

FIT5216: Modelling Discrete Optimization Problems

Assignment 2: Testing Vaccines

1 Overview

For this assignment, your task is to write a MiniZinc model for a given problem specification.

- Submit your work to the MiniZinc auto grading system (using the submit button in the MiniZinc IDE). **You must submit via the IDE to be graded and receive marks.**
- Submit your model (copy and paste the contents of the .mzn file) using the Moodle assignment.

You have to submit by the due date (20th April 2020, 11:59pm), using MiniZinc and using the Moodle assignment, to receive full marks. You can submit as often as you want before the due date. Late submissions without special consideration receive a penalty of 10% per day. Submissions are not accepted more than 3 days after the original deadline.

This is an **individual assignment**. Your submission has to be **entirely your own work**. We will use similarity detection software to detect any attempt at collusion, and the **penalties are quite harsh**. If in doubt, contact your teaching team with any questions!

2 Problem Statement

In the race to find an effective vaccine for COVID-19 the World Health Organization is trialing different potential vaccines developed by different medical research groups around the world¹. The vaccines are given as an enumerated type:

```
enum VACCINE;
```

To determine the effectiveness of each vaccine they need to trial it across many groups of test participants with many different features.

The participants are defined by features

```
enum AGE = { BABY, CHILD, YOUTH, ADULT, SENIOR, AGED, ANCIENT };
enum GENDER = { FEMALE, MALE, OTHER };
enum HEALTH = { GOOD, POOR, COMPROMISED };
enum EXPOSURE = { LOW, AVERAGE, HIGH, EXTREME };
```

The participants are divided into groups with similar features. There are m groups, and all of the participants in a group have the same features. The groups are not all the same size. The data is available as

¹This is a **fictional scenario**, and completely made up! It does not reflect any actual testing going on at the moment. But this type of problem, called an *experiment design*, is quite common in practice.

```

int: m; % number of groups
set of int: GROUP = 1..m;
array[GROUP] of AGE: age;           % age of the group
array[GROUP] of GENDER: gender;     % gender of the group
array[GROUP] of HEALTH: health;     % underlying health of the group
array[GROUP] of EXPOSURE: exposure; % exposure for the group
array[GROUP] of int: size;          % number of people in the group

```

The aim is to assign treatments to groups in order to gain the most possible knowledge of the effectiveness of different potential vaccines.

The decisions are then for each group, the set of vaccines to be applied to members of the group. Each participant will get only one vaccine, but we want to trial different vaccines for each group to get the most information in this experiment.

```
array[GROUP] of var set of VACCINE: x;
```

The constraints on the assignment are as follows:

1. In order for the tests to be reliable we need to replicate the test on at least `minsize` people in each group. So we can't assign a number of vaccines to a group bigger than the size dictates. For example, if `minsize=4`, and a group has 14 participants, we can assign at most 3 vaccines to the group (since $3 \times \text{minsize} \leq 14$, but $4 \times \text{minsize} > 14$).
2. To find the effect of vaccines on different ages the decision must balance vaccine applications across the age types. Since the number of groups of each age type are very different we give minimum `age_group_min` and maximum `age_group_max` for the each age type. Each vaccine must be applied to a number of groups for each age type within these bounds.
3. If a group is tested with k vaccines then the number of people tested for each vaccine is `size div k`, that is each vaccine is applied evenly. We want to balance the number of people trialled for each vaccine. For any pair of vaccines v_1 and v_2 , the difference between the total number of people trialled for v_1 and the total number of people trialled for v_2 cannot be more than `max_people_diff`;
4. To avoid any possibility that the trial is gender biased, for each gender type we must have exactly the same number of groups of that type for each vaccine.
5. To improve the statistical predictive ability of the trial we need to spread out the use of vaccines across different groups. For any two groups the maximum number of vaccines they can share is at most `max_share_vaccines`.

The data for these constraints is given by

```

int: minsize;
array[AGE] of int: age_group_min;
array[AGE] of int: age_group_max;
int: max_people_diff;
int: max_share_vaccines;

```

The objective of the trial is to discover the most information possible. The effectiveness of the vaccine will be tested most efficiently in participants with lower health and higher exposure, since here the statistical likelihood of the participant being critically affected by COVID-19 is higher.

The information gain for each vaccine in the trial is calculated by adding the information value for each group by its health and exposure. The data is given as

```
array[HEALTH] of int: health_information;
array[EXPOSURE] of int: exposure_information;
```

For a group g treated with vaccine v we gain information about that vaccine equal to the

$$health_information[g] \times exposure_information[g]$$

units. The objective is to maximise the minimum information gain over all vaccines.

For example a sample data set is

```
VACCINE = { ACOV, RETROV, ANTIFLU };
```

```
m = 10;
age = [BABY, CHILD, YOUTH, ADULT, AGED, YOUTH, BABY, ADULT, ADULT, ANCIENT];
gender = [MALE, FEMALE, OTHER, MALE, FEMALE, MALE, FEMALE, MALE, MALE, FEMALE];
health = [GOOD, POOR, GOOD, COMPROMISED, GOOD, GOOD, POOR, POOR, POOR, COMPROMISED];
exposure = [LOW, HIGH, LOW, AVERAGE, EXTREME, LOW, HIGH, AVERAGE, AVERAGE, AVERAGE];
size = [10, 35, 10, 100, 46, 60, 18, 145, 170, 12];

minsize = 9;
age_group_min = [0,0,0,1,0,0,0];
age_group_max = [1,1,1,2,1,1,1];
max_people_diff = 20;
max_share_vaccines = 2;

health_information = [1,2,4];
exposure_information = [1,2,5,7];
```

A solution (not necessarily the best) is given by

```
x = [{ANTIFLU}, {RETROV, ANTIFLU}, {}, {ANTIFLU}, {RETROV, ANTIFLU}, {ACOV, RETROV},
      {ACOV}, {ACOV, RETROV}, {ACOV, RETROV, ANTIFLU}, {ACOV}];
```

Let's examine the constraints.

The maximum number of vaccines applicable to the first, third and last group is 1 because of their size, and that holds. The maximum number for the 7th group is 2, that also holds. All the rest can have 3.

Each number of groups for each age type is given by

AGE	BABY	CHILD	YOUTH	ADULT	SENIOR	AGED	ANCIENT
ACOV	1	0	1	1	0	0	1
RETROV	0	1	0	2	0	1	0
ANTIFLU	1	1	0	2	0	1	0

Clearly the min and max constraints hold.

The total number of participants trialled with each vaccine is 188, 198 and 206 with the maximum difference only 18.

The number of groups trialled for each gender and vaccine is

GENDER	FEMALE	MALE	OTHER
ACOV	2	3	0
RETROV	2	3	0
ANTIFLU	2	3	0

so each vaccine is trialled on an equal number of groups for each gender.

The maximum number of vaccines shared by any group is 2, since there is only one group trialling all three vaccines.

The minimum information is for vaccine RETROV with 26 information units. The information gain for RETROV for each group is

GROUP	2	5	6	8	9
health	POOR	GOOD	GOOD	POOR	POOR
exposure	HIGH	EXTREME	LOW	AVERAGE	AVERAGE
info	10	7	1	4	4

Write a MiniZinc model `vaccine.mzn` which models the above problem.

3 Full Objective

To achieve full marks you need to tackle this extension. Grading of the solutions will be done based on this “true form” of the objective.

The aim is really to test the vaccine on as many different categories of health and exposure as possible. The true information gain is more complicated than discussed above.

For a group g treated with vaccine v we gain information about that vaccine equal to the $health_information[g] \times exposure_information[g]$ units. But we only gain this from the first group with this combination of HEALTH and EXPOSURE, for each other group treated by this vaccine with the same combination we simply get information of 1 unit. The objective is to maximise the minimum information gain over all vaccines.

For the solution above then minimum information is for vaccine RETROV with 23 information units. The information gain for RETROV for each group is

GROUP	2	5	6	8	9
health	POOR	GOOD	GOOD	POOR	POOR
exposure	HIGH	EXTREME	LOW	AVERAGE	AVERAGE
info	10	7	1	4*	4*

where the last two groups have the same combination so we replace one of the 4s by 1.

4 Instructions

Edit the provided `mzn` model files to solve the problems described above. You are provided with some sample data files to try your model on. Your implementations can be tested locally by using

the *Run* icon in the MINIZINC IDE or by using,

```
minizinc ./modelname.mzn ./datafile.dzn
```

at the command line.

5 Marking

The marks are automatically calculated.

The submission has 8 marks for locally tested data and 14 for model testing, for a total of 22 marks.