**Praktikum Komputasi Statistika I**

**Pertemuan 4**

**Program Linear**

## Pendahuluan

Program linear adalah metode penyelesaian masalah optimisasi yang diterapkan pada fungsi dan kendala yang berbentuk linear. Seringkali ditemukan kasus pada dunia nyata dimana kita ingin memaksimumkan (atau meminimumkan) suatu fungsi yang memiliki beberapa kendala didalamnya.

Dalam modul ini, permasalahan yang diselesaikan harus berbentuk persamaan atau pertidaksamaan linear.

Contoh

a. Pertidaksamaan $2x_1 + 5x_2 \leq 12$ dan $x_1 + x_2 \geq 0$ adalah pertidaksamaan linear.
b. Pertidaksamaan $5x_1 + 3x_2x_3 \leq 12$ bukan pertidaksamaan linear.
c. Persamaan $x_1 + x_2 + 5x_3 = 34$ adalah persamaan linear
d. Persamaan $x_1^2 + x_2^2 + x_3^2 + x_1x_2x_3 = 300$ bukan persamaan linear

## Model Program Linear

Terdapat tiga komponen utama dalam model program linear.

a. Variabel Keputusan
   Langkah pertama pemodelan PL adalah dengan menentukan variabel yang diputuskan.

b. Fungsi Tujuan (*Objective Function*)
   Tujuan yang diharapkan, bisa memaksimumkan atau meminimumkan suatu fungsi linear yang terdiri dari variabel keputusan.

c. Kendala (*Constraints*)
   Nilai variabel keputusan harus memenuhi kendala-kendala yang ada. Lebih lanjut, dalam modul ini, semua variabel juga harus bernilai tak negatif (*non-negativity constraint*).

Bentuk kanonik dari permasalahan PL dengan *n* variabel dan *m* kendala adalah sebagai berikut.

| **Maksimumkan** | **Meminimumkan** |
|---|---|
| **Max** $\boldsymbol{C = c_1x_1 + c_2x_2 + \cdots + c_nx_n}$ | Min $C = c_1x_1 + c_2x_2 + \cdots + c_nx_n$ |
| **dengan kendala** | dengan kendala |

| | |
|---|---|
| $\boldsymbol{a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1}$ | $a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \geq b_1$ |
| $\boldsymbol{a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2}$ | $a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \geq b_2$ |
| ... | ... |
| $\boldsymbol{a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{1n}x_n \leq b_m}$ | $a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{1n}x_n \geq b_m$ |
| $\boldsymbol{x_i \geq 0}$ | $x_i \geq 0$ |

Dalam notasi matriks juga dapat dituliskan sebagai

$Max\ C = c^T\boldsymbol{x}$ dengan kendala $A\boldsymbol{x} \leq \boldsymbol{b}$ dan $\boldsymbol{x} \geq 0$

atau

$Min\ C = c^T\boldsymbol{x}$ dengan kendala $A\boldsymbol{x} \geq \boldsymbol{b}$ dan $\boldsymbol{x} \geq 0$

dimana A adalah matriks koefisien kendala, x adalah vektor variabel, b adalah vektor nilai batas kendala, dan c adalah vektor koefisien fungsi tujuan.

---

**Contoh 1**

Sebuah pabrik menyusun dua alat berbeda untuk mengurangi emisi gas $CO_2$ dan $SO_2$. Alat pertama dengan harga \$5 mampu mengurangi 1 unit gas $CO_2$ dan 1 unit gas $SO_2$. Alat kedua dengan harga \$8 mampu mengurangi $SO_2$ dengan jumlah yang sama seperti alat pertama dan mengurangi dua kali lebih banyak $CO_2$. Pabrik diwajibkan untuk mengurangi emisi paling tidak sebanyak 10 unit $CO_2$ dan 14 unit $SO_2$. Tentukan biaya optimal yang harus dikeluarkan oleh pabrik!

a. Variabel Keputusan
   Dalam kasus ini, ingin diputuskan banyaknya alat yang harus dibeli perusahaan. Misalkan $x_1$ adalah banyaknya alat pertama yang digunakan dan $x_2$ adalah banyaknya alat kedua yang digunakan.

b. Fungsi Tujuan
   Tujuan yang diharapkan adalah meminimumkan biaya pembelian alat.
   $$min: C = 5x_1 + 8x_2$$
   Koefisien variabel di atas diperoleh dengan mengingat bahwa harga $x_1$ adalah \$5 per unit dan $x_2$ adalah \$8 per unit.

c. Kendala
   Kendala yang pertama adalah pabrik diwajibkan mengurangi emisi $CO_2$ sekurang-kurangnya 10 unit. Telah diketahui bahwa alat pertama dan kedua mampu mengurangi jumlah $CO_2$ yang sama, yakni 1 unit. Dengan demikian,
   $$x_1 + x_2 \geq 10$$

Kendala yang kedua adalah pabrik diwajibkan mengurangi emisi $SO_2$ sekurang-kurangnya 14 unit. Telah diketahui bahwa alat pertama mampu mengurangi 1 unit $SO_2$ dan alat kedua mampu mengurangi 2 unit $SO_2$. Dengan demikian,

$$x_1 + 2x_2 \geq 14$$

Kendala yang terakhir adalah kendala non-negatif,

$$x_1 \geq 0, \;\; x_2 \geq 0$$

Model PL yang diperoleh dapat dituliskan dalam bentuk kanonik sebagai

Minimumkan $C = 5x_1 + 8x_2$

dengan kendala

$-x_1 - x_2 \leq -10$

$-x_1 - 2x_2 \leq -14$

$x_1, x_2 \geq 0$

atau dalam bentuk matriks,

$$A = \begin{pmatrix} -1 & -1 \\ -1 & -2 \end{pmatrix}, \;\; x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \qquad b = \begin{pmatrix} -10 \\ -14 \end{pmatrix}, \qquad C = \begin{pmatrix} 5 \\ 8 \end{pmatrix}$$

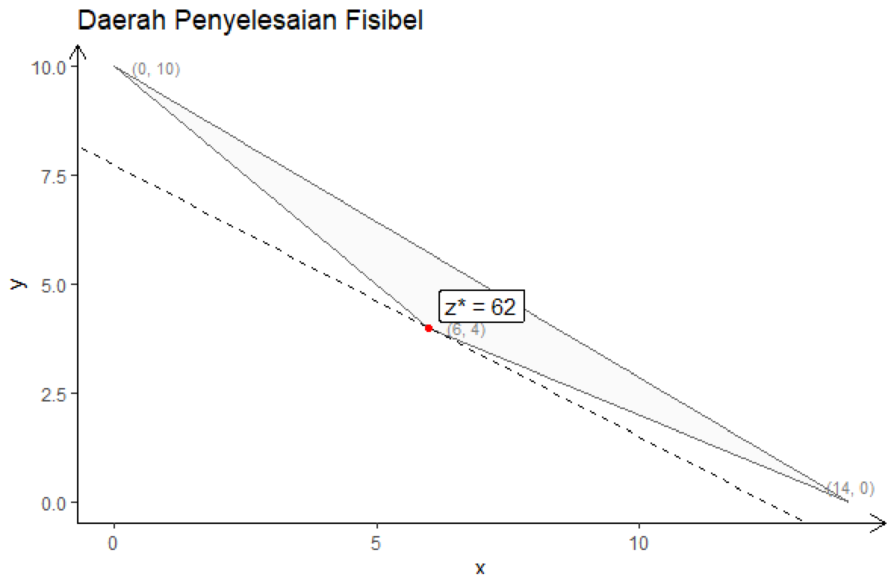
Tujuan kita adalah ingin menentukan nilai $x_1$ dan $x_2$ yang dapat meminimumkan *C* dan memenuhi semua kendala. Untuk menampilkan daerah fisibel (daerah penyelesaian yang memenuhi semua kendala), digunakan fungsi plotPolytope(...) dalam library gMOIP.

Definisikan matriks A, vektor b, dan vektor *C* terlebih dahulu. Kemudian, jalankan fungsi plotPolytope sebagai berikut (sesuaikan argumen crit dengan fungsi tujuan yakni “max” atau “min”).

```
#Plot daerah fisibel
require(gMOIP)
A = matrix(c(-1,-1,
        -1,-2), nrow = 2, byrow = TRUE)
b = c(-10, -14)
C = c(5, 8)

plotPolytope(
  A,
  b,
  C,
  type = rep("c", ncol(A)),
  crit = "min",
  faces = rep("c", ncol(A)),
  plotFaces = TRUE,
  plotFeasible = TRUE,
```

```
  plotOptimum = TRUE,
  labels = "coord"
) + ggplot2::xlab("x") + ggplot2::ylab("y") + ggplot2::ggtitle("Daerah Penyelesaian Fisibel")
```




Semua titik-titik $(x_1, x_2)$ dalam area di atas mencakup semua kombinasi yang memenuhi semua kendala. Nilai $(x_1, x_2)$ yang meminimumkan fungsi C adalah $(6, 4)$. Nilai C (biaya) yang minimum adalah sebesar 62 (ditandai dengan titik berwarna merah). Artinya, pabrik tersebut harus membeli sebanyak 6 alat pertama dan 4 alat kedua untuk mengurangi emisi CO2 dan SO2 dengan harga minimum, yakni $62.

Garis putus-putus menunjukkan garis *isocost*. Garis ini berisi seluruh kombinasi $(x_1, x_2)$ yang dapat meminimumkan fungsi C. Pada permasalahan memaksimumkan, garis ini disebut *isoprofit*.

---

**Contoh 2**

Sebuah penjara sedang memikirkan kombinasi makanan yang harus diberikan pada tahanannya. Makanan yang diberikan adalah kombinasi dari susu, kacang, dan apel. Penjara tersebut ingin meminimumkan biaya yang harus dikeluarkan sekaligus memenuhi persyaratan gizi minimum yang ditetapkan pemerintah. Biaya, kandungan gizi, dan kebutuhan gizi harian ditunjukkan pada tabel berikut.

| | Susu | Kacang | Apel | Persyaratan Gizi Minimum |
|---|---|---|---|---|
| Niacin | 3.2 | 4.9 | 0.8 | 13.0 |
| Thiamin | 1.12 | 1.3 | 0.19 | 1.5 |
| Vitamin A | 32.0 | 0.0 | 93.0 | 45.0 |
| Biaya | 2.00 | 0.20 | 0.25 | |

a. Variabel Keputusan

Hal yang ingin diputuskan adalah banyaknya susu, kacang, dan apel yang harus dibeli. Dimisalkan

$$x_1: banyaknya\ susu\ yang\ dibeli$$
$$x_2: banyaknya\ kacang\ yang\ dibeli$$
$$x_3: banyaknya\ apel\ yang\ dibeli$$

b. Fungsi Tujuan

Ingin diminimumkan biaya pembelian makanan.

$$min: C = 2x_1 + 0.2x_2 + 0.25x_3$$

Koefisien di atas diperoleh dari biaya masing-masing jenis makanan (susu seharga \$2.00, kacang seharga \$0.20, dan apel seharga \$0.25.

c. Kendala

Kendala yang ada adalah setiap kandungan gizi (Niacin, Thiamin, dan Vit A) harus mencapai syarat minimum.

| | |
|---|---|
| $3.2x_1 + 4.9x_2 + 0.8x_3 \geq 13$ | (syarat Niacin) |
| $1.12x_1 + 1.3x_2 + 0.19x_3 \geq 1.5$ | (syarat Thiamin) |
| $32x_1 + 93x_3 \geq 45$ | (syarat Vit A) |
| $x_1, x_2, x_3 \geq 0$ | (non-negatif) |

Model di atas sudah dalam bentuk kanonik, dengan

$$A = \begin{pmatrix} 3.2 & 4.9 & 0.8 \\ 1.12 & 1.3 & 0.19 \\ 32 & 0 & 93 \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, b = \begin{pmatrix} 13 \\ 1.5 \\ 45 \end{pmatrix}, C = \begin{pmatrix} 2 \\ 0.2 \\ 0.25 \end{pmatrix}$$

Perlu diperhatikan bahwa metode grafik (seperti pada Contoh 1) hanya berlaku untuk masalah PL dengan 2 variabel. Apabila ditemukan 3 variabel / lebih, digunakan metode lainnya. Dalam contoh ini, digunakan fungsi lp(...) dalam library lpSolve. Metode ini tidak mengharuskan kendala dituliskan dalam bentuk kanonik.

Sama seperti sebelumnya, definisikan matriks A, vektor b, dan C. Selanjutnya, definisikan vektor tanda yang berisi tanda pertidaksamaan (atau persamaan) dari masing-masing kendala. Atur argumen “min” atau “max” dalam fungsi lp, kemudian tampilkan hasilnya.

```
#LPSolve
require(lpSolve)
A = matrix(c(3.2, 4.9, 0.8,
        1.12, 1.3, 0.19,
        32, 0, 93), nrow = 3, byrow = TRUE)
b = c(13, 1.5, 45)
tanda = c(">=", ">=", ">=")
C = c(2, 0.2, 0.25)
```

```
solusi = lp("min", C, A, tanda, b)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

```
> solusi$objval #Nilai optimal fungsi C
[1] 0.6357801
> solusi$solution #Nilai x yang mengoptimalkan
[1] 0.000000 2.574062 0.483871
```

Artinya, biaya minimum untuk membeli makanan bagi tahanan adalah sebesar $0.6357801, dengan kacang sebanyak 2.574062 unit, apel sebanyak 0.483871 unit, dan susu sebanyak 0 unit.

Dalam kasus ini, kita tidak perlu membulatkan hasil yang diperoleh ke dalam bilangan bulat (meskipun makanan tidak masuk akal jika berjumlah desimal). Untuk memperoleh solusi dalam bentuk bilangan bulat akan dibahas pada subbab selanjutnya (*Integer Programming*).

---

Terdapat beberapa kasus khusus yang mungkin ditemui dalam masalah PL.

a. Tidak ada penyelesaian fisibel (*Infeasibility*)
   Kasus ini ditemui ketika semua kendala tidak dapat dipenuhi. Oleh karenanya, tidak ada penyelesaian yang mungkin.

   Contoh:
   Maksimumkan $C = 5x_1 + 8x_2$
   dengan kendala
   $x_1 + x_2 \geq 2$
   $x_1 + x_2 \leq 1$
   $x_1, x_2 \geq 0$

```
#LPSolve
require(lpSolve)
A = matrix(c(1, 1,
             1, 1), nrow = 2, byrow = TRUE)
b = c(2,1)
tanda = c(">=", "<=")
C = c(5, 8)

solusi = lp("max", C, A, tanda, b)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

```
> solusi$objval #Nilai optimal fungsi C
[1] 0
> solusi$solution #Nilai x yang mengoptimalkan
[1] 0 0
```

b. Penyelesaian optimal tak terbatas (*Unboundedness*)
Kasus ini ditemui ketika daerah fisibel tidak dibatasi sehingga nilai fungsi objektif akan menuju tak hingga.

Contoh:
Maksimumkan $C = 5x_1 + 8x_2$
dengan kendala

$$x_1 + x_2 \geq 2$$
$$x_1 + 2x_2 \geq 3$$
$$x_1, x_2 \geq 0$$

```
#LPSolve
require(lpSolve)
A = matrix(c(1, 1,
        1, 2), nrow = 2, byrow = TRUE)
b = c(2,3)
tanda = c(">=", ">=")
C = c(5, 8)

solusi = lp("max", C, A, tanda, b)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

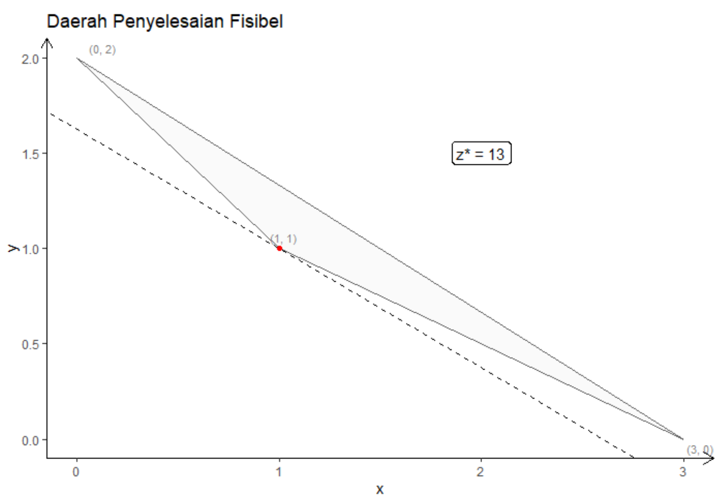
Pada contoh ini, nilai $x_1$ dan $x_2$ yang dapat memaksimumkan C akan menuju tak hingga. Akan tetapi, jika fungsi tujuan dirubah menjadi meminimumkan C, maka solusi optimal dapat ditemukan.

```
#LPSolve
require(lpSolve)
A = matrix(c(1, 1,
        1, 2), nrow = 2, byrow = TRUE)
b = c(2,3)
tanda = c(">=", ">=")
C = c(5, 8)

solusi = lp("min", C, A, tanda, b)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

```
#atau dengan gMOIP
A = matrix(c(-1, -1,
             -1, -2), nrow = 2, byrow = TRUE)
b = c(-2,-3)
C = c(5, 8)

plotPolytope(
  A,
  b,
  C,
  type = rep("c", ncol(A)),
  crit = "min",
  faces = rep("c", ncol(A)),
  plotFaces = TRUE,
  plotFeasible = TRUE,
  plotOptimum = TRUE,
  labels = "coord"
) + ggplot2::xlab("x") + ggplot2::ylab("y") + ggplot2::ggtitle("Daerah Penyelesaian Fisibel")
```




c. (Primal) Degeneracy

Kasus ini ditemukan dalam masalah dengan *m* variabel, dimana terdapat *m* kendala yang saling memotong di satu titik (atau secara mudahnya, terdapat kendala yang 'berlebih').

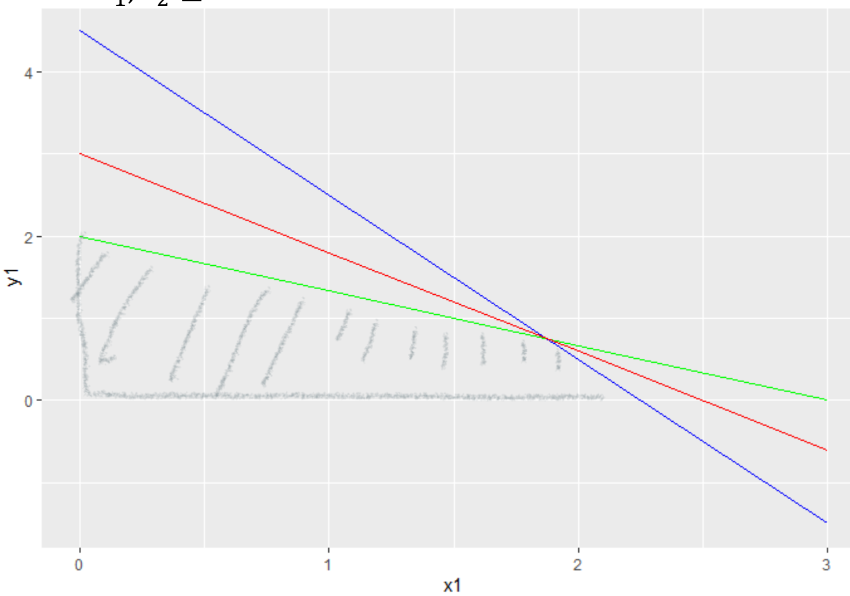
Contoh:

Maksimumkan $C = 2x_1 + 6x_2$

dengan kendala

$$4x_1 + 2x_2 \leq 9$$
$$2x_1 + 3x_2 \leq 6$$

$$6x_1 + 5x_2 \leq 15$$
$$x_1, x_2 \geq 0$$




Daerah fisibel ditandai dengan area yang diarsir. Pada contoh ini, terlihat kendala dengan garis merah tidak diperlukan.

d. Ada penyelesaian optimal alternatif (Multiple Optima / Dual Degeneracy)
Kasus ini terjadi ketika terdapat beberapa solusi yang optimal.

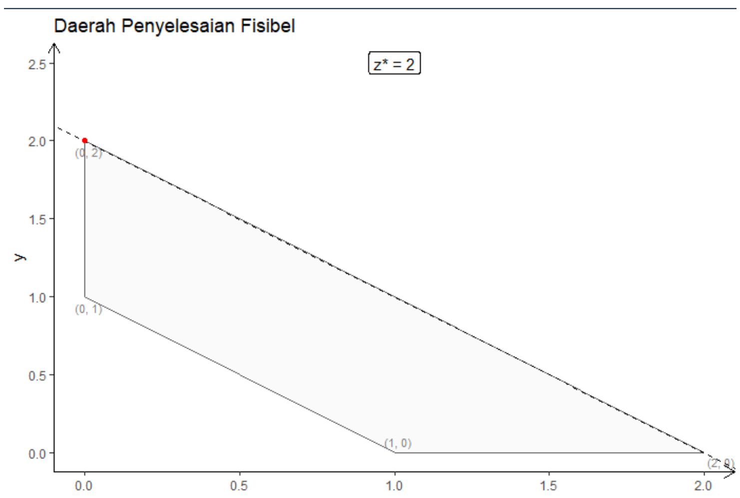
Contoh:
Maksimumkan $C = x_1 + x_2$
dengan kendala

$$x_1 + x_2 \geq 1$$
$$x_1 + x_2 \leq 2$$
$$x_1, x_2 \geq 0$$

```
require(gMOIP)
A = matrix(c(-1, -1,
             1, 1), nrow = 2, byrow = TRUE)
b = c(-1, 2)
C = c(1, 1)

plotPolytope(
  A,
  b,
  C,
  type = rep("c", ncol(A)),
  crit = "max",
  faces = rep("c", ncol(A)),
  plotFaces = TRUE,
  plotFeasible = TRUE,
```

```
plotOptimum = TRUE,
labels = "coord"
) + ggplot2::xlab("x") + ggplot2::ylab("y") + ggplot2::ggtitle("Daerah Penyelesaian Fisibel")
```




Perhatikan bahwa garis putus-putus atau garis *isoprofit* merupakan semua kombinasi $x_1, x_2$) yang memaksimumkan C. Nilai maksimum diperoleh pada $(x_1 = 0, x_2 = 2)$ yang ditandai titik merah. Lebih lanjut, $(x_1 = 2, x_2 = 0)$ juga memaksimumkan C, dan seluruh titik yang berada di garis yang menghubungkan keduanya.

## Unrestricted Variables (urs)

Sebelumnya kita selalu menggunakan *non-negativity* bagi setiap variabel. Ada kalanya suatu variabel dapat bernilai negatif. Secara *default*, fungsi lp(...) tidak bisa mengerjakan permasalahan non-negatif. Oleh karenanya, kita lakukan transformasi sederhana. Jika suatu variabel $x$ adalah *urs* (unrestricted in sign), maka $x$ dapat dituliskan sebagai $x = x_1 - x_2,\ dengan\ x_1 \geq 0,\ x_2 \geq 0$.

**Contoh 3**

$$Min: C = x_1 + 10x_2$$

dengan kendala

$$x_1 + x_2 \geq 2$$

$$x_1 - x_2 \leq 3$$

$$x_1 \geq 0,\ x_2\ urs$$

Dimisalkan $x_2 = x_3 - x_4$, maka permasalahan PL menjadi

$$Min: C = x_1 + 10x_3 - 10x_4$$

dengan kendala

$$x_1 + x_3 - x_4 \geq 2$$

$$x_1 - x_3 + x_4 \leq 3$$

$$x_1 \geq 0, x_3 \geq 0, x_4 \geq 0$$

```
#LPSolve
require(lpSolve)
A = matrix(c(1, 1, -1,
         1, -1, 1), nrow = 2, byrow = TRUE)
b = c(2, 3)
tanda = c(">=", "<=")
C = c(1, 10, -10)

solusi = lp("min", C, A, tanda, b)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

```
> solusi$objval #Nilai optimal fungsi C
[1] -2.5
> solusi$solution #Nilai x yang mengoptimalkan
[1] 2.5 0.0 0.5
```

Diperoleh $x_1 = 2.5, x_3 = 0, x_4 = 0.5, maka\ x_2 = -0.5$.

## Integer Programming

Seringkali nilai variabel keputusan harus dalam bentuk bilangan bulat. Misal $x_1$ adalah banyaknya truk, dan $x_2$ adalah banyaknya bus. Tidak masuk akal jika kita memeroleh solusi optimal dengan banyaknya truk dan bus berbentuk pecahan. Pembulatan sederhana juga seringkali tidak baik untuk dilakukan karena bisa mengakibatkan nilai fungsi objektif yang jauh berbeda.

Permasalahan seperti ini dinamakan integer programming. Fungsi lp(...) memiliki argumen bernama int.vec yang dapat digunakan untuk menentukan variabel mana yang harus bernilai bulat.

**Contoh 4**

$$Min: C = 2x_1 + 3x_2 + 4x_3 - x_4$$

dengan kendala

$$x_1 + 2x_2 \geq 9$$

$$3x_2 + x_3 \geq 9$$

$$x_2 + x_4 \leq 10$$

$$x_i \geq \forall i$$

$x_2, x_4 \in \mathbb{Z}$

Dalam masalah ini, nilai $x_2, x_4$ harus bulat. Sama seperti sebelumnya, definisikan matriks A, vektor b, C, vektor tanda, dan atur argumen “min” atau “max”. Kemudian, definisikan vektor bulat yang berisi indeks variabel yang harus bulat. Sebagai contoh, $x_2$ dan $x_4$ harus bulat, maka indeksnya adalah 2 dan 4.

```
#LPSolve
require(lpSolve)
A = matrix(c(1, 2, 0, 0,
            0, 3, 1, 0,
            0, 1, 0, 1), nrow = 3, byrow = TRUE)
b = c(9, 9, 10)
tanda = c(">=", ">=", "<=")
bulat = c(2, 4)
C = c(2, 3, 4, -1)

solusi = lp("min", C, A, tanda, b, int.vec = bulat)
solusi$objval #Nilai optimal fungsi C
solusi$solution #Nilai x yang mengoptimalkan
```

```
> solusi$objval #Nilai optimal fungsi C
[1] 8
> solusi$solution #Nilai x yang mengoptimalkan
[1] 1 4 0 6
```

Didapatkan solusi $x_1 = 1, x_2 = 4, x_3 = 0, dan\ x_4 = 6$. Nilai optimal (minimum) C adalah 8.

## Catatan Penutup

- Fungsi R lainnya yang dapat digunakan adalah solveLP(...) dalam library linprog.
- Model programming lainnya yang dapat ditelaah lebih lanjut: Quadratic Programming, Masalah Dualitas.
- Buku referensi utama yang digunakan:
  A First Course in Statistical Programming with R – W. John Braun & Duncan J. Murdoch (2007)
- Referensi lainnya:
  Modeling and Solving Linear Programming wirh R – Jose M. Sallan, Oriol Lordan, Vicenc Fernandez (2015).

Latihan Soal

a. Sebuah pesawat yang dimiliki suatu maskapai dapat mengangkut 200 penumpang. Setiap tiket eksekutif memberikan keuntungan sebesar \$1000 dan tiket ekonomi memberikan keuntungan sebesar \$600. Maskapai tersebut menyediakan sekurang-kurangnya 20 tiket eksekutif, akan tetapi, jumlah penumpang ekonomi setidaknya 4 kali lebih banyak daripada penumpang eksekutif. Tuliskan model PL untuk memaksimumkan keuntungan maskapai tersebut!

b. Minimumkan:

$$C = x_1 + x_2$$

dengan kendala

$$-x_1 + x_2 \leq 2$$
$$x_1 + 2x_2 \geq 2$$
$$x_1, x_2 \geq 0$$

c. Maksimumkan:

$$C = x_1 + x_2$$

dengan kendala

$$8x_1 + 3x_2 \leq 24$$
$$x_1 + 7x_2 \leq 35$$
$$-x_1 + x_2 \leq 4$$
$$x_2 \geq -2$$
$$x_1, x_2 \geq 0$$

d. Sebuah toko mewajibkan semua karyawan bekerja 8 jam berturut-turut. Berapa banyak karyawan yang diperlukan untuk memenuhi semua *shift* kerja yang ada pada tabel di bawah ini?

| Periode Shift | Banyaknya Karyawan yang diperlukan |
|---|---|
| 12 a.m. – 4 a.m. | 5 |
| 4 a.m. – 8 a.m. | 7 |
| 8 a.m. – 12 p.m. | 15 |
| 12 p.m. – 4 p.m. | 8 |
| 4 p.m. – 8 p.m. | 12 |
| 8 p.m. – 12 a.m. | 9 |

e. The coach of a swim team needs to assign swimmers to a 200-yard medley relay team (four swimmers, each swim 50 yards of one of the four strokes). Since most of the best swimmers are very fast in more than one stroke, it is not clear which swimmer should be assigned to each of the four strokes. The five fastest swimmers and their best times (in seconds) they have achieved in each of the strokes (for 50 yards) are shown in the table below. How should the swimmers be assigned to make the fastest relay team?

| | Backstroke | Breaststroke | Butterfly | Freestyle |
|---|---|---|---|---|
| Carl | 37.7 | 43.4 | 33.3 | 29.2 |
| Chris | 32.9 | 33.1 | 28.5 | 26.4 |
| David | 33.8 | 42.2 | 38.9 | 29.6 |
| Tony | 37.0 | 34.7 | 30.4 | 28.5 |
| Ken | 34.4 | 41.8 | 32.8 | 31.1 |

f. Maksimumkan

$$C = 6x_1 - 3x_2$$

dengan kendala

$$x_1 + x_2 \leq 1$$
$$2x_1 - x_2 \leq 1$$
$$-x_1 + 2x_2 \leq 1$$
$$x_1, x_2 \geq 0$$

Selidiki juga kasus khususnya (apabila ada).

g. Minimumkan

$$C = x_1 + 3x_2 + 4x_3 + x_4$$

dengan kendala

$$x_1 - 2x_2 \geq 9$$
$$3x_2 + x_3 \geq 9$$
$$x_2 + x_4 \geq 10$$
$$x_1, x_2, x_3, x_4 \geq 0$$

Bandingkan solusi di atas dengan solusi ketika ada salah satu variabel yang bulat! Apakah ada perbedaan diantara solusi yang ada?

Misalkan fungsi C diubah menjadi $C = x_1 - 3x_2 + 4x_3 + x_4$, apa yang terjadi pada solusinya?