Optimization

Lecture 6.2

M.EIC – 2021.2022

Linear Programming

THE ASSIGNMENT PROBLEM

Assignment Problems

The Assignment problems may appear in several situations:

- Assignment of employees to tasks
- Assignment of operations to machines
- Assignment of projects to consultants
- Assignment of vehicles to trips
- Assignment of applicants to jobs
- ...

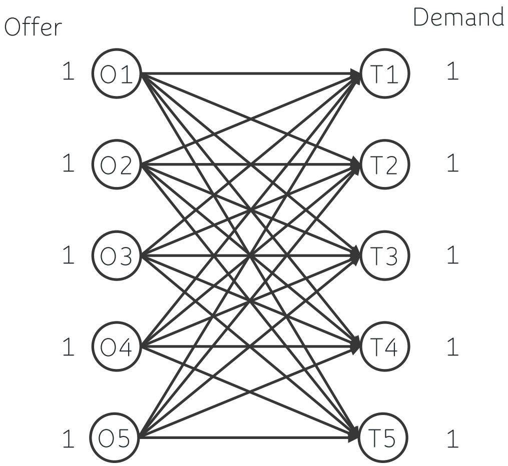
Basically, the classic assignment problem consists in assigning **n** individuals to **n** tasks (a single individual per task and a single task per individual) with the objective of minimizing the total assignment cost, being known the costs c_{ij}, corresponding to assigning the i-th individual (i=1,...n) to the j-the task (j = 1,...,n).

Although the classic assignment problem can be formulated as a special case of a transportation problem (and vice-versa), there is a particular algorithm, more efficient, for the assignment problems, known as the Hungarian Method.

Assignment Problems

Example: A company intends to manufacture a product, for which the manufacturing process involves five tasks. There are five workers available, and each task can be executed by any of the five workers. The execution times of each task for each of the five workers are presented in the following table.

	T1	T2	T3	T4	T5
O1	13	5	9	18	12
O2	13	19	6	13	14
O3	3	2	4	4	5
O4	18	9	13	20	16
O5	12	6	14	19	10



Decision variables: one variable for each arc:

$$X_{ij} = \begin{cases} 1, & \text{if worker i is assigned to task j} \\ 0, & \text{if worker i is not assigned to task j} \end{cases}$$

There is one single constraints for each node, guaranteeing that each worker is assigned to one single task or that each task is assigned to one single worker.

Assignment Problems

Formulation

$$\min\ tempotot = 13x_{11} + 5x_{12} + 9x_{13} + 18x_{14} + 12x_{15} + \\ 13x_{21} + 19x_{22} + 6x_{23} + 13x_{24} + 14x_{25} + \\ \ldots\ldots \\ 13x_{51} + 6x_{52} + 14x_{53} + 19x_{54} + 10x_{55}$$

$$s.a$$

$$\left.\begin{array}{llllllll} x_{11} & +x_{12} & +x_{13} & +x_{14} & +x_{15} & = & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{51} & +x_{52} & +x_{53} & +x_{54} & +x_{55} & = & 1 \end{array}\right\}$$

5 constraints imposing that each worker is totally used

$$\left.\begin{array}{llllllll} x_{11} & +x_{21} & +x_{31} & +x_{41} & +x_{51} & = & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{15} & +x_{25} & x_{35} & x_{45} & x_{55} & = & 1 \end{array}\right\}$$

5 constraints imposing that each task is completely executed

$$x_{ij} \geq 0,\ i = 1,\ldots,\ n;\ j = 1,\ldots,n$$

Generalization

$$\min z = \sum_i \sum_j c_{ij}\ x_{ij}$$

$$\text{s.a} \quad \sum_j x_{ij} = 1 \qquad (i = 1,\ldots, n)$$

$$\sum_i x_{ij} = 1 \qquad (j = 1,\ldots, n)$$

$$x_{ij} \geq 0, \qquad (i = 1,\ldots, n;\ j = 1,\ldots, n)$$

Formulation

$$\min z = \sum_i \sum_j c_{ij}\ x_{ij}$$

$$\text{s.a} \quad \sum_j x_{ij} = 1 \qquad (i = 1,\dots,n)$$

$$\sum_i x_{ij} = 1 \qquad (j = 1,\dots,n)$$

$$x_{ij} \geq 0, \qquad (i = 1,\dots,n;\ j = 1,\dots,n)$$

- The particular structure of the corresponding transportation problem, in which $a_i = 1$ (i=1,…n) and $b_j = 1$ (j= 1,…,n) imposes that the assignment of any individual to a task fulfills simultaneously the offer and demand constraints.
- Hence, any basic feasible solution is "highly degenerated" since from 2n-1 basic variables, only n are positive (the remaining n-1 are null).
- This fact hinders the application of the transportation algorithm ☹

The Hungarian Method, simpler and more efficient, is based on the following theorem:

Theorem: The optimal solution of the assignment problem does not change if a constant is added or subtracted from any row or column of the cost matrix (it only changes the value of objective function).

Let p_i (i=1,...,n) and q_j (j=1,...,n) the constants added or subtracted to the i-th row and j-th column of the cost matrix. The elements of the new cost matrix are: $c'_{ij} = c_{ij} \pm p_i \pm q_j$

The new objective function value is:

$$z' = \sum_i \sum_j c'_{ij} x_{ij} = \sum_i \sum_j \left(c_{ij} \pm p_i \pm q_j\right) x_{ij} = \sum_i \sum_j c_{ij} x_{ij} \pm \sum_i p_i \sum_j x_{ij} \pm \sum_j q_j \sum_i x_{ij} =$$
$$= z \pm \sum_i p_i \pm \sum_j q_j = z \pm \text{constant}$$

This shows that the z minimum differs from the z' minimum by a constant (independent of the variables), so the optimal solutions coincide.

Conclusion: The aim is to obtain a new assignment problem, equivalent to the original one, in which the costs (for the positive variables n) are zero.

Hungarian Method

1st step: For each row, select the element with the minimum cost and subtract it from all the elements in that row.

Initial matrix

	T1	T2	T3	T4
O1	2	6	7	4
O2	9	8	4	2
O3	2	7	5	7
O4	0	0	0	0

	T1	T2	T3	T4
O1	0	4	5	2
O2	7	6	2	0
O3	0	5	3	5
O4	0	0	0	0

Hungarian Method - Step 1 (cont.)

Verify if the solution is optimal:

(i) Find the minimum number of lines (horizontal or vertical), **n**, that cover all the zeros in the matrix. We can use the following heuristic:

- *Choose any row or column with a single zero;*
  - *- If you chose a row, draw a line passing through the column where the zero appears;*
  - *- If you chose a column, draw a line passing through the row where the zero appears.*
- *Continue until all the zeros have a line.*

(ii) If **n** is equal to the matrix dimension, the current solution is optimal;

(iii) If **n** is less than the matrix dimension, go to **Step 2**.

	T1	T2	T3	T4
O1	0	4	5	2
O2	7	6	2	0
O3	0	5	3	5
O4	0	0	0	0

n=3 < 4, hence the solution is not optimal

Hungarian Method – Step 2

2nd step: For each column, select the element with the minimum cost and subtract it from all the elements in that column.
Verify if the solution is optimal. If not, go to **Step 3**.

	T1	T2	T3	T4
O1	0	4	5	2
O2	7	6	2	0
O3	0	5	3	5
O4	0	0	0	0

→

	T1	T2	T3	T4
O1	0	4	5	2
O2	7	6	2	0
O3	0	5	3	5
O4	0	0	0	0

$n=3 < 4$, hence the solution is not optimal

Hungarian Method - Step 3

3rd step: Identify the minimum number that has no lines passing through it (**t**).

Update the cost matrix applying the following rules:

- Subtract **t** from the elements with no lines passing through them.
- The elements with one single line passing through them remain unchanged.
- Add **t** to the elements that have two lines passing through them.

Verify if the solution is optimal. If not repeat **Step 3**.

	T1	T2	T3	T4
O1	0	4	5	2
O2	7	6	2	0
O3	0	5	3	5
O4	0	0	0	0

Minimum number with no lines: t= 2

	T1	T2	T3	T4
O1	0	2	3	0
O2	9	6	2	0
O3	0	3	1	3
O4	2	0	0	0

n=3 < 4, the solution is not optimal

Hungarian Method - Step 3 (cont.)

	T1	T2	T3	T4
O1	0	2	3	0
O2	9	6	2	0
O3	0	3	1	3
O4	2	0	0	0

Minimum number with no lines: t= 1

	T1	T2	T3	T4
O1	0	1	2	0
O2	9	5	1	0
O3	0	2	0	3
O4	3	0	0	1

n=4 = 4, the solution is optimal

☺

Hungarian Method- Identifying the optimal solution

- Select a group of **n** zeros (being **n** the matrix dimension) in different rows and columns.
  At the end of the algorithm, it is always possible to find such a group. Otherwise, the number of lines passing through all the zeros would be less than the dimension of the matrix..
- The optimal solution is obtained making $x_{ij} = 1$ in the positions corresponding to the selected zeros and $x_{ij} = 0$ in the remaining positions..

	T1	T2	T3	T4
O1	0	1	2	0
O2	9	5	1	0
O3	0	2	0	2
O4	3	0	0	1

Optimal assignment:

Worker 1 $\longrightarrow$ Task 1
Worker 2 $\longrightarrow$ Task 4
Worker 3 $\longrightarrow$ Task 3
Worker 4 (dummy) $\longrightarrow$ Task 2

Hungarian Method- Identifying the optimal solution

- The solution obtained at the end of the Hungarian Method is optimal because:
  - ➢ The objective function value is zero;
  - ➢ It could not be less than zero because all the coefficients and all the variables are non-negative.
- Since the last matrix was obtained from the first one by adding and subtracting constant values to rows and columns, the optimal solution of the last matrix is equal to the optimal solution of the initial problem.

  To determine the objective function value (total time needed), we use the original matrix and add up the values for the selected assignment.

	T1	T2	T3	T4
O1	2	6	7	4
O2	9	8	4	2
O3	2	7	5	7
O4	0	0	0	0

Minimum total time = 2+2+5+0 = 9
Task 2 was not performed

The Hungarian Method is applicable only to minimization problems with an equal number of rows and columns.

Special cases that can be converted to assignment problems:

- The number of workers (columns) is different form the number of tasks (rows): create dummy rows or columns with null coefficients in the objective function.
- One of more workers (columns) cannot be assigned to some tasks (rows): assign a infinite value to the corresponding objective function value.
- One or more workers (columns) can be assigned to more than one task (row): for each worker create a number of rows equal to the number of tasks he can execute.
- There are negative coefficients in the objective function: Let -C be the most negative coefficient in the original problem: $-C = \min_{i,j}(c_{ij})$.

  Create a new matrix with coefficients: $c^*_{ij} = c_{ij} + C$
- Maximization problem (with positive coefficients): Let C be the highest value of the objective function for the original problem: $C = \max_{i,j}(c_{ij})$.

  Create a new matrix with coefficients $c^*_{ij} = C - c_{ij}$ and minimize the new problem

Bottleneck Assignment Problem

- This is a variant of the Assignment Problem in which the tasks are executed simultaneously.
- In this case, the objective is to minimize the time needed to execute the longest task (instead of minimizing the sum of times associated to the tasks).

1st Step: Choose arbitrarily a feasible assignment and determine the time (T) needed to execute the longest task.

Example: We can choose, for example, the optimal assignment of the previous problem.

	T1	T2	T3	T4
O1	2	6	7	4
O2	9	8	4	2
O3	2	7	5	7
O4	0	0	0	0

Time of the longest task in the chosen assignment: T=5

Bottleneck Assignment Problem (cont.)

2nd **Step**: Define a new matrix with the following elements:

$$\begin{cases} 0, \text{ if the corresponding element in the original elelment is less than T } (c_{ij} < T) \\ 1, \text{ otherwise } (c_{ij} >= T) \end{cases}$$

Old matrix

Time of the longest task in the chosen assignment: T=5

	T1	**T2**	**T3**	**T4**
O1	2	6	7	4
O2	9	8	4	2
O3	2	7	5	7
O4	0	0	0	0

New matrix

	T1	**T2**	**T3**	**T4**
O1	0	1	1	0
O2	1	1	0	0
O3	0	1	1	1
O4	0	0	0	0

Bottleneck Assignment Problem (cont.)

3rd **Step**: Determine **n**, the minimum number of lines (horizontal or vertical) covering all the zeros in the new matrix.

- If **n** is less than the dimension of the matrix, the solution is optimal.
- If **n** is equal to the dimension of the matrix, the solution is not optimal.
  Therefore, we can define a new set of zeros, all of them in different rows and columns, defining a new assignment in which all the times are inferior to the longest time of the previous solution.
- While the solution is not optimal repeat **steps 2 and 3**.

	T1	**T2**	**T3**	**T4**
O1	0	1	1	0
O2	1	1	0	0
O3	0	1	1	1
O4	0	0	0	0

n=4, the solution is not optimal

	T1	**T2**	**T3**	**T4**
O1	0	1	1	0
O2	1	1	0	0
O3	0	1	1	1
O4	0	0	0	0

	T1	T2	T3	T4
O1	2	6	7	4
O2	9	8	4	2
O3	2	7	5	7
O4	0	0	0	0

T=4

	T1	T2	T3	T4
O1	0	1	1	1
O2	1	1	1	0
O3	0	1	1	1
O4	0	0	0	0

n=3 < 4 , the solution is optimal

This assignment solution is optimal because it is not possible to define a new solution in which all the times are inferior to T=4.

Optimal solution:

Worker 1 → Task 4
Worker 2 → Task 3
Worker 3 → Task 1
Worker 4 (dummy) → Task 2