Percolation of Matérn II Point Processes on a Random Graph

Ryan Honea

Austin Peay State University

February 9, 2018

1. Introduction
    - Definitions
    - Statement of the Problem
    - Motivation
2. Methodology
    - Method of Determining Solutions
    - Algorithmic Analysis
3. Results
    - Visual Analysis
    - Logistic Model
4. Conclusion
    - Final Thoughts
    - References
    - Acknowledgments

Introduction

1. What is a point process and what's a random graph?
   - A Point Process describes how points are distributed across a graph. Ex: Poisson Point Processes are distributed with Poisson probability in the x and y coordinates.
   - Random graphs are random ensembles of vertices and edge.
2. What is a Matérn II Point Process?
   - Otherwise referred to a hard-core shell point processes, Matérn II point processes simulate real objects
   - This is done by giving points a radius and effectively making circles.
   - Any new point that overlaps an old point deletes the first point to simulate real objects maintaining space.

Introduction

1. What is percolation?
   - Percolation is the study of connected clusters on a random graph.
   - I am looking for an infinite cluster, or a percolating cluster, in which vertices extend from the origin into infinity.

Statement of the Problem

I attempt to find the following values:

1. The critical probability in which an infinite cluster exists for different factors of r (or the radius required for an edge between vertices)
2. The critical probability in which an infinite cluster exists for different factors of a (or the radius of each Matérn II point)
3. The critical probability in which an infinite cluster exists for different factors of p (the probability an edge exists given distance between points is $\leq r$)
4. A function $\Phi(p, r, a)$ that approximates the probability Φ that an infinite cluster exists.
5. The algorithmic complexity of the simulation.

Motivation

1. The study of graphs is incredibly important to the fields of epidemiology, some fields of botany, and networking.
2. With increasingly difficult problems, simulation as a method for solution is becoming increasingly common.
3. Matérn II point processes are a great way to simulate real objects that take up space.

Method of Determining Solutions

I used the following methods to complete my research objectives.

1. Develop a simulation in C++ to determine if there exists a connection to a vertex outside of a set radius from the origin given different factor levels of p, r, and a.
2. Analyze the results in R, and determine a function for Φ while checking the residuals of the estimated function.
3. Hone in on a close range in which critical probabilities reside.
4. I will analytically solve for best and worst case complexities of the algorithm and use Visual Studio analytics for average time complexity.

Motivation for Analysis

It's important to find a solution that will quickly simulate this process so that enough replicates can be found for strong certainty.

Example of simulation time problem:

1. Time Passed for 1000 R Simulations in Parallel: ≈Six Hours
2. Time Passed for 1000 C++ Simulations in Parallel: ≈Three Minutes

This comes from the methods used in R packages not being optimized complexity-wise and compiler optimizations.

Complexity Analysis: C++ Algorithm

Primary complexity of C++ algorithm comes from the Matérn II Thinning Process.

- Primary Operation - Comparison: Distance(Point1, Point2 ¡ a)
- (i from 1 to n - 1) and (j from i + 1 to n)
- Computations $= \frac{n(n+1)}{2}$

The other complexity comes from creating the adjacency matrix

- Primary Operation - Allocating to Adjacency Matrix by Comparison
- distance(point1, point2) ¡ r and (random number between 0 and 1) ¡ p
- (i from 1 to $n_{II} - 1$ and (j from i + 1 to n_{II})
- Computations $= 2(\frac{n_{II}(n_{II}+1)}{2}) = n_{II}(n_{II} + 1)$

Note that average graph traversal time is n

Overall Complexity

Note that n is the initial number of points while n_{II} is the number of points after Matérn II thinning.

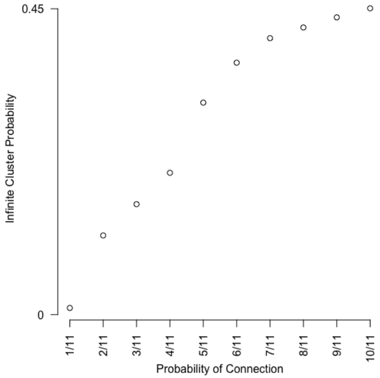
$$\frac{n(n+1)}{2} + n_{II}(n_{II}+1) + n_{II} \in \Theta(n^2)$$

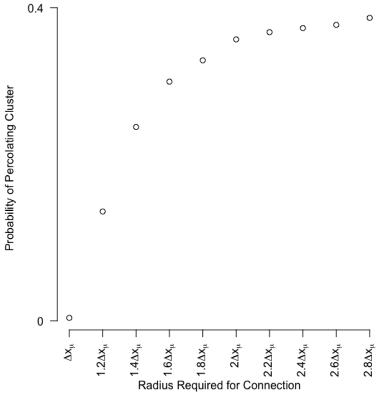
R Problems:

- Lack of compiler optimization and spatial usage
- R-algorithm to check for paths in graph structure also checked for shortest path with Floyd's algorithm by default which is an $\Theta(n^3)$ algorithm.

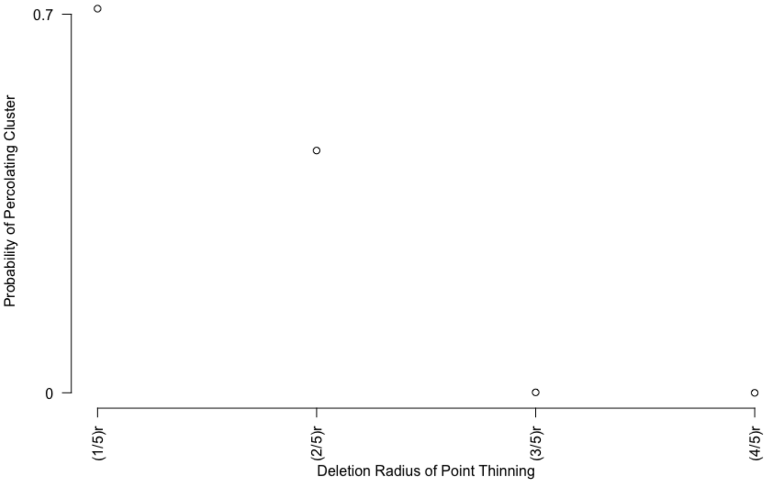
For $n > 1, \Theta(n^2) < \Theta(n^3)$, so algorithm speed is improved by a degree of n times not including spacial complexity issues.

Visual Results

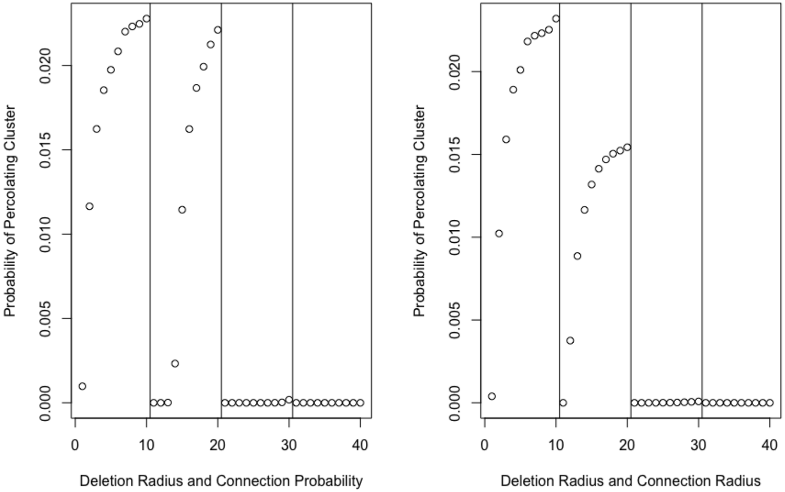




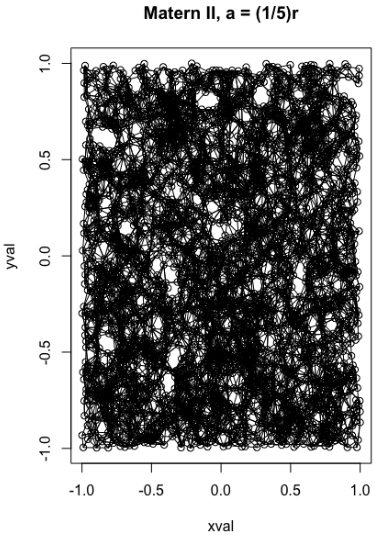
Visual Results

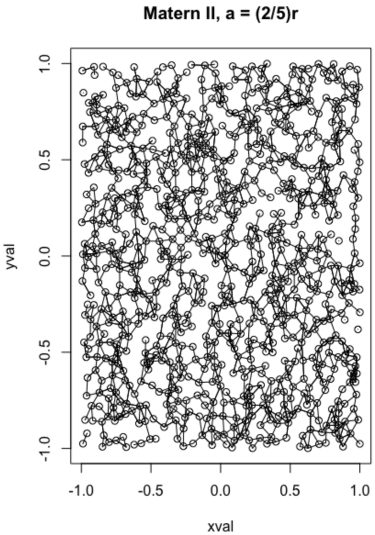


Visual Results

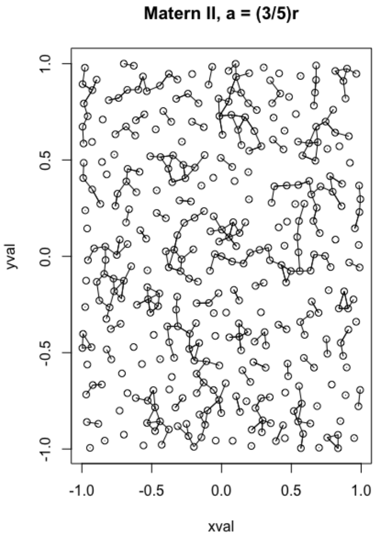


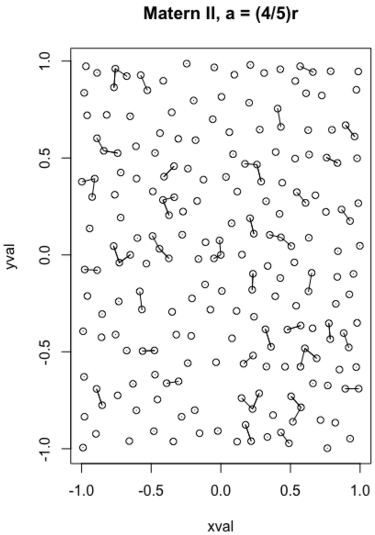
Visual Results





Visual Results





Visual Results

1. The effect of a on p and r's probability is pretty clear
2. Logistic Model will need to account for the interactions of a

Logistic Summary

Full logistic model showed that p and $p : a$ had little impact on model. Upon removal, the model is thus:

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -11.822189   0.065949 -179.26   <2e-16 ***
r             1.640490   0.012455  131.71   <2e-16 ***
a             2.562077   0.028343   90.39   <2e-16 ***
r:p           0.963367   0.003870  248.93   <2e-16 ***
r:a          -1.566532   0.009616 -162.91   <2e-16 ***
r:a:p        -0.219323   0.001178 -186.20   <2e-16 ***
```

Model Information

1. Model formulated based on training set of 2500 replicates per factor level.
2. Model training on data set that also has 2500 replicates per factor level.
3. Does not account for any potential phase shifts that might have seemed clear in visual representation
4. Model had accuracy on testing set of 97.9%.

$$\Phi(p_c, r_c, a_c) = -11.822 + 1.64r_c + 2.56a_c + .96r_c p_c - 1.57r_c a_c - .21r_c a_c p_c$$

for a final decision boundary of

$$p(\Phi) = \frac{1}{1 + e^{-\Phi}} \qquad \text{Cluster} \begin{cases} \text{exists,} & p(\Phi) \geq 0.5 \\ \text{does not exist,} & o/w \end{cases}$$

Conclusion

1. The model, while already accurate, received some improvement by accounting for what appeared to be a clear phase shift in a.

$$\Phi_{Imp.} = \Phi(p, r, a)\mathbb{1}\{a < 1/2\}$$

2. Other phase shifts were not seen in this simulation but hopefully others can be found later
3. $\Phi_{Imp.}$ improved the accuracy of the model by about .2% giving a final confidence interval of accuracy at (97.2%, 99.0%) with 95% confidence.
4. Using new knowledge of Generalized Linear Models, this model can likely be improved(or simplified).

References

1. Broadbent, S.; Hammersly, J. (1957), Percolation processes I. Crystals and mazes, Proceedings of the Cambridge Philosophical Society 53: 629641
2. Matérn, B.; Spatial Variation
3. Penrose, M. Connectivity of soft random geometric graphs. The Annals of Applied Probability. 2016, 2, 968-1028
4. Dr. Matthew Jones's Wisdom

Acknowledgments

- Austin Peay State University Department of Mathematics and Statistics
- Dr. Jones for being a constant support network when I struggled
- The Department of Computer Science for letting me use their machines for months to simulate results
- Dr. Elarde for helping me parallelize my code