

UPDATE PROGRESS PENELITIAN MANDIRI DALAM SAINS KOMPUTASI

Contoh Kasus Binary Programming|
Membuat Jadwal Tatap Muka Terbatas di Sekolah

Mohammad Rizka Fadhli
NIM: 20921004

15 October 2021

PENDAHULUAN

Pada minggu ini saya akan memberikan *update progress* penelitian mandiri dalam sains komputasi I berupa contoh penggunaan *binary programming* dalam memecahkan masalah *real*. Sebagaimana yang telah dijelaskan pada presentasi sebelumnya, *binary programming* erat kaitannya dengan permasalahan *matching* antara suatu hal (kita akan sebut sebagai indeks) dengan hal (indeks) lainnya. Saya akan berikan contoh terkait penjadwalan **Pembelajaran Tatap Muka Terbatas** (PTMT) yang terjadi di suatu sekolah dasar.

MASALAH

Suatu sekolah memiliki kelas berisi 20 orang siswa. Mereka hendak menggelar PTMT dengan aturan sebagai berikut:

1. PTMT digelar dari Senin hingga Jumat (5 hari).
2. Dalam sehari, siswa yang boleh hadir dibatasi 4-8 orang saja.
3. Dalam seminggu, diharapkan siswa bisa hadir 2-3 kali.
4. Siswa yang hadir di selang sehari baru bisa hadir kembali.

Saat membuat jadwal berdasarkan aturan di atas, tiba-tiba beberapa orang tua murid menelepon sang guru. Ternyata didapatkan:

1. 3 orang siswa hanya bisa hadir di **Senin dan Jumat**.
2. 2 orang siswa ingin hadir di **Rabu** dan hari lainnya bebas.

Berdasarkan semua kondisi yang ada ini, sang guru harus membuat jadwal kunjungan siswa.

Dari uraian di atas, kita bisa membuat model optimisasinya sebagai berikut:

Saya definisikan $x_{i,j} \in (0, 1)$ sebagai bilangan biner di mana $i \in \{1, 2, \dots, 20\}$ menandakan siswa dan $j \in \{1, 2, \dots, 5\}$ menandakan hari. Berlaku:

$$x_{i,j} = \begin{cases} 0, & \text{siswa } i \text{ tidak masuk di hari } j \\ 1, & \text{siswa } i \text{ masuk di hari } j \end{cases}$$

Objective Function

Tujuan utama kita adalah memaksimalkan siswa yang hadir.

$$\max \sum_{j=1}^5 \sum_{i=1}^{20} x_{i,j}$$

Constraints

Dalam sehari, ada pembatasan jumlah siswa yang hadir.

$$4 \leq \sum_i x_{i,j} \leq 8, j \in \{1, 2, \dots, 5\}$$

Dalam seminggu, siswa hadir dalam frekuensi tertentu.

$$2 \leq \sum_j x_{i,j} \leq 3, i \in \{1, 2, \dots, 20\}$$

Ada jeda sehari agar siswa bisa masuk kembali.

$$x_{i,j} + x_{i,j+1} \leq 1$$

Ada 3 orang yang hanya bisa masuk di hari Senin dan Jumat.

$$x_{i,1} + x_{i,5} = 2, i = 1, 2, 3$$

Ada 2 orang yang masuk di hari Rabu sedangkan hari lainnya bebas.

$$x_{i,3} = 1, i = 4, 5$$

Jangan lupa bahwa $x_{i,j} \geq 0$.

Sekarang kita akan coba selesaikan model di atas dengan menggunakan `library(ompr)`. Berikut adalah skrip di **R**-nya:

```

rm(list=ls())

library(dplyr)
library(ompr)
library(ompr.roi)
library(ROI.plugin.glpk)

bin_prog =
  MIPModel() %>%
  # menambah variabel
  add_variable(x[i,j],
    i = 1:20,
    j = 1:5,
    type = "binary",
    lb = 0) %>%
  # membuat objective function
  set_objective(sum_expr(x[i,j],
    i = 1:20,
    j = 1:5),
    "max") %>%
  # menambah constraints
  # max kapasitas kelas
  add_constraint(sum_expr(x[i,j], i = 1:20) >= 4,
    j = 1:5) %>%
  add_constraint(sum_expr(x[i,j], i = 1:20) <= 8,
    j = 1:5) %>%
  # frek kunjungan siswa
  add_constraint(sum_expr(x[i,j], j = 1:5) >= 2,
    i = 1:20) %>%
  add_constraint(sum_expr(x[i,j], j = 1:5) <= 3,
    i = 1:20) %>%
  # jeda sehari
  add_constraint(x[i,j] + x[i,j+1] <= 1,
    i = 1:20,
    j = 1:4) %>%
  # 3 anak di senin dan jumat
  add_constraint(x[i,1] + x[i,5] == 2,
    i = 1:3) %>%
  # 2 anak di hari rabu
  add_constraint(x[i,3] == 1,
    i = 4:5)

bin_prog

```

```
## Mixed integer linear optimization problem
## Variables:
##   Continuous: 0
##   Integer: 0
##   Binary: 100
## Model sense: maximize
## Constraints: 135

hasil =
  bin_prog %>%
  solve_model(with_ROI(solver = "glpk",
                       verbose = T))

## <SOLVER MSG> ----
## GLPK Simplex Optimizer, v4.65
## 135 rows, 100 columns, 568 non-zeros
##      0: obj = -0.000000000e+00 inf =  6.800e+01 (30)
##     77: obj =  4.000000000e+01 inf =  0.000e+00 (0)
##    * 78: obj =  4.000000000e+01 inf =  0.000e+00 (0)
## OPTIMAL LP SOLUTION FOUND
## GLPK Integer Optimizer, v4.65
## 135 rows, 100 columns, 568 non-zeros
## 100 integer variables, all of which are binary
## Integer optimization begins...
## Long-step dual simplex will be used
## + 78: mip =      not found yet <=          +inf      (1; 0)
## + 78: >>>>  4.000000000e+01 <=  4.000000000e+01  0.0% (1; 0)
## + 78: mip =  4.000000000e+01 <=      tree is empty  0.0% (0; 1)
## INTEGER OPTIMAL SOLUTION FOUND
## <!SOLVER MSG> ----
```

Berikut adalah hasilnya:

Jadwal Kunjungan Siswa

Berikut adalah jadwal kunjungan siswa perhari:

```
rekap =
  hasil %>%
  get_solution(x[i,j]) %>%
  filter(value == 1) %>%
  rename(siswa = i,
         hari = j)

rekap %>%
  group_by(hari) %>%
  summarise(presensi = paste(siswa,collapse = ",")) %>%
```

```
ungroup() %>%
knitr::kable("simple",caption = "Jadwal Kunjungan Siswa")
```

Table 1: Jadwal Kunjungan Siswa

hari	presensi
1	1,2,3,10,11,12,13,20
2	6,7,8,9,14,15,16,17
3	4,5,10,11,12,13,18,19
4	6,7,8,9,14,15,16,17
5	1,2,3,4,5,18,19,20

Rekap Presensi Siswa

Berikut adalah rekap frekuensi kedatangan siswa selama seminggu:

```
rekap %>%
  group_by(siswa) %>%
  tally() %>%
  ungroup() %>%
  rename("jumlah kehadiran" = n) %>%
knitr::kable("simple",caption = "Rekap Presensi Siswa")
```

Table 2: Rekap Presensi Siswa

siswa	jumlah kehadiran
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	2
17	2
18	2

siswa	jumlah kehadiran
19	2
20	2

Ternyata semua siswa mendapatkan kesempatan yang sama untuk bisa hadir ke sekolah.

Dari model di atas, kita bisa melakukan modifikasi jikalau ada peraturan baru yang akan diterapkan di sekolah tersebut atau ada kondisi lain yang terjadi di beberapa orang siswa tertentu.

Selain itu kita bisa melakukan *sensitivity test* dengan cara mengubah-ubah parameter yang ada sehingga bisa mendapatkan gambaran tentang sistem yang ada. Apakah ada perubahan solusi saat ada perubahan nilai parameter?