top \boldsymbol{\lambda},$$

from which we obtain

$$\boldsymbol{\lambda} = (\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^\top)^{-1}(\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} - \boldsymbol{d}).$$

Substituting back into the equation for $\boldsymbol{x}$, we obtain

$$\boldsymbol{x} = \boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} - \boldsymbol{Q}^{-1}\boldsymbol{C}^\top(\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^\top)^{-1}(\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} - \boldsymbol{d}).$$

b. Rewrite the objective function as

$$\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{A}\boldsymbol{x} + \frac{1}{2}\|\boldsymbol{b}\|^2.$$

As before, write $\boldsymbol{Q} = \boldsymbol{A}^\top \boldsymbol{A}$. Completing the squares and setting $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b}$, the objective function can be written as

$$\frac{1}{2}\boldsymbol{y}^\top \boldsymbol{Q}\boldsymbol{y} + \text{const.}$$

Hence, the problem can be converted to the equivalent QP:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\boldsymbol{y}^\top \boldsymbol{Q}\boldsymbol{y} \\ \text{subject to} & \boldsymbol{C}\boldsymbol{y} = \boldsymbol{d} - \boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b}. \end{array}$$

The solution to this QP is

$$\boldsymbol{y}^* = \boldsymbol{Q}^{-1}\boldsymbol{C}^\top (\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^\top)^{-1}(\boldsymbol{d} - \boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b}).$$

Hence, the solution to the original problem is:

$$\begin{aligned} \boldsymbol{x}^* &= \boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b} + \boldsymbol{Q}^{-1}\boldsymbol{C}^\top (\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^\top)^{-1}(\boldsymbol{d} - \boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top \boldsymbol{b}) \\ &= (\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{A}^\top \boldsymbol{b} + (\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{C}^\top (\boldsymbol{C}(\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{C}^\top)^{-1}(\boldsymbol{d} - \boldsymbol{C}(\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{A}^\top \boldsymbol{b}), \end{aligned}$$

which agrees with the solution obtained in part a.

20.18

Write $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{c}^\top \boldsymbol{x} + d$ (actually, we could have ignored d) and $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$. We have

$$Df(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{Q} - \boldsymbol{c}^\top, \qquad D\boldsymbol{h}(\boldsymbol{x}) = -\boldsymbol{A}.$$

The Lagrange condition is

$$\begin{aligned} \boldsymbol{x}^{*\top}\boldsymbol{Q} - \boldsymbol{c}^\top - \boldsymbol{\lambda}^{*\top}\boldsymbol{A} &= \boldsymbol{0}^\top \\ \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^* &= \boldsymbol{0}. \end{aligned}$$

From the first equation we get

$$\boldsymbol{x}^* = \boldsymbol{Q}^{-1}(\boldsymbol{A}^\top \boldsymbol{\lambda}^* + \boldsymbol{c}).$$

Multiplying both sides by $\boldsymbol{A}$ and using the second equation (constraint), we get

$$(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)\boldsymbol{\lambda}^* + \boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{c} = \boldsymbol{b}.$$

Since $\boldsymbol{Q} > 0$ and $\boldsymbol{A}$ is of full rank, we can write

$$\boldsymbol{\lambda}^* = (\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{c}).$$

Hence,

$$\boldsymbol{x}^* = \boldsymbol{Q}^{-1}\boldsymbol{c} + \boldsymbol{Q}^{-1}\boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{c}).$$

Alternatively, we could have rewritten the given problem in our usual quadratic programming form with variable $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{Q}^{-1}\boldsymbol{c}$.

20.19

Clearly, we have $\mathcal{M} = \mathcal{R}(\boldsymbol{B})$, i.e., $\boldsymbol{y} \in \mathcal{M}$ if and only if there exists $\boldsymbol{x} \in \mathbb{R}^m$ such that $\boldsymbol{y} = \boldsymbol{B}\boldsymbol{x}$. Hence

$$\begin{aligned} &\boldsymbol{L} \text{ is positive semidefinite on } \mathcal{M} \\ &\Leftrightarrow \quad \text{for all } \boldsymbol{y} \in \mathcal{M}, \quad \boldsymbol{y}^\top \boldsymbol{L}\boldsymbol{y} \geq 0 \\ &\Leftrightarrow \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^m, \quad (\boldsymbol{B}\boldsymbol{x})^\top \boldsymbol{L}(\boldsymbol{B}\boldsymbol{x}) \geq 0 \\ &\Leftrightarrow \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^m, \quad \boldsymbol{x}^\top (\boldsymbol{B}^\top \boldsymbol{L}\boldsymbol{B})\boldsymbol{x} \geq 0 \\ &\Leftrightarrow \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^m, \quad \boldsymbol{x}^\top \boldsymbol{L}_{\mathcal{M}}\boldsymbol{x} \geq 0 \\ &\Leftrightarrow \quad \boldsymbol{L}_{\mathcal{M}} \geq 0. \end{aligned}$$

For positive definiteness, the same argument applies, with $\geq$ replaced by $>$.

20.20

a. By simple manipulations, we can write

$$x_2 = a^2 x_0 + abu_0 + bu_1.$$

Therefore, the problem is

$$\begin{array}{ll} \text{minimize} & \dfrac{1}{2}(u_0^2 + u_1^2) \\ \text{subject to} & a^2 x_0 + abu_0 + bu_1 = 0. \end{array}$$

Alternatively, we may use a vector notation: writing $\boldsymbol{u} = [u_0, u_1]^\top$, we have

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{u}) \\ \text{subject to} & h(\boldsymbol{u}) = 0, \end{array}$$

where $f(\boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{u}\|^2$, and $h(\boldsymbol{u}) = a^2 x_0 + [ab, b]\boldsymbol{u}$. Since the vector $\nabla h(\boldsymbol{u}) = [ab, b]^\top$ is nonzero for any $\boldsymbol{u}$, then any feasible point is regular. Therefore, by the Lagrange multiplier theorem, there exists $\lambda^* \in \mathbb{R}$ such that

$$\begin{aligned} u_0^* + \lambda^* ab &= 0 \\ u_1^* + \lambda^* b &= 0 \\ a^2 x_0 + abu_0^* + bu_1^* &= 0. \end{aligned}$$

We have three linear equations in three unknowns, that upon solving yields

$$u_0^* = -\frac{a^3 x_0}{b(1+a^2)}, \qquad u_1^* = -\frac{a^2 x_0}{b(1+a^2)}.$$

b. The Hessians of f and h are $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{I}_2$ (2×2 identity matrix) and $\boldsymbol{H}(\boldsymbol{u}) = \boldsymbol{O}$, respectively. Hence, the Hessian of the Lagrangian is $\boldsymbol{L}(\boldsymbol{u}^*, \lambda^*) = \boldsymbol{I}_2$, which is positive definite. Therefore, $\boldsymbol{u}^*$ satisfies the SOSC, and is therefore a strict local minimizer.

20.21

Letting $\boldsymbol{z} = [x_2, u_1, u_2]^\top$, the objective function is $\boldsymbol{z}^\top \boldsymbol{Q}\boldsymbol{z}$, where

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/3 \end{bmatrix}.$$

The linear constraint on $\boldsymbol{z}$ is obtained by writing

$$x_2 = 2x_1 + u_2 = 2(2 + u_1) + u_2,$$

which can be written as $\boldsymbol{A}\boldsymbol{z} = b$, where

$$\boldsymbol{A} = [1, -2, -1], \qquad b = 4.$$

Hence, using the method of Section 20.6, the solution is

$$\boldsymbol{z}^* = \boldsymbol{Q}^{-1}\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)^{-1}b = \begin{bmatrix} 1 \\ -4 \\ -3 \end{bmatrix} \cdot (12)^{-1} \cdot 4 = \begin{bmatrix} 1/3 \\ -4/3 \\ -1 \end{bmatrix}.$$

Thus, the optimal controls are $u_1^* = -4/3$ and $u_2^* = -1$.

20.22

The composite input vector is

$$\boldsymbol{u} = \begin{bmatrix} u_0 & u_1 & u_2 \end{bmatrix}^\top.$$

The performance index J is $J = \frac{1}{2}\boldsymbol{u}^\top \boldsymbol{u}$. To obtain the constraint $\boldsymbol{A}\boldsymbol{u} = b$, where $\boldsymbol{A} \in \mathbb{R}^{1\times 3}$, we proceed as follows. First, we write

$$\begin{aligned} x_2 &= x_1 + 2u_1 \\ &= x_0 + 2u_0 + 2u_1. \end{aligned}$$

Using the above, we obtain

$$\begin{aligned} x_3 &= 9 \\ &= x_2 + 2u_2 \\ &= x_0 + 2u_0 + 2u_1 + 2u_2. \end{aligned}$$

We represent the above in the format $\boldsymbol{A}\boldsymbol{u} = b$ as follows

$$\begin{bmatrix} 2 & 2 & 2 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} = 6.$$

Thus we formulated the problem of finding the optimal control sequence as a constrained optimization problem

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\boldsymbol{u}^\top \boldsymbol{u} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{u} = b. \end{aligned}$$

To solve the above problem, we form the Lagrangian

$$l(\boldsymbol{u}, \lambda) = \frac{1}{2}\boldsymbol{u}^\top \boldsymbol{u} + \lambda \left(\boldsymbol{A}\boldsymbol{u} - b\right),$$

where λ is the Lagrange multiplier. Applying the Lagrange first-order condition yields

$$\boldsymbol{u} + \boldsymbol{A}^\top \lambda = \boldsymbol{0} \quad \text{and} \quad \boldsymbol{A}\boldsymbol{u} = b.$$

From the first of the above conditions, we calculate, $\boldsymbol{u} = -\boldsymbol{A}^\top \lambda$. Substituting the above into the second of the Lagrange conditions gives

$$\lambda = -\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1} b.$$

Combining the last two equations, we obtain a closed-form formula for the optimal input sequence

$$\boldsymbol{u} = \boldsymbol{A}^\top \left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1} b.$$

In our problem,

$$\boldsymbol{u} = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} = \frac{b}{\left(\boldsymbol{A}\boldsymbol{A}^\top\right)} \boldsymbol{A}^\top = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

21. Problems With Inequality Constraints

21.1

a. We form the Lagrangian function,

$$l(\boldsymbol{x}, \mu) = x_1^2 + 4x_2^2 + \mu(4 - x_1^2 - 2x_2^2).$$

The KKT conditions take the form,

$$\begin{aligned}
&D_{\boldsymbol{x}} l(\boldsymbol{x}, \mu) = \begin{bmatrix} 2x_1 - 2\mu x_1 & 8x_2 - 4\mu x_2 \end{bmatrix} = \mathbf{0}^\top \\
&\mu(4 - x_1^2 - 2x_2^2) = 0 \\
&\mu \geq 0 \\
&4 - x_1^2 - 2x_2^2 \leq 0.
\end{aligned}$$

From the first of the above equality, we obtain

$$(1 - \mu)x_1 = 0 \quad (2 - \mu)x_2 = 0.$$

We first consider the case when $\mu = 0$. Then, we obtain the point, $\boldsymbol{x}^{(1)} = \mathbf{0}$, which does not satisfy the constraints.

The next case is when $\mu = 1$. Then we have to have $x_2 = 0$ and using $\mu(4 - x_1^2 - 2x_2^2) = 0$ gives

$$\boldsymbol{x}^{(2)} = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{x}^{(3)} = -\begin{bmatrix} 2 \\ 0 \end{bmatrix}.$$

For the case when $\mu = 2$, we have to have $x_1 = 0$ and we get

$$\boldsymbol{x}^{(4)} = \begin{bmatrix} 0 \\ \sqrt{2} \end{bmatrix} \quad \text{and} \quad \boldsymbol{x}^{(5)} = -\begin{bmatrix} 0 \\ \sqrt{2} \end{bmatrix}.$$

b. The Hessian of l is

$$\boldsymbol{L} = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix} + \mu \begin{bmatrix} -2 & 0 \\ 0 & -4 \end{bmatrix}.$$

When $\mu = 1$,

$$\boldsymbol{L} = \begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix}.$$

We next find the subspace

$$\begin{aligned}
\tilde{T} &= T = \{\boldsymbol{y} : \begin{bmatrix} \pm 4 & 0 \end{bmatrix} \boldsymbol{y} = 0\} \\
&= \{\boldsymbol{y} = a \begin{bmatrix} 0 & 1 \end{bmatrix}^\top : a \in \mathbb{R}\}.
\end{aligned}$$

We then check for positive definiteness of $\boldsymbol{L}$ on $\tilde{T}$,

$$\boldsymbol{y}^\top \boldsymbol{L} \boldsymbol{y} = a^2 \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 4a^2 > 0.$$

Hence, $\boldsymbol{x}^{(2)}$ and $\boldsymbol{x}^{(3)}$ satisfy the SOSC to be strict local minimizers.

When $\mu = 2$,

$$\boldsymbol{L} = \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix},$$

and

$$T = \{\boldsymbol{y} = a \begin{bmatrix} 1 & 0 \end{bmatrix}^\top : a \in \mathbb{R}\}.$$

We have

$$\boldsymbol{y}^\top \boldsymbol{L}\boldsymbol{y} = -2a^2 < 0.$$

Thus, $\boldsymbol{x}^{(4)}$ and $\boldsymbol{x}^{(5)}$ do not satisfy the SONC to be minimizers.

In summary, only $\boldsymbol{x}^{(2)}$ and $\boldsymbol{x}^{(3)}$ are strict local minimizers.

21.2

a. We first find critical points by applying the Karush-Kuhn-Tucker conditions, which are

$$\begin{aligned} 2x_1 - 2 - 2\mu_1 x_1 + 5\mu_2 &= 0 \\ 2x_2 - 10 + \frac{1}{5}\mu_1 + \frac{1}{2}\mu_2 &= 0 \\ \mu_1\left(\frac{1}{5}x_2 - x_1^2\right) + \mu_2\left(5x_1 + \frac{1}{2}x_2 - 5\right) &= 0 \\ \boldsymbol{\mu} &\geq 0. \end{aligned}$$

We have to check four possible combinations.

Case 1: ($\mu_1 = 0$, $\mu_2 = 0$) Solving the first and second Karush-Kuhn-Tucker equations yields $\boldsymbol{x}^{(1)} = [1, 5]^\top$. However, this point is not feasible and is therefore not a candidate minimizer.

Case 2: ($\mu_1 > 0$, $\mu_2 = 0$) We have two possible solutions:

$$\begin{aligned} \boldsymbol{x}^{(2)} &= [-0.98, 4.8]^\top \qquad \mu_1^{(2)} = 2.02 \\ \boldsymbol{x}^{(3)} &= [-0.02, 0]^\top \qquad \mu_1^{(3)} = 50. \end{aligned}$$

Both $\boldsymbol{x}^{(2)}$ and $\boldsymbol{x}^{(3)}$ satisfy the constraints, and are therefore candidate minimizers.

Case 3: ($\mu_1 = 0$, $\mu_2 > 0$) Solving the corresponding equation yields:

$$\boldsymbol{x}^{(4)} = [0.5050, 4.9505]^\top \qquad \mu_1^{(4)} = 0.198.$$

The point $\boldsymbol{x}^{(4)}$ is not feasible, and hence is not a candidate minimizer.

Case 4: ($\mu_1 > 0$, $\mu_2 > 0$) We have two solutions:

$$\begin{aligned} \boldsymbol{x}^{(5)} &= [0.732, 2.679]^\top \qquad \boldsymbol{\mu}^{(5)} = [13.246, 3.986]^\top \\ \boldsymbol{x}^{(6)} &= [-2.73, 37.32]^\top \qquad \boldsymbol{\mu}^{(6)} = [188.8, -204]^\top. \end{aligned}$$

The point $\boldsymbol{x}^{(5)}$ is feasible, but $\boldsymbol{x}^{(6)}$ is not.

We are left with three candidate minimizers: $\boldsymbol{x}^{(2)}$, $\boldsymbol{x}^{(3)}$, and $\boldsymbol{x}^{(5)}$. It is easy to check that they are regular. We now check if each satisfies the second order conditions. For this, we compute

$$\boldsymbol{L}(\boldsymbol{x}, \boldsymbol{\mu}) = \begin{bmatrix} 2 - 2\mu_1 & 0 \\ 0 & 2 \end{bmatrix}.$$

For $\boldsymbol{x}^{(2)}$, we have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(2)}, \boldsymbol{\mu}^{(2)}) &= \begin{bmatrix} -2.04 & 0 \\ 0 & 2 \end{bmatrix} \\ \tilde{T}(\boldsymbol{x}^{(2)}) &= \{a[-0.1021, 1]^\top : a \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = a[-0.1021, 1]^\top \in \tilde{T}(\boldsymbol{x}^{(2)})$ with $a \neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(2)}, \boldsymbol{\mu}^{(2)})\boldsymbol{y} = 1.979a^2 > 0.$$

Thus, by the SOSC, $\boldsymbol{x}^{(2)}$ is a strict local minimizer.

For $\boldsymbol{x}^{(3)}$, we have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(3)}, \boldsymbol{\mu}^{(3)}) &= \begin{bmatrix} -97.958 & 0 \\ 0 & 2 \end{bmatrix} \\ \tilde{T}(\boldsymbol{x}^{(3)}) &= \{a[-4.898, 1]^\top : a \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = a[-4.898, 1]^\top \in \tilde{T}(\boldsymbol{x}^{(3)})$ with $a \neq 0$. Then,

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(3)}, \boldsymbol{\mu}^{(3)})\boldsymbol{y} = -2347.9a^2 < 0.$$

Thus, $\boldsymbol{x}^{(3)}$ does not satisfy the SOSC. In fact, in this case, we have $T(\boldsymbol{x}^{(3)}) = \tilde{T}(\boldsymbol{x}^{(3)})$, and hence $\boldsymbol{x}^{(3)}$ does not satisfy the SONC either. We conclude that $\boldsymbol{x}^{(3)}$ is not a local minimizer. We can easily check that $\boldsymbol{x}^{(3)}$ is not a local maximizer either.

For $\boldsymbol{x}^{(5)}$, we have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(5)}, \boldsymbol{\mu}^{(5)}) &= \begin{bmatrix} -24.4919 & 0 \\ 0 & 2 \end{bmatrix} \\ \tilde{T}(\boldsymbol{x}^{(5)}) &= \{\boldsymbol{0}\}. \end{aligned}$$

The SOSC is trivially satisfied, and therefore $\boldsymbol{x}^{(5)}$ is a strict local minimizer.

b. The Karush-Kuhn-Tucker conditions are:

$$\begin{aligned} 2x_1 - \mu_1 - \mu_3 &= 0 \\ 2x_2 - \mu_2 - \mu_3 &= 0 \\ -x_1 &\leq 0 \\ -x_2 &\leq 0 \\ -x_1 - x_2 + 5 &\leq 0 \\ -\mu_1 x_1 - \mu_2 x_2 + \mu_3(-x_1 - x_2 + 5) &= 0 \\ \mu_1, \mu_2, \mu_3 &\geq 0. \end{aligned}$$

It is easy to verify that the only combination of Karush-Kuhn-Tucker multipliers resulting in a feasible point is $\mu_1 = \mu_2 = 0$, $\mu_3 > 0$. For this case, we obtain $\boldsymbol{x}^* = [2.5, 2.5]^\top$, $\boldsymbol{\mu}^* = [0, 0, 5]^\top$. We have

$$\boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\mu}^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} > 0.$$

Hence, $\boldsymbol{x}^*$ is a strict local minimizer (in fact, the only one for this problem).

c. The Karush-Kuhn-Tucker conditions are:

$$\begin{aligned} 2x_1 + 6x_2 - 4 + 2\mu_1 x_1 + 2\mu_2 &= 0 \\ 6x_1 - 2 + 2\mu_1 - 2\mu_2 &= 0 \\ x_1^2 + 2x_2 - 1 &\leq 0 \\ 2x_1 - 2x_2 - 1 &\leq 0 \\ \mu_1(x_1^2 + 2x_2 - 1) + \mu_2(2x_1 - 2x_2 - 1) &= 0 \\ \mu_1, \mu_2 &\geq 0. \end{aligned}$$

It is easy to verify that the only combination of Karush-Kuhn-Tucker multipliers resulting in a feasible point is $\mu_1 = 0$, $\mu_2 > 0$. For this case, we obtain $\boldsymbol{x}^* = [9/14, 2/14]^\top$, $\boldsymbol{\mu}^* = [0, 13/14]^\top$. We have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\mu}^*) &= \begin{bmatrix} 2 & 6 \\ 6 & 0 \end{bmatrix} \\ \tilde{T}(\boldsymbol{x}^*) &= \{a[1, 1] : a \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = a[1, 1] \in \tilde{T}(\boldsymbol{x}^*)$ with $a \neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\mu}^*)\boldsymbol{y} = 14a^2 > 0.$$

Hence, $\boldsymbol{x}^*$ is a strict local minimizer (in fact, the only one for this problem).

21.3

The Karush-Kuhn-Tucker conditions are:

$$\begin{aligned} 2x_1 + 2\lambda x_1 + 2\lambda x_2 + 2\mu x_1 &= 0 \\ 2x_2 + 2\lambda x_1 + 2\lambda x_2 - \mu &= 0 \\ x_1^2 + 2x_1 x_2 + x_2^2 - 1 &= 0 \\ x_1^2 - x_2 &\leq 0 \\ \mu(x_1^2 - x_2) &= 0 \\ \mu &\geq 0. \end{aligned}$$

We have two cases to consider.

Case 1: ($\mu > 0$) Substituting $x_2 = x_1^2$ into the third equation and combining the result with the first two yields two possible points:

$$\begin{aligned} \boldsymbol{x}^{(1)} &= [-1.618, 0.618]^\top \qquad \mu^{(1)} = -3.7889 \\ \boldsymbol{x}^{(2)} &= [2.618, 0.382]^\top \qquad \mu^{(2)} = -0.2111. \end{aligned}$$

Note that the resulting μ values violate the condition $\mu > 0$. Hence, neither of the points are minimizers (although they are candidates for maximizers).

Case 2: ($\mu = 0$) Subtracting the second equation from the first yields $x_1 = x_2$, which upon substituting into the third equation gives two possible points:

$$\boldsymbol{x}^{(3)} = [1/2, 1/2]^\top, \qquad \boldsymbol{x}^{(4)} = [-1/2, 1/2]^\top.$$

Note that $\boldsymbol{x}^{(4)}$ is not a feasible point, and is therefore not a candidate minimizer.

Therefore, the only remaining candidate is $\boldsymbol{x}^{(3)}$, with corresponding $\lambda^{(3)} = -1/2$ (and $\mu = 0$). We now check second order conditions. We have

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{x}^{(3)}, 0, \lambda^{(3)}) &= \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \\ \tilde{T}(\boldsymbol{x}^{(3)}) &= \{a[1, -1]^\top : a \in \mathbb{R}\}. \end{aligned}$$

Let $\boldsymbol{y} = a[1, -1]^\top \in \tilde{T}(\boldsymbol{x}^{(3)})$ with $a \neq 0$. Then

$$\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^{(3)}, 0, \lambda^{(3)})\boldsymbol{y} = 4a^2 > 0.$$

Therefore, by the SOSC, $\boldsymbol{x}^{(3)}$ is a strict local minimizer.

21.4

The optimization problem is:

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{e}_n^\top \boldsymbol{p} \\ \text{subject to} \quad & \boldsymbol{G}\boldsymbol{p} \geq P\boldsymbol{e}_m \\ & \boldsymbol{p} \geq \boldsymbol{0}, \end{aligned}$$

where $\boldsymbol{G} = [g_{i,j}]$, $\boldsymbol{e}_n = [1, \ldots, 1]^\top$ (with n components), and $\boldsymbol{p} = [p_1, \ldots, p_n]^\top$. The KKT condition for this problem is:

$$\begin{aligned} \boldsymbol{e}_n^\top - \boldsymbol{\mu}_1^\top \boldsymbol{G} - \boldsymbol{\mu}_2^\top &= \boldsymbol{0}^\top \\ \boldsymbol{\mu}_1^\top (P\boldsymbol{e}_m - \boldsymbol{G}\boldsymbol{p}) - \boldsymbol{\mu}_2^\top \boldsymbol{p} &= 0 \\ \boldsymbol{G}\boldsymbol{p} &\geq P\boldsymbol{e}_m \\ \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{p} &\geq \boldsymbol{0}. \end{aligned}$$

21.5

a. We have $f(\boldsymbol{x}) = x_2 - (x_1 - 2)^3 + 3$ and $g(\boldsymbol{x}) = 1 - x_2$. Hence, $\nabla f(\boldsymbol{x}) = [-3(x_1 - 2)^2, 1]^\top$ and $\nabla g(\boldsymbol{x}) = [0, -1]^\top$. The KKT condition is

$$\begin{aligned} \mu &\geq 0 \\ -3(x_1 - 2)^2 &= 0 \\ 1 - \mu &= 0 \\ \mu(1 - x_2) &= 0 \\ 1 - x_2 &\leq 0. \end{aligned}$$

The only solution to the above conditions is $\boldsymbol{x}^* = [2, 1]^\top$, $\mu^* = 1$.

To check if $\boldsymbol{x}^*$ is regular, we note that the constraint is active. We have $\nabla g(\boldsymbol{x}^*) = [0, -1]^\top$, which is nonzero. Hence, $\boldsymbol{x}^*$ is regular.

b. We have

$$\boldsymbol{L}(\boldsymbol{x}^*, \mu^*) = \boldsymbol{F}(\boldsymbol{x}^*) + \mu^* \boldsymbol{G}(\boldsymbol{x}^*) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Hence, the point $\boldsymbol{x}^*$ satisfies the SONC.

c. Since $\mu^* > 0$, we have $\tilde{T}(\boldsymbol{x}^*, \mu^*) = T(\boldsymbol{x}^*) = \{\boldsymbol{y} : [0, -1]\boldsymbol{y} = 0\} = \{\boldsymbol{y} : y_2 = 0\}$, which means that $\tilde{T}$ contains nonzero vectors. Hence, the SOSC does not hold at $\boldsymbol{x}^*$.

21.6

a. Write $f(\boldsymbol{x}) = x_2$, $g(\boldsymbol{x}) = -(x_2 + (x_1 - 1)^2 - 3)$. We have $\nabla f(\boldsymbol{x}) = [0, 1]^\top$ and $\nabla g(\boldsymbol{x}) = [-2(x_1 - 1), -1]^\top$. The KKT conditions are:

$$\begin{aligned} \mu &\geq 0 \\ -2\mu(x_1 - 1) &= 0 \\ 1 - \mu &= 0 \\ \mu(x_2 + (x_1 - 1)^2 - 3) &= 0 \\ x_2 + (x_1 - 1)^2 + 3 &\leq 0. \end{aligned}$$

From the third equation we get $\mu = 1$. The second equation then gives $x_1 = 1$, and the fourth equation gives $x_2 = 3$. Therefore, the only point that satisfies the KKT condition is $\boldsymbol{x}^* = [1, 3]^\top$, with a KKT multiplier of $\mu^* = 1$.

b. Note that the constraint $x_2 + (x_1 - 1)^2 + 3 \geq 0$ is active at $\boldsymbol{x}^*$. We have $T(\boldsymbol{x}^*) = \{\boldsymbol{y} : [0, -1]\boldsymbol{y} = 0\} = \{\boldsymbol{y} : y_2 = 0\}$, and $N(\boldsymbol{x}^*) = \{\boldsymbol{y} : \boldsymbol{y} = [0, -1]z,\ z \in \mathbb{R}\} = \{\boldsymbol{y} : y_1 = 0\}$. Because $\mu^* > 0$, we have $\tilde{T}(\boldsymbol{x}^*) = T(\boldsymbol{x}^*) = \{\boldsymbol{y} : y_2 = 0\}$.

c. We have

$$\boldsymbol{L}(\boldsymbol{x}^*, \mu^*) = \boldsymbol{O} + 1 \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix}.$$

From part b, $T(\boldsymbol{x}^*) = \{\boldsymbol{y} : y_2 = 0\}$. Therefore, for any $\boldsymbol{y} \in T(\boldsymbol{x}^*)$, $\boldsymbol{y}^\top \boldsymbol{L}(\boldsymbol{x}^*, \mu^*)\boldsymbol{y} = -2y_1^2 \leq 0$, which means that $\boldsymbol{x}^*$ does not satisfy the SONC.

21.7

a. We need to consider two optimization problems. We first consider the minimization problem

$$\begin{aligned} \text{minimize} \quad & (x_1 - 2)^2 + (x_2 - 1)^2 \\ \text{subject to} \quad & x_1^2 - x_2 \leq 0 \\ & x_1 + x_2 - 2 \leq 0 \\ & -x_1 \leq 0. \end{aligned}$$

Then, we form the Lagrangian function

$$l(\boldsymbol{x}, \boldsymbol{\mu}) = (x_1 - 2)^2 + (x_2 - 1)^2 + \mu_1(x_1^2 - x_2) + \mu_2(x_1 + x_2 - 2) + \mu_3(-x_1).$$

The KKT condition takes the form

$$\begin{aligned}
&\nabla_{\boldsymbol{x}} l(\boldsymbol{x}, \boldsymbol{\mu}) = \begin{bmatrix} 2(x_1 - 2) + 2\mu_1 x_1 + \mu_2 - \mu_3 & 2(x_2 - 1) - \mu_1 + \mu_2 \end{bmatrix} = \mathbf{0}^T \\
&\mu_1(x_1^2 - x_2) = 0 \\
&\mu_2(x_1 + x_2 - 2) = 0 \\
&\mu_3(-x_1) = 0 \\
&\mu_i \geq 0.
\end{aligned}$$

The point $\boldsymbol{x}^* = \mathbf{0}$ satisfies the above conditions for $\mu_1 = -2$, $\mu_2 = 0$, and $\mu_3 = -4$. Thus the point $\boldsymbol{x}^*$ does not satisfy the KKT conditions for minimum.

We next consider the maximization problem

$$\begin{aligned}
\text{minimize} \quad & -(x_1 - 2)^2 - (x_2 - 1)^2 \\
\text{subject to} \quad & x_1^2 - x_2 \leq 0 \\
& x_1 + x_2 - 2 \leq 0 \\
& -x_1 \leq 0.
\end{aligned}$$

The Lagrangian function for the above problem is,

$$l(\boldsymbol{x}, \boldsymbol{\mu}) = -(x_1 - 2)^2 - (x_2 - 1)^2 + \mu_1(x_1^2 - x_2) + \mu_2(x_1 + x_2 - 2) + \mu_3(-x_1).$$

The KKT condition takes the form

$$\begin{aligned}
&\nabla_{\boldsymbol{x}} l(\boldsymbol{x}, \boldsymbol{\mu}) = \begin{bmatrix} -2(x_1 - 2) + 2\mu_1 x_1 + \mu_2 - \mu_3 & -2(x_2 - 1) - \mu_1 + \mu_2 \end{bmatrix} = \mathbf{0}^\top \\
&\mu_1(x_1^2 - x_2) = 0 \\
&\mu_2(x_1 + x_2 - 2) = 0 \\
&\mu_3(-x_1) = 0 \\
&\mu_i \geq 0.
\end{aligned}$$

The point $\boldsymbol{x}^* = \mathbf{0}$ satisfies the above conditions for $\mu_1 = 2$, $\mu_2 = 0$, and $\mu_3 = 4$. Hence, the point $\boldsymbol{x}^*$ satisfies the KKT conditions for maximum.

b. We next compute the Hessian, with respect to $\boldsymbol{x}$, of the lagrangian to obtain

$$\boldsymbol{L} = \boldsymbol{F} + \mu_1^* \boldsymbol{G}_1 = \begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix} + \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix}$$

which is indefinite on $\mathbb{R}^2$. We next find the subspace

$$\begin{aligned}
\tilde{T} &= \left\{ \boldsymbol{y} : \begin{bmatrix} \nabla g_1(\boldsymbol{x}^*)^\top \\ \nabla g_3(\boldsymbol{x}^*)^\top \end{bmatrix} \boldsymbol{y} = \mathbf{0} \right\} = \left\{ \boldsymbol{y} : \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} \boldsymbol{y} = \mathbf{0} \right\} \\
&= \{\mathbf{0}\},
\end{aligned}$$

That is, $\tilde{T}$ is a trivial subspace that consists only of the zero vector. Thus the SOSC for $\boldsymbol{x}^*$ to be a strict local maximizer is trivially satisfied.

21.8

a. Write $h(\boldsymbol{x}) = x_1 - x_2$, $g(\boldsymbol{x}) = -x_1$. We have $Df(\boldsymbol{x}) = [x_2^2, 2x_1x_2]$, $Dh(\boldsymbol{x}) = [1, -1]$, $Dg(\boldsymbol{x}) = [-1, 0]$.

Note that all feasible points are regular. The KKT condition is:

$$\begin{aligned} x_2^2 + \lambda - \mu &= 0 \\ 2x_1x_2 - \lambda &= 0 \\ \mu x_1 &= 0 \\ \mu &\geq 0 \\ x_1 - x_2 &= 0 \\ x_1 &\geq 0. \end{aligned}$$

We first try $x_1 = x_1^* = 0$ (active inequality constraint). Substituting and manipulating, we have the solution $x_1^* = x_2^* = 0$ with $\mu^* = 0$, which is a legitimate solution. If we then try $x_1 = x_1^* > 0$ (inactive inequality constraint), we find that there is no consistent solution to the KKT condition. Thus, there is only one point satisfying the KKT condition: $\boldsymbol{x}^* = \mathbf{0}$.

c. The tangent space at $\boldsymbol{x}^* = \mathbf{0}$ is given by

$$T(\mathbf{0}) = \{\boldsymbol{y} : [1, -1]\boldsymbol{y} = 0, [-1, 0]\boldsymbol{y} = 0\} = \{\mathbf{0}\}.$$

Therefore, the SONC holds for the solution in part a.

d. We have

$$\boldsymbol{L}(\boldsymbol{x}, \lambda, \mu) = \begin{bmatrix} 0 & 2x_2 \\ 2x_2 & 2x_1 \end{bmatrix}.$$

Hence, at $\boldsymbol{x}^* = 0$, we have $\boldsymbol{L}(\mathbf{0}, 0, 0) = \boldsymbol{O}$. Since the active constraint at $\boldsymbol{x}^* = \mathbf{0}$ is degenerate, we have

$$\tilde{T}(\mathbf{0}, 0) = \{\boldsymbol{y} : [1, -1]\boldsymbol{y} = 0\},$$

which is nontrivial. Hence, for any nonzero vector $\boldsymbol{y} \in \tilde{T}(\mathbf{0}, 0)$, we have $\boldsymbol{y}^\top \boldsymbol{L}(\mathbf{0}, 0, 0)\boldsymbol{y} = 0 \not> 0$. Thus, the SOSC does not hold for the solution in part a.

21.9

a. The KKT condition for the problem is:

$$\begin{aligned} (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top \boldsymbol{A} + \lambda \boldsymbol{e}^\top - \boldsymbol{\mu}^\top &= \mathbf{0}^\top \\ \boldsymbol{\mu}^\top \boldsymbol{x} &= 0 \\ \boldsymbol{\mu} &\geq \mathbf{0} \\ \boldsymbol{e}^\top \boldsymbol{x} - 1 &= 0 \\ \boldsymbol{x} &\geq \mathbf{0} \end{aligned}$$

where $\boldsymbol{e} = [1, \ldots, 1]^\top$.

b. A feasible point $\boldsymbol{x}^*$ is regular in this problem if the vectors $\boldsymbol{e}$, $\boldsymbol{e}_i$, $i \in J(\boldsymbol{x}^*)$ are linearly independent, where $J(\boldsymbol{x}^*) = \{i : x_i^* = 0\}$ and $\boldsymbol{e}_i$ is the vector with 0 in all components except the ith component, which is 1.

In this problem, all feasible points are regular. To see this, note that $\mathbf{0}$ is not feasible. Therefore, any feasible point results in the set $J(\boldsymbol{x}^*)$ having fewer than n elements, which implies that the vectors $\boldsymbol{e}$, $\boldsymbol{e}_i$, $i \in J(\boldsymbol{x}^*)$ are linearly independent.

21.10

By the KKT Theorem, there exists $\mu^* \geq 0$ such that

$$\begin{aligned} (\boldsymbol{x}^* - \boldsymbol{x}_0) + \mu^* \nabla g(\boldsymbol{x}^*) &= 0 \\ \mu^* g(\boldsymbol{x}^*) &= 0. \end{aligned}$$

Premultiplying both sides of the first equation by $(\boldsymbol{x}^* - \boldsymbol{x}_0)^\top$, we obtain

$$\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \mu^* (\boldsymbol{x}^* - \boldsymbol{x}_0)^\top \nabla g(\boldsymbol{x}^*) = 0.$$

Since $\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 > 0$ (because $g(\boldsymbol{x}_0) > 0$) and $\mu^* \geq 0$, we deduce that $(\boldsymbol{x}^* - \boldsymbol{x}_0)^\top \nabla g(\boldsymbol{x}^*) < 0$ and $\mu^* > 0$. From the second KKT condition above, we conclude that $g(\boldsymbol{x}^*) = 0$.

21.11

a. By inspection, we guess the point $[2, 2]^\top$ (drawing a picture may help).

b. We write $f(\boldsymbol{x}) = (x_1 - 3)^2 + (x_2 - 4)^2$, $g_1(\boldsymbol{x}) = -x_1$, $g_2(\boldsymbol{x}) = -x_2$, $g_3(\boldsymbol{x}) = x_1 - 2$, $g_4(\boldsymbol{x}) = x_2 - 2$, $\boldsymbol{g} = [g_1, g_2, g_3, g_4]^\top$. The problem becomes

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}. \end{array}$$

We now check the SOSC for the point $\boldsymbol{x}^* = [2, 2]^\top$. We have two active constraints: g_3, g_4. Regularity holds, since $\nabla g_3(\boldsymbol{x}^*) = [1, 0]^\top$ and $\nabla g_4(\boldsymbol{x}^*) = [0, 1]^\top$. We have $\nabla f(\boldsymbol{x}^*) = [-2, -4]^\top$. We need to find a $\boldsymbol{\mu}^* \in \mathbb{R}^4$, $\boldsymbol{\mu}^* \geq \boldsymbol{0}$, satisfying FONC. From the condition $\boldsymbol{\mu}^{*\top} \boldsymbol{g}(\boldsymbol{x}^*) = 0$, we deduce that $\mu_1^* = \mu_2^* = 0$. Hence, $Df(\boldsymbol{x}^*) + \boldsymbol{\mu}^{*\top} D\boldsymbol{g}(\boldsymbol{x}^*) = \boldsymbol{0}^\top$ if and only if $\boldsymbol{\mu}^* = [0, 0, 2, 4]^\top$. Now,

$$\boldsymbol{F}(\boldsymbol{x}^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \qquad [\boldsymbol{\mu}^* \boldsymbol{G}(\boldsymbol{x}^*)] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Hence

$$\boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\mu}^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

which is positive definite on $\mathbb{R}^2$. Hence, SOSC is satisfied, and $\boldsymbol{x}^*$ is a strict local minimizer.

21.12

The KKT condition is

$$\begin{aligned} \boldsymbol{x}^\top \boldsymbol{Q} + \boldsymbol{\mu}^\top \boldsymbol{A} &= \boldsymbol{0}^\top \\ \boldsymbol{\mu}^\top (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) &= 0 \\ \boldsymbol{\mu} &\geq \boldsymbol{0} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} &\leq \boldsymbol{0}. \end{aligned}$$

Postmultiplying the first equation by $\boldsymbol{x}$ gives

$$\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{\mu}^\top \boldsymbol{A}\boldsymbol{x} = 0.$$

We note from the second equation that $\boldsymbol{\mu}^\top \boldsymbol{A}\boldsymbol{x} = \boldsymbol{\mu}^\top \boldsymbol{b}$. Hence,

$$\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{\mu}^\top \boldsymbol{b} = 0.$$

Since $\boldsymbol{Q} > 0$, the first term is nonnegative. Also, the second term is nonnegative because $\boldsymbol{\mu} \geq \boldsymbol{0}$ and $\boldsymbol{b} \geq \boldsymbol{0}$. Hence, we conclude that both terms must be zero. Because $\boldsymbol{Q} > 0$, we must have $\boldsymbol{x} = \boldsymbol{0}$.

Aside: Actually, we can deduce that the only solution to the KKT condition must be $\boldsymbol{0}$, as follows. The problem is convex; thus, the only points satisfying the KKT condition are global minimizers. However, we see that $\boldsymbol{0}$ is a feasible point, and is the only point for which the objective function value is 0. Further, the objective function is bounded below by 0. Hence, $\boldsymbol{0}$ is the only global minimizer.

21.13

a. We have one scalar equality constraint with $h(\boldsymbol{x}) = [c, d]^\top \boldsymbol{x} - e$ and two scalar inequality constraints with $\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{x}$. Hence, there exists $\boldsymbol{\mu}^* \in \mathbb{R}^2$ and $\lambda^* \in \mathbb{R}$ such that

$$\begin{aligned} \boldsymbol{\mu}^* &\geq 0 \\ a + c\lambda^* - \mu_1^* &= 0 \\ b + d\lambda^* - \mu_2^* &= 0 \\ \boldsymbol{\mu}^{*\top} \boldsymbol{x}^* &= 0 \\ cx_1^* + dx_2^* &= e \\ \boldsymbol{x}^* &\geq 0. \end{aligned}$$

b. Because $\boldsymbol{x}^*$ is a basic feasible solution, and the equality constraint precludes the point $\mathbf{0}$, exactly one of the inequality constraints is active. The vectors $\nabla h(\boldsymbol{x}^*) = [c, d]^\top$ and $\nabla g_1 = [1, 0]^\top$ are linearly independent. Similarly, the vectors $\nabla h(\boldsymbol{x}^*) = [c, d]^\top$ and $\nabla g_2 = [0, 1]^\top$ are linearly independent. Hence, $\boldsymbol{x}^*$ must be regular.

c. The tangent space is given by

$$\begin{aligned} T(\boldsymbol{x}^*) &= \{\boldsymbol{y} \in \mathbb{R}^n : Dh(\boldsymbol{x}^*)\boldsymbol{y} = \mathbf{0},\ Dg_j(\boldsymbol{x}^*)\boldsymbol{y} = 0,\ j \in J(\boldsymbol{x}^*)\} \\ &= \mathcal{N}(\boldsymbol{M}), \end{aligned}$$

where $\boldsymbol{M}$ is a matrix with the first row equal to $Dh(\boldsymbol{x}^*) = [c, d]$, and the second row is either $Dg_1 = [1, 0]$ or $Dg_2 = [0, 1]$. But, as we have seen in part b, rank $\boldsymbol{M} = 2$ Hence, $T(\boldsymbol{x}^*) = \{\mathbf{0}\}$.

d. Recall that we can take $\boldsymbol{\mu}^*$ to be the relative cost coefficient vector (i.e., the KKT conditions are satisfied with $\boldsymbol{\mu}^*$ being the relative cost coefficient vector). If the relative cost coefficients of all nonbasic variables are strictly positive, then $\mu_j^* > 0$ for all $j \in J(\boldsymbol{x}^*)$. Hence, $\tilde{T}(\boldsymbol{x}^*, \boldsymbol{\mu}^*) = T(\boldsymbol{x}^*) = \{\mathbf{0}\}$, which implies that $\boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) > 0$ on $\tilde{T}(\boldsymbol{x}^*, \boldsymbol{\mu}^*)$. Hence, the SOSC is satisfied.

21.14

Let $\boldsymbol{x}^*$ be a solution. Since $\boldsymbol{A}$ is of full rank, $\boldsymbol{x}^*$ is regular. The KKT Theorem states that $\boldsymbol{x}^*$ satisfies:

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \mathbf{0} \\ \boldsymbol{c}^\top + \boldsymbol{\mu}^{*\top}\boldsymbol{A} &= \mathbf{0} \\ \boldsymbol{\mu}^{*\top}\boldsymbol{A}\boldsymbol{x}^* &= 0. \end{aligned}$$

If we postmultiply the second equation by $\boldsymbol{x}^*$ and subtract the third from the result, we get

$$\boldsymbol{c}^\top \boldsymbol{x}^* = 0.$$

21.15

a. We can write the LP as

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}, \\ & \boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}, \end{aligned}$$

where $f(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x}$, $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$, and $\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{x}$. Thus, we have $Df(\boldsymbol{x}) = \boldsymbol{c}^\top$, $D\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{A}$, and $D\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{I}$. The Karush-Kuhn-Tucker conditions for the above problem have the form: if $\boldsymbol{x}^*$ is a local minimizer, then there exists $\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$ such that

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \mathbf{0} \\ \boldsymbol{c}^\top + \boldsymbol{\lambda}^{*\top}\boldsymbol{A} - \boldsymbol{\mu}^{*\top} &= \mathbf{0}^\top \\ \boldsymbol{\mu}^{*\top}\boldsymbol{x}^* &= 0. \end{aligned}$$

b. Let $\boldsymbol{x}^*$ be an optimal feasible solution. Then, $\boldsymbol{x}^*$ satisfies the Karush-Kuhn-Tucker conditions listed in part a. Since $\boldsymbol{\mu}^* \geq \mathbf{0}$, then from the second condition in part a, we obtain $(-\boldsymbol{\lambda}^*)^\top \boldsymbol{A} \leq \boldsymbol{c}^\top$. Hence, $\bar{\boldsymbol{\lambda}} = -\boldsymbol{\lambda}^*$ is a feasible solution to the dual (see Chapter 17). Postmultiplying the second condition in part a by $\boldsymbol{x}^*$, we have

$$0 = \boldsymbol{c}^\top \boldsymbol{x}^* + \boldsymbol{\lambda}^{*\top}\boldsymbol{A}\boldsymbol{x} - \boldsymbol{\mu}^{*\top}\boldsymbol{x}^* = \boldsymbol{c}^\top \boldsymbol{x}^* + \boldsymbol{\lambda}^{*\top}\boldsymbol{b}$$

which gives

$$\boldsymbol{c}^\top \boldsymbol{x}^* = \bar{\boldsymbol{\lambda}}^\top \boldsymbol{b}.$$

Hence, $\bar{\boldsymbol{\lambda}}$ achieves the same objective function value for the dual as $\boldsymbol{x}^*$ for the primal.

c. From part a, we have $\boldsymbol{\mu}^{*\top} = \boldsymbol{c}^\top - \bar{\boldsymbol{\lambda}}\boldsymbol{A}$. Substituting this into $\boldsymbol{\mu}^{*\top}\boldsymbol{x}^* = 0$ yields the desired result.

21.16

By definition of $J(\boldsymbol{x}^*)$, we have $g_i(\boldsymbol{x}^*) < 0$ for all $i \notin J(\boldsymbol{x}^*)$. Since by assumption g_i is continuous for all i, there exists $\varepsilon > 0$ such that $g_i(\boldsymbol{x}) < 0$ for all $i \notin J(\boldsymbol{x}^*)$ and all $\boldsymbol{x}$ in the set $B = \{\boldsymbol{x} : \|\boldsymbol{x} - \boldsymbol{x}^*\| < \varepsilon\}$. Let $S_1 = \{\boldsymbol{x} : \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}, g_j(\boldsymbol{x}) \leq 0, j \in J(\boldsymbol{x}^*)\}$. We claim that $S \cap B = S_1 \cap B$. To see this, note that clearly $S \cap B \subset S_1 \cap B$. To show that $S_1 \cap B \subset S \cap B$, suppose $\boldsymbol{x} \in S_1 \cap B$. Then, by definition of S_1 and B, we have $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$, $g_j(\boldsymbol{x}) \leq 0$ for all $j \in J(\boldsymbol{x}^*)$, and $g_i(\boldsymbol{x}) < 0$ for all $i \notin J(\boldsymbol{x}^*)$. Hence, $\boldsymbol{x} \in S \cap B$.

Since $\boldsymbol{x}^*$ is a local minimizer of f over S, and $S \cap B \subset S$, $\boldsymbol{x}^*$ is also a local minimizer of f over $S \cap B = S_1 \cap B$. Hence, we conclude that $\boldsymbol{x}^*$ is a regular local minimizer of f on S_1. Note that $S' \subset S_1$, and $\boldsymbol{x}^* \in S'$. Therefore, $\boldsymbol{x}^*$ is a regular local minimizer of f on S'.

21.17

Write $f(\boldsymbol{x}) = x_1^2 + x_2^2$, $g_1(\boldsymbol{x}) = x_1^2 - x_2 - 4$, $g_2(\boldsymbol{x}) = x_2 - x_1 - 2$, and $\boldsymbol{g} = [g_1, g_2]^\top$. We have $\nabla f(\boldsymbol{x}) = [2x_1, 2x_2]^\top$, $\nabla g_1(\boldsymbol{x}) = [2x_1, -1]^\top$, $\nabla g_2(\boldsymbol{x}) = [-1, 1]^\top$, and $D^2 f(\boldsymbol{x}) = \operatorname{diag}[2, 2]$. We compute

$$\nabla f(\boldsymbol{x}) + \boldsymbol{\mu}^\top \nabla \boldsymbol{g}(\boldsymbol{x}) = [2x_1 + 2\mu_1 x_1 - \mu_2, 2x_2 - \mu_1 + \mu_2]^\top.$$

We use the FONC to find critical points. Rewriting $\nabla f(\boldsymbol{x}) + \boldsymbol{\mu}^\top \nabla \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}$, we obtain

$$x_1 = \frac{\mu_2}{2 + 2\mu_1}, \qquad x_2 = \frac{\mu_1 - \mu_2}{2}.$$

We also use $\boldsymbol{\mu}^\top \boldsymbol{g}(\boldsymbol{x}) = 0$ and $\boldsymbol{\mu} \geq \boldsymbol{0}$, giving

$$\mu_1(x_1^2 - x_2 - 4) = 0, \qquad \mu_2(x_2 - x_1 - 2) = 0.$$

The vector $\boldsymbol{\mu}$ has two components; therefore, we try four different cases.

Case 1: ($\mu_1 > 0$, $\mu_2 > 0$) We have

$$x_1^2 - x_2 - 4 = 0, \qquad x_2 - x_1 - 2 = 0.$$

We obtain two solutions: $\boldsymbol{x}^{(1)} = [-2, 0]^\top$ and $\boldsymbol{x}^{(2)} = [3, 5]^\top$. For $\boldsymbol{x}^{(1)}$, the two FONC equations give $\mu_1 = \mu_2$ and $-2(2+2\mu_1) = \mu_1$, which yield $\mu_1 = \mu_2 = -4/5$. This is not a legitimate solution since we require $\boldsymbol{\mu} \geq \boldsymbol{0}$. For $\boldsymbol{x}^{(2)}$, the two FONC equations give $\mu_1 - \mu_2 = 10$ and $3(2 + 2\mu_1) = \mu_2$, which yield $\boldsymbol{\mu} = [-16/5, -66/5]$. Again, this is not a legitimate solution.

Case 2: ($\mu_1 = 0$, $\mu_2 > 0$) We have

$$x_2 - x_1 - 2 = 0, \qquad x_1 = \frac{\mu_2}{2}, \qquad x_2 = -\frac{\mu_2}{2}.$$

Hence, $x_1 = -x_2$, and thus $\boldsymbol{x} = [-1, 1]$, $\mu_2 = -2$. This is not a legitimate solution since we require $\boldsymbol{\mu} \geq \boldsymbol{0}$.

Case 3: ($\mu_1 > 0$, $\mu_2 = 0$) We have

$$x_1^2 - x_2 - 4 = 0, \qquad x_1 = 0, \qquad x_2 = \frac{\mu_1}{2}.$$

Therefore, $x_2 = -4$, $\mu_1 = -8$, and again we don't have a legitimate solution.

Case 4: ($\mu_1 = 0$, $\mu_2 = 0$) We have $x_1 = x_2 = 0$, and all constraints are inactive. This is a legitimate candidate for the minimizer. We now apply the SOSC. Note that since the candidate is an interior point of the constraint set, the SOSC for the problem is equivalent to the SOSC for unconstrained optimization. The Hessian matrix $D^2 f(\boldsymbol{x}) = \operatorname{diag}[2, 2]$ is symmetric and positive definite. Hence, by the SOSC, the point $\boldsymbol{x}^* = [0, 0]^\top$ is the strict local minimizer (in fact, it is easy to see that it is a global minimizer).

21.18

Write $f(\boldsymbol{x}) = x_1^2 + x_2^2$, $g_1(\boldsymbol{x}) = -x_1 + x_2^2 + 4$, $g_2(\boldsymbol{x}) = x_1 - 10$, and $\boldsymbol{g} = [g_1, g_2]^\top$. We have $\nabla f(\boldsymbol{x}) = [2x_1, 2x_2]^\top$, $\nabla g_1(\boldsymbol{x}) = [-1, 2x_2]^\top$, $\nabla g_2(\boldsymbol{x}) = [1, 0]^\top$, $D^2 f(\boldsymbol{x}) = \operatorname{diag}[2, 2]$, $D^2 g_1(\boldsymbol{x}) = \operatorname{diag}[0, 2]$, and $D^2 g_2(\boldsymbol{x}) = \boldsymbol{O}$. We compute

$$\nabla f(\boldsymbol{x}) + \boldsymbol{\mu}^\top \nabla \boldsymbol{g}(\boldsymbol{x}) = [2x_1 - \mu_1 + \mu_2, 2x_2 + 2\mu_1 x_2]^\top.$$

We use the FONC to find critical points. Rewriting $\nabla f(\boldsymbol{x}) + \boldsymbol{\mu}^\top \nabla \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}$, we obtain

$$x_1 = \frac{\mu_1 - \mu_2}{2}, \qquad x_2(1 + \mu_1) = 0.$$

Since we require $\boldsymbol{\mu} \geq \mathbf{0}$, we deduce that $x_2 = 0$. Using $\boldsymbol{\mu}^\top \boldsymbol{g}(\boldsymbol{x}) = 0$ gives

$$\mu_1(-x_1 + 4) = 0, \qquad \mu_2 = 0.$$

We are left with two cases.

Case 1: ($\mu_1 > 0$, $\mu_2 = 0$) We have $-x_1 + 4 = 0$, and $\mu_1 = 8$, which is a legitimate candidate.

Case 2: ($\mu_1 = 0$, $\mu_2 = 0$) We have $x_1 = x_2 = 0$, which is not a legitimate candidate, since it is not a feasible point.

We now apply SOSC to our candidate $\boldsymbol{x} = [4, 0]^\top$, $\boldsymbol{\mu} = [8, 0]^\top$. Now,

$$\boldsymbol{L}([4, 0]^\top, [8, 0]^\top) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} + 8 \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 18 \end{bmatrix},$$

which is positive definite on all of $\mathbb{R}^2$. The point $[4, 0]^\top$ is clearly regular. Hence, by the SOSC, $\boldsymbol{x}^* = [4, 0]$ is a strict local minimizer.

21.19

Write $f(\boldsymbol{x}) = x_1^2 + x_2^2$, $g_1(\boldsymbol{x}) = -x_1 - x_2^2 + 4$, $g_2(\boldsymbol{x}) = 3x_2 - x_1$, $g_3(\boldsymbol{x}) = -3x_2 - x_1$ and $\boldsymbol{g} = [g_1, g_2, g_3]^\top$. We have $\nabla f(\boldsymbol{x}) = [2x_1, 2x_2]^\top$, $\nabla g_1(\boldsymbol{x}) = [-1, -2x_2]^\top$, $\nabla g_2(\boldsymbol{x}) = [-1, 3]^\top$, $\nabla g_2(\boldsymbol{x}) = [-1, -3]^\top$, $D^2 f(\boldsymbol{x}) = \operatorname{diag}[2, 2]$, $D^2 g_1(\boldsymbol{x}) = \operatorname{diag}[0, -2]$, and $D^2 g_2(\boldsymbol{x}) = D^2 g_3(\boldsymbol{x}) = \boldsymbol{O}$.

From the figure, we see that the two candidates are $\boldsymbol{x}^{(1)} = [3, 1]$ and $\boldsymbol{x}^{(2)} = [3, -1]$. Both points are easily verified to be regular.

For $\boldsymbol{x}^{(1)}$, we have $\mu_3 = 0$. Now,

$$Df(\boldsymbol{x}^{(1)}) + \boldsymbol{\mu}^\top D\boldsymbol{g}(\boldsymbol{x}^{(1)}) = [6 - \mu_1 - \mu_2, 2 - 2\mu_1 + 3\mu_2] = \mathbf{0}^\top,$$

which yields $\mu_1 = 4$, $\mu_2 = 2$. Now, $\tilde{T}(\boldsymbol{x}^{(1)}) = \{\mathbf{0}\}$. Therefore, any matrix is positive definite on $\tilde{T}(\boldsymbol{x}^{(1)})$. Hence, by the SOSC, $\boldsymbol{x}^{(1)}$ is a strict local minimizer.

For $\boldsymbol{x}^{(2)}$, we have $\mu_2 = 0$. Now,

$$Df(\boldsymbol{x}^{(1)}) + \boldsymbol{\mu}^\top D\boldsymbol{g}(\boldsymbol{x}^{(1)}) = [6 - \mu_1 - \mu_3, -2 + 2\mu_1 - 3\mu_3] = \mathbf{0}^\top,$$

which yields $\mu_1 = 4$, $\mu_3 = 2$. Now, again we have $\tilde{T}(\boldsymbol{x}^{(2)}) = \{\mathbf{0}\}$. Therefore, any matrix is positive definite on $\tilde{T}(\boldsymbol{x}^{(2)})$. Hence, by the SOSC, $\boldsymbol{x}^{(2)}$ is a strict local minimizer.

21.20

a. Write $f(\boldsymbol{x}) = 3x_1$ and $g(\boldsymbol{x}) = 2 - x_1 - x_2^2$. We have $\nabla f(\boldsymbol{x}^*) = [3, 0]^\top$ and $\nabla g(\boldsymbol{x}^*) = [-1, 0]^\top$. Hence, letting $\mu^* = 3$, we have $\nabla f(\boldsymbol{x}^*) + \mu^* \nabla g(\boldsymbol{x}^*) = 0$. Note also that $\mu^* \geq 0$ and $\mu^* g(\boldsymbol{x}^*) = 0$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ satisfies the KKT (first order necessary) condition.

b. We have $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{O}$ and $\boldsymbol{G}(\boldsymbol{x}^*) = \operatorname{diag}[0, -2]$. Hence, $\boldsymbol{L}(\boldsymbol{x}^*, \mu^*) = \boldsymbol{O} + 3\operatorname{diag}[0, -2] = \operatorname{diag}[0, -6]$. Also, $T(\boldsymbol{x}^*) = \{\boldsymbol{y} : [-1, 0]\boldsymbol{y} = 0\} = \{\boldsymbol{y} : y_1 = 0\}$. Hence, $\boldsymbol{x}^* = [2, 0]^\top$ does not satisfy the second order necessary condition.

c. No. Consider points of the form $\boldsymbol{x} = [-x_2^2 + 2, x_2]^\top$, $x_2 \in \mathbb{R}$. Such points are feasible, and could be arbitrarily close to $\boldsymbol{x}^*$. However, for such points $\boldsymbol{x} \neq \boldsymbol{x}^*$,

$$f(\boldsymbol{x}) = 3(-x_2^2 + 2) = 6 - 6x_2^2 < 6 = f(\boldsymbol{x}^*).$$

Hence, $\boldsymbol{x}^*$ is not a local minimizer.

21.21

The KKT condition for the problem is

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \mathbf{0} \\ \boldsymbol{x}^* + \lambda^* \boldsymbol{a} - \boldsymbol{\mu}^* &= \mathbf{0} \\ \boldsymbol{\mu}^{*\top} \boldsymbol{x}^* &= 0. \end{aligned}$$

Premultiplying the second KKT condition above by $\boldsymbol{\mu}^{*\top}$ and using the third condition, we get

$$\lambda^* \boldsymbol{\mu}^{*\top} \boldsymbol{a} = \|\boldsymbol{\mu}^*\|^2.$$

Also, premultiplying the second KKT condition above by $\boldsymbol{x}^{*\top}$ and using the feasibility condition $\boldsymbol{a}^\top \boldsymbol{x}^* = b$, we get

$$\lambda^* = -\frac{\|\boldsymbol{x}^*\|^2}{b} < 0.$$

We conclude that $\boldsymbol{\mu}^* = \boldsymbol{0}$. For if not, the equation $\lambda^* \boldsymbol{\mu}^{*\top} \boldsymbol{a} = \|\boldsymbol{\mu}^*\|^2$ implies that $\boldsymbol{\mu}^{*\top} \boldsymbol{a} < 0$, which contradicts $\boldsymbol{\mu}^* \geq \boldsymbol{0}$ and $\boldsymbol{a} \geq \boldsymbol{0}$.

Rewriting the second KKT condition with $\boldsymbol{\mu}^* = \boldsymbol{0}$ yields

$$\boldsymbol{x}^* = -\lambda^* \boldsymbol{a}.$$

Using the feasibility condition $\boldsymbol{a}^\top \boldsymbol{x}^* = b$, we get

$$\boldsymbol{x}^* = \boldsymbol{a} \frac{b}{\|\boldsymbol{a}\|^2}.$$

21.22

a. Suppose $(x_1^*)^2 + (x_2^*)^2 < 1$. Then, the point $\boldsymbol{x}^* = [x_1^*, x_2^*]^\top$ lies in the interior of the constraint set $\Omega = \{\boldsymbol{x} : \|\boldsymbol{x}\|^2 \leq 1\}$. Hence, by the FONC for unconstrained optimization, we have that $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$, where $f(\boldsymbol{x}) = \|\boldsymbol{x} - [a, b]^\top\|^2$ is the objective function. Now, $\nabla f(\boldsymbol{x}^*) = 2(\boldsymbol{x}^* - [a, b]^\top) = \boldsymbol{0}$, which implies that $\boldsymbol{x}^* = [a, b]^\top$ which violates the assumption $(x_1^*)^2 + (x_2^*)^2 < 1$.

b. First, we need to show that $\boldsymbol{x}^*$ is a regular point. For this, note that if we write the constraint as $g(\boldsymbol{x}) = \|\boldsymbol{x}\|^2 - 1 \leq 0$, then $\nabla g(\boldsymbol{x}^*) = 2\boldsymbol{x}^* \neq \boldsymbol{0}$. Therefore, $\boldsymbol{x}^*$ is a regular point. Hence, by the Karush-Kuhn-Tucker theorem, there exists $\mu \in \mathbb{R}$, $\mu \geq 0$, such that

$$\nabla f(\boldsymbol{x}^*) + \mu \nabla g(\boldsymbol{x}^*) = \boldsymbol{0},$$

which gives

$$\boldsymbol{x}^* = \frac{1}{1+\mu} \begin{bmatrix} a \\ b \end{bmatrix}.$$

Hence, $\boldsymbol{x}^*$ is unique, and we can write $x_1^* = \alpha a$, $x_2^* = \alpha b$, where $\alpha = 1/(1+\mu) \geq 0$.

c. Using part b and the fact that $\|\boldsymbol{x}^*\| = 1$, we get $\|\boldsymbol{x}^*\|^2 = \alpha^2 \|[a, b]^\top\|^2 = 1$, which gives $\alpha = 1/\|[a, b]^\top\| = 1/\sqrt{a^2 + b^2}$.

21.23

a. The Karush-Kuhn-Tucker conditions for this problem are

$$\begin{aligned} 2x_1^* + \mu^* \exp(x_1^*) &= 0 \\ 2(x_2^* + 1) - \mu^* &= 0 \\ \mu^*(\exp(x_1^*) - x_2^*) &= 0 \\ \exp(x_1^*) &\leq x_2^* \\ \mu^* &\geq 0. \end{aligned}$$

b. From the second equation in part a, we obtain $\mu^* = 2(x_2^* + 1)$. Since $x_2^* \geq \exp(x_1^*) > 0$, then $\mu^* > 0$. Hence, by the third equation in part a, we obtain $x_2^* = \exp(x_1^*)$.

c. Since $\mu^* = 2(x_2^* + 1) = 2(\exp(x_1^*) + 1)$, then by the first equation in part a, we have

$$2x_1^* + 2(\exp(x_1^*) + 1)\exp(x_1^*) = 0$$

which implies

$$x_1^* = -(\exp(2x_1^*) + \exp(x_1^*)).$$

Since $\exp(x_1^*), \exp(2x_1^*) > 0$, then $x_1^* < 0$, and hence $\exp(x_1^*), \exp(2x_1^*) < 1$. Therefore, $x_1^* > -2$.

21.24

a. We rewrite the problem as

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g(\boldsymbol{x}) \leq 0, \end{array}$$

where $f(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x}$ and $g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|^2 - 1$. Hence, $\nabla f(\boldsymbol{x}) = \boldsymbol{c}$ and $\nabla g(\boldsymbol{x}) = \boldsymbol{x}$. Note that $\boldsymbol{x}^* \neq \boldsymbol{0}$ (for otherwise it would not be feasible), and therefore it is a regular point. By the KKT theorem, there exists $\mu^* \leq 0$ such that $\boldsymbol{c} = \mu^* \boldsymbol{x}^*$ and $\mu^* g(\boldsymbol{x}^*) = 0$. Since $\boldsymbol{c} \neq \boldsymbol{0}$, we must have $\mu^* \neq 0$. Therefore, $g(\boldsymbol{x}^*) = 0$, which implies that $\|\boldsymbol{x}^*\|^2 = 2$.

b. From part a, we have $\alpha^2 \|\boldsymbol{e}\|^2 = 2$. Since $\|\boldsymbol{e}\|^2 = n$, we have $\alpha = \sqrt{2/n}$.
To find $\boldsymbol{c}$, we use

$$4 = \boldsymbol{c}^\top \boldsymbol{x}^* + 8 = \mu^* \|\boldsymbol{x}^*\|^2 + 8 = 2\mu^* + 8,$$

and thus $\mu^* = -2$. Hence, $\boldsymbol{c} = -2\alpha \boldsymbol{e} = -\left(2\sqrt{2/n}\right)\boldsymbol{e}$.

21.25

We can represent the equivalent problem as

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g(\boldsymbol{x}) \leq 0, \end{array}$$

where $g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{h}(\boldsymbol{x})\|^2$. Note that

$$\nabla g(\boldsymbol{x}) = D\boldsymbol{h}(\boldsymbol{x})^\top \boldsymbol{h}(\boldsymbol{x}).$$

Therefore, the KKT condition is:

$$\begin{aligned} \mu^* &\geq 0 \\ \nabla f(\boldsymbol{x}^*) + \mu^* D\boldsymbol{h}(\boldsymbol{x}^*)^\top \boldsymbol{h}(\boldsymbol{x}^*) &= \boldsymbol{0} \\ \mu^* \|\boldsymbol{h}(\boldsymbol{x}^*)\| &= 0. \end{aligned}$$

Note that for a feasible point $\boldsymbol{x}^*$, we have $\boldsymbol{h}(\boldsymbol{x}^*) = \boldsymbol{0}$. Therefore, the KKT condition becomes

$$\begin{aligned} \mu^* &\geq 0 \\ \nabla f(\boldsymbol{x}^*) &= \boldsymbol{0}. \end{aligned}$$

Note that $\nabla g(\boldsymbol{x}^*) = \boldsymbol{0}$. Therefore, any feasible point $\boldsymbol{x}^*$ is not regular. Hence, the KKT theorem cannot be applied in this case. This should be clear, since obviously $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ is not necessary for optimality in general.

22. Convex Optimization Problems

22.1

The given function is a quadratic, which we represent in the form

$$f = \boldsymbol{x}^\top \begin{bmatrix} -1 & -\alpha & 1 \\ -\alpha & -1 & -2 \\ 1 & -2 & -5 \end{bmatrix} \boldsymbol{x}.$$

A quadratic function is concave if and only if it is negative-semidefinite. Equivalently, if and only if its negative is positive-semidefinite. On the other hand, a symmetric matrix is positive semidefinite if and only

if all its principal minors, not just the leading principal minors, are nonnegative. Thus we will determine the range of the parameter α for which

$$-f = \boldsymbol{x}^\top \begin{bmatrix} 1 & \alpha & -1 \\ \alpha & 1 & 2 \\ -1 & 2 & 5 \end{bmatrix} \boldsymbol{x}.$$

is positive-semidefinite. It is easy to see that the three first-order principal minors (diagonal elements of $\boldsymbol{F}$) are all positive. There are three second-order principal minors. Only one of them, the leading principal minor, is a function of the parameter α,

$$\det \boldsymbol{F}(1\colon !2, 1\colon !2) = \det \begin{bmatrix} 1 & \alpha \\ \alpha & 1 \end{bmatrix} = 1 - \alpha^2.$$

The above second-order leading principal minor is nonnegative if and only if

$$\alpha \in [-1, 1].$$

The other second-order principal minors are

$$\det \boldsymbol{F}(1\colon !3, 1\colon !3) \qquad \text{and} \qquad \det \boldsymbol{F}(2\colon !3, 2\colon !3),$$

and they are positive. There is only one third-order principal minor, $\det \boldsymbol{F}$, where

$$\begin{aligned} \det \boldsymbol{F} &= \det \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} - \alpha \det \begin{bmatrix} \alpha & -1 \\ 2 & 5 \end{bmatrix} - \det \begin{bmatrix} \alpha & -1 \\ 1 & 2 \end{bmatrix} \\ &= 1 - \alpha(5\alpha + 2) - (2\alpha + 1) \\ &= 1 - 5\alpha^2 - 2\alpha - 2\alpha - 1 \\ &= -5\alpha^2 - 4\alpha. \end{aligned}$$

The third-order principal minor is nonnegative if and only if, $-\alpha(5\alpha + 4) \geq 0$, that is, if and only if

$$\alpha \in [-4/5, 0].$$

Combining this with $\alpha \in [-1, 1]$ from above, we conclude that the function f is negative-semidefinite, equivalently, the quadratic function f is concave, if and only if

$$\alpha \in [-4/5, 0].$$

22.2

We have

$$\begin{aligned} \phi(\alpha) &= \frac{1}{2}(\boldsymbol{x} + \alpha\boldsymbol{d})^\top \boldsymbol{Q}(\boldsymbol{x} + \alpha\boldsymbol{d}) - (\boldsymbol{x} + \alpha\boldsymbol{d})\boldsymbol{b} \\ &= \frac{1}{2}(\boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d})\alpha^2 + \boldsymbol{d}^\top(\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b})\alpha + \left(\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{b}\right). \end{aligned}$$

This is a quadratic function of α. Since $\boldsymbol{Q} > 0$, then

$$\frac{d^2\phi}{d\alpha^2}(\alpha) = \boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d} > 0$$

and hence by Theorem 22.5, ϕ is strictly convex.

22.3

Write $f(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}$, where

$$\boldsymbol{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$. Then, $\boldsymbol{x} = [a_1, ma_1]^\top$ and $\boldsymbol{y} = [a_2, ma_2]^\top$ for some $a_1, a_2 \in \mathbb{R}$. By Proposition 22.1, it is enough to show that $(\boldsymbol{y} - \boldsymbol{x})^\top \boldsymbol{Q}(\boldsymbol{y} - \boldsymbol{x}) \geq 0$. By substitution,

$$(\boldsymbol{y} - \boldsymbol{x})^\top \boldsymbol{Q}(\boldsymbol{y} - \boldsymbol{x}) = m(a_2 - a_1)^2 \geq 0,$$

which completes the proof.

22.4

Let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0, 1)$. Then, $h(\boldsymbol{x}) = h(\boldsymbol{y}) = c$. By convexity of Ω, $h(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) = c$. Therefore,

$$h(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) = \alpha h(\boldsymbol{x}) + (1-\alpha)h(\boldsymbol{y})$$

and so h is convex over Ω. We also have

$$-h(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) = \alpha(-h(\boldsymbol{x})) + (1-\alpha)(-h(\boldsymbol{y})),$$

which shows that $-h$ is convex, and thus h is concave.

22.5

At $x = 0$, for $\xi \in [-1, 1]$, we have, for all $y \in \mathbb{R}$,

$$|y| \geq |0| + \xi(y - 0) = \xi y.$$

Thus, in this case any ξ in the interval $[-1, 1]$ is a subgradient of f at $x = 0$.

At $x = 1$, $\xi = 1$ is the only subgradient of f, because, for all $y \in \mathbb{R}$,

$$|y| \geq 1 + \xi(y - 1) = y.$$

22.6

Let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0, 1)$. For convenience, write $\bar{f} = \max\{f_1, \ldots, f_\ell\} = \max_i f_i$. We have

$$\begin{aligned}
\bar{f}(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) &= \max_i f_i(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \\
&\leq \max_i \left(\alpha f_i(\boldsymbol{x}) + (1-\alpha) f_i(\boldsymbol{y})\right) \qquad \text{by convexity of each } f_i \\
&\leq \alpha \max_i f_i(\boldsymbol{x}) + (1-\alpha) \max_i f_i(\boldsymbol{y}) \qquad \text{by property of max} \\
&= \alpha\bar{f}(\boldsymbol{x}) + (1-\alpha)\bar{f}(\boldsymbol{y})
\end{aligned}$$

which implies that $\bar{f}$ is convex.

22.7

$\Rightarrow$: This is true by definition.

$\Leftarrow$: Let $\boldsymbol{d} \in \mathbb{R}^n$ be given. We want to show that $\boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d} \geq 0$. Now, fix some vector $\boldsymbol{x} \in \Omega$. Since Ω is open, there exists $\alpha \neq 0$ such that $\boldsymbol{y} = \boldsymbol{x} - \alpha\boldsymbol{d} \in \Omega$. By assumption,

$$0 \leq (\boldsymbol{y} - \boldsymbol{x})^\top \boldsymbol{Q}(\boldsymbol{y} - \boldsymbol{x}) = \alpha^2 \boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d}$$

which implies that $\boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d} \geq 0$.

22.8

Yes, the problem is a convex optimization problem.

First we show that the objective function $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ is convex. We write

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top(\boldsymbol{A}^\top\boldsymbol{A})\boldsymbol{x} - (\boldsymbol{b}^\top\boldsymbol{A})\boldsymbol{x} + \text{constant}$$

which is a quadratic function with Hessian $\boldsymbol{A}^\top\boldsymbol{A}$. Since the Hessian $\boldsymbol{A}^\top\boldsymbol{A}$ is positive semidefinite, the objective function f is convex.

Next we show that the constraint set is convex. Consider two feasible points $\boldsymbol{x}$ and $\boldsymbol{y}$, and let $\lambda \in (0, 1)$. Then, $\boldsymbol{x}$ and $\boldsymbol{y}$ satisfy $\boldsymbol{e}^\top\boldsymbol{x} = 1, \boldsymbol{x} \geq \boldsymbol{0}$ and $\boldsymbol{e}^\top\boldsymbol{y} = 1, \boldsymbol{y} \geq \boldsymbol{0}$, respectively. We have

$$\boldsymbol{e}^\top(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) = \lambda\boldsymbol{e}^\top\boldsymbol{x} + (1-\lambda)\boldsymbol{e}^\top\boldsymbol{y} = \lambda + (1-\lambda) = 1.$$

Moreover, each component of $\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}$ is given by $\lambda x_i + (1-\lambda)y_i$, which is nonnegative because every term here is nonnegative. Hence, $\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}$ is a feasible point, which shows that the constraint set is convex.

22.9

We need to show that Ω is a convex set, and f is a convex function on Ω.

To show that Ω is a convex set, we need to show that for any $\boldsymbol{y}, \boldsymbol{z} \in \Omega$ and $\alpha \in (0,1)$, we have $\alpha\boldsymbol{y}+(1-\alpha)\boldsymbol{z} \in \Omega$. Let $\boldsymbol{y}, \boldsymbol{z} \in \Omega$ and $\alpha \in (0,1)$. Thus, $y_1 = y_2 \geq 0$ and $z_1 = z_2 \geq 0$. Hence,

$$\boldsymbol{x} = \alpha\boldsymbol{y} + (1-\alpha)\boldsymbol{z} = \begin{bmatrix} \alpha y_1 + (1-\alpha)z_1 \\ \alpha y_2 + (1-\alpha)z_2 \end{bmatrix}$$

Now,

$$x_1 = \alpha y_1 + (1-\alpha)z_1 = \alpha y_2 + (1-\alpha)z_2 = x_2,$$

and since $\alpha, 1-\alpha \geq 0$,

$$x_1 \geq 0.$$

Hence, $\boldsymbol{x} \in \Omega$ and therefore Ω is convex.

To show that f is convex on Ω, we need to show that for any $\boldsymbol{y}, \boldsymbol{z} \in \Omega$ and $\alpha \in [0,1]$, $f(\alpha\boldsymbol{y} + (1-\alpha)\boldsymbol{z}) \leq \alpha f(\boldsymbol{y}) + (1-\alpha)f(\boldsymbol{z})$. Let $\boldsymbol{y}, \boldsymbol{z} \in \Omega$ and $\alpha \in [0,1]$. Thus, $y_1 = y_2 \geq 0$ and $z_1 = z_2 \geq 0$, so that $f(\boldsymbol{y}) = y_1^3$ and $f(\boldsymbol{z}) = z_1^3$. Also, $\alpha^3 \leq \alpha$ and $(1-\alpha)^3 \leq (1-\alpha)$. We have

$$\begin{aligned} f(\alpha\boldsymbol{y} + (1-\alpha)\boldsymbol{z}) &= (\alpha y_1 + (1-\alpha)z_1)^3 \\ &= \alpha^3 y_1^3 + (1-\alpha)^3 z_1^3 + 3\alpha^2 x_1^2(1-\alpha)y_1 + 3\alpha x_1(1-\alpha)^2 y_1^2 \\ &\leq \alpha y_1^3 + (1-\alpha)z_1^3 + \max(y_1, z_1)(\alpha^3 - \alpha + (1-\alpha)^3 - (1-\alpha) \\ &\qquad + 3\alpha^2(1-\alpha) + 3\alpha(1-\alpha)^2) \\ &= \alpha y_1^3 + (1-\alpha)z_1^3 \\ &= \alpha f(\boldsymbol{y}) + (1-\alpha)f(\boldsymbol{z}). \end{aligned}$$

Hence, f is convex.

22.10

Since the problem is a convex optimization problem, we know for sure that any point of the form $\alpha\boldsymbol{y}+(1-\alpha)\boldsymbol{z}$, $\alpha \in (0,1)$, is a global minimizer. However, any other point may or may not be a minimizer. Hence, the largest set of points $G \subset \Omega$ for which we can be sure that every point in G is a global minimizer, is given by

$$G = \{\alpha\boldsymbol{y} + (1-\alpha)\boldsymbol{z} : 0 \leq \alpha \leq 1\}.$$

22.11

a. Let f be the objective function and Ω the constraint set. Consider the set $\Gamma = \{\boldsymbol{x} \in \Omega : f(\boldsymbol{x}) \leq 1\}$. This set contains all three of the given points. Moreover, by Lemma 22.1, Γ is convex. Now, if we take the average of the first two points (which is a convex combination of them), the resulting point $(1/2)[1,0,0]^\top + (1/2)[0,1,0]^\top = (1/2)[1,1,0]^\top$ is in Γ, because Γ is convex. Similarly, the point $(2/3)(1/2)[1,1,0]^\top + (1/3)[0,0,1]^\top = (1/3)[1,1,1]^\top$ is also in Γ, because Γ is convex. Hence, the objective function value of $(1/3)[1,1,1]^\top$ must be ≤ 1.

b. If the three points are all global minimizers, then the point $(1/3)[1,1,1]^\top$, which must cannot have higher objective function value than the given three points (by part a), must also be a global minimizer.

22.12

a. The Lagrange condition for the problem is given by:

$$\begin{aligned} \boldsymbol{x}^\top\boldsymbol{Q} + \boldsymbol{\lambda}^\top\boldsymbol{A} &= \boldsymbol{0} \\ \boldsymbol{A}\boldsymbol{x} &= \boldsymbol{b}. \end{aligned}$$

From the first equation above, we obtain

$$\boldsymbol{x} = \boldsymbol{Q}^{-1}\boldsymbol{A}^\top\boldsymbol{\lambda}.$$

Applying the second equation (constraint on $\boldsymbol{x}$), we have

$$(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)\boldsymbol{\lambda} = \boldsymbol{b}.$$

Since rank $\boldsymbol{A} = m$, the matrix $\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top$ is invertible. Therefore, the only solution to the Lagrange condition is

$$\boldsymbol{x} = \boldsymbol{Q}^{-1}\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{Q}^{-1}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}.$$

b. The point in part a above is a global minimizer because the problem is a convex optimization problem (by problem 1, the constraint set is convex; the objective function is convex because its Hessian, $\boldsymbol{Q}$, is positive definite).

22.13

By Theorem 22.4, for all $\boldsymbol{x} \in \Omega$, we have

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + Df(\boldsymbol{x}^*)(\boldsymbol{x} - \boldsymbol{x}^*).$$

Substituting $Df(\boldsymbol{x}^*)$ from the equation $Df(\boldsymbol{x}^*) + \sum_{j\in J(\boldsymbol{x}^*)} \mu_j^* \boldsymbol{a}_j^\top = \boldsymbol{0}^\top$ into the above inequality yields

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) - \sum_{j\in J(\boldsymbol{x}^*)} \mu_j^* \boldsymbol{a}_j^\top(\boldsymbol{x} - \boldsymbol{x}^*).$$

Observe that for each $j \in J(\boldsymbol{x}^*)$,

$$\boldsymbol{a}_j^\top \boldsymbol{x}^* + b_j = 0,$$

and for each $\boldsymbol{x} \in \Omega$,

$$\boldsymbol{a}_j^\top \boldsymbol{x} + b_j \geq 0.$$

Hence, for each $j \in J(\boldsymbol{x}^*)$,

$$\boldsymbol{a}_j^\top(\boldsymbol{x} - \boldsymbol{x}^*) \geq 0.$$

Since $\mu_j^* \leq 0$, we get

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) - \sum_{j\in J(\boldsymbol{x}^*)} \mu_j^* \boldsymbol{a}_j^\top(\boldsymbol{x} - \boldsymbol{x}^*) \geq f(\boldsymbol{x}^*)$$

and the proof is completed.

22.14

a. Let $\Omega = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^\top \boldsymbol{x} \geq b\}$, $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \Omega$, and $\lambda \in [0, 1]$. Then, $\boldsymbol{a}^\top \boldsymbol{x}_1 \geq b$ and $\boldsymbol{a}^\top \boldsymbol{x}_2 \geq b$. Therefore,

$$\begin{aligned} \boldsymbol{a}^\top(\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) &= \lambda\boldsymbol{a}^\top\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{a}^\top\boldsymbol{x}_2 \\ &\geq \lambda b + (1-\lambda)b \\ &= b \end{aligned}$$

which means that $\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 \in \Omega$. Hence, Ω is a convex set.

b. Rewrite the problem as

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g(\boldsymbol{x}) \leq 0 \end{aligned}$$

where $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^2$ and $g(\boldsymbol{x}) = b - \boldsymbol{a}^\top\boldsymbol{x}$. Now, $\nabla g(\boldsymbol{x}) = -\boldsymbol{a} \neq \boldsymbol{0}$. Therefore, any feasible point is regular. By the Karush-Kuhn-Tucker theorem, there exists $\mu^* \geq 0$ such that

$$\begin{aligned} 2\boldsymbol{x}^* - \mu^*\boldsymbol{a} &= \boldsymbol{0} \\ \mu^*(b - \boldsymbol{a}^\top\boldsymbol{x}^*) &= 0. \end{aligned}$$

Since $\boldsymbol{x}^*$ is a feasible point, then $\boldsymbol{x}^* \neq \boldsymbol{0}$. Therefore, by the first equation, we see that $\mu^* \neq 0$. The second equation then implies that $b - \boldsymbol{a}^\top\boldsymbol{x}^* = 0$.

c. By the first Karush-Kuhn-Tucker equation, we have $\boldsymbol{x}^* = \mu^* \boldsymbol{a}/2$. Since $\boldsymbol{a}^\top \boldsymbol{x}^* = b$, then $\mu^* \boldsymbol{a}^\top \boldsymbol{a}/2 = \boldsymbol{a}^\top \boldsymbol{x}^* = b$, and therefore $\mu^* = 2b/\|\boldsymbol{a}\|^2$. Since $\boldsymbol{x}^* = \mu^* \boldsymbol{a}/2$ then $\boldsymbol{x}^*$ is uniquely given by $\boldsymbol{x}^* = b\boldsymbol{a}/\|\boldsymbol{a}\|^2$.

22.15

a. Let $f(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x}$ and $\Omega = \{\boldsymbol{x} : \boldsymbol{x} \geq \boldsymbol{0}\}$. Suppose $\boldsymbol{x}, \boldsymbol{y} \in \Omega$, and $\alpha \in (0, 1)$. Then, $\boldsymbol{x}, \boldsymbol{y} \geq \boldsymbol{0}$. Hence, $\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \geq \boldsymbol{0}$, which means $\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \in \Omega$. Furthermore,

$$\boldsymbol{c}^\top(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) = \alpha\boldsymbol{c}^\top\boldsymbol{x} + (1-\alpha)\boldsymbol{c}^\top\boldsymbol{y}.$$

Therefore, f is convex. Hence, the problem is a convex programming problem.

b. $\Rightarrow$: We use contraposition. Suppose $c_i < 0$ for some i. Let $\boldsymbol{d} = [0, \ldots, 1, \ldots, 0]$, where 1 appears in the ith component. Clearly $\boldsymbol{d}$ is a feasible direction for any point $\boldsymbol{x}^* \geq \boldsymbol{0}$. However, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) = \boldsymbol{d}^\top \boldsymbol{c} = c_i < 0$. Therefore, the FONC does not hold, and any point $\boldsymbol{x}^* \geq \boldsymbol{0}$ cannot be a minimizer.

$\Leftarrow$: Suppose $\boldsymbol{c} \geq \boldsymbol{0}$. Let $\boldsymbol{x}^* = \boldsymbol{0}$, and $\boldsymbol{d}$ a feasible direction at $\boldsymbol{x}^*$. Then, $\boldsymbol{d} \geq \boldsymbol{0}$. Hence, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) \geq \boldsymbol{0}$. Therefore, by Theorem 22.7, $\boldsymbol{x}^*$ is a solution.

The above also proves that if a solution exists, then $\boldsymbol{0}$ is a solution.

c. Write $\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{x}$ so that the constraint can be expressed as $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$.

$\Rightarrow$: We have $D\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{I}$, which has full rank. Therefore, any point is regular. Suppose a solution $\boldsymbol{x}^*$ exists. Then, by the KKT theorem, there exists $\boldsymbol{\mu}^* \geq \boldsymbol{0}$ such that $\boldsymbol{c}^\top - \boldsymbol{\mu}^{*\top} = \boldsymbol{0}^\top$ and $\boldsymbol{\mu}^{*\top}\boldsymbol{x}^* = 0$. Hence, $\boldsymbol{c} = \boldsymbol{\mu}^* \geq \boldsymbol{0}$.

$\Leftarrow$: Suppose $\boldsymbol{c} \geq \boldsymbol{0}$. Let $\boldsymbol{x}^* = \boldsymbol{0}$ and $\boldsymbol{\mu}^* = \boldsymbol{c}$. Then, $\boldsymbol{\mu}^* \geq \boldsymbol{0}$, $\boldsymbol{c}^\top - \boldsymbol{\mu}^{*\top} = \boldsymbol{0}^\top$, and $\boldsymbol{\mu}^{*\top}\boldsymbol{x}^* = 0$, i.e., the KKT condition is satisfied. By part a, $\boldsymbol{x}^*$ is a solution to the problem.

The above also proves that if a solution exists, then $\boldsymbol{0}$ is a solution.

22.16

a. The standard form problem is

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0}, \end{aligned}$$

which can be written as

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}, \\ & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \end{aligned}$$

where $f(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x}$, $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$, and $\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{x}$. Thus, we have $Df(\boldsymbol{x}) = \boldsymbol{c}^\top$, $D\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{A}$, and $D\boldsymbol{g}(\boldsymbol{x}) = -\boldsymbol{I}$. The Karush-Kuhn-Tucker conditions for the above problem has the form:

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \boldsymbol{0} \\ \boldsymbol{c}^\top + \boldsymbol{\lambda}^{*\top}\boldsymbol{A} - \boldsymbol{\mu}^{*\top} &= \boldsymbol{0}^\top \\ \boldsymbol{\mu}^{*\top}\boldsymbol{x}^* &= 0. \end{aligned}$$

b. The Karush-Kuhn-Tucker conditions are sufficient for optimality in this case because the problem is a convex optimization problem, i.e., the objective function is a convex function, and the feasible set is a convex set.

c. The dual problem is

$$\begin{aligned} &\text{maximize} \, \boldsymbol{\lambda}^\top \boldsymbol{b} \\ &\text{subject to} \, \boldsymbol{\lambda}^\top \boldsymbol{A} \leq \boldsymbol{c}^\top. \end{aligned}$$

c. Let

$$\boldsymbol{\mu}^* = \boldsymbol{c}^\top - \boldsymbol{\lambda}^{*\top}\boldsymbol{A}.$$

Since $\boldsymbol{\lambda}^*$ is feasible for the dual, we have $\boldsymbol{\mu}^* \geq \mathbf{0}$. Rewriting the above equation, we get

$$\boldsymbol{c}^\top - \boldsymbol{\lambda}^{*\top}\boldsymbol{A} - \boldsymbol{\mu}^* = \mathbf{0}.$$

The Complementary Slackness condition $(\boldsymbol{c}^\top - \boldsymbol{\lambda}^{*\top}\boldsymbol{A})\boldsymbol{x}^* = \mathbf{0}$ can be written as $\boldsymbol{\mu}^{*\top}\boldsymbol{x}^* = \mathbf{0}$. Therefore, the Karush-Kuhn-Tucker conditions hold. By part b, $\boldsymbol{x}^*$ is optimal.

22.17

a. We can treat $\boldsymbol{s}^{(1)}$ and $\boldsymbol{s}^{(2)}$ as vectors in $\mathbb{R}^n$. We have

$$S_a = \{\boldsymbol{s} : \boldsymbol{s} = x_1\boldsymbol{s}^{(1)} + x_2\boldsymbol{s}^{(2)}, x_1, x_2 \in \mathbb{R}, \quad s_i \geq a, i = 1\ldots, n\}.$$

Let $\boldsymbol{a} = [a, \ldots, a]^\top$. The optimization problem is:

$$\begin{array}{ll} \text{minimize} & \dfrac{1}{2}(x_1^2 + x_2^2) \\ \text{subject to} & x_1\boldsymbol{s}^{(1)} + x_2\boldsymbol{s}^{(2)} \geq \boldsymbol{a}. \end{array}$$

b. The KKT conditions are:

$$\begin{array}{rcl} x_1 - \boldsymbol{\mu}^\top\boldsymbol{s}^{(1)} & = & 0 \\ x_2 - \boldsymbol{\mu}^\top\boldsymbol{s}^{(2)} & = & 0 \\ \boldsymbol{\mu}^\top(x_1\boldsymbol{s}^{(1)} + x_2\boldsymbol{s}^{(2)} - \boldsymbol{a}) & = & \mathbf{0} \\ \boldsymbol{\mu} & \geq & \mathbf{0} \\ x_1\boldsymbol{s}^{(1)} + x_2\boldsymbol{s}^{(2)} & \geq & \boldsymbol{a}. \end{array}$$

c. Yes, because the Hessian of the Lagrangian is $\boldsymbol{I}$ (identity), which is positive definite.

d. Yes, this is a convex optimization problem. The objective function is quadratic, with identity Hessian (hence positive definite). The constraint set is of the form $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{a}$, and hence is a linear variety.

22.18

a. We first show that the set of probability vectors

$$\Omega = \{\boldsymbol{q} \in \mathbb{R}^n : q_1 + \cdots + q_n = 1,\ q_i > 0,\ i = 1, \ldots, n\}$$

is a convex set. Let $\boldsymbol{y}, \boldsymbol{z} \in \Omega$, so $y_1 + \cdots + y_n = 1$, $y_i > 0$, $z_1 + \cdots + z_n = 1$, and $z_i > 0$. Let $\alpha \in (0, 1)$ and $\boldsymbol{x} = \alpha\boldsymbol{y} + (1 - \alpha)\boldsymbol{z}$. We have

$$\begin{array}{rcl} x_1 + \cdots + x_n & = & \alpha y_1 + (1-\alpha)z_1 + \cdots + \alpha y_n + (1-\alpha)z_n \\ & = & \alpha(y_1 + \cdots + y_n) + (1-\alpha)(z_1 + \cdots + z_n) \\ & = & \alpha + (1 - \alpha) \\ & = & 1. \end{array}$$

Also, because $y_i > 0$, $z_i > 0$, $\alpha > 0$, and $1 - \alpha > 0$, we conclude that $x_i > 0$. Thus, $\boldsymbol{x} \in \Omega$, which shows that Ω is convex.

b. We next show that the function f is a convex function on Ω. For this, we compute

$$\boldsymbol{F}(\boldsymbol{q}) = \begin{bmatrix} \frac{p_1}{q_1^2} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{p_n}{q_n^2} \end{bmatrix},$$

which shows that $\boldsymbol{F}(\boldsymbol{q}) > 0$ for all $\boldsymbol{q}$ in the open set $\{\boldsymbol{q} : q_i > 0,\ i = 1, \ldots, n\}$, which contains Ω. Therefore, f is convex on Ω.

c. Fix a probability vector $\boldsymbol{p}$. Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & p_1 \log\left(\dfrac{p_1}{x_1}\right) + \cdots + p_n \log\left(\dfrac{p_n}{x_n}\right) \\ \text{subject to} & x_1 + \cdots + x_n = 1 \\ & x_i > 0,\ i = 1, \ldots, n \end{array}$$

By parts a and b, the problem is a convex optimization problem. We ignore the constraint $x_i > 0$ and write down the Lagrange conditions for the equality-constraint problem:

$$\begin{aligned} -\frac{p_i}{x_i^*} + \lambda^* &= 0,\ i = 1, \ldots, n \\ x_1^* + \cdots + x_n^* &= 1. \end{aligned}$$

Rewrite the first set of equations as $x_i^* = p_i/\lambda^*$. Combining this with the constraint and the fact that $p_1 + \cdots + p_n = 1$, we obtain $\lambda^* = 1$, which means that $x_i^* = p_i$. Therefore, the unique global minimizer is $\boldsymbol{x}^* = \boldsymbol{p}$.

Note that $f(\boldsymbol{x}^*) = 0$. Hence, we conclude that $f(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x} \in \Omega$. Moreover, $f(\boldsymbol{x}) = 0$ if and only if $\boldsymbol{x}^* = 0$. This proves the required result.

d. Given two probability vectors $\boldsymbol{p}$ and $\boldsymbol{q}$, the number

$$D(\boldsymbol{p}, \boldsymbol{q}) = p_1 \log\left(\frac{p_1}{q_1}\right) + \cdots + p_n \log\left(\frac{p_n}{q_n}\right)$$

is called the *relative entropy* (or *Kullback-Liebler divergence*) between $\boldsymbol{p}$ and $\boldsymbol{q}$. It is used in information theory to measure the "distance" between two probability vectors. The result of part c justifies the use of D as a measure of "distance" (although D is not a *metric* because it is not symmetric).

22.19

We claim that a solution exists, and that it is unique. To prove the first claim, choose $\varepsilon > 0$ such that there exists $\boldsymbol{x} \in \Omega$ satisfying $\|\boldsymbol{x} - \boldsymbol{z}\| < \varepsilon$. Consider the modified problem

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{x} - \boldsymbol{z}\| \\ \text{subject to} & \boldsymbol{x} \in \Omega \bigcap \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{z}\| \leq \varepsilon\}. \end{array}$$

If this modified problem has a solution, then clearly so does the original problem. The objective function here is continuous, and the constraint set is closed and bounded. Hence, by Weierstrass's Theorem, a solution to the problem exists.

Let f be the objective function. Next, we show that f is convex (and hence the problem is a convex optimization problem). Let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0, 1)$. Then,

$$\begin{aligned} f(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) &= \|\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} - \boldsymbol{z}\| \\ &= \|\alpha(\boldsymbol{x} - \boldsymbol{z}) + (1-\alpha)(\boldsymbol{y} - \boldsymbol{z})\| \\ &\leq \alpha\|\boldsymbol{x} - \boldsymbol{z}\| + (1-\alpha)\|\boldsymbol{y} - \boldsymbol{z}\| \\ &= \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{y}), \end{aligned}$$

which shows that f is convex.

To prove uniqueness, let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be solutions to the problem. Then, by convexity, $\boldsymbol{x}_3 = (\boldsymbol{x}_1 + \boldsymbol{x}_2)/2$ is also a solution. But

$$\begin{aligned} \|\boldsymbol{x}_3 - \boldsymbol{z}\| &= \left\| \frac{\boldsymbol{x}_1 + \boldsymbol{x}_2}{2} - \boldsymbol{z} \right\| \\ &= \left\| \frac{\boldsymbol{x}_1 - \boldsymbol{z}}{2} + \frac{\boldsymbol{x}_2 - \boldsymbol{z}}{2} \right\| \\ &\leq \frac{1}{2}(\|\boldsymbol{x}_1 - \boldsymbol{z}\| + \|\boldsymbol{x}_2 - \boldsymbol{z}\|) \\ &= \|\boldsymbol{x}_3 - \boldsymbol{z}\|, \end{aligned}$$

from which we conclude that the triangle inequality above holds with equality, implying that $\boldsymbol{x}_1 - \boldsymbol{z} = \alpha(\boldsymbol{x}_2 - \boldsymbol{z})$ for some $\alpha \geq 0$. Because $\|\boldsymbol{x}_1 - \boldsymbol{z}\| = \|\boldsymbol{x}_2 - \boldsymbol{z}\| = \alpha\|\boldsymbol{x}_1 - \boldsymbol{z}\|$, we have $\alpha = 1$. From this, we obtain $\boldsymbol{x}_1 = \boldsymbol{x}_2$, which proves uniqueness.

22.20

a. Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{n\times}$ be symmetric and $\boldsymbol{A} \geq 0$, $\boldsymbol{B} \geq 0$. Fix $\alpha \in (0,1)$, $\boldsymbol{x} \in \mathbb{R}^n$, and let $\boldsymbol{C} = \alpha\boldsymbol{A} + (1-\alpha)\boldsymbol{B}$. Then,

$$\begin{aligned} \boldsymbol{x}^\top \boldsymbol{C}\boldsymbol{x} &= \boldsymbol{x}^\top[\alpha\boldsymbol{A} + (1-\alpha)\boldsymbol{B}]\boldsymbol{x} \\ &= \alpha\boldsymbol{x}^\top\boldsymbol{A}\boldsymbol{x} + (1-\alpha)\boldsymbol{x}^\top\boldsymbol{B}\boldsymbol{x}. \end{aligned}$$

Since $\boldsymbol{x}^\top\boldsymbol{A}\boldsymbol{x} \geq 0$, $\boldsymbol{x}^\top\boldsymbol{B}\boldsymbol{x} \geq 0$, and $\alpha, (1-\alpha) > 0$ by assumption, then $\boldsymbol{x}^\top\boldsymbol{C}\boldsymbol{x} \geq 0$, which proves the required result.

b. We first show that the constraint set $\Omega = \{\boldsymbol{x} : \boldsymbol{F}_0 + \sum_{j=1}^n x_j\boldsymbol{F}_j \geq 0\}$ is convex. So, let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0,1)$. Let $\boldsymbol{z} = \alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}$. Then,

$$\begin{aligned} \boldsymbol{F}_0 + \sum_{j=1}^n z_j\boldsymbol{F}_j &= \boldsymbol{F}_0 + \sum_{j=1}^n [\alpha x_j + (1-\alpha)y_j]\boldsymbol{F}_j \\ &= \boldsymbol{F}_0 + \alpha\sum_{j=1}^n x_j\boldsymbol{F}_j + (1-\alpha)\sum_{j=1}^n y_j\boldsymbol{F}_j \\ &= \alpha[\boldsymbol{F}_0 + \sum_{j=1}^n x_j\boldsymbol{F}_j] + (1-\alpha)[\boldsymbol{F}_0 + \sum_{j=1}^n y_j\boldsymbol{F}_j]. \end{aligned}$$

By assumption, we have

$$\begin{aligned} \boldsymbol{F}_0 + \sum_{j=1}^n x_j\boldsymbol{F}_j &\geq 0 \\ \boldsymbol{F}_0 + \sum_{j=1}^n y_j\boldsymbol{F}_j &\geq 0. \end{aligned}$$

By part c, we conclude that

$$\boldsymbol{F}_0 + \sum_{j=1}^n z_j\boldsymbol{F}_j \geq 0,$$

which implies that $\boldsymbol{z} \in \Omega$.

To show that the objective function $f(\boldsymbol{x}) = \boldsymbol{c}^\top\boldsymbol{x}$ is convex on Ω, let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0,1)$. Then,

$$\begin{aligned} f(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) &= \boldsymbol{c}^\top(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \\ &= \alpha\boldsymbol{c}^\top\boldsymbol{x} + (1-\alpha)\boldsymbol{c}^\top\boldsymbol{y} \\ &= \alpha f(\boldsymbol{x}) + (1-\alpha)f(\boldsymbol{y}) \end{aligned}$$

which shows that f is convex.

c. The objective function is already in the required form. To rewrite the constraint, let $a_{i,j}$ be the (i,j)th entry of $\boldsymbol{A}$, $i = 1,\ldots,m$, $j = 1,\ldots,n$. Then, the constraint $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ can be written as

$$a_{i,1}x_1 + a_{i,2}x_2 + \cdots + a_{i,n}x_n \geq b_i, \qquad i = 1,\ldots,m$$

Now form the diagonal matrices

$$\begin{aligned} \boldsymbol{F}_0 &= \text{diag}\{-b_1,\ldots,-b_m\} \\ \boldsymbol{F}_j &= \text{diag}\{a_{1,j},\ldots,a_{m,j}\}, \qquad j = 1,\ldots,n. \end{aligned}$$

Note that a diagonal matrix is positive semidefinite if and only if every diagonal element is nonnegative. Hence, the constraint $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ can be written as $\boldsymbol{F}_0 + \sum_{j=1}^n x_j \boldsymbol{F}_j \geq 0$. The left hand side is a diagonal matrix, and the ith diagonal element is simply $-b_i + a_{i,1}x_1 + a_{i,2}x_2 + \cdots + a_{i,n}x_n$.

22.21

a. We have

$$\Omega = \{\boldsymbol{x} : x_1 + \ldots + x_n = 1;\ x_1, \ldots, x_n > 0;\ x_1 \geq 2x_i,\ i = 2, \ldots, n\}.$$

So let $\boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0, 1)$. Consider $\boldsymbol{z} = \alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}$. We have

$$\begin{aligned} z_1 + \cdots + z_n &= \alpha x_1 + (1-\alpha)y_1 + \cdots + \alpha x_n + (1-\alpha)y_n \\ &= \alpha(x_1 + \cdots + x_n) + (1-\alpha)(y_1 + \cdots + y_n) \\ &= \alpha + 1 - \alpha \\ &= 1. \end{aligned}$$

Moreover, for each i, because $x_i > 0$, $y_i > 0$, $\alpha > 0$ and $1 - \alpha > 0$, we have $z_i > 0$. Finally, for each i,

$$z_1 = \alpha x_1 + (1-\alpha)y_1 \geq \alpha 2x_i + (1-\alpha)2y_i = 2z_i.$$

Hence, $\boldsymbol{z} \in \Omega$, which implies that Ω is convex.

b. We first show that the negative of the objective function is convex. For this, we will compute its Hessian, which turns out to be a diagonal matrix with ith diagonal entry $1/x_i^2$, which is strictly positive. Hence, the Hessian is positive definite, which implies that the negative of the objective function is convex.

Combining the above with part a, we conclude that the problem is a convex optimization problem. Hence, the FONC (for set constraints) is necessary and sufficient. Let $\boldsymbol{x}$ be a given allocation. The FONC at $\boldsymbol{x}$ is $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}) \geq 0$ for all feasible directions $\boldsymbol{d}$ at $\boldsymbol{x}$. But because Ω is convex, the FONC can be written as $(\boldsymbol{y} - \boldsymbol{x})^\top \nabla f(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{y} \in \Omega$. Computing $\nabla f(\boldsymbol{x})$ for $f(\boldsymbol{x}) = -\sum_{i=1}^n \log(x_i)$, we get the proportional fairness condition.

22.22

a. We rewrite the problem into a minimization problem by multiplying the objective function by -1. Thus, the new objective function is the sum of the functions $-U_i$. Because each U_i is concave, $-U_i$ is convex, and hence their sum is convex.

To show that the constraint set $\Omega = \{\boldsymbol{x} : \boldsymbol{e}^\top \boldsymbol{x} \leq C\}$ (where $\boldsymbol{e} = [1, \ldots, 1]^\top$) is convex, let $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \Omega$, and $\lambda \in [0, 1]$. Then, $\boldsymbol{e}^\top \boldsymbol{x}_1 \leq C$ and $\boldsymbol{e}^\top \boldsymbol{x}_2 \leq C$. Therefore,

$$\begin{aligned} \boldsymbol{e}^\top(\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) &= \lambda\boldsymbol{e}^\top\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{e}^\top\boldsymbol{x}_2 \\ &\leq \lambda C + (1-\lambda)C \\ &= C \end{aligned}$$

which means that $\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 \in \Omega$. Hence, Ω is a convex set.

b. Because the problem is a convex optimization problem, the following KKT condition is necessary and sufficient for $\boldsymbol{x}^*$ to be a global minimizers:

$$\begin{aligned} \mu^* &\geq 0 \\ -U_i'(x_i^*) + \mu^* &= 0, \qquad i = 1, \ldots, n \\ \mu^* \left(\sum_{i=1}^n x_i^* - C \right) &= 0 \\ \sum_{i=1}^n x_i^* &\leq C. \end{aligned}$$

Note that because $-U_i(x_i) + \mu^* x_i$ is a convex function of x_i, the second line above can be written as $x_i^* = \arg\max_x (U_i(x) - \mu^* x)$.

c. Because each U_i is concave and increasing, we conclude that $\sum_{i=1}^n x_i^* = C$; for otherwise we could increase some x_i^* and hence $U_i(x_i^*)$ and also $\sum_{i=1}^n U_i(x_i^*)$, contradicting the optimality of $\boldsymbol{x}^*$.

22.23

First note that the optimization problem that you construct cannot be a convex problem (for otherwise, the FONC implies that $\boldsymbol{x}^*$ is a global minimizer, which then implies that the SONC holds). Let $f(\boldsymbol{x}) = x_2$, $g(\boldsymbol{x}) = -(x_2 + x_1^2)$, and $\boldsymbol{x}^* = \boldsymbol{0}$. Then, $\nabla f(\boldsymbol{x}^*) = [0,1]^\top$. Any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^*$ is of the form $\boldsymbol{d} = [d_1, d_2]^\top$ with $d_2 \geq 0$. Hence, $\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) \geq 0$, which shows that the FONC holds.

Because $\nabla g(\boldsymbol{x}^*) = -[0,1]$ (so $\boldsymbol{x}^*$ is regular), we see that if $\mu^* = 1$, then $\nabla f(\boldsymbol{x}^*) + \mu^* \nabla g(\boldsymbol{x}^*) = 0$, and so the KKT condition holds.

Because $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{O}$, the SONC for set constraint Ω holds. However,

$$\boldsymbol{L}(\boldsymbol{x}^*, \mu^*) = \boldsymbol{O} + \mu^* \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix} < 0$$

and $T(\boldsymbol{x}^*) = \{\boldsymbol{y} : y_2 = 0\}$, which shows that the SONC for inequality constraint $g(\boldsymbol{x}) \leq 0$ does not hold.

22.24

a. Let $\boldsymbol{x}_0$ and $\boldsymbol{\mu}_0$ be feasible points in the primal and dual, respectively. Then, $\boldsymbol{g}(\boldsymbol{x}_0) \leq \boldsymbol{0}$ and $\boldsymbol{\mu}_0 \geq \boldsymbol{0}$, and so $\boldsymbol{\mu}_0^\top \boldsymbol{g}(\boldsymbol{x}_0) \leq 0$. Hence,

$$\begin{aligned} f(\boldsymbol{x}_0) &\geq f(\boldsymbol{x}_0) + \boldsymbol{\mu}_0^\top \boldsymbol{g}(\boldsymbol{x}_0) \\ &= l(\boldsymbol{x}_0, \boldsymbol{\mu}_0) \\ &\geq \min_{\boldsymbol{x} \in \mathbb{R}^n} l(\boldsymbol{x}, \boldsymbol{\mu}_0) \\ &= q(\boldsymbol{\mu}_0). \end{aligned}$$

b. Suppose $f(\boldsymbol{x}_0) = q(\boldsymbol{\mu}_0)$ for feasible points $\boldsymbol{x}_0$ and $\boldsymbol{\mu}_0$. Let $\boldsymbol{x}$ be any feasible point in the primal. Then, by part a, $f(\boldsymbol{x}) \geq q(\boldsymbol{\mu}_0) = f(\boldsymbol{x}_0)$. Hence $\boldsymbol{x}_0$ is optimal in the primal.

Similarly, let $\boldsymbol{\mu}$ be any feasible point in the dual. Then, by part a, $q(\boldsymbol{\mu}) \leq f(\boldsymbol{x}_0) = q(\boldsymbol{\mu}_0)$. Hence $\boldsymbol{\mu}_0$ is optimal in the dual.

c. Let $\boldsymbol{x}^*$ be optimal in the primal. Then, by the KKT Theorem, there exists $\boldsymbol{\mu}^* \in \mathbb{R}^m$ such that

$$\begin{aligned} \nabla_x l(\boldsymbol{x}^*, \boldsymbol{\mu}^*) = (Df(\boldsymbol{x}^*) + \boldsymbol{\mu}^{*\top} D\boldsymbol{g}(\boldsymbol{x}^*))^\top &= \boldsymbol{0} \\ \boldsymbol{\mu}^{*\top} \boldsymbol{g}(\boldsymbol{x}^*) &= \boldsymbol{0} \\ \boldsymbol{\mu}^* &\geq \boldsymbol{0}. \end{aligned}$$

Therefore, $\boldsymbol{\mu}^*$ is feasible in the dual. Further, we note that $l(\cdot, \boldsymbol{\mu}^*)$ is a convex function (because f is convex, $\boldsymbol{\mu}^{*\top} \boldsymbol{g}$ is convex being the sum of convex functions, and hence l is the sum of two convex functions). Hence, we have $l(\boldsymbol{x}^*, \boldsymbol{\mu}^*) = \min_{\boldsymbol{x} \in \mathbb{R}^n} l(\boldsymbol{x}, \boldsymbol{\mu}^*)$. Therefore,

$$\begin{aligned} q(\boldsymbol{\mu}^*) &= \min_{\boldsymbol{x} \in \mathbb{R}^n} l(\boldsymbol{x}, \boldsymbol{\mu}^*) \\ &= l(\boldsymbol{x}^*, \boldsymbol{\mu}^*) \\ &= f(\boldsymbol{x}^*) + \boldsymbol{\mu}^{*\top} \boldsymbol{g}(\boldsymbol{x}^*) \\ &= f(\boldsymbol{x}^*). \end{aligned}$$

By part b, $\boldsymbol{\mu}^*$ is optimal in the dual.

22.25

a. The Schur complement of $\boldsymbol{M}(1,1)$ is

$$\begin{aligned} \Delta_{11} &= \boldsymbol{M}(2\colon !3, 2\colon !3) - \boldsymbol{M}(2\colon 3, 1)\boldsymbol{M}(1,1)^{-1}\boldsymbol{M}(1, 2\colon !3) \\ &= \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} - \begin{bmatrix} \gamma \\ -1 \end{bmatrix} \begin{bmatrix} \gamma & -1 \end{bmatrix} \\ &= \begin{bmatrix} 1 - \gamma^2 & 2 + \gamma \\ 2 + \gamma & 4 \end{bmatrix}. \end{aligned}$$

b. The Schur complement of $\boldsymbol{M}(2\colon !3, 2\colon !3)$ is

$$\begin{aligned}
\Delta_{22} &= \boldsymbol{M}(1,1) - \boldsymbol{M}(1,2\colon !3)\boldsymbol{M}(2\colon !3, 2\colon 3)^{-1}\boldsymbol{M}(2\colon !3, 1) \\
&= 1 - \begin{bmatrix}\gamma & -1\end{bmatrix}\begin{bmatrix}1 & 2\\ 2 & 5\end{bmatrix}^{-1}\begin{bmatrix}\gamma\\ -1\end{bmatrix} \\
&= 1 - \begin{bmatrix}\gamma & -1\end{bmatrix}\begin{bmatrix}5 & -2\\ -2 & 1\end{bmatrix}\begin{bmatrix}\gamma\\ -1\end{bmatrix} \\
&= -\gamma(5\gamma + 4).
\end{aligned}$$

22.26

Let

$$\boldsymbol{P} = \begin{bmatrix}x_1 & x_2\\ x_2 & x_3\end{bmatrix}$$

and let

$$\boldsymbol{P}_1 = \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}, \quad \boldsymbol{P}_2 = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}, \quad \text{and} \quad \boldsymbol{P}_3 = \begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}.$$

Then, we can represent the Lyapunov inequality $\boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} < 0$ as

$$\begin{aligned}
\boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} &= x_1\left(\boldsymbol{A}^\top\boldsymbol{P}_1 + \boldsymbol{P}_1\boldsymbol{A}\right) + x_2\left(\boldsymbol{A}^\top\boldsymbol{P}_2 + \boldsymbol{P}_2\boldsymbol{A}\right) \\
&\quad + x_3\left(\boldsymbol{A}^\top\boldsymbol{P}_3 + \boldsymbol{P}_3\boldsymbol{A}\right) \\
&= -x_1\boldsymbol{F}_1 - x_2\boldsymbol{F}_2 - x_3\boldsymbol{F}_3 \\
&< 0,
\end{aligned}$$

where

$$\boldsymbol{F}_i = -\boldsymbol{A}^\top\boldsymbol{P}_i - \boldsymbol{P}_i\boldsymbol{A}, \quad i = 1, 2, 3.$$

Equivalently,

$$\boldsymbol{P} = \boldsymbol{P}^\top \quad \text{and } \boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} < 0$$

if and only if

$$\boldsymbol{F}(\boldsymbol{x}) = x_1\boldsymbol{F}_1 + x_2\boldsymbol{F}_2 + x_3\boldsymbol{F}_3 > 0.$$

22.27

The quadratic inequality

$$\boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} + \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^\top\boldsymbol{P} < 0$$

can be equivalently represented by the following LMI:

$$\begin{bmatrix}-\boldsymbol{R} & \boldsymbol{B}^\top\boldsymbol{P}\\ \boldsymbol{P}\boldsymbol{B} & \boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}\end{bmatrix} < 0,$$

or as the following LMI:

$$\begin{bmatrix}\boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} & \boldsymbol{P}\boldsymbol{B}\\ \boldsymbol{B}^\top\boldsymbol{P} & -\boldsymbol{R}\end{bmatrix} < 0.$$

It is easy to verify, using Schur complements, that the above two LMIs are equivalent to the following quadratic inequality:

$$\begin{bmatrix}\boldsymbol{A}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} + \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^\top\boldsymbol{P} & \boldsymbol{O}\\ \boldsymbol{O} & -\boldsymbol{R}\end{bmatrix} < 0.$$

22.28

The MATLAB code is as follows:

```
A =[-0.9501   -0.4860   -0.4565
   -0.2311   -0.8913   -0.0185
   -0.6068   -0.7621   -0.8214];
setlmis([]);
P=lmivar(1,[3 1]);
lmiterm([1 1 1 P],1,A,'s')
lmiterm([2 1 1 0],0.1)
lmiterm([-2 1 1 P],1,1)
lmiterm([3 1 1 P],1,1)
lmiterm([-3 1 1 0],1)
lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
P=dec2mat(lmis,xfeas,P)
```

23. Algorithms for Constrained Optimization

23.1

a. By drawing a simple picture, it is easy to see that $\mathbf{\Pi}[\boldsymbol{x}] = \boldsymbol{x}/\|\boldsymbol{x}\|$, provided $\boldsymbol{x} \neq \mathbf{0}$.

b. By inspection, we see that the solutions are $[0,1]^\top$ and $[0,-1]^\top$. (Or use Rayleigh's inequality.)

c. Now,

$$\boldsymbol{x}^{(k+1)} = \mathbf{\Pi}[\boldsymbol{x}^{(k)} + \alpha \nabla f(\boldsymbol{x}^{(k)})] = \beta_k(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{Q}\boldsymbol{x}^{(k)}) = \beta_k(\boldsymbol{I} + \alpha \boldsymbol{Q})\boldsymbol{x}^{(k)},$$

where $\beta_k = 1/\|(\boldsymbol{I} + \alpha \boldsymbol{Q})\boldsymbol{x}^{(k)}\|$. For the particular given form of $\boldsymbol{Q}$, we have

$$\begin{aligned} x_1^{(k+1)} &= \beta_k(1+\alpha)x_1^{(k)} \\ x_2^{(k+1)} &= \beta_k(1+2\alpha)x_2^{(k)}. \end{aligned}$$

Hence,

$$y^{(k+1)} = \left(\frac{1+\alpha}{1+2\alpha}\right) y^{(k)}.$$

d. Assuming $x_2^{(0)} \neq 0$, $y^{(0)}$ is well defined. Hence, by part c, we can write

$$y^{(k)} = \left(\frac{1+\alpha}{1+2\alpha}\right)^k y^{(0)}.$$

Because $\alpha > 0$,

$$\frac{1+\alpha}{1+2\alpha} < 1,$$

which implies that $y^{(k)} \to 0$. But

$$\begin{aligned} 1 &= \|\boldsymbol{x}^{(k)}\| \\ &= \sqrt{(x_1^{(k)})^2 + (x_2^{(k)})^2} \\ &= \sqrt{(x_2^{(k)})^2 \left(\frac{(x_1^{(k)})^2}{(x_2^{(k)})^2} + 1\right)} \\ &= |x_2^{(k)}| \sqrt{(y^{(k)})^2 + 1}, \end{aligned}$$

which implies that

$$|x_2^{(k)}| = \frac{1}{\sqrt{(y^{(k)})^2 + 1}}.$$

Because $y^{(k)} \to 0$, we have $|x_2^{(k)}| \to 1$. By the expression for $x_2^{(k+1)}$ in part c, we see that the sign of $x_2^{(k)}$ does not change with k. Hence, we deduce that either $x_2^{(k)} \to 1$ or $x_2^{(k)} \to -1$. This also implies that $x_1^{(k)} \to 0$. Hence, $\boldsymbol{x}^{(k)}$ converges to a solution to the problem.

e. If $x_2^{(0)} = 0$, then $x_2^{(k)} = 0$ for all k, which means that $x_1^{(k)} = 1$ or -1 for all k. In this case, the algorithm is stuck at the initial condition $[1, 0]^\top$ or $[-1, 0]^\top$ (which are in fact the minimizers).

23.2

a. Yes. To show: Suppose that $\boldsymbol{x}^{(k)}$ is a global minimizer of the given problem. Then, for all $\boldsymbol{x} \in \Omega$, $\boldsymbol{x} \neq \boldsymbol{x}^{(k)}$, we have $\boldsymbol{c}^\top \boldsymbol{x} \geq \boldsymbol{c}^\top \boldsymbol{x}^{(k)}$. Rewriting, we obtain $\boldsymbol{c}^\top(\boldsymbol{x} - \boldsymbol{x}^{(k)}) \geq 0$. Recall that

$$\begin{aligned} \boldsymbol{\Pi}[\boldsymbol{x}^{(k)} - \nabla f(\boldsymbol{x}^{(k)})] &= \arg\min_{\boldsymbol{x} \in \Omega} \|\boldsymbol{x} - (\boldsymbol{x}^{(k)} - \nabla f(\boldsymbol{x}^{(k)}))\|^2 \\ &= \arg\min_{\boldsymbol{x} \in \Omega} \|\boldsymbol{x} - \boldsymbol{x}^{(k)} + \boldsymbol{c}\|^2. \end{aligned}$$

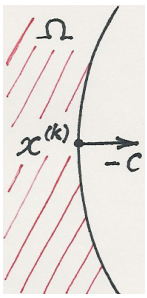
But, for any $\boldsymbol{x} \in \Omega$, $\boldsymbol{x} \neq \boldsymbol{x}^{(k)}$,

$$\begin{aligned} \|\boldsymbol{x} - \boldsymbol{x}^{(k)} + \boldsymbol{c}\|^2 &= \|\boldsymbol{x} - \boldsymbol{x}^{(k)}\|^2 + \|\boldsymbol{c}\|^2 + 2\boldsymbol{c}^\top(\boldsymbol{x} - \boldsymbol{x}^{(k)}) \\ &> \|\boldsymbol{c}\|^2, \end{aligned}$$

where we used the facts that $\|\boldsymbol{x} - \boldsymbol{x}^{(k)}\|^2 > 0$ and $\boldsymbol{c}^\top(\boldsymbol{x} - \boldsymbol{x}^{(k)}) \geq 0$. On the other hand, $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k)} + \boldsymbol{c}\|^2 = \|\boldsymbol{c}\|^2$. Hence,

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{\Pi}[\boldsymbol{x}^{(k)} - \nabla f(\boldsymbol{x}^{(k)})] = \boldsymbol{x}^{(k)}.$$

b. No. Counterexample:



23.3

a. Suppose $\boldsymbol{x}^{(k)}$ satisfies the FONC. Then, $\nabla f(\boldsymbol{x}^{(k)}) = \boldsymbol{0}$. Hence, $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)}$. Conversely, suppose $\boldsymbol{x}^{(k)}$ does not satisfy the FONC. Then, $\nabla f(\boldsymbol{x}^{(k)}) \neq \boldsymbol{0}$. Hence, $\alpha_k > 0$, and so $\boldsymbol{x}^{(k+1)} \neq \boldsymbol{x}^{(k)}$.

b. Case (i): Suppose $\boldsymbol{x}^{(k)}$ is a corner point. Without loss of generality, take $\boldsymbol{x}^{(k)} = [1, 1]^\top$. (We can do this because any other corner point can be mapped to this point by changing variables x_i to $-x_i$ as appropriate.) Note that any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^{(k)} = [1, 1]^\top$ satisfies $\boldsymbol{d} \leq \boldsymbol{0}$. Therefore,

$$\begin{aligned} \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} &\Leftrightarrow -\nabla f(\boldsymbol{x}^{(k)}) \geq \boldsymbol{0} \\ &\Leftrightarrow \boldsymbol{d}^\top \nabla f(\boldsymbol{x}^{(k)}) \geq 0 \text{ for all feasible } \boldsymbol{d} \text{ at } \boldsymbol{x}^{(k)} \\ &\Leftrightarrow \boldsymbol{x}^{(k)} \text{ satisfies FONC.} \end{aligned}$$

Case (ii): Suppose $\boldsymbol{x}^{(k)}$ is not a corner point (i.e., is an edge point). Without loss of generality, take $\boldsymbol{x}^{(k)} \in \{\boldsymbol{x} : x_1 = 1, -1 < x_2 < 1\}$. (We can do this because any other edge point can be mapped to this point by changing variables x_i to $-x_i$ as appropriate.) Note that any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^{(k)} \in \{\boldsymbol{x} : x_1 = 1, -1 < x_2 < 1\}$ satisfies $d_1 \leq 0$. Therefore,

$$\begin{aligned} \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} &\Leftrightarrow -\nabla f(\boldsymbol{x}^{(k)}) = [a, 0]^\top, \ a > 0 \\ &\Leftrightarrow \boldsymbol{d}^\top \nabla f(\boldsymbol{x}^{(k)}) \geq 0 \text{ for all feasible } \boldsymbol{d} \text{ at } \boldsymbol{x}^{(k)} \\ &\Leftrightarrow \boldsymbol{x}^{(k)} \text{ satisfies FONC.} \end{aligned}$$

23.4

By definition of $\boldsymbol{\Pi}$, we have

$$\begin{aligned}\boldsymbol{\Pi}[\boldsymbol{x}_0 + \boldsymbol{y}] &= \underset{\boldsymbol{x}\in\Omega}{\arg\min}\, \|\boldsymbol{x} - (\boldsymbol{x}_0 + \boldsymbol{y})\| \\ &= \underset{\boldsymbol{x}\in\Omega}{\arg\min}\, \|(\boldsymbol{x} - \boldsymbol{x}_0) - \boldsymbol{y}\|.\end{aligned}$$

By Exercise 6.7, we can write

$$\underset{\boldsymbol{x}\in\Omega}{\arg\min}\, \|(\boldsymbol{x} - \boldsymbol{x}_0) - \boldsymbol{y}\| = \boldsymbol{x}_0 + \underset{\boldsymbol{z}\in\mathcal{N}(\boldsymbol{A})}{\arg\min}\, \|\boldsymbol{z} - \boldsymbol{y}\|.$$

The term $\arg\min_{\boldsymbol{z}\in\mathcal{N}(\boldsymbol{A})} \|\boldsymbol{z} - \boldsymbol{y}\|$ is simply the orthogonal projection of $\boldsymbol{y}$ onto $\mathcal{N}(\boldsymbol{A})$. By Exercise 6.7, we have

$$\underset{\boldsymbol{z}\in\mathcal{N}(\boldsymbol{A})}{\arg\min}\, \|\boldsymbol{z} - \boldsymbol{y}\| = \boldsymbol{P}\boldsymbol{y},$$

where $\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}$. Hence,

$$\boldsymbol{\Pi}[\boldsymbol{x}_0 + \boldsymbol{y}] = \boldsymbol{x}_0 + \boldsymbol{P}\boldsymbol{y}.$$

23.5

Since $\alpha_k \geq 0$ is a minimizer of $\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} - \alpha\boldsymbol{P}\boldsymbol{g}^{(k)})$, we apply the FONC to $\phi_k(\alpha)$ to obtain

$$\phi_k'(\alpha) = (\boldsymbol{x}^{(k)} - \alpha\boldsymbol{P}\boldsymbol{g}^{(k)})^\top\boldsymbol{Q}(-\boldsymbol{P}\boldsymbol{g}^{(k)}) - \boldsymbol{b}^\top(-\boldsymbol{P}\boldsymbol{g}^{(k)}).$$

Therefore, $\phi_k'(\alpha) = 0$ if $\alpha\boldsymbol{g}^{(k)\top}\boldsymbol{P}\boldsymbol{Q}\boldsymbol{P}\boldsymbol{g}^{(k)} = (\boldsymbol{x}^{(k)\top}\boldsymbol{Q} - \boldsymbol{b}^\top)\boldsymbol{P}\boldsymbol{g}^{(k)}$. But

$$\boldsymbol{x}^{(k)\top}\boldsymbol{Q} - \boldsymbol{b}^\top = \boldsymbol{g}^{(k)\top}.$$

Hence

$$\alpha_k = \frac{\boldsymbol{g}^{(k)\top}\boldsymbol{P}\boldsymbol{g}^{(k)}}{\boldsymbol{g}^{(k)\top}\boldsymbol{P}\boldsymbol{Q}\boldsymbol{P}\boldsymbol{g}^{(k)}}.$$

23.6

By Exercise 23.5, the projected steepest descent algorithm applied to this problem takes the form

$$\begin{aligned}\boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \boldsymbol{P}\boldsymbol{x}^{(k)} \\ &= (\boldsymbol{I}_n - \boldsymbol{P})\boldsymbol{x}^{(k)} \\ &= \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{x}^{(k)}.\end{aligned}$$

If $\boldsymbol{x}^{(0)} \in \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$, then $\boldsymbol{A}\boldsymbol{x}^{(0)} = \boldsymbol{b}$, and hence

$$\boldsymbol{x}^{(1)} = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}\boldsymbol{b}$$

which solves the problem (see Section 12.3).

23.7

a. Define

$$\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} - \alpha\boldsymbol{P}\nabla f(\boldsymbol{x}^{(k)}))$$

By the Chain Rule,

$$\begin{aligned}\phi_k'(\alpha_k) &= \frac{d\phi_k}{d\alpha}(\alpha_k) \\ &= -(\nabla f(\boldsymbol{x}^{(k)} - \alpha_k\boldsymbol{P}\nabla f(\boldsymbol{x}^{(k)})))^\top\boldsymbol{P}\nabla f(\boldsymbol{x}^{(k)}) \\ &= -(\nabla f(\boldsymbol{x}^{(k+1)}))^\top\boldsymbol{P}\nabla f(\boldsymbol{x}^{(k)}).\end{aligned}$$

Since α_k minimizes ϕ_k, $\phi_k'(\alpha_k) = 0$, and thus $\boldsymbol{g}^{(k+1)\top}\boldsymbol{P}\boldsymbol{g}^{(k)} = 0$.

b. We have $\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)} = -\alpha_k \boldsymbol{P}\boldsymbol{g}^{(k)}$ and $\boldsymbol{x}^{(k+2)} - \boldsymbol{x}^{(k+1)} = -\alpha_{k+1}\boldsymbol{P}\boldsymbol{g}^{(k+1)}$. Therefore,

$$\begin{aligned}(\boldsymbol{x}^{(k+2)} - \boldsymbol{x}^{(k+1)})^\top(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}) &= \alpha_{k+1}\alpha_k \boldsymbol{g}^{(k+1)\top}\boldsymbol{P}^\top\boldsymbol{P}\boldsymbol{g}^{(k)} \\ &= \alpha_{k+1}\alpha_k \boldsymbol{g}^{(k+1)\top}\boldsymbol{P}\boldsymbol{g}^{(k)} \\ &= 0\end{aligned}$$

by part a, and the fact that $\boldsymbol{P} = \boldsymbol{P}^\top = \boldsymbol{P}^2$.

23.8

a. minimize $f(\boldsymbol{x}) + \gamma P(\boldsymbol{x})$.

b. Suppose $\boldsymbol{x}^\gamma \notin \Omega$. Then, $P(\boldsymbol{x}^\gamma) > 0$ by definition of P. Because $\boldsymbol{x}^\gamma$ is a global minimizer of the unconstrained problem, we have

$$\begin{aligned}f(\boldsymbol{x}^\gamma) + \gamma P(\boldsymbol{x}^\gamma) &\leq f(\boldsymbol{x}^*) + \gamma P(\boldsymbol{x}^*) \\ &= f(\boldsymbol{x}^*),\end{aligned}$$

which implies that

$$f(\boldsymbol{x}^\gamma) \leq f(\boldsymbol{x}^*) - \gamma P(\boldsymbol{x}^\gamma) < f(\boldsymbol{x}^*).$$

23.9

We use the penalty method. First, we construct the unconstrained objective function with penalty parameter γ:

$$f(\boldsymbol{x}) = x_1^2 + 2x_2^2 + \gamma(x_1 + x_2 - 3)^2.$$

Because f is a quadratic with positive definite quadratic term, it is easy to find its minimizer:

$$\boldsymbol{x}_\gamma = \frac{1}{1 + 2/(3\gamma)}\begin{bmatrix}2\\1\end{bmatrix}.$$

For example, we can obtain the above by solving the FONC:

$$\begin{aligned}2(1+\gamma)x_1 + 2\gamma x_2 - 6\gamma &= 0 \\ 2\gamma x_1 + 2(2+\gamma)x_2 - 6\gamma &= 0.\end{aligned}$$

Now letting $\gamma \to \infty$, we obtain

$$\boldsymbol{x}^* = \begin{bmatrix}2\\1\end{bmatrix}.$$

(It is easy to verify, using other means, that this is indeed the correct solution.)

23.10

Using the penalty method, we construct the unconstrained problem

$$\text{minimize}\, x + \gamma(\max(a - x, 0))^2$$

To find the solution to the above problem, we use the FONC. It is easy to see that the solution x^* satisfies $x^* < a$. The derivative of the above objective function in the region $x < a$ is $1 + 2\gamma(x - a)$. Thus, by the FONC, we have $x^* = a - 1/(2\gamma)$. Since the true solution is at a, the difference is $1/(2\gamma)$. Therefore, for $1/(2\gamma) \leq \varepsilon$, we need $\gamma \geq 1/(2\varepsilon)$. The smallest such γ is $1/(2\varepsilon)$.

23.11

a. We have

$$\frac{1}{2}\|\boldsymbol{x}\|^2 + \gamma\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 = \frac{1}{2}\boldsymbol{x}^\top\begin{bmatrix}1+2\gamma & 2\gamma\\ 2\gamma & 1+2\gamma\end{bmatrix}\boldsymbol{x} - \boldsymbol{x}^\top\begin{bmatrix}2\gamma\\2\gamma\end{bmatrix} + \gamma.$$

The above is a quadratic with positive definite Hessian. Therefore, the minimizer is

$$\begin{aligned} \boldsymbol{x}_\gamma^* &= \begin{bmatrix} 1+2\gamma & 2\gamma \\ 2\gamma & 1+2\gamma \end{bmatrix}^{-1} \begin{bmatrix} 2\gamma \\ 2\gamma \end{bmatrix} \\ &= \frac{1}{2+1/2\gamma} \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \end{aligned}$$

Hence,

$$\lim_{\gamma \to \infty} \boldsymbol{x}_\gamma^* = \frac{1}{2} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

The solution to the original constrained problem is (see Section 12.3)

$$\boldsymbol{x}^* = \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1} \boldsymbol{b} = \frac{1}{2} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

b. We represent the objective function of the associated unconstrained problem as

$$\frac{1}{2}\|\boldsymbol{x}\|^2 + \gamma \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 = \frac{1}{2}\boldsymbol{x}^\top \left(\boldsymbol{I}_n + 2\gamma \boldsymbol{A}^\top \boldsymbol{A}\right) \boldsymbol{x} - \boldsymbol{x}^\top \left(2\gamma \boldsymbol{A}^\top \boldsymbol{b}\right) + \gamma \boldsymbol{b}^\top \boldsymbol{b}.$$

The above is a quadratic with positive definite Hessian. Therefore, the minimizer is

$$\begin{aligned} \boldsymbol{x}_\gamma^* &= \left(\boldsymbol{I}_n + 2\gamma \boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \left(2\gamma \boldsymbol{A}^\top \boldsymbol{b}\right) \\ &= \left(\frac{1}{2\gamma} \boldsymbol{I}_n + \boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b}. \end{aligned}$$

Let $\boldsymbol{A} = \boldsymbol{U} \begin{bmatrix} \boldsymbol{S} & \boldsymbol{O} \end{bmatrix} \boldsymbol{V}^\top$ be the singular value decomposition of $\boldsymbol{A}$. For simplicity, denote $\varepsilon = 1/2\gamma$. We have

$$\begin{aligned} \boldsymbol{x}_\gamma^* &= \left(\varepsilon \boldsymbol{I}_n + \boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b} \\ &= \left(\varepsilon \boldsymbol{I}_n + \boldsymbol{V} \begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{O} \end{bmatrix} \boldsymbol{U}^\top \boldsymbol{U} \begin{bmatrix} \boldsymbol{S} & \boldsymbol{O} \end{bmatrix} \boldsymbol{V}^\top\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b} \\ &= \left(\varepsilon \boldsymbol{I}_n + \boldsymbol{V} \begin{bmatrix} \boldsymbol{S}^2 & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix} \boldsymbol{V}^\top\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b} \\ &= \boldsymbol{V} \begin{bmatrix} \varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2 & \boldsymbol{O} \\ \boldsymbol{O} & \varepsilon \boldsymbol{I}_{n-m} \end{bmatrix}^{-1} \boldsymbol{V}^\top \boldsymbol{A}^\top \boldsymbol{b}. \end{aligned}$$

Note that

$$\boldsymbol{V}^\top \boldsymbol{A}^\top = \begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{O} \end{bmatrix} \boldsymbol{U}^\top.$$

Also,

$$\begin{bmatrix} \varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2 & \boldsymbol{O} \\ \boldsymbol{O} & \varepsilon \boldsymbol{I}_{n-m} \end{bmatrix}^{-1} = \begin{bmatrix} (\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} & \boldsymbol{O} \\ \boldsymbol{O} & \frac{1}{\varepsilon} \boldsymbol{I}_{n-m} \end{bmatrix},$$

where $(\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1}$ is diagonal. Hence,

$$\begin{aligned}
\boldsymbol{x}_\gamma^* &= \left(\varepsilon \boldsymbol{I}_n + \boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b} \\
&= \boldsymbol{V} \begin{bmatrix} (\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} & \boldsymbol{O} \\ \boldsymbol{O} & \frac{1}{\varepsilon} \boldsymbol{I}_{n-m} \end{bmatrix} \begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{O} \end{bmatrix} \boldsymbol{U}^\top \\
&= \boldsymbol{V} \begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{O} \end{bmatrix} (\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top \\
&= \boldsymbol{V} \begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{O} \end{bmatrix} \boldsymbol{U}^\top \boldsymbol{U} (\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top \\
&= \boldsymbol{A}^\top \boldsymbol{U} (\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top.
\end{aligned}$$

Note that as $\gamma \to \infty$, $\varepsilon \to 0$, and

$$\boldsymbol{U}(\varepsilon \boldsymbol{I}_m + \boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top \to \boldsymbol{U}(\boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top.$$

But,

$$\boldsymbol{U}(\boldsymbol{S}^2)^{-1} \boldsymbol{U}^\top = \left(\boldsymbol{U} \boldsymbol{S}^2 \boldsymbol{U}^\top\right)^{-1} = (\boldsymbol{A}\boldsymbol{A}^\top)^{-1}.$$

Therefore,

$$\boldsymbol{x}_\gamma^* \to \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top)^{-1} \boldsymbol{b} = \boldsymbol{x}^*.$$

24. Multi-Objective Optimization

24.1

The MATLAB code is as follows:

```
function multi_op
%MULTI_OP, illustrates multi-objective optimization.

clear
clc
disp ('                    ')
disp ('This is a demo illustrating multi-objective optimization.')
disp ('The numerical example is a modification of the example')
disp ('from the 2002 book by A. Osyczka,')
disp ('Example 5.1 on pages 101--105')
disp ('------------------------------------------------------------')
disp ('Select the population size denoted POPSIZE, for example, 50.')
disp ('                    ')
POPSIZE=input('Population size POPSIZE  = ');
disp ('------------------------------------------------------------')
disp ('Select the number of iterations denoted NUMITER; e.g., 10.')
disp ('                    ')
NUMITER=input('Number of iterations NUMITER  = ');
disp ('                    ')
disp ('------------------------------------------------------------')

% Main
for i = 1:NUMITER
    fprintf('Working on Iteration %.0f...\n',i)
   xmat = genxmat(POPSIZE);
   if i~=1
      for j = 1:length(xR)
```

```
         xmat = [xmat;xR{j}];
      end
   end
   [xR,fR] = Select_P(xmat);
   fprintf('Number of Pareto solutions: %.0f\n',length(fR))

end
disp ('                    ')
disp ('------------------------------------------------------------')

fprintf(' Pareto solutions \n')
celldisp(xR)
disp ('                    ')
disp ('------------------------------------------------------------')

fprintf(' Objective vector values \n')
celldisp(fR)
xlabel('f_1','Fontsize',16)
ylabel('f_2','Fontsize',16)
title('Pareto optimal front','Fontsize',16)
set(gca,'Fontsize',16)
grid
for i=1:length(xR)
 xx(i)=xR{i}(1);
 yy(i)=xR{i}(2);
end
XX=[xx; yy];
figure
axis([1 7 5 10])
hold on
for i=1:size(XX,2)
plot(XX(1,i),XX(2,i),'marker','o','markersize',6)
end
xlabel('x_1','Fontsize',16)
ylabel('x_2','Fontsize',16)
title('Pareto optimal solutions','Fontsize',16)
set(gca,'Fontsize',16)
grid
hold off
figure
axis([-2 10 2 13])
hold on
plot([2 6],[5 5],'marker','o','markersize',6)
plot([6 6],[5 9],'marker','o','markersize',6)
plot([2 6],[9 9],'marker','o','markersize',6)
plot([2 2],[5 9],'marker','o','markersize',6)
for i=1:size(XX,2)
plot(XX(1,i),XX(2,i),'marker','x','markersize',10)
end
x1=-2:.2:10;
x2=2:.2:13;
[X1, X2]=meshgrid(x1,x2);
Z1=-X1.^2 - X2;
v=[0 -5 -7 -10 -15 -20 -30 -40 -60];
cs1=contour(X1,X2,Z1,v);
clabel(cs1)
Z2=X1+X2.^2;
v2=[20 25 35 40 60 80 100 120];
cs2=contour(X1,X2,Z2,v2);
```

```
clabel(cs2)
xlabel('x_1','Fontsize',16)
ylabel('x_2','Fontsize',16)
title('Level sets of f_1 and f_2, and Pareto optimal
points','Fontsize',16)
set(gca,'Fontsize',16)
grid
hold off

function xmat0 = genxmat(POPSIZE)
xmat0 = rand(POPSIZE,2);
xmat0(:,1) = xmat0(:,1)*4+2;
xmat0(:,2) = xmat0(:,2)*4+5;

function [xR,fR] = Select_P(xmat)

% Declaration
J = size(xmat,1);

% Init
Rset = [1];
j = 1;
isstep7 = 0;

% Step 1
x{1} = xmat(1,:);
f{1} = evalfcn(x{1});

% Step 2
while j < J
   j = j+1;

   % Step 3
   r = 1;
   rdel = [];
   q = 0;
   R = length(Rset);

   for k = 1:size(xmat,1)
       x{k} = xmat(k,:);
       f{k} = evalfcn(x{k});
   end

   % Step 4
   while 1
   %for r=1:R
      if all(f{j}<f{Rset(r)})
         q = q+1;
         rdel = [rdel r];

      else

         % Step 5
         if all(f{j}>=f{Rset(r)})
           break
         end
     end

      % Step 6
```

```
      r=r+1;
      if r > R
         isstep7 = 1;
         break
      end
  end

   % Step 7
   if isstep7 == 1
      isstep7 = 0;
      if (q~=0)
         Rset(rdel) =[];
         Rset = [Rset j];
      else

         %Step 8
         Rset = [Rset j];
      end
  end

   for k = 1:size(xmat,1)
       x{k} = xmat(k,:);
       f{k} = evalfcn(x{k});
   end

   R = length(Rset);
end

% Return the Pareto solution.
for i = 1:length(Rset)
   xR{i} = x{Rset(i)};
   fR{i} = f{Rset(i)};
end


x1 = [];
y1 = [];
x2  = [];
y2  = [];
for k = 1:size(xmat,1)
  if ismember(k,Rset)
        x1 = [x1 f{k}(1)];
        y1 = [y1 f{k}(2)];
    else
        x2 = [x2 f{k}(1)];
        y2 = [y2 f{k}(2)];
    end
end
%newplot
plot(x1,y1,'xr',x2,y2,'.b')
drawnow

function y = f1(x)
% y = x(1)^2+x(2);
% The above function is the original function in the Osyczka's 2002
% book,
% (Example 5.1, page 101).
% Its negative makes a much more interesting example.
```

```
y = -(x(1)^2+x(2));

function y = f2(x)
y = x(1)+x(2)^2;

function y = evalfcn(x)
y(1) = f1(x);
y(2) = f2(x);
```

24.2

a. We proceed using contraposition. Assume that $\boldsymbol{x}^*$ is not Pareto optimal. Therefore, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. Since $\boldsymbol{c} > \boldsymbol{0}$,

$$\boldsymbol{c}^\top \boldsymbol{f}(\boldsymbol{x}^*) > \boldsymbol{c}^\top \boldsymbol{f}(\hat{\boldsymbol{x}}),$$

which implies that $\boldsymbol{x}^*$ is not a global minimizer for the weighted-sum problem.

For the converse, consider the following counterexample: $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : \|\boldsymbol{x}\| \geq 1, \boldsymbol{x} \geq \boldsymbol{0}\}$ and $\boldsymbol{f}(\boldsymbol{x}) = [x_1, x_2]^\top$. It is easy to see that the Pareto front is $\{\boldsymbol{x} : \|\boldsymbol{x}\| = 1, \boldsymbol{x} \geq \boldsymbol{0}\}$ (i.e., the part of the unit circle in the nonnegative quadrant). So $\boldsymbol{x}^* = (1/\sqrt{2})[1, 1]^\top$ is a Pareto minimizer. However, there is no $\boldsymbol{c} > \boldsymbol{0}$ such that $\boldsymbol{x}^*$ is a global minimizer of the weighted-sum problem. To see this, fix $\boldsymbol{c} > \boldsymbol{0}$ (assuming $c_1 \leq c_2$ without loss of generality) and consider the objective function value $f(\boldsymbol{x}^*) = (c_1 + c_2)/\sqrt{2}$ for the weighted-sum problem. Now, the point $\boldsymbol{x}_0 = [1, 0]^\top$ is also a feasible point. Moreover $f(\boldsymbol{x}_0) = c_1 \leq (c_1 + c_2)/2 \leq f(\boldsymbol{x}^*)$. So $\boldsymbol{x}^*$ is not a global minimizer of the weighted-sum problem.

b. We proceed using contraposition. Assume that $\boldsymbol{x}^*$ is not Pareto optimal. Therefore, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. By assumption, for all $i = 1, \ldots, \ell$, $f_i(\boldsymbol{x}^*) \geq 0$, which implies that

$$\sum_{i=1}^{\ell} (f_i(\boldsymbol{x}^*))^p > \sum_{i=1}^{\ell} (f_i(\hat{\boldsymbol{x}}))^p$$

(because $p > 0$). Hence, $\boldsymbol{x}^*$ is not a global minimizer for the minimum-norm problem.

For the converse, consider the following counterexample: $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : x_1 + 2x_2 \geq 2, \boldsymbol{x} \geq \boldsymbol{0}\}$ and $\boldsymbol{f}(\boldsymbol{x}) = [x_1, x_2]^\top$. It is easy to see that the Pareto front is $\{\boldsymbol{x} : x_1 + 2x_2 = 2, \boldsymbol{x} \geq \boldsymbol{0}\}$. So $\boldsymbol{x}^* = [1, 1/2]^\top$ is a Pareto minimizer. However, there is no $p > 0$ such that $\boldsymbol{x}^*$ is a global minimizer of the minimum-norm problem. To see this, fix $p > 0$ and consider the objective function value $f(\boldsymbol{x}^*) = 1 + (1/2)^p$ for the minimum-norm problem. Now, the point $\boldsymbol{x}_0 = [0, 1]^\top$ is also a feasible point. Moreover $f(\boldsymbol{x}_0) = 1 \leq 1 + (1/2)^p = f(\boldsymbol{x}^*)$. So $\boldsymbol{x}^*$ is not a global minimizer of the minimum-norm problem.

c. For the first part, consider the following counterexample: $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : x_1 + x_2 \geq 2, \boldsymbol{x} \geq \boldsymbol{0}\}$ and $\boldsymbol{f}(\boldsymbol{x}) = [x_1, x_2]^\top$. The Pareto front is $\{\boldsymbol{x} : x_1 + x_2 = 2, \boldsymbol{x} \geq \boldsymbol{0}\}$, and $\boldsymbol{x}^* = [1/2, 3/4]^\top$ is a Pareto minimizer. But

$$\begin{aligned} f(\boldsymbol{x}^*) &= \max\{f_1(\boldsymbol{x}^*), f_2(\boldsymbol{x}^*)\} \\ &= \max\{1/2, 3/4\} \\ &= 3/4. \end{aligned}$$

However, $\boldsymbol{x}_0 = [1, 1]^\top$ is also a feasible point, and $f(\boldsymbol{x}_0) = 1 < f(\boldsymbol{x}^*)$. Hence, $\boldsymbol{x}^*$ is not a global minimizer of the minimax problem.

For the second part, suppose $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : x_1 \leq 2\}$ and $\boldsymbol{f}(\boldsymbol{x}) = [x_1, 2]^\top$. Then, for any $\boldsymbol{x} \in \Omega$, $\max\{f_1(\boldsymbol{x}), f_2(\boldsymbol{x})\} = 2$. So any $\boldsymbol{x}^* \in \mathbb{R}^2$ is a global minimizer of the minimax (single-objective) problem.

However, consider another point $\hat{\boldsymbol{x}}$ such that $\hat{x}_1 < x_1^*$. Then, $f_1(\hat{\boldsymbol{x}}) < f_1(\boldsymbol{x}^*)$ and $f_2(\hat{\boldsymbol{x}}) = f_2(\boldsymbol{x}^*)$. Hence, $\boldsymbol{x}^*$ is not a Pareto minimizer.

In fact, in the above example, no Pareto minimizer exists. However, if we set $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : 1 \leq x_1 \leq 2\}$, then the counterexample is still valid, but in this case any point of the form $[1, x_2]^\top$ is a Pareto minimizer.

24.3

Let

$$f(\boldsymbol{x}) = \boldsymbol{c}^{*\top} \boldsymbol{f}(\boldsymbol{x})$$

where $\boldsymbol{x} \in \Omega = \{\boldsymbol{x} : \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}\}$. The function f is convex because all the functions f_i are convex and $c_i^* > 0$, $i = 1, 2, \ldots, \ell$. We can represent the given first-order condition in the following form: for any feasible direction $\boldsymbol{d}$ at $\boldsymbol{x}^*$, we have

$$\boldsymbol{d}^\top \nabla f(\boldsymbol{x}^*) \geq 0.$$

By Theorem 22.7, the point $\boldsymbol{x}^*$ is a global minimizer of f over Ω. Therefore,

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

That is,

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

To finish the proof, we now assume that $\boldsymbol{x}^*$ is not Pareto optimal and the above condition holds. We then proceed using the proof by contradiction. Because, by assumption, $\boldsymbol{x}^*$ is not Pareto optimal, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. Since for all $i = 1, 2, \ldots, \ell$, $c_i^* > 0$, we must have

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) > \sum_{i=1}^{\ell} c_i^* f_i(\hat{\boldsymbol{x}}),$$

which contradicts the above condition, $\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$. This completes the proof. (See also Exercise 24.2, part a.)

24.4

Let

$$f(\boldsymbol{x}) = \boldsymbol{c}^{*\top} \boldsymbol{f}(\boldsymbol{x})$$

where $\boldsymbol{x} \in \Omega = \{\boldsymbol{x} : \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}\}$. The function f is convex because all the functions f_i are convex and $c_i^* > 0$, $i = 1, 2, \ldots, \ell$. We can represent the given Lagrange condition in the form

$$\begin{aligned} D f(\boldsymbol{x}^*) + \boldsymbol{\lambda}^{*\top} D \boldsymbol{g}(\boldsymbol{x}^*) &= \boldsymbol{0}^\top \\ \boldsymbol{h}(\boldsymbol{x}^*) &= \boldsymbol{0}. \end{aligned}$$

By Theorem 22.8, the point $\boldsymbol{x}^*$ is a global minimizer of f over Ω. Therefore,

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

That is,

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

To finish the proof, we now assume that $\boldsymbol{x}^*$ is not Pareto optimal and the above condition holds. We then proceed using the proof by contradiction. Because, by assumption, $\boldsymbol{x}^*$ is not Pareto optimal, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. Since for all $i = 1, 2, \ldots, \ell$, $c_i^* > 0$, we must have

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) > \sum_{i=1}^{\ell} c_i^* f_i(\hat{\boldsymbol{x}}),$$

which contradicts the above condition, $\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$. This completes the proof. (See also Exercise 24.2, part a.)

24.5

Let

$$f(\boldsymbol{x}) = \boldsymbol{c}^{*\top} \boldsymbol{f}(\boldsymbol{x})$$

where $\boldsymbol{x} \in \Omega = \{\boldsymbol{x} : \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}\}$. The function f is convex because all the functions f_i are convex and $c_i^* > 0$, $i = 1, 2, \ldots, \ell$. We can represent the given KKT condition in the form

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \boldsymbol{0} \\ Df(\boldsymbol{x}^*) + \boldsymbol{\mu}^{*\top} D\boldsymbol{g}(\boldsymbol{x}^*) &= \boldsymbol{0}^\top \\ \boldsymbol{\mu}^{*\top} \boldsymbol{g}(\boldsymbol{x}^*) &= 0 \\ \boldsymbol{g}(\boldsymbol{x}^*) &\leq \boldsymbol{0}. \end{aligned}$$

By Theorem 22.9, the point $\boldsymbol{x}^*$ is a global minimizer of f over Ω. Therefore,

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

That is,

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

To finish the proof, we now assume that $\boldsymbol{x}^*$ is not Pareto optimal and the above condition holds. We then proceed using the proof by contradiction. Because, by assumption, $\boldsymbol{x}^*$ is not Pareto optimal, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. Since for all $i = 1, 2, \ldots, \ell$, $c_i^* > 0$, we must have

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) > \sum_{i=1}^{\ell} c_i^* f_i(\hat{\boldsymbol{x}}),$$

which contradicts the above condition, $\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$. This completes the proof. (See also Exercise 24.2, part a.)

24.6

Let

$$f(\boldsymbol{x}) = \boldsymbol{c}^{*\top} \boldsymbol{f}(\boldsymbol{x})$$

where $\boldsymbol{x} \in \Omega = \{\boldsymbol{x} : \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}, \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}\}$. The function f is convex because all the functions f_i are convex and $c_i^* > 0$, $i = 1, 2, \ldots, \ell$. We can represent the given KKT-type condition in the form

$$\begin{aligned} \boldsymbol{\mu}^* &\geq \boldsymbol{0} \\ Df(\boldsymbol{x}^*) + \boldsymbol{\lambda}^{*\top} D\boldsymbol{h}(\boldsymbol{x}^*) + \boldsymbol{\mu}^{*\top} D\boldsymbol{g}(\boldsymbol{x}^*) &= \boldsymbol{0}^\top \\ \boldsymbol{\mu}^{*\top} \boldsymbol{g}(\boldsymbol{x}^*) &= 0 \\ \boldsymbol{h}(\boldsymbol{x}^*) &= \boldsymbol{0} \\ \boldsymbol{g}(\boldsymbol{x}^*) &\leq \boldsymbol{0}. \end{aligned}$$

By Theorem 22.9, the point $\boldsymbol{x}^*$ is a global minimizer of f over Ω. Therefore,

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

That is,

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \Omega.$$

To finish the proof, we now assume that $\boldsymbol{x}^*$ is not Pareto optimal and the above condition holds. We then proceed using the proof by contradiction. Because, by assumption, $\boldsymbol{x}^*$ is not Pareto optimal, there exists a point $\hat{\boldsymbol{x}} \in \Omega$ such that

$$f_i(\hat{\boldsymbol{x}}) \leq f_i(\boldsymbol{x}^*) \quad \text{for all} \quad i = 1, 2, \ldots, \ell$$

and for some j, $f_j(\hat{\boldsymbol{x}}) < f_j(\boldsymbol{x}^*)$. Since for all $i = 1, 2, \ldots, \ell$, $c_i^* > 0$, we must have

$$\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) > \sum_{i=1}^{\ell} c_i^* f_i(\hat{\boldsymbol{x}}),$$

which contradicts the above condition, $\sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x}^*) \leq \sum_{i=1}^{\ell} c_i^* f_i(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$. This completes the proof. (See also Exercise 24.2, part a.)

24.7

The given minimax problem is equivalent to the problem given in the hint:

$$\begin{array}{ll} \text{minimize} & z \\ \text{subject to} & f_i(\boldsymbol{x}) - z \leq 0, \ i = 1, 2. \end{array}$$

Suppose $[\boldsymbol{x}^{*\top}, z^*]^\top$ is a local minimizer for the above problem (which is equivalent to $\boldsymbol{x}^*$ being a local minimizer to for the original problem). Then, by the KKT Theorem, there exists $\boldsymbol{\mu}^{*\top} \geq \mathbf{0}$, where $\boldsymbol{\mu}^* \in \mathbb{R}^2$, such that

$$\begin{aligned} [\mathbf{0}^\top, 1] + \boldsymbol{\mu}^{*\top} \begin{bmatrix} [\nabla f_1(\boldsymbol{x}^*)^\top, -1] \\ [\nabla f_2(\boldsymbol{x}^*)^\top, -1] \end{bmatrix} &= \mathbf{0}^\top \\ \boldsymbol{\mu}^{*\top} \begin{bmatrix} f_1(\boldsymbol{x}^*) - z^* \\ f_2(\boldsymbol{x}^*) - z^* \end{bmatrix} &= 0. \end{aligned}$$

Rewriting the first equation above, we get

$$\mu_1^* \nabla f_1(\boldsymbol{x}^*) + \mu_2^* \nabla f_2(\boldsymbol{x}^*) = \mathbf{0}, \qquad \mu_1^* + \mu_2^* = 1.$$

Rewriting the second equation, we get

$$\mu_i^* (f_i(\boldsymbol{x}^*) - z^*) = 0, \ i = 1, 2.$$

Suppose $f_i(\boldsymbol{x}^*) < \max\{f_1(\boldsymbol{x}^*), f_2(\boldsymbol{x}^*)\}$, where $i \in \{1, 2\}$. Then, $z^* > f_i(\boldsymbol{x}^*)$. Hence, by the above equation we conclude that $\mu_i^* = 0$.