1 The assignment problem

We have $n$ workers and $n$ jobs. Each worker is assigned to exactly one job. There is a cost $c_{ij}$ for assigning worker $i$ to job $j$. What is the minimum cost assignment?

Formulate as a binary integer program:

$$\min_{x \in \mathbb{R}^{m \times n}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, m \text{ (each worker has one job)}$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad j = 1, \ldots, n \text{ (each job has one worker)}$$

$x_{ij}$ binary $i = 1, \ldots, m, j = 1, \ldots, n$

The linear optimization relaxation of this problem is obtained by replacing the restriction that each $x_{ij}$ be binary by the constraints $0 \leq x_{ij} \leq 1$. Further, the equality constraints force all $x_{ij} \leq 1$, so these upper bound constraints are redundant and can be omitted. This gives the relaxation:

$$\min_{x \in \mathbb{R}^{m \times n}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, m \text{ (each worker has one job)}$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad j = 1, \ldots, n \text{ (each job has one worker)}$$

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

This is a transportation problem. As we saw earlier, this problem can be solved by the simplex algorithm. Further, because of the structure of the problem, all the basic feasible solutions are integral when the supply $a_i$ and demand $b_j$ quantities are integral: the initial solution found by the northwest corner rule is integral, and all the updates are integral since they push an integral amount of flow around a cycle.

Thus, we can solve the assignment problem by solving its relaxation.

2 An example assignment problem

Our example problem has costs

$$\begin{array}{cccc}
\text{workers} & 7 & 4^1 & 4^0 & 6 \\
7 & 5 & 8 & 7^1 \\
4 & 5 & 8 & 7^1 \\
4^1 & 6 & 6^0 & 8 \\
5 & 4^1 & 2^0 & \\
\end{array}$$
We used a greedy approach to find an initial feasible solution: pick the cheapest available job, until all the workers have jobs. So we make the following assignments in order:

\[ x_{43} = 1, \quad x_{31} = 0, \quad x_{12} = 1, \quad x_{24} = 1. \]

We then obtain a basic feasible solution by adding three edges as basic variables with \( x_{ij} = 0 \).

The initial basic feasible solution is illustrated below; note that it constitutes a spanning tree. It has value \( 5 + 7 + 4 + 1 = 17 \).

![Graph showing initial basic feasible solution with spanning tree]

### 3 Solve the assignment problem example

#### 3.1 First iteration

We first find the dual variables. We have dual variables \( u_i, i = 1, \ldots, 4 \) for the workers, and dual variables \( v_j, j = 1, \ldots, 4 \) for the jobs. The dual variables satisfy

\[ u_i + v_j = c_{ij} \quad \text{for basic} \; x_{ij}. \]

We set \( u_1 = 0 \) and then determine the other dual variables through a chain reaction.

<table>
<thead>
<tr>
<th>jobs</th>
<th>7</th>
<th>5</th>
<th>4</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>5</td>
<td>8</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>6</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>v_1</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

We can now find the reduced costs from

\[
\text{reduced cost is } c_{ij} - u_i - v_j
\]

<table>
<thead>
<tr>
<th>jobs</th>
<th>5</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
The nonbasic variable \( x_{22} \) has reduced cost \(-2\), so we bring it into the basis. The edge completes a unique cycle, with the other edges in the cycle being some of the current basic variables. The cycle is
\[ s_2 \rightarrow d_2 \rightarrow s_1 \rightarrow d_3 \rightarrow s_4 \rightarrow d_4 \rightarrow s_2. \]
We adjust flow around the cycle.

The largest possible value of \( t \) is \( t = 1 \). We have a choice of variable to leave the basis: with \( t = 1 \), we get \( x_{12} = x_{24} = x_{43} = 0 \). We arbitrarily choose \( x_{12} \) to become nonbasic. We can update the primal solution and then update the dual solution.

Update reduced costs:
\[
\bar{c}_{ij} \leftarrow \bar{c}_{ij} - u_i - v_j
\]
Since all the reduced costs are nonnegative, we are optimal. The optimal assignment is:
- worker 1 does job 3,
- worker 2 does job 2,
- worker 3 does job 1,
- worker 4 does job 4.
Total cost is \( 4 + 5 + 4 + 2 = 15 \).
4 Variants of the transportation problem

4.1 Unequal supply and demand

A transportation problem has $m$ sources supplying $a_i$ for source $i$ and $n$ destinations with demands $b_j$. So far, we’ve assumed total supply and total demand are equal. If they are not equal, we can introduce a dummy node, with costs $c_{ij} = 0$ to ship in or out of the dummy node. For example

<table>
<thead>
<tr>
<th>sources</th>
<th>destinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>40 20 30</td>
</tr>
<tr>
<td>70</td>
<td>1 3 4</td>
</tr>
</tbody>
</table>

total supply: 120
total demand: 90

We introduce a dummy demand node with demand 30.

<table>
<thead>
<tr>
<th>sources</th>
<th>destinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>40 20 30 30</td>
</tr>
<tr>
<td>70</td>
<td>1 3 4 0</td>
</tr>
</tbody>
</table>

total supply: 120
total demand: 120

The excess supply is shipped to the dummy demand node.

4.2 Unusable edges

It may be that some edges cannot be used. These edges can be given infinite cost. In an implementation, we need to choose a finite value, typically called a big-$M$ value. This should be large enough so that an optimal solution does not use the edge. One possibility is to choose the big-$M$ to be at least as large as the sum of all the other edge costs.

<table>
<thead>
<tr>
<th>40 40 20</th>
<th>40 40 20</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>50 4 2</td>
<td>50 4 2</td>
</tr>
<tr>
<td>30 2 5</td>
<td>30 2 5</td>
</tr>
</tbody>
</table>

Edge (2,3) set $c_{23} > \sum$ of other edge weights

not usable

unusable