1 Uncapacitated Metric Facility Location Problem (UMFL).

- Let $L$ be a set of locations.
- Let $F \subseteq L$ be a set of potential facility locations.
- Let $C \subseteq L$ be a set of clients (cities).
- Let $c : L \times L \rightarrow \mathbb{R}^+$ be a distance function. Alternatively think of the function as describing the cost of assigning city $j$ to facility $i$.
- Let $f : F \rightarrow \mathbb{R}^+$ be a cost function describing the cost of opening a facility at $i$.
- Let $\phi : C \rightarrow F \forall j \in C$ be an assignment function. I.e. $\phi(j) = i$ if city $j$ is assigned to facility $i$.

**Problem:** Determine a set of facilities to open and an assignment of all cities to the open facilities that minimizes the total opening and distance cost.

**Notation 1.1.**

$f(i) = f_i$, $c(i,j) = c_{ij}$

The problem is uncapacitated as there is no bound on how many cities an open facility can serve. It is metric as $c$ is defining a metric. This especially means:

\[
\begin{align*}
    c_{ij} &= c_{ji} & \forall i, j \in L \\
    c_{ij} &\leq c_{ik} + c_{kj} & \forall i, j, k \in L \quad \text{(triangle inequality)}
\end{align*}
\]

2 IP Formulation.

Let $y_i = \begin{cases} 
1 & \text{if a facility is opened at } i \in F \\
0 & \text{otherwise}
\end{cases}$

Let $x_{ij} = \begin{cases} 
1 & \text{if } \phi(j) = i \\
0 & \text{otherwise}
\end{cases}$
Minimize $Z(x, y) = F(x, y) + C(x, y) = \sum_{i \in \mathcal{F}} f_i \cdot y_i + \sum_{j \in \mathcal{C}, i \in \mathcal{F}} c_{ij} \cdot x_{ij}$

s.t.

$$\sum_{i \in \mathcal{F}} x_{ij} = 1 \quad \forall j \in \mathcal{C} \quad (1)$$
$$x_{ij} \leq y_i \quad \forall i \in \mathcal{F}, j \in \mathcal{C} \quad (2)$$
$$x_{ij}, y_i \in \{0, 1\} \quad \forall i \in \mathcal{F}, j \in \mathcal{C} \quad (3)$$

The constraint (1) ensures that all cities get assigned to a facility. The constraint (2) ensures that the assigned facilities are open. The LP-relaxation is obtained by changing (3) into:

$$x_{ij}, y_i \geq 0 \quad \forall i \in \mathcal{F}, j \in \mathcal{C}$$

the upper bound is unnecessary.

3 Approximation Algorithm for UMFL.

Algorithm Idea.

The algorithm will take an optimal solution for the LP-relaxation $(x^*, y^*)$ and change this into a feasible solution for the IP problem. This will consist of two operations:

$$(x^*, y^*) \xrightarrow{\text{filter}} (x, y) \xrightarrow{\text{round}} (\hat{x}, \hat{y})$$

where $(x, y)$ and $(\hat{x}, \hat{y})$ are feasible solutions to the LP- and IP-problem respectively. A high-level description of the algorithm steps are:

1. Greedily choose the city, $c_{\text{min}}$ which is cheapest i.e has the lowest overall distance cost.
2. Choose the cheapest facility location $\alpha$ among those fractionally opened locations which the city is fractionally assigned to.
3. Open a facility $f_\alpha$ at $\alpha$ completely.
4. Assign the city completely to the $f_\alpha$ facility (and only to this facility).
5. Assign all cities which are fractionally assigned to some facility locations in the neighbourhood of $c_{\text{min}}$ completely to the the $f_\alpha$ facility.
6. Update collection of unassigned cities and repeat from step 1)

The reason for the filtering is step 5). If some city "far away" is assigned with a very small $x_{ij} > 0$ value to a neighbouring facility, it will be very costly to assign this city to the newly opened facility. This is avoided by ensuring $x_{ij} = 0$ if $c_{ij}$ is "large".
Filtering.

Definition 3.1. Let
\[ \Delta_j = \sum_{i \in F} c_{ij} \cdot x_{ij} \quad \forall j \in C \]

Definition 3.2. \( \forall j \in C \) let \( B_j = \{ i \in F \mid c_{ij} < 2\Delta_j \} \) This describes a neighbourhood or ”ball” around each city containing facility locations with ”small” distances.

Lemma 3.3. Given a solution \((x', y')\) of the LP-problem, there exists a feasible solution to the LP-problem \((x, y)\) such that:

i) \( x_{ij} > 0 \Rightarrow c_{ij} < 2\Delta_j \) (I.e. \( c_{ij} ”is small”\))

ii) \( Z(x, y) \leq 2 \cdot Z(x', y') \)

Proof. \( \forall i \in F, \ j \in C \) let
\[ x_{ij} = \begin{cases} \frac{x'_{ij}}{\sum_{i \in B_j} x'_{ij}} & \text{if } i \in B_j \\ 0 & \text{otherwise} \end{cases} \]
\[ y_i = \min\{1, 2y'_i\} \]

Observation 3.4. \( x_{ij} \leq 1 \quad \forall i \in F, \ j \in C \)

Claim 3.5. \((x, y)\) is a feasible solution fulfilling Lemma 3.3 i).

Check for constraint (1)
\[ \sum_{i \in F} x_{ij} = \sum_{i \in B_j} x_{ij} + \sum_{i \notin B_j} x_{ij} = \sum_{i \in B_j} \frac{x'_{ij}}{\sum_{i \in B_j} x'_{ij}} + \sum_{i \notin B_j} 0 = 1 + 0 = 1 \]

Check for constraint (2)

Case 1. \( y_i = 1 \) follows from observation 3.4

Case 2. \( y_i = 2y'_i \). We have \( \sum_{i \in F} x'_{ij} = 1 \) as \((x'_{ij}, y'_i)\) is a solution to the LP-problem. Interpret \( x'_{ij} \) as a probability distribution for ”assigning j to i” and \( c_{ij} \) as a “distance” random variable.

Theorem 3.6 (Markov Inequality). Let \( X \) be a positive, random variable. Let \( a > 0 \) then
\[ Pr \left[ X \geq a \right] \leq \frac{E[X]}{a} \]
Using the Markov Inequality we get:

\[
\sum_{i \in B_j} x_{ij}' = Pr \left[ c_{ij} \geq 2\Delta_j \right] \leq \frac{\Delta_j}{2\Delta_j} = \frac{1}{2}
\]

\[
\sum_{i \in B_j} x_{ij}' \geq \frac{1}{2}
\]

\[
x_{ij} \leq 2x_{ij}' \leq 2y_i' = y_i
\]

The last line following from the definition of \( x_{ij} \) and from \((x', y')\) being a feasible solution. This proves claim 3.5 and per construction part i) of Lemma 3.3

\[
Z(x, y) = F(x, y) + C(x, y)
\]

\[
F(x, y) = \sum_{i \in F} f_i \cdot y_i \leq \sum_{i \in F} f_i \cdot 2y_i' = 2F(x', y')
\]

\[
C(x, y) = \sum_{j \in C} \sum_{i \in F} c_{ij} \cdot x_{ij} \leq \sum_{j \in C} \sum_{i \in F} c_{ij} \cdot 2x_{ij}' = 2C(x', y')
\]

\[
Z(x, y) \leq 2Z(x', y')
\]

This proves part ii) of Lemma 3.3.

\[\square\]

**Algorithm.**

Let \((x', y')\) denote the constructed solution to the IP-problem.

**Step 1.** Solve the relaxed LP-problem getting optimal solution \((x^*, y^*)\).

**Step 2.** Filter \((x^*, y^*) \rightarrow (x, y)\).

**Step 3.** Define \( \Delta_j = \sum_{i \in F} c_{ij} x_{ij} \) and \( B_j = \{ i \in F \mid c_{ij} < \Delta j \} \).

**Observation 3.7.** No factor 2 in definition of \( B_j \) and \( \Delta_j \leq 2 \cdot \Delta j^* \), \( \forall j \in C \)

**Step 4.** While \( C \neq \emptyset \) do

- Chose minimal overall cost city:

\[
j \leftarrow \min_j \Delta j
\]
Consider neighbourhood $B_j$. Let $\alpha$ be the facility location $i \in B_j$ with smallest opening cost ($f_\alpha$ is minimum.)

- Open facility at $\alpha$ ($y'_\alpha = 1$).
- Assign city $j$ to $\alpha$ ($\phi(j) = \alpha, \ x'_{ij} = 1$ for $i = \alpha$ and $x'_{ij} = 0$ for $i \neq \alpha$)
- Update $C \leftarrow C \setminus \{j\}$.

Consider all other neighbourhoods $\overline{B_j}$ for which $B_j \cap \overline{B_j} \neq \emptyset \Rightarrow \exists \bar{i} \in F : \bar{i} \in B_j$ and $\bar{i} \in \overline{B_j}$

- Assign city $\bar{j}$ to $\alpha$ ($\phi(\bar{j}) = \alpha, \ x'_{i\bar{j}} = 1$ for $i = \alpha$ and $x'_{i\bar{j}} = 0$ for $i \neq \alpha$)
- Update $C \leftarrow C \setminus \{\bar{j}\}$.

**Step 5.** Output $\{\alpha \mid y'_\alpha = 1\}$ and $\phi$.

**Algorithm Analysis.**

**Claim 3.8.** The algorithm is a 6-approximation.

**Proof.**

**Termination and Feasibility:** The number of cities is final and in each iteration at least one city is removed from the set of unassigned cities. The algorithm returns a feasible solution as each city has been assigned to an open facility location.
**Opening Cost:** Consider a round of the algorithm choosing city $j$.

Using the choice of $\alpha$ and that $(x, y)$ is a filtered solution we have for all facility locations in $B_j$:

$$\sum_{i \in B_j} f_i \cdot y_i \geq \sum_{i \in B_j} f_\alpha \cdot y_i \geq f_\alpha \cdot \sum_{i \in B_j} y_i = f_\alpha \cdot \sum_{i \in \mathcal{F}} x_{ij} = \text{opening cost of algorithm.}$$

Let $\{\overline{B_1}, \overline{B_2}, \ldots, \overline{B_n}\}$ be all the $\overline{B_j}$ sets intersecting with $B_j$. Define a union of disjoint sets:

$$\overline{B} = \bigcup_{i,k \in 1\ldots n} (\overline{B_i} \setminus \bigcup_{k < i} \overline{B_k})$$

We have for the facility locations in $\overline{B} \setminus B_j$:

$$\sum_{i \in \overline{B} \setminus B_j} f_i \cdot y_i \geq 0 = \text{opening cost of algorithm.}$$

The algorithm "touches" each facility location exactly once, either selecting or dropping it $\Rightarrow$ summing over all algorithm rounds and using Lemma 3.3 we get:

$$\text{Summed opening cost of algorithm} \leq \sum_{i \in \mathcal{F}} f_i \cdot y_i = F(x, y) \leq 2 F(x^*, y^*) \quad (4)$$

**Connection Cost:** For all cities we either have

a) The city, $j$ is assigned to a facility in its own neighbourhood: $\Rightarrow$ connection cost for $j \leq \Delta j$

b) The city, $\bar{j}$ is assigned to a facility in the neighbourhood of another city, $j \Rightarrow$ connection cost for $j \leq \frac{\Delta j}{d_{2}} + \frac{\Delta j}{d_{3}} + \frac{\Delta j}{d_{4}} \leq 3 \Delta \bar{j}$ (see Figure 1 on the previous page)

Using a) and b) and observation 3.7 we get:

$$C(x', y') = \sum_{j \in \mathcal{C}} c_{ij} \cdot x'_{ij} \leq \sum_{j \in \mathcal{C}} 3 \Delta j \leq \sum_{j \in \mathcal{C}} 6 \Delta j^* = 6 \cdot \sum_{j \in \mathcal{C}} c_{ij} \cdot x^*_{ij} = 6 C(x^*, y^*) \quad (5)$$

(4) and (5) gives: Algorithm cost $\leq 2 F(x^*, y^*) + 6 C(x^*, y^*) \leq 6 Z(x^*, y^*) \leq 6 OPT_{UMFL} \quad \square$