Spatial Patterns Underlying Facility Location Problems

Visualization and Classification

Hannah Bakker
hannah.bakker@kit.edu
Stefan Nickel
stefan.nickel@kit.edu
Karlsruhe Institute of Technology, Institute for Operations Research

I. Motivation

Despite facility location problems being perfectly suited for effective visualizations of instances and solutions, relatively few works make use of this option. This is, not the least, because many of the benchmark instances provided by literature that are frequently used do not provide exact locations in terms of coordinates that would allow for visualization in a 2-dimensional space. Instead, these instances only provide the input that is required for the mathematical programming formulation, the transport cost matrix $\{D\}_{I\times J}$ with I being the set of candidate locations, J the set of customers, and d_{ij} the unit transportation costs.

In the following, we briefly outline how based on this sparse information, one can retrieve approximate coordinates in two dimensions for all customers and candidates that allow for an effective visualization of the underlying spatial relationships. In addition, we provide code for classifying the estimated spatial point pattern.

We see the following two use cases for our code:

1) Apprehend and visualize spatial relationships between candidates and customers in instances from benchmark instances for which exact locations are not available.
2) Apprehend implied spatial relationships in real-world instances for which exact locations are known, yet not Euclidean distances are used to derive $\{D\}_{I\times J}$, and thus the visualization on a map may be misleading.

Along with a suggestion on how to provide effective visualizations that encourage an intuitive understanding of the underlying spatial pattern,

II. Coordinates via Multi-dimensional scaling

We suggest retrieving coordinates via multi-dimensional scaling (MDS), a dimensionality reduction technique. To the best of our knowledge, [1] first used MDS to approximate coordinates of location instances in a 2D plane. MDS is a dimensionality reduction technique commonly used in machine learning. It analyses the similarity or dissimilarity of data in the original high-dimensional space and then attempts to model it in a lower-dimensional space. We implement MDS using the `sklearn.manifold` package, taking the unit transport costs $\{D\}_{I\times J}$ as input features. As there are usually more customers than facilities, we consider each customer as an observation and their distances to the respective facilities as features. This leaves us with a $J \times I$-dimensional feature matrix. Distances among facilities are usually not provided in the transport cost matrix of benchmark instances, so they are estimated according to Algorithm 1. For each pair of facilities i and i', we successively derive upper and lower bounds (UB and LB) on their distance with the help of triangular inequalities. We then approximate their distance as the average between the upper and lower bound. The parameter p determines how many customers closest to facility i are considered to improve UB and LB. The resulting extended transport cost matrix has dimensions $((|J| + |I|) \times |I|)$. With MDS it is projected to a $(|J| + |I|) \times 2$-dimensional matrix. The two columns serve as a basis for the (x, y)-coordinates in a plane.

Algorithm 1 Estimate distances between two facilities i, i'

Require: i, i', $d_{ij_{j=1,\dots,J}}$, $d_{i'j_{j=1,\dots,J}}$, p
Ensure: $\tilde{d}_{ii'}$
1: $UB \leftarrow +inf$, $LB \leftarrow 0$
2: Sort j according to increasing distances $d_{ij_{j=1,\dots,J}}$.
3: **for** $j \in 1 \dots, p$ **do**
4: $\quad UB \leftarrow \min\{UB, d_{ij} + d_{ij'}\}$,
5: $\quad LB \leftarrow \max\{LB, |d_{ij} - d_{ij'}|\}$.
6: **end for**
7: $\tilde{d}_{ii'} \leftarrow (UB + LB)/2$
8: $\theta_{old} \leftarrow \theta$

III. Classification of spatial point patterns

Taking the estimated coordinates of candidates and customers as input, we classify the spatial point pattern underlying a problem instance as either "clustered", "random", or "evenly" distributed with the help of a two-sided t-test on the average nearest neighbor distance. The test measures the degree to which the average distance of any point to its nearest neighbor differs from the expected average distance under spatial randomness, a situation in which any point has had the same chance of occurring on any sub-area as any other point.

Let D_O denote the observed average nearest neighbor distance. D_O is compared to the expected nearest neighbor distance D_E in a randomly dispersed population of $I + J$ points with specified density ρ, known to be

$$D_E = \frac{1}{2\sqrt{\rho}} \tag{1}$$

with a standard deviation σ_{D_E} of

$$\sigma_{D_E} = \frac{0.26136}{\sqrt{N\rho}}. \tag{2}$$

The density ρ is obtained as the number of points divided by the area [3].

Given D_O, D_E, and σ_{D_E}, it is possible to perform a two-sided t-test on the hypothesis that D_O equals D_E and thereby the null hypothesis that the mean distance to the nearest neighbor in the observed distribution cannot be distinguished from the mean distance under spatial randomness. This test evaluates the test statistic

$$z = \frac{D_o - D_E}{\sigma_{D_E}} \tag{3}$$

based on the quantiles of the standard normal distribution. This implies that z-values whose absolute value exceeds 1.96 (2.58) allow one to reject the null hypothesis of spatial randomness with a confidence level of 95% (99%). A detailed description of the hypothesis test for complete spatial randomness can be found in [2].

The test allows one to gain information on which spatial distribution one observes instead of spatial randomness. If the test statistic is negative, then the observed mean distance to the nearest neighbor is smaller than expected under randomness, indicating clusters of points at certain sub-spaces. We classify these patterns as "clustered". If the test statistic is positive, the distance is larger than expected, indicating a more evenly dispersed distribution of points. We classify these patterns as "even". If the null hypothesis cannot be rejected, we classify the pattern as "random".

Considering candidates and customers as data points, we use the coordinates obtained according to the previous section as a proxy for the location of candidates and customers in a hypothetical plane. We determine the Euclidean distances between all pairs of points and take the minimum distance per point as its estimated nearest neighbor distance. We take the area within the convex hull of the estimated points as a proxy for A.

References

[1] Guazzelli, C. S., Cunha, C. B.: Exploring K- best solutions to enrich network design decision-making. *Omega* 78:139–164 (2018). doi:10.1016/j.omega.2017.06.009 1

[2] Clark, P. J., Evans, F. C.: Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations. *Ecology* 35(4):445–453 (1954). doi:10.2307/1931034 2

[3] Hertz, P.: Über den gegenseitigen durchschnittlichen Abstand von Punkten, die mit bekannter mittlerer Dichte im Raume angeordnet sind. *Mathematische Annalen* 67(3):1432–1807 (1909). doi:10.1007/BF01450410 2