Regional Co-location Pattern Detection

Hashim Mohamed
moham992@umn.edu

Sihan Wei
wei00114@umn.edu

May 12th, 2020

Abstract

An important spacial data science problem is the measurement prevalence of spatial interactions between types of spatial features. Questions like "Do these two species tend to build nests close together in this region" are often answered with utilization of spatial co-location patterns and their related prevalence measurements. The related but less explored problem of regional co-location pattern detection flips the problem around; Instead of measuring the prevalence of interaction within a given region, the goal is the identification of arbitrary sub-regions that meet a certain co-location pattern prevalence measure. Finding regional co-location patterns has significance in many fields such as environmental protection, public security and public health; however, significant computational challenges arise mainly in regards with the exponential number of candidate regions in the solution set. Existing approaches utilize partitioning and cluster based regions that may miss finding certain prevalent regions, sacrificing completeness for computation speed. More recent developments have been made in the case of finding regionalities defined by orthogonally aligned rectangles. These utilize properties of such regions to toward efficiently pruning the search space with while maintaining completeness. Our study involves the exploration ways to improve upon these methods. We began with a detailed exploration of the remaining search space to identify and categorize patterns. Then we we proposed the Unique Quadruplet Enumeration algorithm that utilizes one such identified pattern to outperform previous methods in testing with both synthetic data and a real world crime.

1 Introduction

1.1 Key Concepts

Co-location Patterns represent set of potentials spatial **features** related to others through a spatial **relation** (e.g., within certain euclidean distance) [1]. Figure 1 shows a dataset consisting of instances of two spatial features, f_A (yellow circle) and f_B (blue triangle). There are 7 instances of f_A and 8 instances of f_B while only exists one co-location pattern $\{f_A, f_B\}$, with 4 instances. Note: co-location pattern are not limited to just two instances

they can form cliques / chains of arbitrarily high number of feature instances as long as each feature instance meets the relation criteria.

Participation ratio $pr(C, f_i)$ is the ratio of instances of f_i participating in a co-location pattern C to the total number of f_i instances in the same region. For example, in Figure 1, for pattern $C = \{f_A, f_B\}$, we have $pr(C, f_A) = \frac{4}{7}$, $pr(C, f_B) = \frac{4}{8}$

Participation index is a type of prevalence measure for a collocation pattern. $pi(C) := min_i\{pr(C, f_i)\}$ for all $f_i's$ participating in pattern C. In our example $pi(C) = min\{pr(C, f_A), pr(C, f_B)\} = \frac{4}{8}$.

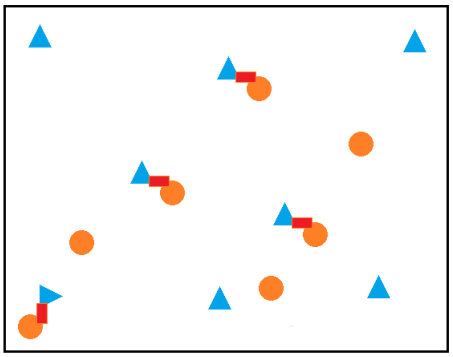
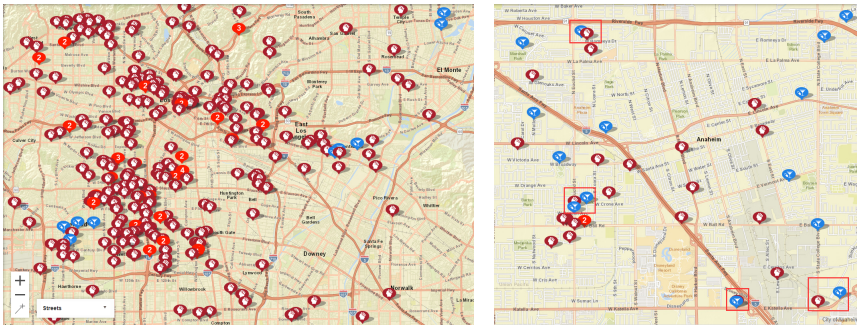


Figure 1: Two interacting feature types

Minimal Orthogonal Bounding Rectangle (MOBRs) Ar the smallest rectangle with boundaries parallel to the coordinate axis minimally bounds a set of points or shapes [2].

1.2 Motivation

Finding regional co-location patterns is a difficult data mining problem with significance in many fields such as environmental protection, public security and public health. Global co-location patterns often do not provide us useful insight for certain relationships. If a co-location pattern is not frequent relative to all input instances, its global participation index in the entire data-set may be rather low. However, we may be interested in a co-location pattern have a higher participation index in some subset of the data-set. For example, Figure 2 is the crime map of Los Angeles, where the red icon represents assault while the blue icon refers to crime related to drugs or alcohol. Most of the feature instances are not participating in a relation leading to . But in (b) it shows that in some certain areas of the city, we could still find the existence of such patterns.



(a) Global co-location pattern may be neglected

(b) Regional co-location pattern could be detected

Figure 2: A crime map of Los Angeles. (Source: `https://www.crimemapping.com/`)

1.3 Related Work and Challenges

Many methods have been proposed to solve variants of the regional collocation pattern detection problem. Data unaware methods systematically divide the study area into sub localities with such as grids [3] or quadtrees [4]. These methods are incomplete because it may miss potential regions crossing grids. The second class use data aware methods such as those that define regions with clusters of co-location instances. In [5], the authors start with randomly selecting representatives from the dataset and use product of z-scores of the relevant continuous variables as a interest measure. [6, 7] both detect co-location patterns using neighborhoods as localities. These methods are often computationally complex and sacrifice completeness as regions in low density areas without object or co-location instance concentrations may not be detected by the cluster finding algorithms used.

Major challenges include the fact that these data sets tend to be very large with many potential patterns to be explored therefor computational complexity is paramount. When pursuing computational efficiency, we also need to maintain completeness, which means that we need to ensure no potential region is missed. Another challenge is that is that exact details of the type of regions to be found and the participation identifier used is domain specific so it is difficult to produce a generalized solution.

1.4 Our Contributions

In our study we expanded work of [8]. We began with a detailed exploration of the remaining search space to identify and categorize patterns. Then we proposed the Unique Quadruplet Enumeration algorithm that utilizes one such identified pattern to outperform the previous Quad method in experimentation with both synthetic data and a real world crime. We also introduce a number of new pruning metrics that lay the ground work to developing even better algorithms in future.

2 Problem Definition and Previously Proposed Approaches

2.1 Problem Definition

Input:

- A set of spatial objects.
- A spatial relation on the objects.
- A participation index threshold θ.
- A co-location instance number threshold γ

Output: Regional co-location patterns within participation index threshold $\geq \theta$ and the number of instances within threshold $\geq \gamma$.

Objective: Computational efficiency.

Constraints:

- Correctness and completeness of the result set.
- The co-location instance number threshold $\gamma \geq 2$.
- The locality of a local co-location pattern is the MOBR of its co-location instances.

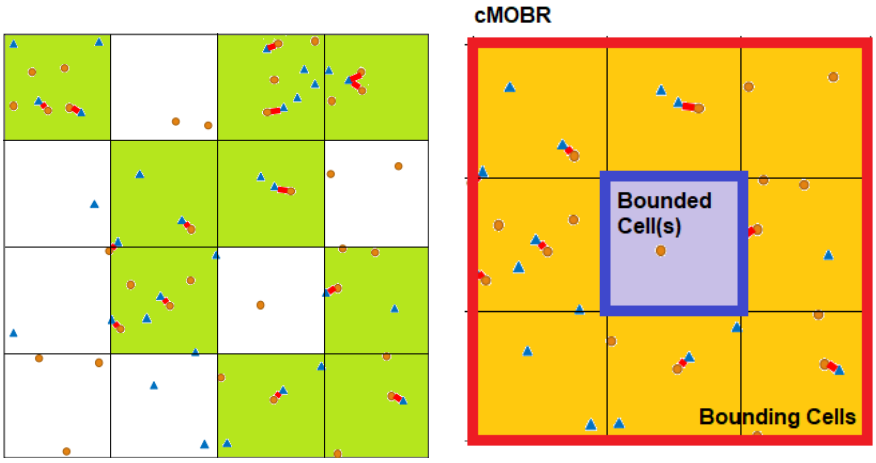
2.2 The Quadruplet Algorithm

A naive approach to solve the regional co-location pattern problem is to enumerate all arbitrary subsets of the input instances and generate an MOBR for each subset. The computational cost for such a method is exponential in the number of feature instances. Even with just rectangular axis aligned regionalities, if there are n instances of a co-location pattern, there will be 2^n subsets, resulting in 2^n rectangle enumerations. In [8], a novel algorithm, the quad-element lgorithm is proposed that significantly decrease the search space, Given a set s of n points in a two-dimensional plane, the set of MOBRs for arbitrary subsets of s is the same as the set of MOBRs for arbitrary subsets with cardinality ≤ 4 of s.

2.3 QGFR Algorithm overview

In [8], the authors refine the Quad algorithm by partitioning the area into grid cells. An active cell is a cell overlapping co-location instances, as shown in Figure 2.3(a). After getting all active cells, the authors use their MOBRs (cMOBR) as an approximation of the MOBRs of $C's$ instances. In this way, we could further reduce the computational cost of enumerating

a co-location pattern's MOBRs from $O(2^n)$ to $O(n^4)$.Figure 2.3(b) shows that there are two types of cells in a cMOBR. We call cells adjacent to the boundary of cMOBR (yellow cells) bounding cells while the rest are bounded cells (blue cells).



(a) 9 active grid cells contain 13 co-location pattern instances

(b) Bounding and bounded cells

Figure 3: An example of cMOBR

To eliminate the cMOBRs where no iMOBR is eligible, the authors introduce an upper bound, $\zeta(< C, cMOBR >, f)$ for the participation ratio of a feature f in a regional co-location pattern composed of a pattern C and any iMOBR in a cMOBR of C.

$$\zeta(< C, cMOBR >, f) = \frac{po(C, f, cMOBR)}{o(f, bounded) + po(C, f, bounding)}$$

3 Improving Further

In tackling the problem of improving the above mentioned algorithms we analysed the possible solution regions of the quad algorithm to try to spot patterns that can help us further prune the solution set.

3.1 The Redundant Enumeration Problem

One pattern we discovered is for practically any data distribution we tried, the quad algorithm would enumerate the same rectangles multiple times. On average it the same rectangle was enumerated 5 times. Each enumeration after the first is completely redundant since if it were a solution, it would already be in the solution set and conversely if it were not it would already have been rejected. The reason for this over counting is shown in the figure to the right. Many mobrs groupings of three colocation patterns is often equal to an mobr of just two of their points. This is not always the case and it is possible to have upto groups of 4 colocation pattern instances that do not share their mobr therefore we cannot use this property to lower the complexity of the enumeration.

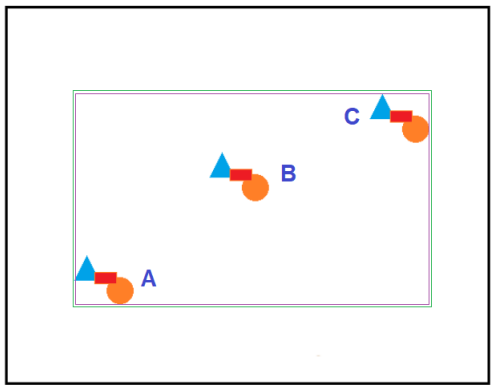


Figure 4: Redundant enumeration problem: MOBR(A,B,C) = MOBR(A,C)

Many distributions were tested including comparing datasets with a small number of large clusters and datasets with large numbers of small clusters. Other distributions included measuring the property clusters alone without noise, on a data set consisting of uniform randomly distributed points, on a data distribution that was 100 times wider than it was tall and on the Chicago crime data set[9]. The results are summerized in the table below:

Table 1: Percent unique across different test sets

Data Distribution	Percent unique
Few large clusters	23.02872566 %
Many small clusters	18.83499984 %
No noise	18.1319422 %
Uniform Randomness	18.96598394 %
Skewed region	17.52591731 %
Chicago Crime Data set	19.78769909 %

All distributions feature around the same 80% redundant figure. Further testing was done to attempt to see the effect on increasing co-location instance numbers in this property. The results are shown below. The property

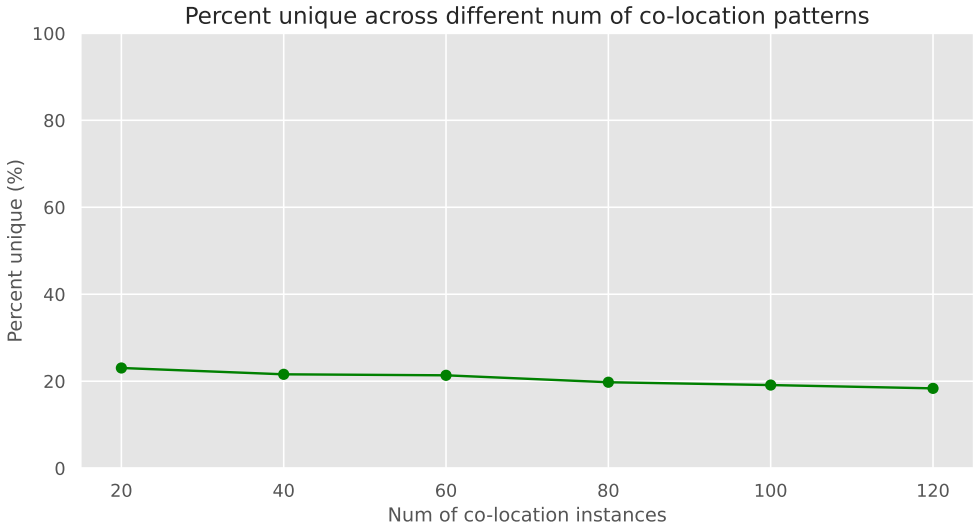


Figure 5: Percentage of unique MOBR's across a number of different distributions

3.2 The Unique-Quad Algorithm

We explored a number of possible solutions to this problem.

High memory trade off solution: The fastest way to account for the redundant enumeration problem is by pre-allocating a data-structure with a minimum of one bit per each mobr to be enumerated. This would allow checking if a certain rectangle was enumerated before in a single constant time memory read operation. It however requires O(n^4) space complexity. Even for a small example of just 50 col-location instances that at least 6 Gigabits of memory needed.

Medium memory trade off solution: Another solution is to create a hashed set of seen solutions. The O(n) space solution will require an extra O(log(n)) computation to check if the given enumeration hos been done before. If an ennumeration has been done before it can break early avoiding the unnecessary O(nlog(n)) interception range search needed afterwards

Constant time geometric solution:One observation made is that with special consideration on the order of enumeration and how certain internal properties of co-location instances are stored, constant time redundancy checking can be done solely one the geometric properties of the co-location instances themselves. While not as fast as the singe memory read of the high memory trade off Solution, it is still an O(1).

Special care has to be taken into account to avoid this step of the algorithm degrading into O(n) cost. The ordering of enumeration was set to first enumerate on combinations of single instances then pairs then triples then quads. The first step of enumerating each collocation instance on it's own is required as sometimes a co-location instance's clique has enough members on it's own to meet the solution threshold. This is the first step is where the mobr around the members of each co-location instance clique is generated and stored. From this

point forward when calculating the mobr around groups of co-location pattern instances, this singe collocation instance mobr is rather than the points within. Since the quad algorithm maxes out to only enumerating 4 co-location pattern instances, such operations will always occur in constant time.

With these considerations in place there are many ways to preform the redundancy check. The simplest is for each group of n ¡= 4 co-location instances being enumerated to calculate the mobr around the upto n choose n-1 instances and comparing it to the mobr around all n instances. IF they all match then the instances are redundant. This one check before each enumeration along with the maintaining of the above mentioned properties to the quad algorithm form the new unique-quad algorithm,

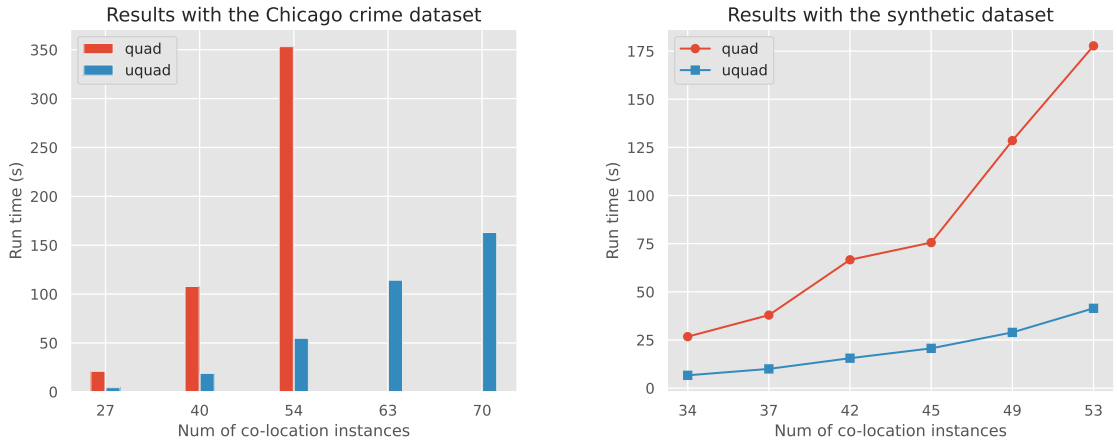
4 Experimentation

4.1 Methedology

The primary goal of the experimentation was to evaluate the relative performance of the proposed uQuad algorithm over the base Quad algorithm [8]. Experimentation was conducted on both synthetic and subsets of a real Chicago crime data set[9]. Multiple tests were conducted with increasing number of co-location instances to attempt measure the the relatively.

A number of key changes were made compared to experimentation in [8, 10]. First was the aggregated point process used, we used the Thomas Cluster Process. The Thomas Cluster Process distributes according to a normal distribution unlike Poisson cluster process and Matérn's cluster processes which generate cluster points uniformly within circles around the cluster centers [11]. This avoids the sharp falloff of points around edges of clusters. Another change is that instead of inserting artificial co-location pattern instances on top of the randomly generated data, we got the desired number of co-location pattern instances by dynamically increasing the neighbor distance threshold for co-location patterns until a value along the desired number. This allowed us to experiment on different problem sizes without changing the distribution.

4.2 Results



(a) Results with the Chicago crime dataset

(b) Results with the synthetic dataset

Figure 6: Performance of Quad vs uQuad on different test sets and number of co-location instances

The results of experimentation are summarized in the figure above that shows performance of the Quad and uQuad algorithms on synthetic and real-world data. The graph shoes significant decreased processing time of the uQuad algorithm over the Quad in all preformed tests. The change in the relative performance as the number of co-location instances increased also suggest that the performance gap increases with more complex datasets.

5 Conclusions

In our study we expanded upon We began with a detailed exploration of the remaining search space to identify and categorize patterns. Then we we proposed the Unique Quadruplet Enumeration algorithm that utilizes one such identified pattern to outperform the previous Quad method in experimentation with both synthetic data and a real world crime.

In the process of developing the uquad algorithm We partially developed some loosely related bounds that can be use full that lay the ground work to developing even better algorithms in future. These are listed in the following Addendum Section. Planned future work involves further exploring our proposed new pruning metrics developing algorithms that utilize them.

6 Addendum: Potentially useful bounds

6.1

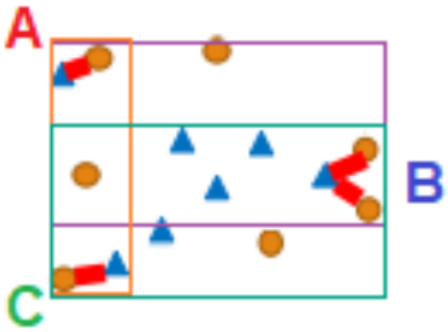


Figure 7: The MOBR around any combination of 3 or more co-location patterns instances is equal to the union of the MOBR around each pair of patterns

We can construct a bound on the participation ratio for the combination of all three instances:

$$pr(< C, (A \cup B \cup C) >, f) \leq 3*(pr(< C, (A \cup B) >, f) + pr(< C, (A \cup C) >, f) + pr(< C, (B \cup C) >, f))$$

The bound is rather loose but is still applicable as in many sub-regions have relatively few co-location pattern instances and have extremely low participation index values. It is constructed based on the worst case pessimistic view that the the mobr around the 3 instances deos not intersect any more participating instances and that all non participating instances are arranged in the worst case arrangement where they were all each counted toward the calculation of the regions around each pair of instances.

This can be expanded to combinations of 4 co-location instances where their MOBR is equal to the to the union of the MOBR around each triplet. This allows us to skip the enumeration of certain combinations of 3 and 4 co-location instances in the quad or uquad algorithms. This can be done by storing the results of each pair of co-location instances in $O(n^2)$ space before moving on to triples or quads.

6.2

In cases when A and B, then we could deduce the the bound of $(A \cup B)$ in O(1) time with the following:

Lemma 1. *The upper bound, $\xi\,(< C, (A \cup B) >, f)$, for any two regions that share an edge, the participation ratio of feature f in a regional co-location pattern composed of a pattern C*

and a cMOBR $(A \cup B)$ *is*

$$pr(< C, (A \cup B) >, f) \leq \max(pr(< C, A >, f), pr(< C, B >, f))$$

Proof.

The participation ratio for any feature in the combined region is:

$$pr(< C, (A \cup B) >, f) = \frac{po(f, C, A) + po(f, C, B)}{o(f, A) + o(f, B)}$$

WLOG, assume that region B is the region with the larger participation ratio:

$$\frac{po(f, C, A)}{o(f, A)} \leq \frac{po(f, C, B)}{o(f, B)} = \max(pr(< C, A >, f), pr(< C, B >, f))$$

simplifying

$$\begin{aligned} po(f, C, A)o(f, B) &\leq po(f, C, B)(o(f, A)) \\ o(f, B)\left(po(f, C, A) + po(f, C, B)\right) &\leq po(f, C, B)\left(o(f, A) + o(f, B)\right) \\ \frac{po(f, C, A) + po(f, C, B)}{o(f, A) + o(f, B)} &\leq \frac{po(f, C, B)}{o(f, B)} \end{aligned}$$

Therefore

$$pr(< C, (A \cup B) >, f) \leq \max(pr(< C, A >, f), pr(< C, B >, f))$$

□

Given a participation index threshold θ, if $pr(< C, (A \cup B) >, f) < \theta$, there will be no eligible regions in $(A \cup B)$. This could help us further reduce the computational cost of future algorithms.

References

[1] Y. Huang, S. Shekhar, and H. Xiong, "Discovering colocation patterns from spatial data sets: a general approach," *IEEE Transactions on Knowledge and data engineering*, vol. 16, no. 12, pp. 1472–1485, 2004.

[2] J. Wood, *Minimum Bounding Rectangle*, pp. 1232–1233. Cham: Springer International Publishing, 2017.

[3] S. Wang, Y. Huang, and X. S. Wang, "Regional co-locations of arbitrary shapes," in *International Symposium on Spatial and Temporal Databases*, pp. 19–37, Springer, 2013.

[4] M. Celik, J. M. Kang, and S. Shekhar, "Zonal co-location pattern discovery with dynamic parameters," in *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pp. 433–438, IEEE, 2007.

[5] C. F. Eick, R. Parmar, W. Ding, T. F. Stepinski, and J.-P. Nicot, "Finding regional co-location patterns for sets of continuous variables in spatial datasets," in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, pp. 1–10, 2008.

[6] P. Mohan, S. Shekhar, J. A. Shine, J. P. Rogers, Z. Jiang, and N. Wayant, "A neighborhood graph based approach to regional co-location pattern discovery: A summary of results," in *Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems*, pp. 122–132, 2011.

[7] M. Deng, J. Cai, Q. Liu, Z. He, and J. Tang, "Multi-level method for discovery of regional co-location patterns," *International Journal of Geographical Information Science*, vol. 31, no. 9, pp. 1846–1870, 2017.

[8] Y. Li and S. Shekhar, "Local Co-location Pattern Detection: A Summary of Results," in *10th International Conference on Geographic Information Science (GIScience 2018)* (S. Winter, A. Griffin, and M. Sester, eds.), vol. 114 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 10:1–10:15, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018.

[9] "Chicago police department. crimes - 2001 to present, 2017.." `https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2`. Accessed: 2020-04-28.

[10] S. Barua and J. Sander, "Mining statistically significant co-location and segregation patterns," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26, pp. 1185–1199, 05 2014.

[11] "Simulating a thomas cluster point process." `https://hpaulkeeler.com/simulating-a-thomas-cluster-point-process/`. Accessed: 2020-04-28.