Plan for today

• Transportation Problem
  • Unbalanced

• Assignment Problem
  • Formulation
  • Connections to Transportation problem
  • Polygamy vs Monogamy?!
  • Solution method: Hungarian algorithm
TRANSPORTATION PROBLEM

... where we see an algorithm to solve the transportation problem
Transportation Problem: General Case

- Single product to be transported from supply to demand
- Transportation Cost per unit from $S_i$ to $D_j$ is $c_{ij} (\geq 0)$ for all $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$
- Supply constraints: Entire supply at every $S_i$ must be distributed to the demand points
- Demand constraints: Entire demand at every $D_j$ must be received from the supply points

Question: How much to transport from each supply point to each demand point to minimize cost?
Formulation of the general case transportation problem

\[
\min Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij}
\]

\[
\sum_{j=1}^{n} x_{ij} = s_i \text{ for every } i \in \{1, \ldots, m\}
\]

\[
\sum_{i=1}^{m} x_{ij} = d_j \text{ for every } j \in \{1, \ldots, n\}
\]

\[x_{ij} \geq 0 \text{ for every } i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}\]
ALGORITHM FOR THE TRANSPORTATION PROBLEM

- Can use Simplex method, but it is typically slow
- So we have a special-purpose algorithm
  … known as the “Transportation Simplex Method”
# A Convenient Representation of the Transportation Problem

<table>
<thead>
<tr>
<th></th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>$D_5$</th>
<th>Supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>16</td>
<td>16</td>
<td>13</td>
<td>22</td>
<td>17</td>
<td>50</td>
</tr>
<tr>
<td>$S_2$</td>
<td>14</td>
<td>14</td>
<td>13</td>
<td>19</td>
<td>15</td>
<td>60</td>
</tr>
<tr>
<td>$S_3$</td>
<td>19</td>
<td>19</td>
<td>20</td>
<td>23</td>
<td>99</td>
<td>50</td>
</tr>
<tr>
<td>$S_4$</td>
<td>99</td>
<td>0</td>
<td>99</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>Demand</td>
<td>30</td>
<td>20</td>
<td>70</td>
<td>30</td>
<td>60</td>
<td></td>
</tr>
</tbody>
</table>

Per unit shipment cost $c_{34}$
A Convenient Representation of the Transportation Problem

<table>
<thead>
<tr>
<th></th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>$D_5$</th>
<th>Supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>16</td>
<td>16</td>
<td>13</td>
<td>22</td>
<td>17</td>
<td>50</td>
</tr>
<tr>
<td>$S_2$</td>
<td>14</td>
<td>14</td>
<td>13</td>
<td>19</td>
<td>15</td>
<td>60</td>
</tr>
<tr>
<td>$S_3$</td>
<td>19</td>
<td>19</td>
<td>20</td>
<td>23</td>
<td>99</td>
<td>50</td>
</tr>
<tr>
<td>$S_4$</td>
<td>99</td>
<td>0</td>
<td>99</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>Demand</td>
<td>30</td>
<td>20</td>
<td>70</td>
<td>30</td>
<td>60</td>
<td></td>
</tr>
</tbody>
</table>

We will avoid writing the variables and
- simply write the allocations in the respective cells
- simply write the relevant allocations

Test your understanding:
What is the objective value of this solution?

$Z = ??$
Transportation Simplex Method: Two Stage Algorithm

Initialization:
Identify an initial **allocation** (i.e., an initial basic feasible solution)
Northwest corner rule

Iterations:
Iterative method to arrive at an optimal solution

\[
\begin{align*}
\min Z &= \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &= s_i \text{ for every } i \in \{1, \ldots, m\} \\
\sum_{i=1}^{m} x_{ij} &= d_j \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\geq 0 \text{ for every } i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}
\end{align*}
\]
TRANSPORTATION SIMPLEX METHOD

Initialization: identifying an initial basic feasible solution
## Northwest Corner Rule

<table>
<thead>
<tr>
<th></th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>$D_5$</th>
<th>Supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>16</td>
<td>16</td>
<td>13</td>
<td>22</td>
<td>17</td>
<td>50 20</td>
</tr>
<tr>
<td>$S_2$</td>
<td>14</td>
<td>14</td>
<td>0</td>
<td>60</td>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td>$S_3$</td>
<td>19</td>
<td>19</td>
<td>20</td>
<td>23</td>
<td>99</td>
<td>50 40 10</td>
</tr>
<tr>
<td>$S_4$</td>
<td>99</td>
<td>0</td>
<td>99</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
</tbody>
</table>

| Demand | 30 | 20 | 70 | 30 | 60 50 | $Z = 3460$ |

1. Start from northwest corner
2. Allocate as much as possible at $x_{ij}$ while meeting demand and supply constraints
3. If allocation exhausts all supply from $S_i$, move one row down
4. Else if $S_i$ has any supply remaining, move one column to the right
Basic Feasible Solution to a Transportation Problem

A basic feasible solution is an allocation satisfying the following conditions:
1. Supply, demand constraints and non-negativity constraints are satisfied
2. There are exactly $m + n - 1$ allocations
3. The allocations do not form a loop

the corresponding variables are basic variables
TRANSPORTATION SIMPLEX METHOD

Iterations: To arrive at an optimal solution
Transportation Problem Instances have Integral Opt Solutions

**Observation:** If the supplies and demands are integers, then the transportation problem always has an integral optimum solution.

- In fact, the Transportation Simplex Method always terminates with an optimum solution whose values are integers

**Proof:**
- Starting allocation (initial basic feasible solution) values obtained by Northwest Corner rule are integers
- Each iteration of the transportation simplex method transforms an allocation with integer values to another allocation with integer values
- So final optimum solution values are integers
THE UNBALANCED CASE
Transportation Problem: General Case

Supply points $S_i$ to Demand points $D_j$

- Single product to be transported from supply to demand
- $c_{ij} \geq 0$
- Supply Constraints: $S_i$ can supply exactly $s_i$ units
- Demand Constraints: $D_j$ has to receive exactly $d_j$ units

Question: How much to transport from each supply point to each demand point to minimize cost?

Balanced Transportation Problem

Feasible only if $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$
Transportation Problem: General Case

**Supply**
- Supply points: \( S_1, \ldots, S_m \)
- \( S_i \) can supply at most \( s_i \) units

**Demand**
- Demand points: \( D_1, \ldots, D_n \)
- \( D_j \) has to receive at least \( d_j \) units

**Costs**
- \( c_{ij} \): Cost of transporting from \( S_i \) to \( D_j \)

**Question:** How much to transport from each supply point to each demand point to minimize cost?

**Constraints:**
- Single product to be transported from supply to demand
- \( c_{ij} \geq 0 \)

**Feasibility Conditions:**
- \( \sum_{i=1}^{m} s_i \geq \sum_{j=1}^{n} d_j \)

**Unbalanced Transportation Problem**

Unbalanced: Feasible only if \( \sum_{i=1}^{m} s_i \geq \sum_{j=1}^{n} d_j \)
Formulation

\[
\begin{align*}
\text{min } Z &= \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &\leq s_i \text{ for every } i \in \{1, \ldots, m\} \\
\sum_{i=1}^{m} x_{ij} &\geq d_j \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\geq 0 \text{ for every } i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}
\end{align*}
\]

\[
\begin{align*}
\text{min } Z &= \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &\leq s_i \text{ for every } i \in \{1, \ldots, m\} \\
\sum_{i=1}^{m} x_{ij} &\geq d_j \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\geq 0 \text{ for every } i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}
\end{align*}
\]

• Since shipping costs \(c_{ij}\) are non-negative, we can assume that no demand point receives more than its required demand
• Now we have excess supply availability
• Introduce dummy demand point \(D_{n+1}\) with demand requirement \(d_{n+1} = \sum_{i=1}^{m} s_i - \sum_{j}^{n} d_j\) and shipping costs \(c_{i,n+1} = 0\) for every supply \(S_i\)
• Resulting transportation problem is balanced
Transportation Problem: General Case

Supply points

- Supply points $S_1, S_i, S_m$
- Demand points $D_1, D_j, D_n$
- Cost $c_{ij}$
- Supply $s_i$
- Demand $d_j$

- Single product to be transported from supply to demand
- $c_{ij} \geq 0$
- Supply Constraints: $S_i$ can supply at most $s_i$ units
- Demand Constraints: $D_j$ has to receive at least $d_j$ units

Unbalanced Transportation Problem

- Feasible only if $\sum_{i=1}^{m} s_i \geq \sum_{j=1}^{n} d_j$

Question: How much to transport from each supply point to each demand point to minimize cost?
Formulation

\[
\min Z = \sum_{i=1}^{m} \sum_{j=1}^{n+1} c_{ij}x_{ij}
\]

\[
\sum_{j=1}^{n+1} x_{ij} = s_i \text{ for every } i \in \{1, \ldots, m\}
\]

\[
\sum_{i=1}^{m} x_{ij} = d_j \text{ for every } j \in \{1, \ldots, n+1\}
\]

\[
x_{ij} \geq 0 \text{ for every } i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}
\]

where

\[
d_{n+1} := \sum_{i=1}^{m} s_i - \sum_{j}^{n} d_j \text{ and }
\]

\[
c_{i,n+1} := 0 \text{ for every } i = 1, \ldots, m
\]

**Solution Procedure:** This is the balanced transportation problem. We know how to solve the balanced case!
ASSIGNMENT PROBLEM

... where we see the assignment problem and an algorithm to solve it
### Operations … what??

<table>
<thead>
<tr>
<th></th>
<th>Adrian</th>
<th>Miller</th>
<th>Phelps</th>
<th>Murphy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backstroke</td>
<td>51.77s</td>
<td>51.99s</td>
<td>52.33s</td>
<td>51.85s</td>
</tr>
<tr>
<td>Breaststroke</td>
<td>58.86s</td>
<td>58.87s</td>
<td>58.91s</td>
<td>58.95s</td>
</tr>
<tr>
<td>Butterfly</td>
<td>51.59s</td>
<td>51.17s</td>
<td>51.14s</td>
<td>51.83s</td>
</tr>
<tr>
<td>Freestyle</td>
<td>47.85s</td>
<td>48.93s</td>
<td>48.01s</td>
<td>49.31s</td>
</tr>
</tbody>
</table>

**Question:** Who swims which leg of the relay to minimize total time?
Example

- 4 machines and 4 tasks
- Cost for machine $i$ to process task $j$ is $c_{ij}$ as given below
- Each task to be assigned to exactly one machine
- Each machine can process exactly one task

<table>
<thead>
<tr>
<th>Franchisees</th>
<th>Employees</th>
<th>Machines</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$</td>
<td>$T_2$</td>
<td>$T_3$</td>
</tr>
<tr>
<td>$M_1$</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>$M_2$</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>$M_3$</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>$M_4$</td>
<td>3</td>
<td>10</td>
</tr>
</tbody>
</table>

Question: Which task should be assigned to which machine so that the total processing cost is minimized?
Formulation

Step 1: identify decision variables

For \(i \in \{1,2,3,4\}, j \in \{1,2,3,4\}\), let \(x_{ij} = \begin{cases} 1 & \text{if task } j \text{ is assigned to machine } i \\ 0 & \text{otherwise} \end{cases}\)

Step 2: determine the objective function

\[
\min_Z = 5x_{11} + 9x_{12} + 3x_{13} + 6x_{14} \\
+ 8x_{21} + 7x_{22} + 8x_{23} + 2x_{24} \\
+ 6x_{31} + 10x_{32} + 12x_{33} + 7x_{34} \\
+ 3x_{41} + 10x_{42} + 8x_{43} + 6x_{44}
\]

Step 3: identify constraints

\[
\begin{align*}
x_{11} + x_{12} + x_{13} + x_{14} &= 1 \\
x_{21} + x_{22} + x_{23} + x_{24} &= 1 \\
x_{31} + x_{32} + x_{33} + x_{34} &= 1 \\
x_{41} + x_{42} + x_{43} + x_{44} &= 1 \\
x_{11} + x_{21} + x_{31} + x_{41} &= 1 \\
x_{12} + x_{22} + x_{32} + x_{42} &= 1 \\
x_{13} + x_{23} + x_{33} + x_{43} &= 1 \\
x_{14} + x_{24} + x_{34} + x_{44} &= 1 \\
x_{ij} &\in \{0,1\}, i = 1,2,3,4, j = 1,2,3,4
\end{align*}
\]

Machine constraints: Each machine is assigned exactly one task

Task constraints: Each task is assigned to exactly one machine

<table>
<thead>
<tr>
<th></th>
<th>T₁</th>
<th>T₂</th>
<th>T₃</th>
<th>T₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>(M₁)</td>
<td>5</td>
<td>9</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>(M₂)</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>(M₃)</td>
<td>6</td>
<td>10</td>
<td>12</td>
<td>7</td>
</tr>
<tr>
<td>(M₄)</td>
<td>3</td>
<td>10</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>
Formulation

\[
\begin{align*}
\min Z &= 5 \, x_{11} + 9 \, x_{12} + 3 \, x_{13} + 6 \, x_{14} \\
&\quad + 8 \, x_{21} + 7 \, x_{22} + 8 \, x_{23} + 2 \, x_{24} \\
&\quad + 6 \, x_{31} + 10 \, x_{32} + 12 \, x_{33} + 7 \, x_{34} \\
&\quad + 3 \, x_{41} + 10 \, x_{42} + 8 \, x_{43} + 6 \, x_{44}
\end{align*}
\]

\[
x_{11} + x_{12} + x_{13} + x_{14} = 1 \\
x_{21} + x_{22} + x_{23} + x_{24} = 1 \\
x_{31} + x_{32} + x_{33} + x_{34} = 1 \\
x_{41} + x_{42} + x_{43} + x_{44} = 1 \\
x_{11} + x_{21} + x_{31} + x_{41} = 1 \\
x_{12} + x_{22} + x_{32} + x_{42} = 1 \\
x_{13} + x_{23} + x_{33} + x_{43} = 1 \\
x_{14} + x_{24} + x_{34} + x_{44} = 1 \\
x_{ij} \in \{0,1\}, i = 1,2,3,4, j = 1,2,3,4
\]

Note: Every feasible solution has exactly 4 variables that take value one, i.e., non-zero

<table>
<thead>
<tr>
<th></th>
<th>T₁</th>
<th>T₂</th>
<th>T₃</th>
<th>T₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>M₁</td>
<td>5</td>
<td>9</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>M₂</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>M₃</td>
<td>6</td>
<td>10</td>
<td>12</td>
<td>7</td>
</tr>
<tr>
<td>M₄</td>
<td>3</td>
<td>10</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

A non-zero variable is also known as an assignment
Assignment Problem: General Case

- Number of machines = Number of tasks, say they are = \( n \)
- Given: Cost \( c_{ij} \) (\( \geq 0 \)) associated with machine \( i \) performing task \( j \)
- Each machine is to be assigned to exactly one task
- Each task is to be performed by exactly one machine

<table>
<thead>
<tr>
<th></th>
<th>( T_1 )</th>
<th>( \ldots )</th>
<th>( T_j )</th>
<th>( \ldots )</th>
<th>( T_n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_1 )</td>
<td>( c_{11} )</td>
<td>( \ldots )</td>
<td>( c_{1j} )</td>
<td>( \ldots )</td>
<td>( c_{1n} )</td>
</tr>
<tr>
<td>( \vdots )</td>
<td>( \vdots )</td>
<td>( \ddots )</td>
<td>( \vdots )</td>
<td>( \ddots )</td>
<td>( \vdots )</td>
</tr>
<tr>
<td>( M_i )</td>
<td>( c_{i1} )</td>
<td>( \ldots )</td>
<td>( c_{ij} )</td>
<td>( \ldots )</td>
<td>( c_{in} )</td>
</tr>
<tr>
<td>( \vdots )</td>
<td>( \vdots )</td>
<td>( \ddots )</td>
<td>( \vdots )</td>
<td>( \ddots )</td>
<td>( \vdots )</td>
</tr>
<tr>
<td>( M_n )</td>
<td>( c_{n1} )</td>
<td>( \ldots )</td>
<td>( c_{nj} )</td>
<td>( \ldots )</td>
<td>( c_{nn} )</td>
</tr>
</tbody>
</table>

Question: Which task should be assigned to which machine so that the total processing cost is minimized?
Formulation

Step 1: identify decision variables

For \( i \in \{1, ..., n\}, j \in \{1, ..., n\}, \) let \( x_{ij} = \begin{cases} 1 & \text{if task } j \text{ is assigned to machine } i \\ 0 & \text{otherwise} \end{cases} \)

Step 2: determine the objective function

\[
\min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}
\]

Step 3: identify constraints

\[
\sum_{j=1}^{n} x_{ij} = 1 \text{ for every } i \in \{1, ..., n\} \quad \text{Machine constraints: Each machine is assigned exactly one task}
\]

\[
\sum_{i=1}^{n} x_{ij} = 1 \text{ for every } j \in \{1, ..., n\} \quad \text{Task constraints: Each task is assigned to exactly one machine}
\]

\( x_{ij} \in \{0,1\} \text{ for every } i \in \{1, ..., n\}, j \in \{1, ..., n\} \)
Formulation of the general case assignment problem

\[
\begin{align*}
\text{min } Z &= \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &= 1 \text{ for every } i \in \{1, \ldots, n\} \\
\sum_{i=1}^{n} x_{ij} &= 1 \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\in \{0,1\} \text{ for every } i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\}
\end{align*}
\]

Note: Every feasible solution has exactly \( n \) non-zero variables

A non-zero variable is also known as an assignment
Assignment problem and Transportation problem

**Problem 1**

\[ \min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}x_{ij} \]

\[ \sum_{j=1}^{n} x_{ij} = 1 \text{ for every } i \in \{1, \ldots, n\} \]

\[ \sum_{i=1}^{n} x_{ij} = 1 \text{ for every } j \in \{1, \ldots, n\} \]

\[ x_{ij} \in \{0,1\} \text{ for every } i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\} \]

**Problem 2**

\[ \min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}x_{ij} \]

\[ \sum_{j=1}^{n} x_{ij} = 1 \text{ for every } i \in \{1, \ldots, n\} \]

\[ \sum_{i=1}^{n} x_{ij} = 1 \text{ for every } j \in \{1, \ldots, n\} \]

\[ x_{ij} \geq 0 \text{ for every } i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\} \]

This problem has an integral optimum solution

- A special case of the transportation problem
  - where \( m = n \), with all demands and all supplies being 1
- Recall: “Integral Optimum Solution Property” of the transportation problem
  - So opt solution to Problem 2 will also give an opt solution to Problem 1
### Assignment problem and Transportation problem

<table>
<thead>
<tr>
<th>Problem 1</th>
<th>Problem 2</th>
</tr>
</thead>
</table>
| \[
\begin{align*}
\min Z &= \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &= 1 \text{ for every } i \in \{1, \ldots, n\} \\
\sum_{i=1}^{n} x_{ij} &= 1 \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\in \{0,1\} \text{ for every } i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\}
\end{align*}
\] |
| \[
\begin{align*}
\min Z &= \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\sum_{j=1}^{n} x_{ij} &= 1 \text{ for every } i \in \{1, \ldots, n\} \\
\sum_{i=1}^{n} x_{ij} &= 1 \text{ for every } j \in \{1, \ldots, n\} \\
x_{ij} &\geq 0 \text{ for every } i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\}
\end{align*}
\] |

- Problem 2 is a “highly” degenerate transportation problem
  - \(n^2\) variables, \(2n\) constraints
    \(\Rightarrow\) any basic feasible solution has \(2n - 1\) basic variables
  - But any feasible solution has only \(n\) non-zero variables
    \(\Rightarrow\) Among the \(2n - 1\) basic variables, \(n\) basic variables take a value of one and the remaining \(n - 1\) basic variables take a value of zero
- Degenerate problems have too many intermediate iterations which do not improve on the objective
- So we do not use the transportation simplex method directly, but use a special-purpose algorithm
THE UNBALANCED CASE

... can be converted to the balanced case
Machine Location Problem

- 4 possible locations, 3 machines
- Cost $c_{ij}$ for installing machine $M_i$ in location $T_j$
- Each machine is to be installed in exactly one location
- Each location can accommodate at most one machine

Question: How to pick 3 locations for the 3 machines to minimize total cost?

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>13</td>
<td>16</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>$M_2$</td>
<td>15</td>
<td>–</td>
<td>13</td>
<td>20</td>
</tr>
<tr>
<td>$M_3$</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>6</td>
</tr>
</tbody>
</table>

Location $T_2$ is not suitable for machine $M_2$. 
Machine Location Problem

- 4 possible locations, 3 machines
- Cost $c_{ij}$ for installing machine $M_i$ in location $T_j$
- Each machine is to be installed in exactly one location
- Each location can accommodate at most one machine

Question: How to pick 3 locations for the 3 machines to minimize total cost?

Formulation as an assignment problem:

1. Introduce big $M$ (recall: big $M = $ large value) for the cost of installing $M_2$ at $T_2$
Machine Location Problem

- 4 possible locations, 3 machines
- Cost \( c_{ij} \) for installing machine \( M_i \) in location \( T_j \)
- Each machine is to be installed in exactly one location
- Each location can accommodate at most one machine

Question: How to pick 3 locations for the 3 machines to minimize total cost?

<table>
<thead>
<tr>
<th></th>
<th>( T_1 )</th>
<th>( T_2 )</th>
<th>( T_3 )</th>
<th>( T_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_1 )</td>
<td>13</td>
<td>16</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>( M_2 )</td>
<td>15</td>
<td>( \text{M} )</td>
<td>13</td>
<td>20</td>
</tr>
<tr>
<td>( M_3 )</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>( M_4 )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Formulation as an assignment problem:
1. Introduce big \( M \) (recall: big \( M = \) large value) for the cost of installing \( M_2 \) at \( T_2 \)
2. Introduce dummy machine \( M_4 \) to account for the extra location

Now the optimum solution for the resulting assignment problem instance with \( n = 4 \) gives an optimum solution to our original problem

Location \( T_2 \) is not suitable for machine \( M_2 \)
HUNGARIAN ALGORITHM

... special purpose algorithm to solve the assignment problem
Observation 1

- Number of machines = Number of tasks, say they are = \( n \)
- Given: Cost \( c_{ij} \geq 0 \) associated with machine \( i \) performing task \( j \)
- Each machine is to be assigned to exactly one task
- Each task is to be performed by exactly one machine

Question: How to make \( n \) assignments to minimize total cost

\[
\min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}
\]

\[
\sum_{j=1}^{n} x_{ij} = 1 \text{ for every } i \in \{1, \ldots, n\}
\]

\[
\sum_{i=1}^{n} x_{ij} = 1 \text{ for every } j \in \{1, \ldots, n\}
\]

\( x_{ij} \in \{0,1\} \) for every \( i \in \{1, \ldots, n\}, j \in \{1, \ldots, n\} \)

Obs 1: Since all \( c_{ij} \)'s are \( \geq 0 \), if we have \( n \) assignments with \( Z = 0 \), then it is an optimum assignment
Observation 2

- Consider first row $M_1$
- This machine has to be assigned to one of the four tasks
- If all tasks uniformly decrease their cost by one dollar for $M_1$, then the optimum assignment will not change, but only the optimum objective value will change (decrease by one dollar)

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>$M_2$</td>
<td>15</td>
<td>$M$</td>
<td>13</td>
<td>20</td>
</tr>
<tr>
<td>$M_3$</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>$M_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$M_2$</td>
<td>15</td>
<td>$M$</td>
<td>13</td>
<td>20</td>
</tr>
<tr>
<td>$M_3$</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>$M_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Obs 2:** If all tasks uniformly decrease their cost by one dollar for $M_1$, then the optimum assignment will not change
Idea

• **Obs 1:** Since all $c_{ij}$s are $\geq 0$, if we have $n$ assignments with $Z = 0$, then it is an optimum assignment

• **Obs 2:** If all tasks uniformly decrease their cost by one dollar for $M_1$, then the optimum solution will not change

• **Approach:** Create as many zeroes in the matrix as possible by uniformly reducing rows and columns
  
  • If we can find $n$ assignments with $Z = 0$ in the reduced matrix then that assignment is optimum
Steps 1 & 2

- Step 1: For each row, subtract row min from that row
- Step 2: For each col, subtract column minimum from that col
- Reduced matrix
  - each row will have at least one zero and
  - each column will have at least one zero
Step 3

- Find a maximum assignment using zero entries in the reduced cost matrix

Choose a maximum possible number of zeroes such that each row and each column has at most one chosen zero

If we can find $n$ assignments in the reduced cost matrix, then we have an optimum

<table>
<thead>
<tr>
<th>Original Matrix</th>
<th>Reduced Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>$M_1$</td>
</tr>
<tr>
<td>13 16 12 11</td>
<td>2 5 1 0</td>
</tr>
<tr>
<td>$M_2$</td>
<td>$M_2$</td>
</tr>
<tr>
<td>15 $M$ 13 20</td>
<td>2 $M$ 0 7</td>
</tr>
<tr>
<td>$M_3$</td>
<td>$M_3$</td>
</tr>
<tr>
<td>5 7 10 6</td>
<td>0 2 5 1</td>
</tr>
<tr>
<td>$M_4$</td>
<td>$M_4$</td>
</tr>
<tr>
<td>0 0 0 0</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>

Optimum solution is $x_{14}^* = 1, x_{23}^* = 1, x_{31}^* = 1, x_{42}^* = 1$

Optimal cost is $11 + 13 + 5 + 0 = 29$

Optimum solution is $x_{14}^* = 1, x_{23}^* = 1, x_{31}^* = 1, x_{42}^* = 1$

Optimal cost is 0
Step 3: How to look for a zero-cost assignment?

Attempt 1: Process the zeroes in arbitrary order and make an assignment when possible

A situation where arbitrary processing ordering fails

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$M_2$</td>
<td>2</td>
<td>$M$</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>$M_3$</td>
<td>0</td>
<td>2</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>$M_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Step 3: How to look for a zero-cost assignment?

Another rule:
Repeat until no more assignments can be made:

a. If any remaining col or row has exactly one zero, then make an assignment
b. If all rows and cols have at least 2 zeroes, then pick an arbitrary assignment

• Once an assignment is made, the other zeros in the corresponding row and column cannot be used for the assignment
  • so cross out the row and column

Watch out: Rule may still not give a maximum assignment

Always inspect for a zero-cost assignment
Steps 1, 2 & 3 may not lead to optimal solution

- An example

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
<th>$T_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>80</td>
<td>0</td>
<td>30</td>
<td>120</td>
<td>0</td>
</tr>
<tr>
<td>$M_2$</td>
<td>80</td>
<td>0</td>
<td>30</td>
<td>120</td>
<td>0</td>
</tr>
<tr>
<td>$M_3$</td>
<td>60</td>
<td>60</td>
<td>$M$</td>
<td>80</td>
<td>0</td>
</tr>
<tr>
<td>$M_4$</td>
<td>60</td>
<td>60</td>
<td>$M$</td>
<td>80</td>
<td>0</td>
</tr>
<tr>
<td>$M_5$</td>
<td>0</td>
<td>90</td>
<td>0</td>
<td>0</td>
<td>$M$</td>
</tr>
</tbody>
</table>

- Cannot reduce further
  ... since every row and every column has at least one zero
- Rule does not give a zero-cost assignment