# Foundation Of Operations Research
*Exercises*

Christian Rossi

Academic Year 2023-2024

## Abstract

Operations Research is the branch of applied mathematics dealing with quantitative methods to analyze and solve complex real-world decision-making problems.

The course covers some of the fundamental concepts and methods of Operations Research pertaining to graph optimization, linear programming and integer linear programming.

The emphasis is on optimization models and efficient algorithms with a wide range of important applications in engineering and management.

# Contents

# Chapter 1

# Exercise session I

### Exercise 1

A bank has a capital of $C$ billions of Euro and two available stocks:

1. With an annual revenue of 15% and risk factor of $\frac{1}{3}$.
2. With an annual revenue of 25% and risk factor of 1.

The risk factor represents the maximum fraction of the stock value that can be lost. A risk factor of 25% implies that, if stocks are bought for 100 euro up to 25 euro can be lost. It is required that at least half of $C$ is risk-free. The amount of money used to buy stocks of two must not be larger than two times that used to buy stocks of one. At least $\frac{1}{6}$ of C must be invested into one.

Give a Linear Programming formulation for the problem of determining an optimal portfolio for which the profit is maximized. Solve the problem graphically.

### Answer of exercise 3

- The decision variable $x_i$ that is the capital invested in each stock ($i = 1, 2$).
- The objective function requires to maximize the expected income, so we have:

$$\max\left(0.15x_1 + 0.25x_2\right)$$

- The constraints are:

- Maximum capital:

$$x_1 + x_2 \leq C$$

- Half of the invested capital is risk-free:

$$\frac{1}{3}x_1 + 1x_2 \leq \frac{C}{2}$$

- The amount of money used to buy stocks of two must not be larger than two times that used to buy stocks of one:

$$x_2 \leq 2x_1$$

- At least $\frac{1}{6}$ of $C$ must be invested into one:

$$x_1 \geq \frac{1}{6}C$$

The region with feasible solutions is the one where all the planes intersects. To find the feasible point where the objective function attains its maximal value, we draw the level curves:

$$f(x_1, x_2) = 0.15x_1 + 0.25x_2 = z$$

that is the set of points whose objective function values is equal to $z$, for any $z$. When $z$ increases, we obtain parallel lines that move towards the direction of $\nabla f(x_1, x_2)$. The last feasible points having a nonempty intersection are the maximizers of $f$ over the feasible set. In this case, there is a single maximizer, $x^*$, at the intersection of lines one and two. So, with the linear system:

$$\begin{cases} x_1 + x_2 = C \\ \frac{1}{3}x_1 + 1x_2 = \frac{C}{2} \end{cases}$$

we obtain that $x^* = \left(\frac{3C}{4}, \frac{C}{4}\right)$, where $f(x^*) = \frac{7C}{40}$.

## Exercise 2

A refinery produces two types of gasoline, mixing three basic oils according to the following gasoline mixture rules:

| | **Oil 1** | **Oil 2** | **Oil 3** | **Revenue** |
|---|---|---|---|---|
| Gasoline A | $\leq 30\%$ | $\geq 40\%$ | - | 5.5 |
| Gasoline B | $\leq 40\%$ | $\geq 10\%$ | - | 4.5 |

The last column of the previous table indicates the profit (euro/barrel). The availability of each type of oil (in barrel) and the cost (euro/barrel) are as follows:

| **Oil** | **Availability** | **Cost** |
|---|---|---|
| 1 | 3 000 | 3 |
| 2 | 2 000 | 6 |
| 3 | 4 000 | 4 |

Give a Linear Programming formulation for the problem of determining a mixture that maximizes the profit (difference between revenues and costs).

### Answer of exercise 2

- Decision variables:
  - $x_{ij}$ is the amount of the $i$-th oil used to produce the $j$-th gasoline, $i \in \{1, 2, 3\}$ and $j \in \{A, B\}$.
  - $y_j$ is the amount of gasoline of type $j$-th that is produced, $j \in \{A, B\}$.
- The objective function needs to maximize the profit that is equal to:

$$\max 5.5y_A + 4.5y_B + 3(x_{1A} + x_{1B}) - 6(x_{2A} + x_{2B}) - 4(x_{3A} + x_{3B})$$

- The constraints are:
  - Availability of 1:

$$x_{1A} + x_{1B} \leq 3000$$

  - Availability of 2:

$$x_{2A} + x_{2B} \leq 2000$$

  - Availability of 3:

$$x_{3A} + x_{3B} \leq 4000$$

- Conservation of A:

$$y_A = x_{1A} + x_{2A} + x_{3A}$$

- Conservation of B:

$$y_B = x_{1B} + x_{2B} + x_{3B}$$

- Minimum quantity of $A$:

$$x_{1A} \leq 0.3y_A$$

- Minimum quantity of $B$:

$$x_{1B} \leq 0.5y_B$$

- Maximum quantity of $A$:

$$x_{2A} \geq 0.4y_A$$

- Maximum quantity of $B$:

$$x_{2B} \geq 0.1y_B$$

- The variable must be non-negative:

$$x_{1A}, x_{2A}, x_{3A}, x_{1B}, x_{2B}, x_{3B}, y_A, y_B \geq 0$$

It is possibly to substitute the variables $y_A$ and $y_B$ with the $x$ variables to have fewer variables.

# Chapter 2

# Laboratory session I

### Exercise 3

A canteen has to plan the composition of the meals that it provides. A meal can be composed of the types of food indicated in the following table. Costs, in Euro per hg, and availabilities, in hg, are also indicated.

| **Food** | **Cost** | **Availability** |
|---|---|---|
| Bread | 0.1 | 4 |
| Milk | 0.5 | 3 |
| Eggs | 0.12 | 1 |
| Meat | 0.9 | 2 |
| Cake | 1.3 | 2 |

A meal must contain at least the following amount of each nutrient:

| Nutrient | Minimal quantity |
|---|---|
| Calories | 600 cal |
| Proteins | 50 g |
| Calcium | 0.7 g |

Each hg of each type of food contains to following amount of nutrients:

| **Food** | **Calories** | **Proteins** | **Calcium** |
|---|---|---|---|
| Bread | 30 cal | 15 g | 0.02 g |
| Milk | 50 cal | 15 g | 0.15 g |
| Eggs | 150 cal | 30 g | 0.05 g |
| Meat | 180 cal | 90 g | 0.08 g |
| Cake | 400 cal | 70 g | 0.01 g |

Give a linear programming formulation for the problem of finding a meal of minimum total cost which satisfies the minimum nutrient requirements.

### Answer of exercise 3

```
# Only for Colab
!pip install mip
# We need to import the package mip
import mip
# Food
I = {’Bread’, ’Milk’, ’Eggs’, ’Meat’, ’Cake’}

# Nutrients
J = {’Calories’, ’Proteins’, ’Calcium’}

# Cost in Euro per hg of food
c = {’Bread’:0.1, ’Milk’:0.5, ’Eggs’:0.12, ’Meat’:0.9, ’Cake’:1.3}

# Availability per hg of food
q = {’Bread’:4, ’Milk’:3, ’Eggs’:1, ’Meat’:2, ’Cake’:2}

# minum nutrients
b = {’Calories’:600, ’Proteins’:50, ’Calcium’:0.7}

# Nutrients per hf of food
a = {  (’Bread’,’Calories’):30,
       (’Milk’,’Calories’):50,
       (’Eggs’,’Calories’):150,
       (’Meat’,’Calories’):180,
       (’Cake’,’Calories’):400,
       (’Bread’,’Proteins’):5,
       (’Milk’,’Proteins’):15,
       (’Eggs’,’Proteins’):30,
       (’Meat’,’Proteins’):90,
       (’Cake’,’Proteins’):70,
       (’Bread’,’Calcium’):0.02,
       (’Milk’,’Calcium’):0.15,
       (’Eggs’,’Calcium’):0.05,
       (’Meat’,’Calcium’):0.08,
       (’Cake’,’Calcium’):0.01}

# Define a empty model
model = mip.Model()
```

```
# Define variables
x = [model.add_var(name = i,lb=0) for i in I]

# Define the objective function
model.objective = mip.minimize(mip.xsum())

# Availability constraint
for i,food in enumerate(I):
  model.add_constr()

# Minum nutrients
for j in J:
  model.add_constr(mip.xsum()>=)

# Optimizing command
model.optimize()

# Optimal objective function value
model.objective.x

# Printing the variables values
for i in model.vars:
  print(i.name)
  print(i.x)
```