BOZ780 Assignment 1

DS de Gouveia
15003079

September 13, 2020

Contents

1 Question 1 2
1.1 Question 1 (a): Graphical solution 2
1.2 Question 1 (b): Define points 3
1.3 Question 1 (c): Pre-emptive 4
1.3.1 Minimise z_1 4
1.3.2 Minimise z_2 5
1.4 Question 1 (d): Efficient frontier 6

2 Question 2 7
2.1 Question 2 (a) - General formulation 7
2.1.1 Sets 7
2.1.2 Parameters 7
2.1.3 Variables 7
2.1.4 Objectives 7
2.1.5 Constraints 7
2.2 Question 2 (b) - Basic goal programming 8
2.2.1 Sets 8
2.2.2 Parameters 8
2.2.3 Variables 8
2.2.4 Objectives 8
2.2.5 Constraints 8
2.2.6 Solution 9
2.3 Question 2 (c) - Pre-emptive goal programming 10
2.3.1 Sets 10
2.3.2 Parameters 10
2.3.3 Variables 10
2.3.4 Objectives 10
2.3.5 Constraints 10
2.3.6 Solution: P1 11
2.3.7 Solution: P2 11
2.3.8 Solution: P3 11
2.4 Question 2 (d) - Weighted sum goal programming 12
2.4.1 Sets 12
2.4.2 Parameters 12
2.4.3 Variables 12
2.4.4 Objectives 12

2.4.5 Constraints 12
2.4.6 Solution 12
2.5 Question 2 (e) - Basic goal programming efficiency adjustment 14
2.5.1 Sets 14
2.5.2 Parameters 14
2.5.3 Variables 14
2.5.4 Objectives 14
2.5.5 Constraints 14
2.5.6 Solution 15

3 Question 1 R Code **16**

4 Question 2 R Code **20**

Question 1

Dashed lines represent $z_1 = 5x_1 - x_2$, dotted lines represent $z_2 = x_1 + 4x_2$ and the feasible region, bounded by solid black lines is shaded gold. The co-ordinate grid in proceeding figures is represented as (x_2, x_1), please take note of the axis.

Question 1 (a): Graphical solution

The solution to z_2 occurs at $(5, 0)$ where the solution for z_1 occurs at $(0, 4)$ both indicated in turquoise. These points do not coincide.

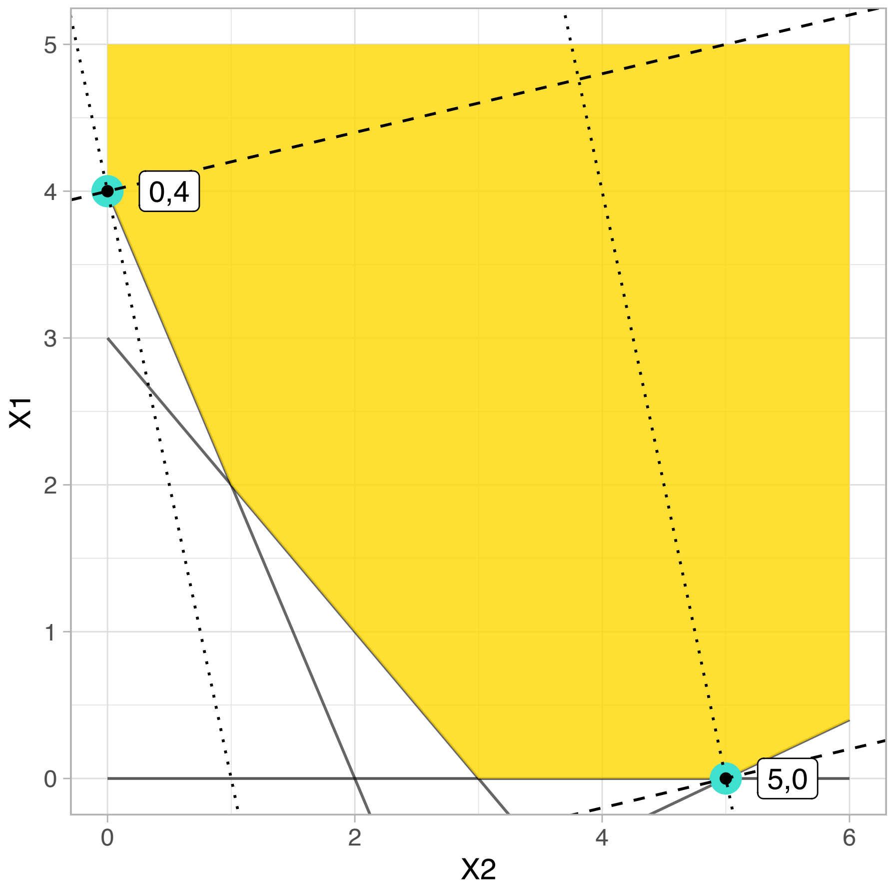

Figure 1: Optimal points of z_1 and z_2 are not located at the same position as the gradient of their respective objective functions are tangential to each other.

Question 1 (b): Define points

Points are defined as efficient if there is no area below the intersection of z_1 (dashed) and z_2 (dotted) that lies within the feasible region.

1. $(5, 0)$ Efficient
2. $(2, 1)$ Efficient
3. $(1, 1)$ Infeasible
4. $(1, 2)$ Efficient
5. $(0, 5)$ Dominated

Point $(0, 5)$ is dominated as point $(0, 4)$ improves on the solution without degrading either objective.

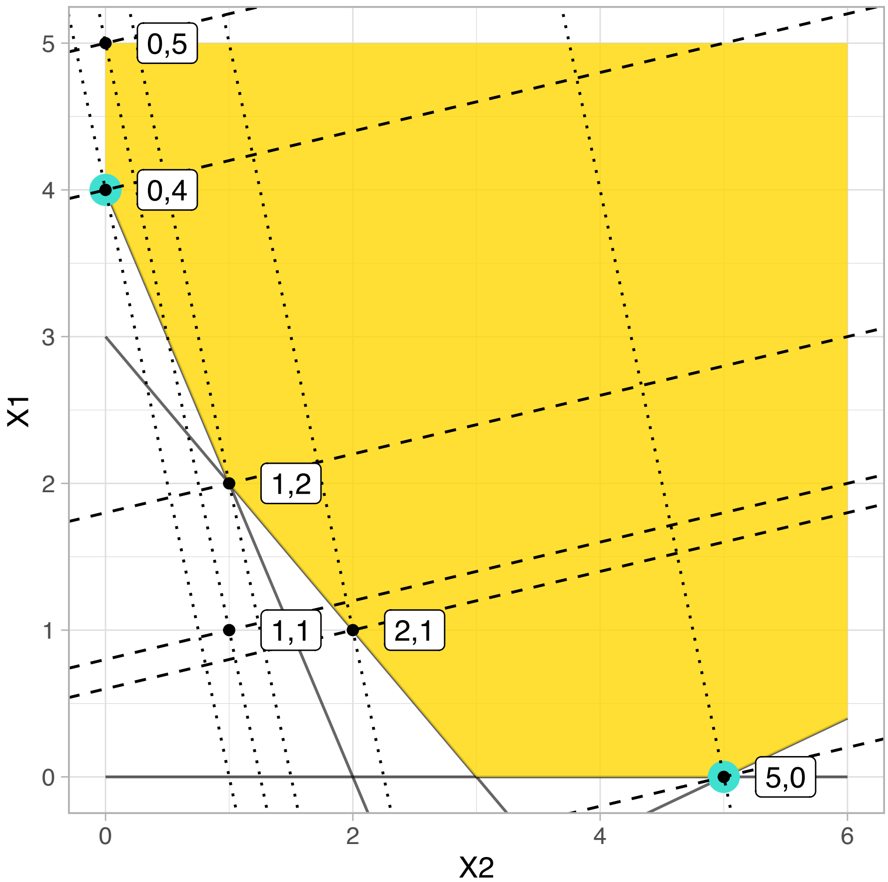

Figure 2: Efficient, dominated and other points

Question 1 (c): Pre-emptive

Minimise z_1

The solution found for z_1 is introduced as a constraint reducing the solution in which z_2 will attempt to find an efficient point in.

$$\min z_1 = 5x_1 - x_2 \tag{1}$$

Subject to:

$$-5x_1 + 2x_2 \leq 10 \tag{2}$$
$$x_1 + x_2 \geq 3 \tag{3}$$
$$x_1 + 2x_2 \geq 4 \tag{4}$$
$$x_1, x_2 \geq 0 \tag{5}$$

$$z_1 = -5 \tag{6}$$
$$x_1 = 0 \tag{7}$$
$$x_2 = 5 \tag{8}$$

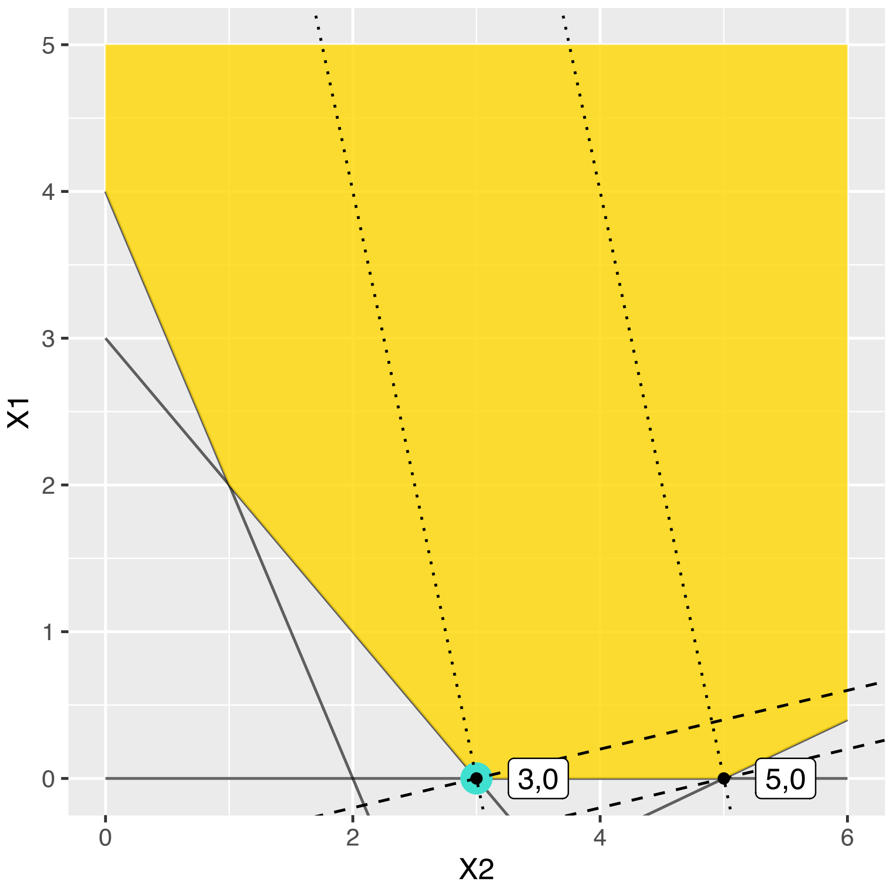

Figure 3: Pre-emptive feasible region with z_1 optimal point at $(5, 0)$.

Minimise z_2

$$\min z_2 = x_1 + 4x_2 \tag{9}$$

Subject to:

$$5x_1 - x_2 \leq -5 \tag{10}$$
$$-5x_1 + 2x_2 \leq 10 \tag{11}$$
$$x_1 + x_2 \geq 3 \tag{12}$$
$$x_1 + 2x_2 \geq 4 \tag{13}$$
$$x_1, x_2 \geq 0 \tag{14}$$

$$z_2 = 12 \tag{15}$$
$$x_1 = 0 \tag{16}$$
$$x_2 = 3 \tag{17}$$

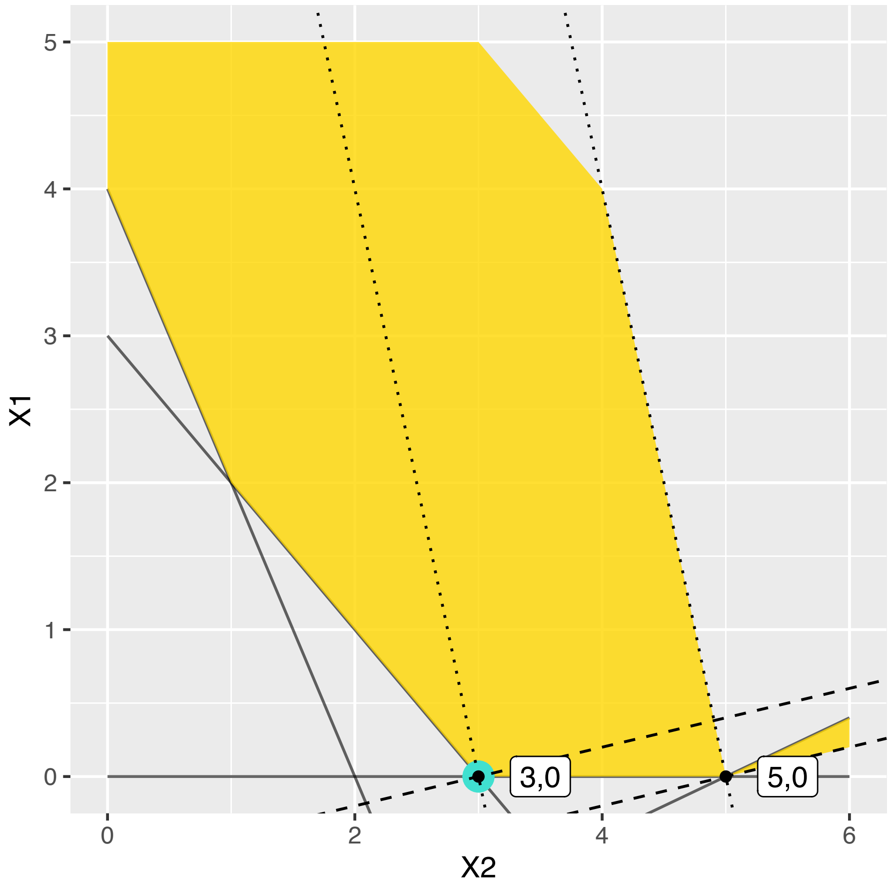

Figure 4: Pre-emptive feasible region including z_1 constraint with z_2 efficient point at $(3, 0)$

.

Question 1 (d): Efficient frontier

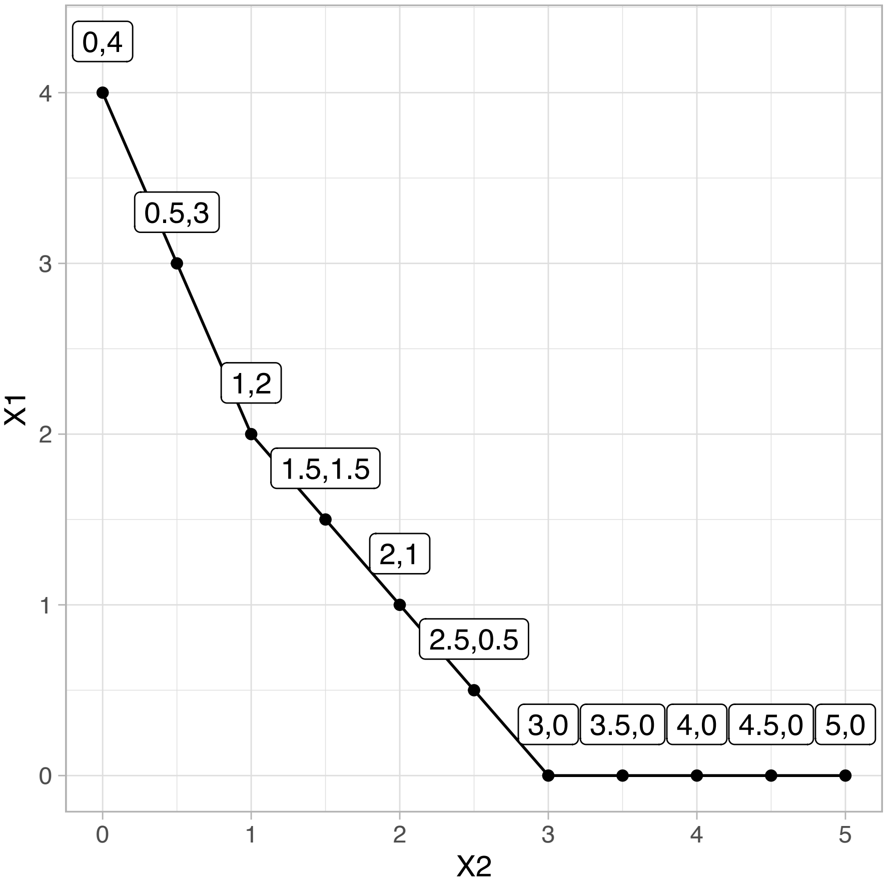

Figure 5: Efficient frontier for x_1 and x_2 in increments of 0.5

Question 2

The following assumptions apply to Question 2:

- The area required for a golf facility is stated as 62 0000 which exceeds the maximum area for site 1 and site 6. The lack of a thousand separator (620 000) and a large deviation from other sites indicates this may be a potential typo. A value of 62 000 has been used instead.
- Weights have been stated as 1000:10:1 which is assumed to reference P1:P2:P3 as priority one to three.

Question 2 (a) - General formulation

Sets

I	set of facility types $I \in (1, \ldots, 4)$ as (Golf, Swimming, Gymnasium, Tennis)
J	set of sites $J \in (1, \ldots, 6)$

Parameters

d_{ij}	user days for facility i on site j	$i \in I, j \in J$
a_j	available land on site j in ft^2	$j \in J$
c_i	construction cost for facility i in \$	$i \in I$
r_i	required land for facility i in ft^2	$i \in I$

Variables

$$x_{ij} \quad \begin{cases} 1 & \text{facility } i \text{ is built on site } j \\ 0 & \text{if else} \end{cases} \quad i \in I, j \in J$$

Objectives

$$\min \sum_{i \in I} \sum_{j \in J} x_{ij} c_i \qquad \text{(construction cost)} \qquad (1)$$

$$\min \sum_{j \in J} a_j - \sum_{i \in I} \sum_{j \in J} x_{ij} r_i \qquad \text{(unused land cost)} \qquad (2)$$

$$\max \sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij} \qquad \text{(user days)} \qquad (3)$$

Constraints

$$\sum_{i \in I} x_{ij} r_i \leq a_j \qquad \forall j \in J \qquad \text{(land capacity)} \qquad (4)$$

$$x_{i1} + x_{i6} = 0 \qquad \forall i \in (2, 3, 4) \qquad \text{(no facilities on site 1 and 6)} \qquad (5)$$

$$x_{1j} = 0 \qquad \forall j \in (2, 3, 4, 5) \qquad \text{(no golf on site 2 to 4)} \qquad (6)$$

$$x_{11} + x_{16} \leq 1 \qquad \text{(either site 1 or 6 can build golf)} \qquad (7)$$

Question 2 (b) - Basic goal programming

Sets

K	set of objectives $K \in (1, 2, 3)$
I	set of facility types $I \in (1, \ldots, 4)$ as (Golf, Swimming, Gymnasium, Tennis)
J	set of sites $J \in (1, \ldots, 6)$

Parameters

d_{ij}	user days for facility i on site j	$i \in I, j \in J$
a_j	available land on site j in ft^2	$j \in J$
c_i	construction cost for facility i in \$	$i \in I$
r_i	required land for facility i in ft^2	$i \in I$

Variables

$$x_{ij} \quad \begin{cases} 1 & \text{facility } i \text{ is built on site } j \\ 0 & \text{if else} \end{cases} \qquad i \in I, j \in J$$

d_k deficiency variable for objective k $\quad k \in K$

Objectives

$$\min \quad \sum_{k \in K} d_k \qquad \text{(deficiencies)} \tag{8}$$

Constraints

$$\begin{aligned} \sum_{j \in J} a_j - \sum_{i \in I} \sum_{j \in J} x_{ij} r_i - d_1 &\leq 40\ 000 \text{ ft}^2 \\ \sum_{i \in I} \sum_{j \in J} x_{ij} r_i + d_1 &\geq \sum_{j \in J} a_j - 40\ 000 \text{ ft}^2 \quad \text{(P1: unused land cost)} \end{aligned} \tag{9}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} c_i - d_2 < \$1.2 \text{ million} \qquad \text{(P2: construction cost)} \tag{10}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij} + d_3 \geq 200\ 000 \text{ days} \qquad \text{(P3: user days)} \tag{11}$$

$$\sum_{i \in I} x_{ij} r_i \leq a_j \qquad \forall j \in J \qquad \text{(land capacity)} \tag{12}$$

$$x_{i1} + x_{i6} = 0 \qquad \forall i \in (2, 3, 4) \qquad \text{(no facilities on site 1 and 6)} \tag{13}$$

$$x_{1j} = 0 \qquad \forall j \in (2, 3, 4, 5) \qquad \text{(no golf on site 2 to 4)} \tag{14}$$

$$x_{11} + x_{16} \leq 1 \qquad \text{(either site 1 or 6 can build golf)} \tag{15}$$

Solution

Metric	**Value**	**Target**
Unused land	234 000 ft^2	40 000 ft^2
Used land	271 000 ft^2	-
Construction cost	\$ 980 000	\$ 1 200 000
User days	148 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_k = \begin{bmatrix} 194\ 000 \\ 0 \\ 52\ 000 \end{bmatrix} \qquad \min(z) = 246\ 000 \tag{16}$$

Question 2 (c) - Pre-emptive goal programming

Sets

K	set of objectives $K \in (1, 2, 3)$
I	set of facility types $I \in (1, \ldots, 4)$ as (Golf, Swimming, Gymnasium, Tennis)
J	set of sites $J \in (1, \ldots, 6)$

Parameters

d_{ij}	user days for facility i on site j	$i \in I, j \in J$
a_j	available land on site j in ft^2	$j \in J$
c_i	construction cost for facility i in \$	$i \in I$
r_i	required land for facility i in ft^2	$i \in I$

Variables

$$x_{ij} \quad \begin{cases} 1 & \text{facility } i \text{ is built on site } j \\ 0 & \text{if else} \end{cases} \qquad i \in I, j \in J$$

d_k deficiency variable for objective k $\quad k \in K$

Objectives

$$\min \quad d_1 \tag{17}$$

$$\min \quad d_2 \qquad \text{(subject to } d_1\text{)} \tag{18}$$

$$\min \quad d_3 \qquad \text{(subject to } d_1 \text{ and } d_2\text{)} \tag{19}$$

Constraints

$$d_1 = 0 \qquad \text{(used in } d_2 \text{ and } d_3\text{)} \tag{20}$$

$$d_2 = 0 \qquad \text{(used in } d_3\text{)} \tag{21}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} r_i + d_1 \geq \sum_{j \in J} a_j - 40\ 000 \text{ ft}^2 \qquad \text{(P1: unused land cost)} \tag{22}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} c_i - d_2 < \$1.2 \text{ million} \qquad \text{(P2: construction cost)} \tag{23}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij} + d_3 \geq 200\ 000 \text{ days} \qquad \text{(P3: user days)} \tag{24}$$

$$\sum_{i \in I} x_{ij} r_i \leq a_j \qquad \forall j \in J \qquad \text{(land capacity)} \tag{25}$$

$$x_{i1} + x_{i6} = 0 \qquad \forall i \in (2, 3, 4) \qquad \text{(no facilities on site 1 and 6)} \tag{26}$$

$$x_{1j} = 0 \qquad \forall j \in (2, 3, 4, 5) \qquad \text{(no golf on site 2 to 4)} \tag{27}$$

$$x_{11} + x_{16} \leq 1 \qquad \text{(either site 1 or 6 can build golf)} \tag{28}$$

Solution: P1

Metric	Value	Target
Unused land	107 000 ft^2	40 000 ft^2
Used land	398 000 ft^2	-
Construction cost	\$ 4 015 000	\$ 1 200 000
User days	277 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_1 = \begin{bmatrix} 67\ 000 \end{bmatrix} \qquad \min(z) = 67\ 000 \tag{29}$$

Solution: P2

A new constraint was included whereby $d_1 <= 67\ 000$.

Metric	Value	Target
Unused land	107 000 ft^2	40 000 ft^2
Used land	398 000 ft^2	-
Construction cost	\$ 4 015 000	\$ 1 200 000
User days	277 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_2 = \begin{bmatrix} 2\ 815\ 000 \end{bmatrix} \qquad \min(z) = 2\ 815\ 000 \tag{30}$$

Solution: P3

Two new constraint were included whereby $d_1 <= 67\ 000$ and $d_2 <= 2\ 815\ 000$.

Metric	Value	Target
Unused land	107 000 ft^2	40 000 ft^2
Used land	398 000 ft^2	-
Construction cost	\$ 4 015 000	\$ 1 200 000
User days	277 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_3 = \begin{bmatrix} 0 \end{bmatrix} \qquad \min(z) = 0 \tag{31}$$

The first priority ensured that area was maximised, inadvertently ensuring more user days (than targeted) could occur. However, the construction cost greatly increased and with the imposed d_1 constraint could not decrease after the second minimisation attempt.

Question 2 (d) - Weighted sum goal programming

Sets

K	set of objectives $K \in (1, 2, 3)$
I	set of facility types $I \in (1, \dots, 4)$ as (Golf, Swimming, Gymnasium, Tennis)
J	set of sites $J \in (1, \dots, 6)$

Parameters

d_{ij}	user days for facility i on site j	$i \in I, j \in J$
a_j	available land on site j in ft^2	$j \in J$
c_i	construction cost for facility i in \$	$i \in I$
r_i	required land for facility i in ft^2	$i \in I$

Variables

$$x_{ij} \quad \begin{cases} 1 & \text{facility } i \text{ is built on site } j \\ 0 & \text{if else} \end{cases} \quad i \in I, j \in J$$

d_k deficiency variable for objective k $\quad k \in K$

Objectives

$$\min \quad 1000d_1 + 10d_2 + d_3 \tag{32}$$

Constraints

$$\sum_{i \in I} \sum_{j \in J} x_{ij} r_i + d_1 \geq \sum_{j \in J} a_j - 40\,000 \text{ ft}^2 \qquad \text{(P1: unused land cost)} \tag{33}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} c_i - d_2 < \$1.2 \text{ million} \qquad \text{(P2: construction cost)} \tag{34}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij} + d_3 \geq 200\,000 \text{ days} \qquad \text{(P3: user days)} \tag{35}$$

$$\sum_{i \in I} x_{ij} r_i \leq a_j \qquad \forall j \in J \qquad \text{(land capacity)} \tag{36}$$

$$x_{i1} + x_{i6} = 0 \qquad \forall i \in (2, 3, 4) \qquad \text{(no facilities on site 1 and 6)} \tag{37}$$

$$x_{1j} = 0 \qquad \forall j \in (2, 3, 4, 5) \qquad \text{(no golf on site 2 to 4)} \tag{38}$$

$$x_{11} + x_{16} \leq 1 \qquad \text{(either site 1 or 6 can build golf)} \tag{39}$$

Solution

Metric	Value	Target
Unused land	107 000 ft^2	40 000 ft^2
Used land	398 000 ft^2	-
Construction cost	\$ 4 015 000	\$ 1 200 000
User days	277 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_k = \begin{bmatrix} 67\ 000 \\ 2\ 814\ 999 \\ 0 \end{bmatrix} \qquad \min(z) = 95\ 150\ 000 \tag{40}$$

Question 2 (e) - Basic goal programming efficiency adjustment

Sets

K	set of objectives $K \in (1, 2, 3)$
I	set of facility types $I \in (1, \ldots, 4)$ as (Golf, Swimming, Gymnasium, Tennis)
J	set of sites $J \in (1, \ldots, 6)$

Parameters

d_{ij}	user days for facility i on site j	$i \in I, j \in J$
a_j	available land on site j in ft^2	$j \in J$
c_i	construction cost for facility i in \$	$i \in I$
r_i	required land for facility i in ft^2	$i \in I$

Variables

$$x_{ij} \quad \begin{cases} 1 & \text{facility } i \text{ is built on site } j \\ 0 & \text{if else} \end{cases} \quad i \in I, j \in J$$

d_k deficiency variable for objective k $\quad k \in K$

Objectives

A small number 0.26 has been selected as the adjustment factor with positive and negative signs applied for minimisation and maximisation respectively.

$$\begin{aligned} \min \quad & \sum_{k \in K} d_k + 0.001(\sum_{i \in I} \sum_{j \in J} x_{ij} c_i) \\ & - 0.26(\sum_{i \in I} \sum_{j \in J} x_{ij} r_i) \\ & - 0.26(\sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij}) \end{aligned} \tag{41}$$

The unused area objective function (2) has been converted from a minimisation of unused area to a maximisation of used area resulting in a -0.26 factor. The outcome of the objective is unchained and the constant a_j is removed whilst the land capacity constraint prevents the area of used land from surpassing what is available.

Constraints

$$\sum_{i \in I} \sum_{j \in J} x_{ij} r_i + d_1 \geq \sum_{j \in J} a_j - 40\ 000 \text{ ft}^2 \qquad \text{(P1: unused land cost)} \tag{42}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} c_i - d_2 < \$1.2 \text{ million} \qquad \text{(P2: construction cost)} \tag{43}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ij} u_{ij} + d_3 \geq 200\ 000 \text{ days} \qquad \text{(P3: user days)} \tag{44}$$

$$\sum_{i \in I} x_{ij} r_i \leq a_j \qquad \forall j \in J \qquad \text{(land capacity)} \tag{45}$$

$$x_{i1} + x_{i6} = 0 \qquad \forall i \in (2, 3, 4) \qquad \text{(no facilities on site 1 and 6)} \tag{46}$$

$$x_{1j} = 0 \qquad \forall j \in (2, 3, 4, 5) \qquad \text{(no golf on site 2 to 4)} \tag{47}$$

$$x_{11} + x_{16} \leq 1 \qquad \text{(either site 1 or 6 can build golf)} \tag{48}$$

Solution

Metric	**Value**	**Target**
Unused land	263 000 ft^2	40 000 ft^2
Used land	242 000 ft^2	-
Construction cost	\$ 680 000	\$ 1 200 000
User days	116 000 days	200 000 days

$$x_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad d_k = \begin{bmatrix} 223\ 000 \\ 0 \\ 84\ 000 \end{bmatrix} \qquad \min(z) = 390\ 720 \tag{49}$$

Including the efficiency adjustment results in the cheaper facility being built over others. The co-efficient values attributed to cost far exceed that of area and user days giving it a greater relative weight.

Question 1 R Code

```
library(tidyverse)
library(gridExtra)
library(lpSolve)
library(extrafont)

### ================ Question 1
w <- 12
h <- 12

lp(direction = "min",
   objective.in = c(1,4),
   const.mat = matrix(c(-5,1,1,0,1,
                        2,1,2,1,0),ncol = 2,byrow = FALSE),
   const.dir = c("<=",">=",">=",">=",">="),
   const.rhs = c(10,3,4,0,0))$solution


lengthX <- 6
lengthY <- 5

x <- 0:lengthX
c1 <- (x)*2/5-2
c2 <- -x +3
c3 <- -2*x+4
c4 <- rep(0,lengthX+1)

df.points <- data.frame(x=c(0,1,1,2,5,0),
                        y=c(5,2,1,1,0,4))
df.pointsOptimal <- data.frame(x=c(0,5),
                               y=c(4,0))
point.coords = paste(df.points$x,df.points$y,sep=",")
z1.intercept <- df.points$y -1/5*df.points$x
z2.intercept <- df.points$y +4 *df.points$x

# ==== Optimal points
pointOptimal.coords = paste(df.pointsOptimal$x,df.pointsOptimal$y,sep=",")
data.frame(x, c1,c2,c3,c4) %>%
  ggplot(aes(x = x))+
  geom_line(aes(y=c1), alpha = 0.6)+
  geom_line(aes(y=c2), alpha = 0.6)+
  geom_line(aes(y=c3), alpha = 0.6)+
  geom_line(aes(y=c4), alpha = 0.6)+
  geom_ribbon(aes(ymin = pmax(c1,c2,c3,c4),
                ymax = lengthY),
                fill = 'gold',
                alpha = 0.8)+
  geom_point(data = df.pointsOptimal, aes(x=x, y=y), color = "turquoise", size = 5)+
```

```
  geom_point(data = df.pointsOptimal, aes(x=x, y=y))+
  coord_cartesian(xlim = c(0,lengthX), ylim = c(0,lengthY))+
  labs(x="X2", y="X1")+
  geom_abline(slope = 1/5, intercept = c(4,-1), linetype = "dashed", color = "black")+
  geom_abline(slope = -4, intercept = c(4,20), linetype = "dotted", color = "black")+
  geom_label(data = df.pointsOptimal,aes(x+.5,y,label=pointOptimal.coords))+
  theme_light()

ggsave("Optimal_points_Z1_Z2.pdf",dpi = "retina", width = w, height = h, units = "cm")

# === Which points are eff

data.frame(x, c1,c2,c3,c4) %>%
  ggplot(aes(x = x))+
  geom_line(aes(y=c1), alpha = 0.6)+
  geom_line(aes(y=c2), alpha = 0.6)+
  geom_line(aes(y=c3), alpha = 0.6)+
  geom_line(aes(y=c4), alpha = 0.6)+
  geom_ribbon(aes(ymin = pmax(c1,c2,c3,c4),
                  ymax = lengthY),
              fill = 'gold',
              alpha = 0.8)+
  geom_point(data = df.pointsOptimal, aes(x=x, y=y), color = "turquoise", size = 5)+
  geom_point(data = df.points, aes(x=x, y=y))+
  coord_cartesian(xlim = c(0,lengthX), ylim = c(0,lengthY))+
  labs(x="X2", y="X1")+
  geom_abline(slope = 1/5, intercept = z1.intercept, linetype = "dashed", color = "black"
  geom_abline(slope = -4, intercept = z2.intercept, linetype = "dotted", color = "black")
  geom_label(data = df.points,aes(x+.5,y,label=point.coords))+
  theme_light()
ggsave("Efficient_dominated_other_points.pdf",dpi = "retina", width = w, height = h, unit

# Question 1 c - pre-emptive

z1 <- lp(direction = "min",
   objective.in = c(5,-1),
   const.mat = matrix(c(-5,1,1,0,1,
                        2,1,2,1,0),ncol = 2,byrow = FALSE),
   const.dir = c("<=",">=",">=",">=",">="),
   const.rhs = c(10,3,4,0,0))

z2 <- lp(direction = "min",
         objective.in = c(1,4),
         const.mat = matrix(c(-5,1,1,0,1,1,0,
                              2,1,2,1,0,0,1),ncol = 2,byrow = FALSE),
         const.dir = c("<=",">=",">=",">=",">=","<=","<="),
         const.rhs = c(10,3,4,0,0,z1$solution[1],z1$solution[2]))

z1
```

```
z2
z1$solution
z2$solution

df.pointsOptimal <- data.frame(y=c(z1$solution[1],z2$solution[1]),
                               x=c(z1$solution[2],z2$solution[2]))
point.coords = paste(df.pointsOptimal$x,df.pointsOptimal$y,sep=",")
z1.intercept <- df.pointsOptimal$y -1/5*df.pointsOptimal$x
z2.intercept <- df.pointsOptimal$y +4 *df.pointsOptimal$x


data.frame(x,c1,c2,c3,c4) %>%
  ggplot(aes(x = x))+
  geom_line(aes(y=c1), alpha = 0.6)+
  geom_line(aes(y=c2), alpha = 0.6)+
  geom_line(aes(y=c3), alpha = 0.6)+
  geom_line(aes(y=c4), alpha = 0.6)+
  geom_ribbon(aes(ymin = pmax(c1,c2,c3,c4),
                  ymax = lengthY),
              fill = 'gold',
              alpha = 0.8)+
  geom_point(data = df.pointsOptimal[2,], aes(x=x, y=y), color = "turquoise", size = 5)+
  geom_point(data = df.pointsOptimal, aes(x=x, y=y))+
  coord_cartesian(xlim = c(0,lengthX), ylim = c(0,lengthY))+
  labs(x="X2", y="X1")+
  geom_abline(slope = 1/5, intercept = z1.intercept, linetype = "dashed", color = "black"
  geom_abline(slope = -4, intercept = z2.intercept, linetype = "dotted", color = "black")
  geom_label(data = df.pointsOptimal,aes(x+.5,y,label=point.coords))

ggsave("Pre_emptive.pdf",dpi = "retina", width = w, height = h, units = "cm")



#Pre-emptive with constraint from past solution
c5 <- 1/5*(x)-1
upper <- rep(lengthY, lengthX+1)


data.frame(x,c1,c2,c3,c4,c5) %>%
  ggplot(aes(x = x))+
  geom_line(aes(y=c1), alpha = 0.6)+
  geom_line(aes(y=c2), alpha = 0.6)+
  geom_line(aes(y=c3), alpha = 0.6)+
  geom_line(aes(y=c4), alpha = 0.6)+
  geom_ribbon(aes(ymin = pmax(c1,c2,c3,c4),
                  ymax = c(5,5,5,5,4,0,0.2)),
              fill = 'gold',
              alpha = 0.8)+
  geom_point(data = df.pointsOptimal[2,], aes(x=x, y=y), color = "turquoise", size = 5)+
```

```
  geom_point(data = df.pointsOptimal, aes(x=x, y=y))+
  coord_cartesian(xlim = c(0,lengthX), ylim = c(0,lengthY))+
  labs(x="X2", y="X1")+
  geom_abline(slope = 1/5, intercept = z1.intercept, linetype = "dashed", color = "black"
  geom_abline(slope = -4, intercept = z2.intercept, linetype = "dotted", color = "black")
  geom_label(data = df.pointsOptimal,aes(x+.5,y,label=point.coords))

ggsave("Pre_emptive2.pdf",dpi = "retina", width = w, height = h, units = "cm")
```

```
# Eff fronteir

z2.seq <- seq(0,5,0.5)
df.eff <- data.frame(Obj=z2.seq, x=z2.seq, y=z2.seq)
n = 1
for (i in z2.seq){

  ef.front <- lp(direction = "min",
     objective.in = c(5,-1),
     const.mat = matrix(c(-5,1,1,0,1,0,
                          2,1,2,1,0,1),ncol = 2,byrow = FALSE),
     const.dir = c("<=",">=",">=",">=",">=","<="),
     const.rhs = c(10,3,4,0,0,i))
  df.eff$Obj[n] <- ef.front$objval
  df.eff$x[n] <- ef.front$solution[2]
  df.eff$y[n] <- ef.front$solution[1]
  n <- n+1
}

df.eff %>%
  ggplot(aes(x=x, y=y))+
  geom_path()+
  geom_point()+
  labs(x="X2", y="X1")+
  geom_label(aes(x,y+.3,label=paste(x,y,sep=",")))+
  theme_light()

ggsave("Efficient_frontier.pdf",dpi = "retina", width = w, height = h, units = "cm")
```

Question 2 R Code

```
library(tidyverse)
library(lpSolve)
matmaker <- function(index,length){
  vec <- rep(0,length)
  vec[index] <- 1
  return(vec)
}

#======================= Q2
#=========== Parameters
i <- 4
j <- 6
k <- 3

uij <- 1000* c(31, 0, 0, 0, 0,27,
                0,25,21,32,32, 0,
                0,37,29,38,28, 0,
                0,20,23,22,20, 0)
aj <- 1000* rep(c(80,70,80,90,120,65),i)
ci <- rep(c(340000, 300000, 840000, 85000),j) %>% matrix(nrow = i, byrow = F) %>% t() %>%
ri <- rep(c(62000 ,29000 ,38000 ,45000),j) %>% matrix(nrow = i, byrow = F)  %>% t() %>% a
dk <- c(0,0,0)

dk.k <- c(rep(0,j*i),c(1,1,1))
uij.k <- c(dij, dk)
aj.k <- c(aj, dk)
ci.k <- c(ci, dk)
ri.k <- c(ri, dk)

len <- length(ri.k)

fn.cost <- function(lpsolve.out){
   cost.landused <- sum(lpsolve.out[1:24]*ri)
   cost.landunused <- -sum(lpsolve.out[1:24]*ri) + sum(1000*c(80,70,80,90,120,65))
   cost.construction <- sum(lpsolve.out[1:24]*ci)
   cost.userdays <- sum(lpsolve.out[1:24]*uij)
   return(paste(paste("Used land",cost.landused,sep = ": "),
                paste("Unused land",cost.landunused,sep = ": "),
                paste("Construction cost",cost.construction,sep = ": "),
                paste("User days",cost.userdays,sep = ": "),sep = "       "))
}

#=========== Constraints
cons1.lhs <- c(ri.k*matmaker(c(1,7,13,19),len),
               ri.k*matmaker(c(1+1,7+1,13+1,19+1),len),
               ri.k*matmaker(c(1+2,7+2,13+2,19+2),len),
               ri.k*matmaker(c(1+3,7+3,13+3,19+3),len),
```

```
                ri.k*matmaker(c(1+4,7+4,13+4,19+4),len),
                ri.k*matmaker(c(1+5,7+5,13+5,19+5),len)) %>% matrix(nrow = 6, byrow = T)
cons1.rhs <- 1000*c(80,70,80,90,120,65)

cons2.lhs <- (rep(1,len)*matmaker(c(7,13,19,12,18,24,2,3,4,5),len))#[1:24] %>% matrix(nro
cons2.rhs <- 0

cons3.lhs <- (rep(1,len)*matmaker(c(1,6),len))#[1:24] %>% matrix(nrow = 4, byrow = T)
cons3.rhs <- 1

base.cons.lhs <- rbind(cons1.lhs,cons2.lhs,cons3.lhs)
base.cons.rhs <- c(cons1.rhs, cons2.rhs, cons3.rhs)
base.cons.dir <- c(rep("<=",j), "=","<=")
#======================== Q2B
q2b.obj <- rep(1,len)*matmaker(c(27,26,25),len)

cons1.lhs <- (ri.k+dk.k*matmaker(25,len))#[1:24] %>% matrix(nrow = 4, byrow = T)
cons1.rhs <- -40000 + (1000*sum(80,70,80,90,120,65))
cons2.lhs <- ci.k-dk.k*matmaker(26,len)
cons2.rhs <- 1200000
cons3.lhs <- uij.k+dk.k*matmaker(27,len)
cons3.rhs <- 200000

q2b.cons.lhs <- rbind(base.cons.lhs, cons1.lhs, cons2.lhs, cons3.lhs)
q2b.cons.rhs <- c(base.cons.rhs, cons1.rhs, cons2.rhs, cons3.rhs)
q2b.cons.dir <- c(base.cons.dir, ">=","<",">=")

q2b.lp <- lp(direction = "min",
   objective.in = q2b.obj,
   const.mat = q2b.cons.lhs,
   const.rhs = q2b.cons.rhs,
   const.dir = q2b.cons.dir,
   binary.vec = 1:24)
q2b.lp
q2b.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2b.lp$solution[25:27]
q2b.lp$solution %>% fn.cost()
#======================== Q2C
#========== P1
q2c.obj.p1 <- rep(1,len)*matmaker(c(25),len)

q2c.lp <- lp(direction = "min",
             objective.in = q2c.obj.p1,
             const.mat = q2b.cons.lhs,
             const.rhs = q2b.cons.rhs,
             const.dir = q2b.cons.dir,
             binary.vec = 1:24)
q2c.lp
q2c.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
```

```
q2c.lp$solution[25:27] %>% as.integer()
q2c.lp$solution %>% fn.cost
#========== P2
q2c.obj.p2 <- rep(1,len)*matmaker(c(26),len)

cons1.lhs <- rep(1,len)*matmaker(25,len)
cons1.rhs <- q2c.lp$solution[25]
cons1.dir <- "<="

q2c.cons.lhs.p2 <- rbind(q2b.cons.lhs, cons1.lhs)
q2c.cons.rhs.p2 <- c(q2b.cons.rhs,cons1.rhs)
q2c.cons.dir.p2 <- c(q2b.cons.dir,cons1.dir)

q2c.lp.p2 <- lp(direction = "min",
             objective.in = q2c.obj.p2,
             const.mat = q2c.cons.lhs.p2,
             const.rhs = q2c.cons.rhs.p2,
             const.dir = q2c.cons.dir.p2,
             binary.vec = 1:24)
q2c.lp.p2
q2c.lp.p2$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2c.lp.p2$solution[25:27] %>% as.integer()
q2c.lp.p2$solution %>% fn.cost()
#========== P3
q2c.obj.p3 <- rep(1,len)*matmaker(c(27),len)

cons1.lhs <- rep(1,len)*matmaker(26,len)
cons1.rhs <- q2c.lp.p2$solution[26]
cons1.dir <- "<="

q2c.cons.lhs.p3 <- rbind(q2c.cons.lhs.p2, cons1.lhs)
q2c.cons.rhs.p3 <- c(q2c.cons.rhs.p2,cons1.rhs)
q2c.cons.dir.p3 <- c(q2c.cons.dir.p2,cons1.dir)

q2c.lp.p3 <- lp(direction = "min",
                objective.in = q2c.obj.p3,
                const.mat = q2c.cons.lhs.p3,
                const.rhs = q2c.cons.rhs.p3,
                const.dir = q2c.cons.dir.p3,
                binary.vec = 1:24)
q2c.lp.p3
q2c.lp.p3$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2c.lp.p3$solution[25:27] %>% as.integer()
q2c.lp.p3$solution %>% fn.cost()
#======================== Q2D
q2d.obj <- rep(1,len)*matmaker(c(27,26,25),len)
q2d.obj[25] <- 1000
q2d.obj[26] <- 10
q2d.obj[27] <- 1
```

```
q2d.lp <- lp(direction = "min",
             objective.in = q2d.obj,
             const.mat = q2b.cons.lhs,
             const.rhs = q2b.cons.rhs,
             const.dir = q2b.cons.dir,
             binary.vec = 1:24)
q2d.lp
q2d.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2d.lp$solution[25:27]
q2d.lp$solution %>% fn.cost()


#======================== Q2E
q2e.obj <- 0.26*(ci.k - uij.k - ri.k)     #ri is treated as maximization
q2e.obj[25:27] <- c(1,1,1)
q2e.lp <- lp(direction = "min",
             objective.in = q2e.obj,
             const.mat = q2b.cons.lhs,
             const.rhs = q2b.cons.rhs,
             const.dir = q2b.cons.dir,
             binary.vec = 1:24)
q2e.lp
q2e.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2e.lp$solution[25:27]
q2e.lp$solution %>% fn.cost()


# Results
q2b.lp$solution %>% fn.cost()
q2b.lp
q2b.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2b.lp$solution[25:27] %>% as.integer()

q2c.lp$solution %>% fn.cost()
q2c.lp
q2c.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2c.lp$solution[25:27] %>% as.integer()

q2c.lp.p2$solution %>% fn.cost()
q2c.lp.p2
q2c.lp.p2$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2c.lp.p2$solution[25:27] %>% as.integer()

q2c.lp.p3$solution %>% fn.cost()
q2c.lp.p3
q2c.lp.p3$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2c.lp.p3$solution[25:27] %>% as.integer()
```

```
q2d.lp$solution %>% fn.cost()
q2d.lp
q2d.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2d.lp$solution[25:27] %>% as.integer()

q2e.lp$solution %>% fn.cost()
q2e.lp
q2e.lp$solution[1:24] %>% matrix(nrow = i, byrow = T)
q2e.lp$solution[25:27] %>% as.integer()
```