Project 2 : TMA4250 : Yawar Mahmood

Problem 1

We will consider three real point pattern datasets with locations in a 2D domain. These are: Cells, Pine trees and Redwood trees. We will be answering different questions around all these datasets.

a)

We start by displaying each point pattern and discuss how the pattern may relate to real processes and behaviour in nature.

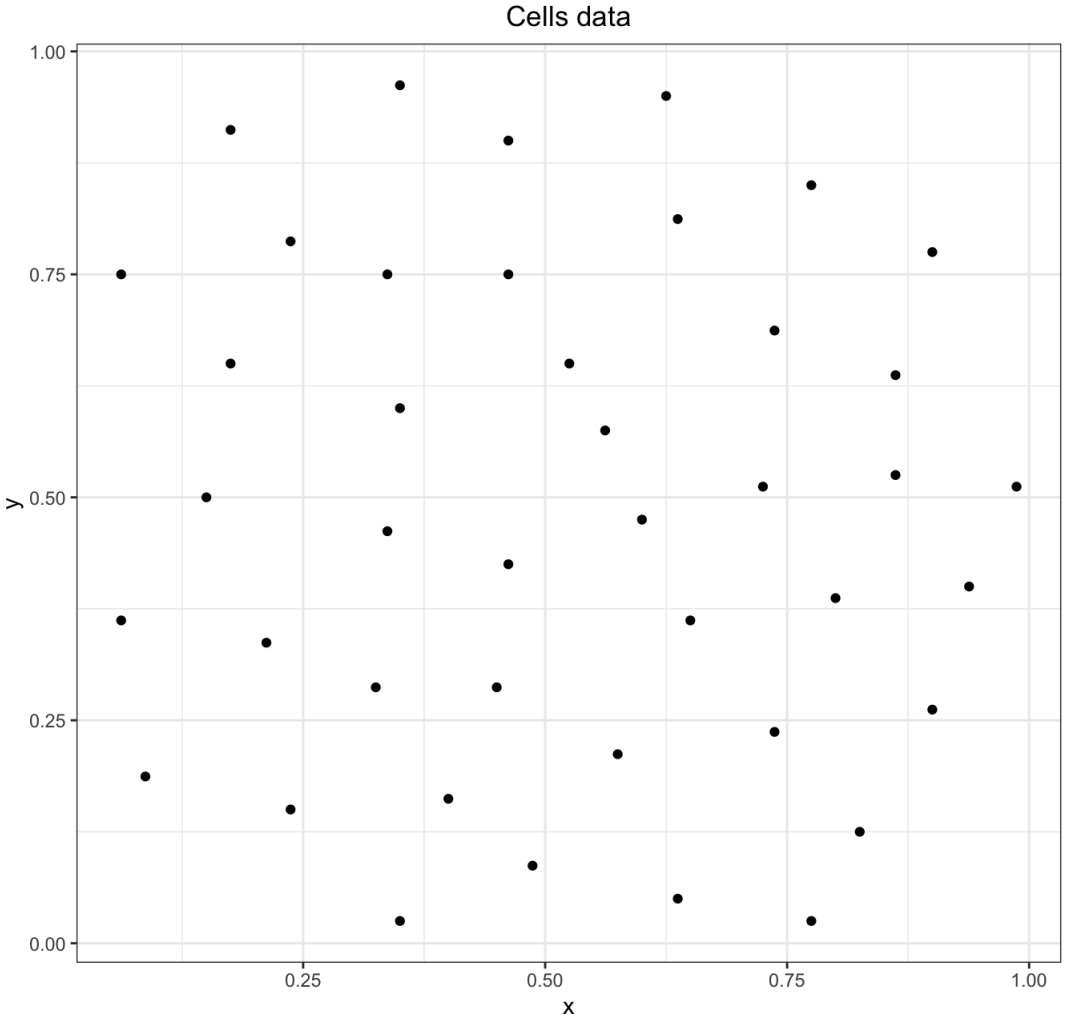


Figure 1: Data points from the Cells data.

The Cells data is evenly distributed, and display repulsion. In this way, the space between each cell is maximized, and is a commom characteristics for cells. If the cells were not evenly distributed, it may have suggested a tumor.

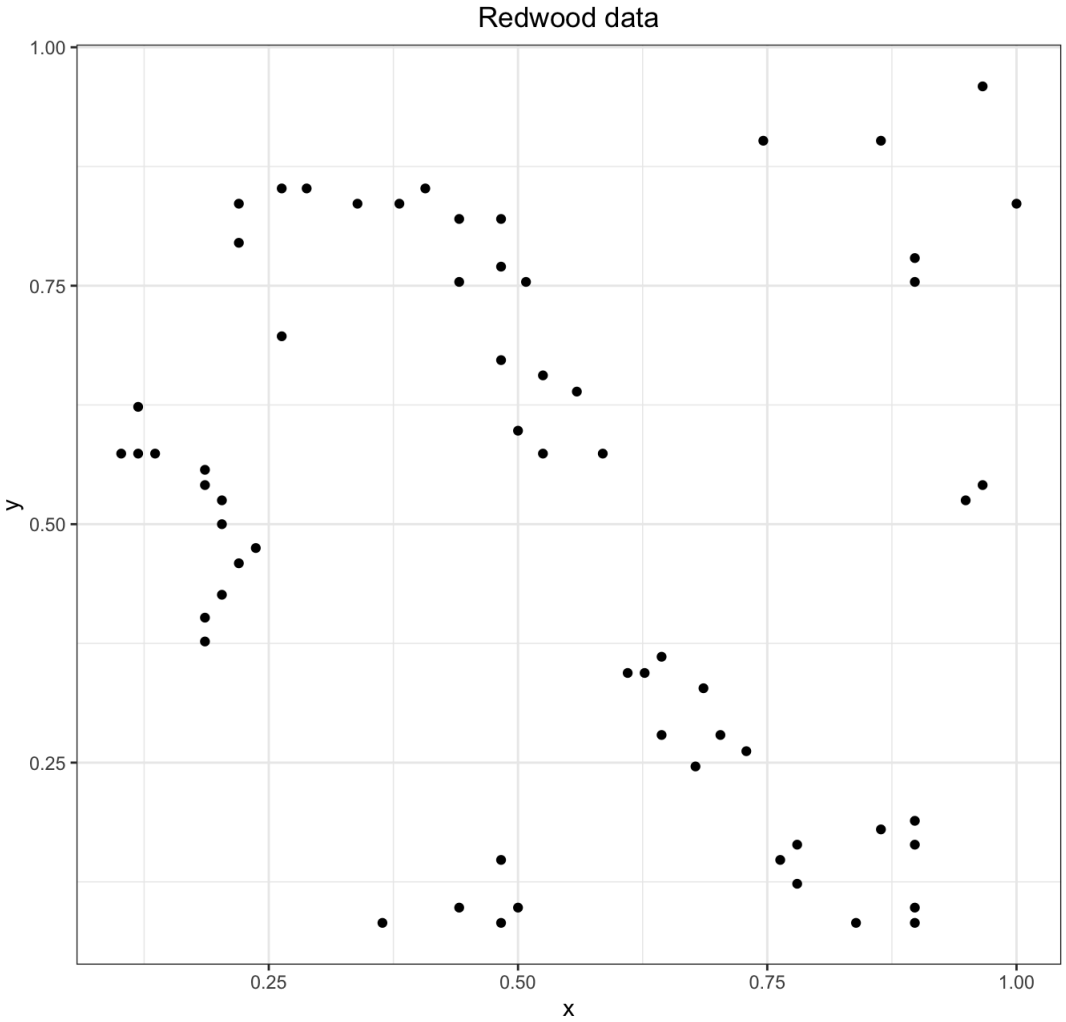


Figure 2: Data points from the Redwood data.

The redwood data shows more clustered tendensies, which can make sence, as young redwood threes grow on their parents roots. As the roots can have limited length, a cluster tendency is obtained.

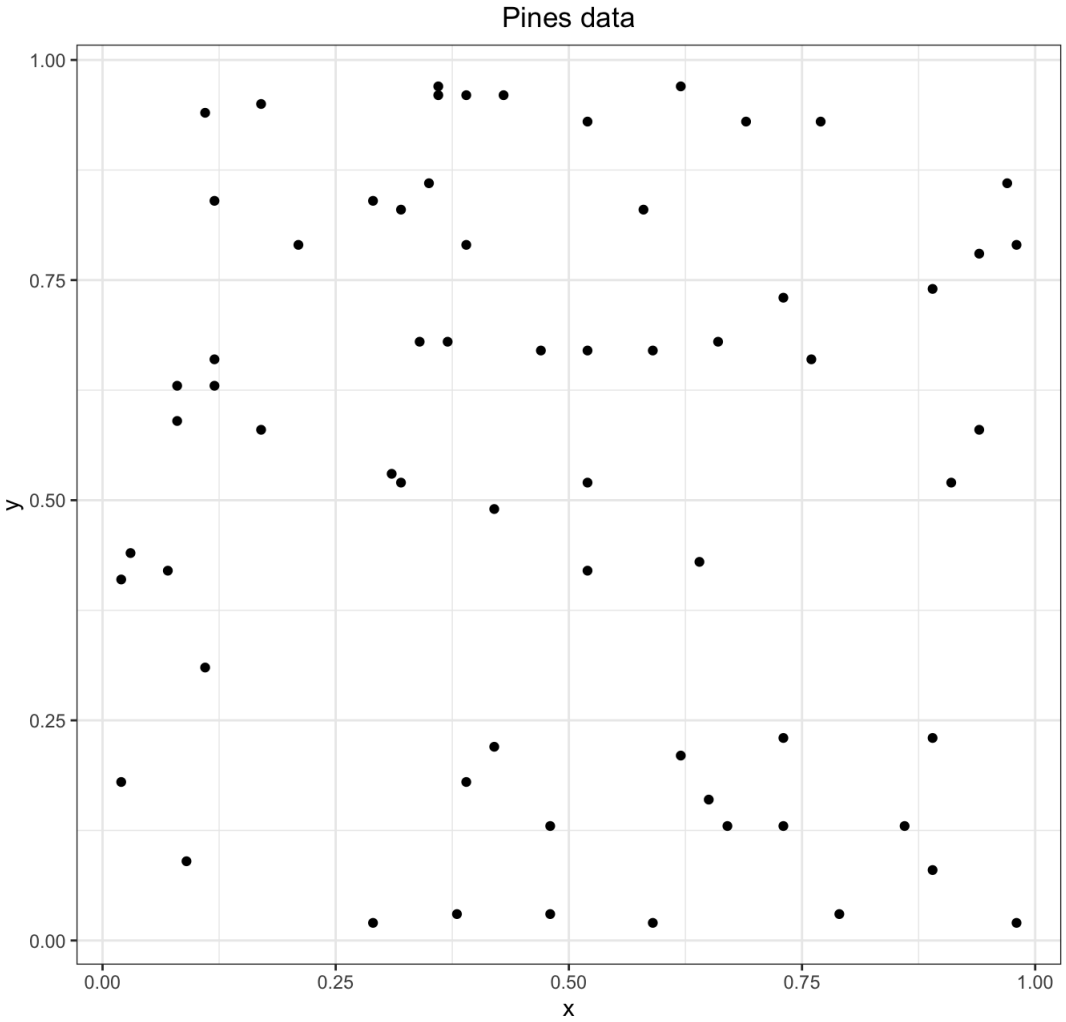


Figure 3: Data points from the Pines data.

The pines data seems to be randomly scattered, which can make sense, as pine trees disperse their seeds usually by wind. Or pine nuts, which are eaten and spread by birds and squirrels. The behavoiur of wind and animals is random, and hence a random distribution of pine trees.

b)

The L-function is used to analyze the distribution of points in a point pattern, helping to identify if a pattern is evenly spaced, random or clustered. For a homogenous Poission Point Process, which serves as a complete random spatial model, the theoretical L-function is a straight line given by $L(t) = t$, where t is the distance from a point.

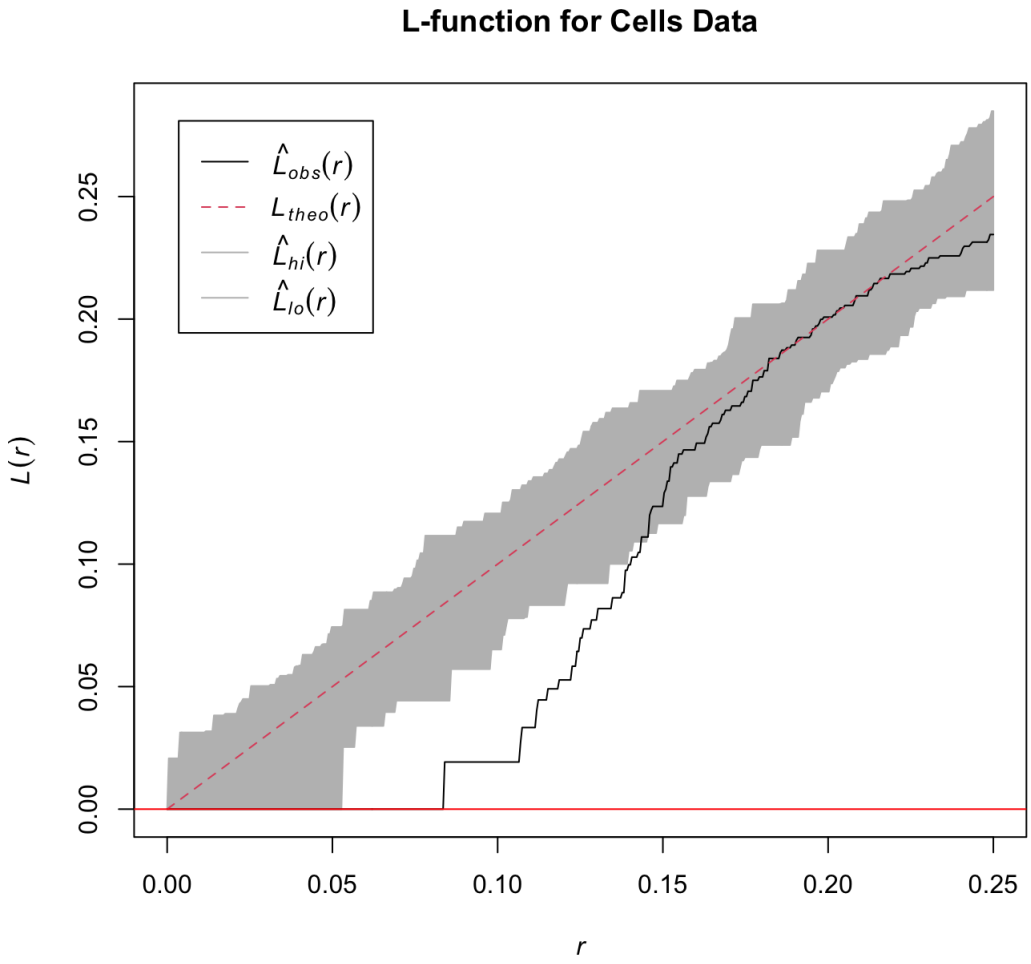


Figure 4: Data points from the Pines data.

If the L-function lies below the theoretical line t, indicates repulsion or points being evenly distributed. This holds up with the discussion about the pattern itself.

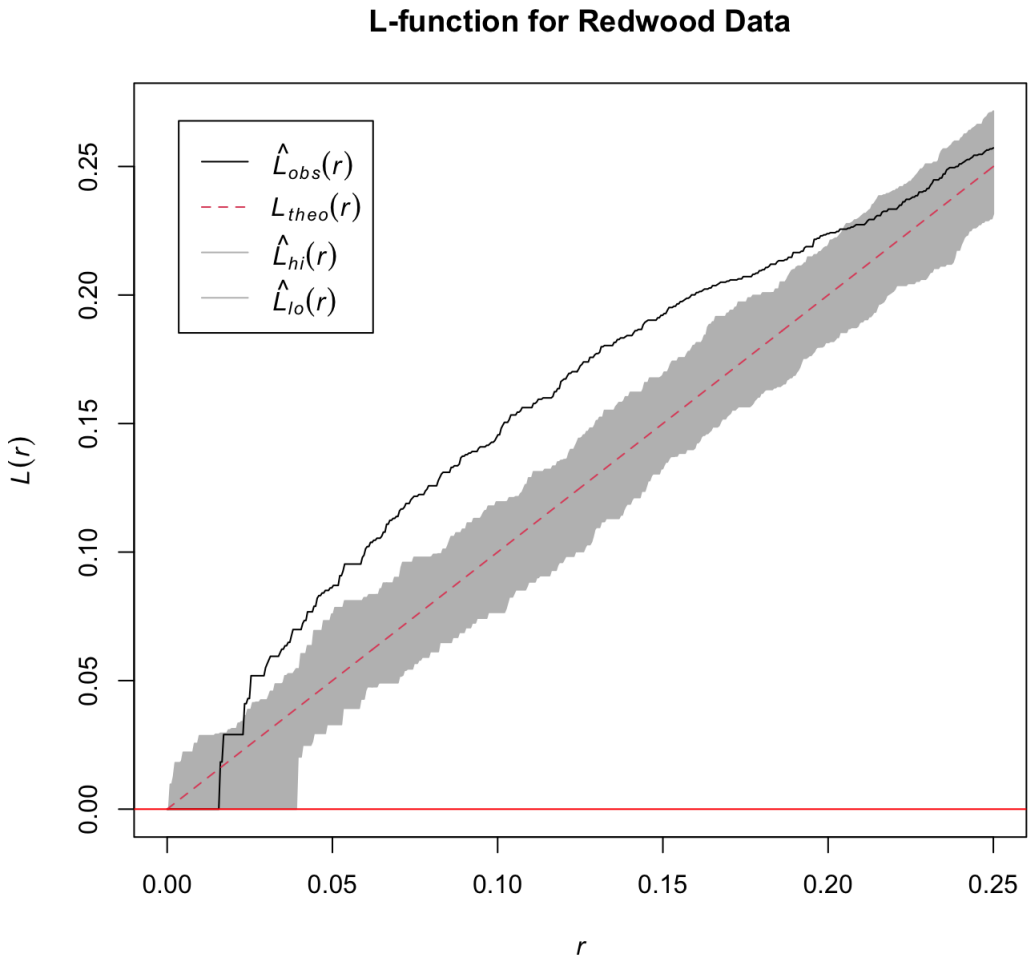


Figure 5: Data points from the Pines data.

If the L-function lies above the theoretical line t, it suggests clustering. This holds up with the discussion about the pattern itself.

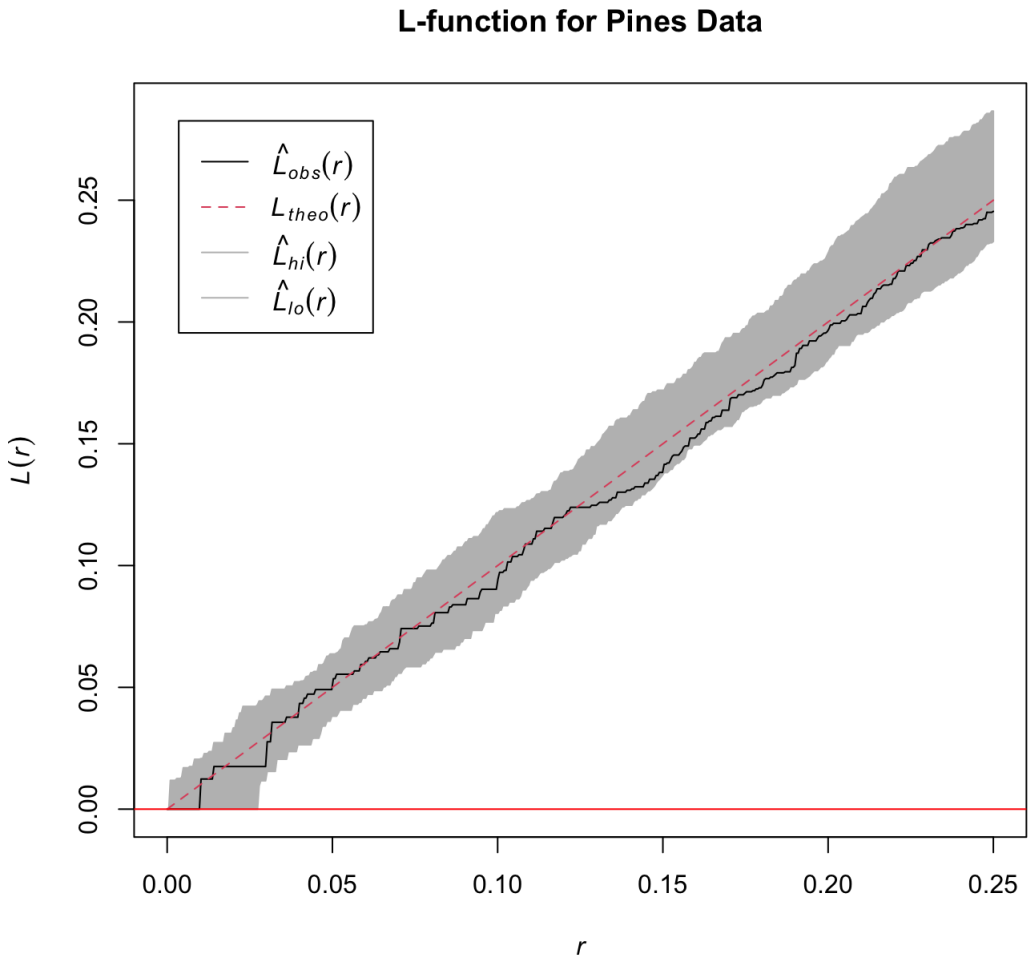


Figure 6: Data points from the Pines data.

When the L-function follows the theoretical line t closely, it suggests that the points may follow a random distribution, indicating a homogenous Poisson Process. This holds up with the discussion about the pattern itself.

c)

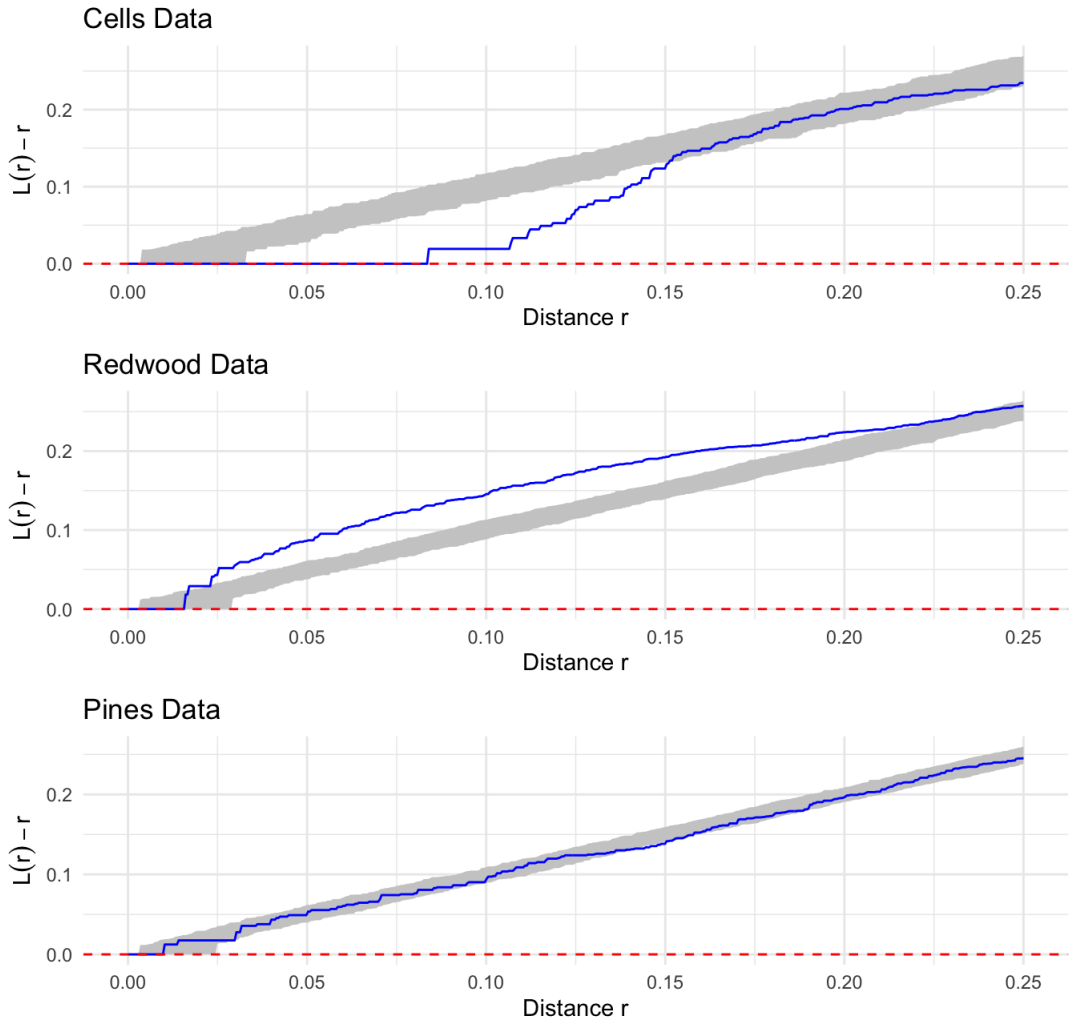


Figure 7: The empirical L-function for each dataset alongside the 90% prediction intervals based on 100 simulations of a homogenous poission point process.

Looking at the cells data, the empirical L-function lies below and outside the 90% prediction interval for smaller values, before it converges with larger distances. All in all, this suggests that the homogenous Poisson Process is not a reasonable model. We will use Strauss repulsion to model this process.

A similar result is obtained for the Redwood data. Here one should use the neymann-scott model to model the clustering.

The only data set which is reasonable to assume a Poisson point process for is the pine dataset, which lies within the 90% confidence interval for all distances.

Problem 2

a)

Let M_{ij} denote the detected number of pine trees, N_{ij} denote the true number of pine trees, and α_{ij} denote the detection probability in grid cell (i, j).

Given that pine trees are detected independently, the observation model $M_{ij}|N_{ij}$ follows a binomial distribution, since each tree has a probability α_{ij} of being detected.

$$M_{ij}|N_{ij} \sim \text{Binomial}(N_{ij}, \alpha_{ij})$$

The probability mass function of $M_{ij}|N_{ij}$ is then:

$$P(M_{ij} = m|N_{ij} = n) = \binom{n}{m} \alpha_{ij}^m (1 - \alpha_{ij})^{n-m}$$

for $m = 0, 1, ..., n$.

In other words, this model assumes that the detection of each tree is an independent Bernoulli trial with success probability α_{ij}.

Let us now display the dataset, that is, the detection probability:

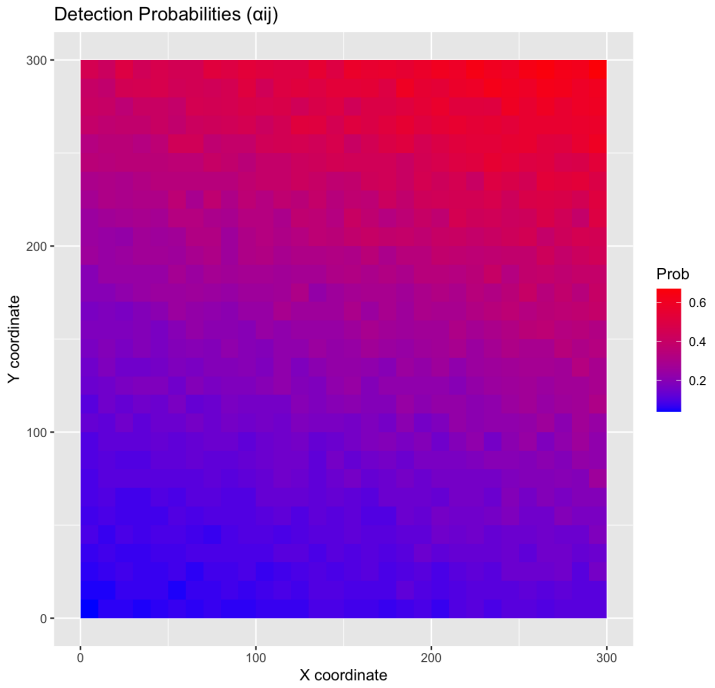


Figure 8: The detection probability

and the counts of detected pine trees:

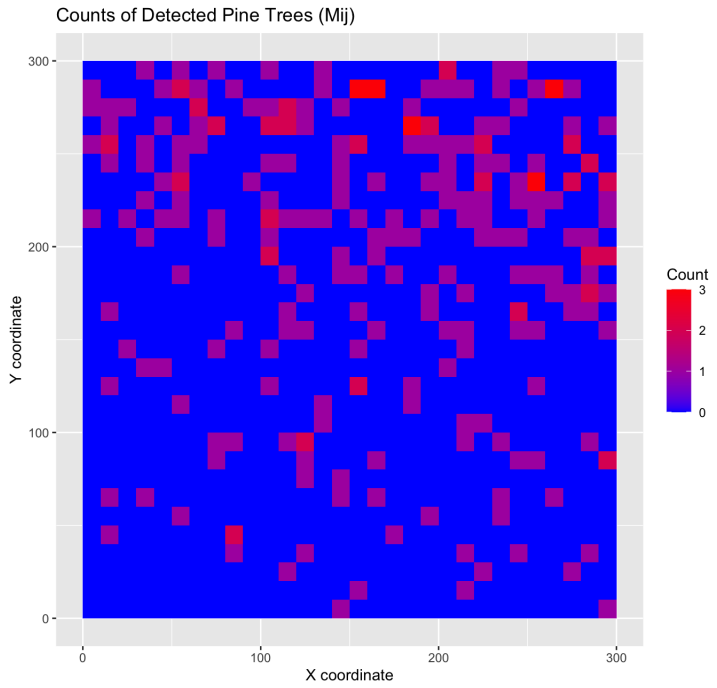


Figure 9: The counts of detected pine trees

These are some of the points one can bring out of such graphs:

- Areas with high detection probabilities that also show high counts suggest that the high number of trees detected is likely accurate.
- Areas with high detection probabilities but low counts could indicate that there are genuinely fewer pine trees in those areas.
- Areas with low detection probabilities but high counts may suggest that the actual number of trees could be even higher, as some may have been missed due to the lower probability of detection.
- Areas with low detection probabilities and low counts could either truly have few or no pine trees, or there could be trees present that were not detected due to the low probability.

One thing is for sure, there are more pines detected where the observation probability is high, and fewer detected where the probability is low.

b)

Assuming a homogeneous Poisson point process, the number pine trees, in any region follows a Poisson distribution. The mean is given by the product of the region's area A and the intensity λ of the process.

Since the observation window is divided into a regular grid of $10mx10m$ cells, each cell with an area of $100m^2$. The expected number of pine trees in each cell is then: 100λ. For each grid cell (i, j), the number of pine trees N_{ij} follows:

$$N_{ij} \sim \text{Poisson}(100\lambda)$$

The Probability mass function of N_{ij} is then given by:

$$P(N_{ij} = n) = \frac{e^{-100\lambda}(100\lambda)^n}{n!}$$

for $n = 0, 1, 2, \ldots$

The joint PMF can easily be found through the fact that the number of points in disjoint subsets of the observation window are independent. Then:

$$P(\mathbf{N} = \mathbf{n}) = \prod_{i=1}^{30}\prod_{j=1}^{30} P(N_{ij} = n_{ij}) = \prod_{i=1}^{30}\prod_{j=1}^{30} \frac{e^{-\lambda\times 100}(\lambda \times 100)^{n_{ij}}}{n_{ij}!}$$

This model and its PMF describe the distribution of the true numbers of pine trees across the observation window under the given assumptions.

c)

To determine an unbiased estimator $\hat{\Lambda}_2$ for λ, we can utilize the observed counts M_{ij} and the detection probabilities α_{ij} for each grid cell. Given the observation window's area of $300\,mx300\,m = 90000\,m^2$ and assuming the independence of detection events, the estimator given by:

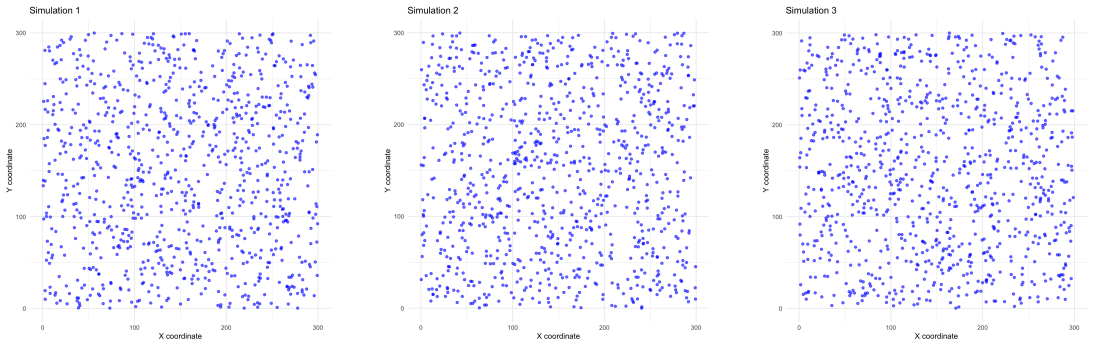
$$\hat{\Lambda}_2 = C(\alpha) \cdot \sum_{i,j} M_{ij}$$

where $C = \frac{1}{\bar{\alpha}}$, where $\bar{\alpha}$ is the average of the detection probabilities α_{ij}.

All in all, we can write it as:

$$\hat{\lambda} = \frac{1}{A} \cdot C \cdot \sum_{i,j} M_{ij} = \frac{1}{90000} \cdot \frac{1}{\bar{\alpha}} \cdot \sum_{i,j} M_{ij}$$

We now want to simulate 3 realizations of the point pattern. We calculate our estimator for λ, which is calculated to 0.0104077.



(a) Simulation 1 (b) Simulation 2 (c) Simulation 3

Figure 10: Simulation results for three different cases.

We see that the points for each plot randomly distributed, suggesting a Poisson Point Process.

d)

N_{ij} follows a Poisson distribution with parameter λA, where A is the area of a grid cell and λ is the intensity of the Poisson point process. The PMF is then:

$$P(N_{ij} = n) = \frac{e^{-\lambda A}(\lambda A)^n}{n!}$$

$M_{ij}|N_{ij} = n$ follows a Binomial distribution with parameters n and α_{ij}, representing the number of successes out of n trials with success probability α_{ij}. The PMF is then:

$$P(M_{ij} = m|N_{ij} = n) = \binom{n}{m} \alpha_{ij}^m (1 - \alpha_{ij})^{n-m}$$

Using Bayes' theorem, the PMF for $N_{ij}|M_{ij} = m$ is:

$$P(N_{ij} = n|M_{ij} = m) = \frac{P(M_{ij} = m|N_{ij} = n)P(N_{ij} = n)}{P(M_{ij} = m)}$$

To find $P(M_{ij} = m)$, we sum over all possible values of n:

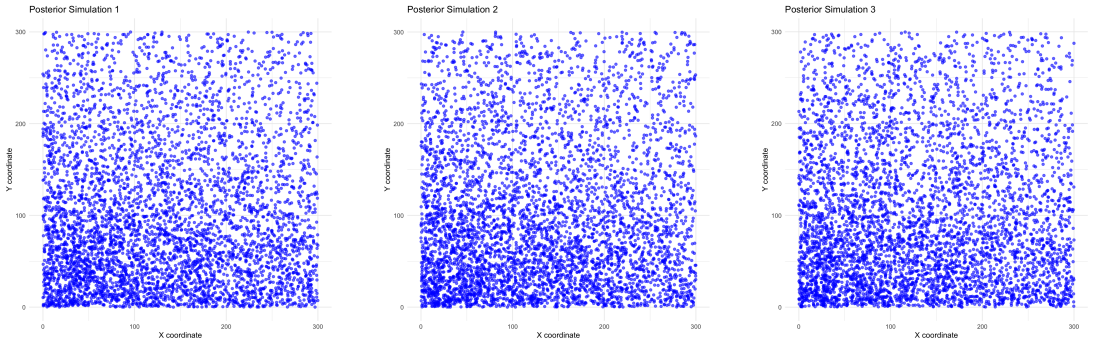
$$P(M_{ij} = m) = \sum_{n=m}^{\infty} P(M_{ij} = m|N_{ij} = n)P(N_{ij} = n)$$

Substituting the expressions for the Poisson and Binomial PMFs, we get:

$$P(M_{ij} = m) = \sum_{n=m}^{\infty} \binom{n}{m} \alpha_{ij}^m (1 - \alpha_{ij})^{n-m} \frac{e^{-\lambda A}(\lambda A)^n}{n!}$$

Thus, the PMF for $N_{ij}|M_{ij} = m$ becomes:

$$P(N_{ij} = n|M_{ij} = m) = \frac{\binom{n}{m} \alpha_{ij}^m (1 - \alpha_{ij})^{n-m} \frac{e^{-\lambda A}(\lambda A)^n}{n!}}{\sum_{n=m}^{\infty} \binom{n}{m} \alpha_{ij}^m (1 - \alpha_{ij})^{n-m} \frac{e^{-\lambda A}(\lambda A)^n}{n!}}$$



(a) Simulation 1 (b) Simulation 2 (c) Simulation 3

Figure 11: Simulation results for three different cases.

We see that compared to the previous simulations without the posterior knowledge, the density of pines are bigger in the bottom region of the area. Why this is, I am unsure.

e)

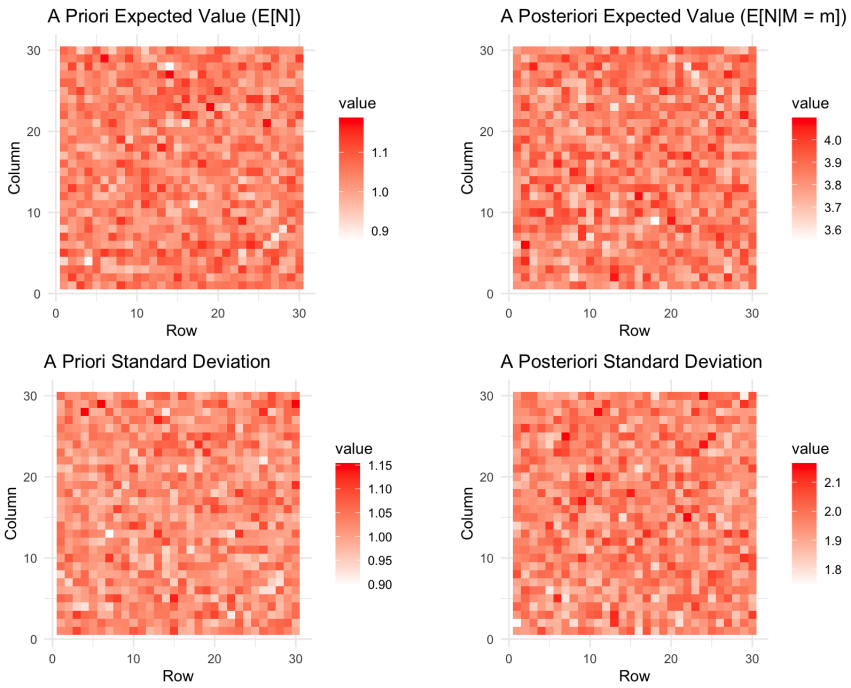


Figure 12: A Priori and A Posteriori Estimated Expected Values and Standard Deviations

The priori expected values (top left) display a uniform distribution of expected counts, with some variations, which most likely reflect the randomness of the Poisson Process.

The priori standard deviations (bottom left) is also uniform, indicating consistent uncertainty acorss the grid.

The posteriori values (top right) have more variability in certain cells showing higher expected counts. This is a result of that the posteriori is conditioned on the observed data.

The posteriori standard deviations (bottom right) vary significantly, with lower standard deviations in cells with more observed points, indicating a higher certainty in these regions due to observed data.

Problem 3

a)

The Neyman-Scott process is a type of cluster point process model used to describe patterns where points are clustered around parent points.

The process starts with a set of parent points that are distributed according to a homogeneous Poisson process. Each parent point then independently generates a random number of daughter points.

The locations of these daughter points are distributed around their parent point according to a normal distribution $N_2(\mathbf{y}, \sigma^2\mathbf{I}_2)$, where $\mathbf{y}$ is the location of the parent point, σ^2 is the variance, and $\mathbf{I}_2$ is the $2x2$ identity matrix. This distribution ensures that daughter points are centered around their parent points, with the spread determined by σ^2.

Let us now look at the model parameters:

- **Lambda** (λ): The intensity of the parent process. It determines the average number of parent points per unit area. Determines numbers of clusters.
- **Mu** (μ): The average number of daughter points per parent, governed by the Poisson distribution. Determines cluster size.
- **Sigma** (σ): The standard deviation gives a indication of the distances of daughter points from their parent, drawn according to a normal distribution. Determines spread of the cluster, the smaller, the tighter the clusters are.

Lets look at how the model works:

1. **Parent Generation**: Initially, parent points are generated according to a Poisson process with intensity λ.
2. **Daughter Generation**: For each parent point, a random number of daughter points are generated. The number of daughters for each parent follows a Poisson distribution with mean μ.
3. **Placement of Daughter Points**: The locations of daughter points relative to their parent point are determined by a normal distribution $N_2(\mathbf{y}, \sigma^2\mathbf{I}_2)$.

Let us now look at potential border problems caused by a finite domain.

Within a finite domain $W \subset \mathbb{R}^2$, border effects present challenges that can influence the accuracy of the simulations.

These constraints placed on the observation window may affect both the distribution of parent points near the borders and hence the placement of daughter points.

- Since parent points can be located close to the borders, their associated daughter points, which are distributed around them, may fall outside the observation window. This can result in a biased distribution of observed daughter points.
- All this can effect the estimation of the Neyman-Scott process parameters, such as the intensity of the parent process (λ) and the average number of daughter points per parent (μ).

We now want to create some realizations of the process. To do so, we must specify the model parameters discussed above.

Looking at the data in Figure 2, we can try to guesstimate the different model parameters.

- **Parent Intensity** (λ): Represents the average number of parent trees or clusters per unit area. There are few clusters, approximately 5-6 clear clusters within the area. A resonable value for λ may then be $0.05 - 0.06$
- **Mean Offspring Per Parent** (μ): Given the variable cluster sizes, some as big as 8-10 while other as small as 3-5, a conservative average is estimated to be around 6 offspring per parent.
- **Offspring Dispersion** (σ): Describes how far offspring are spread around their parent tree. The clusters appear tightly packed with offspring within approximately 0.1, indicating a low dispersion.

We run simulations with these values and get:

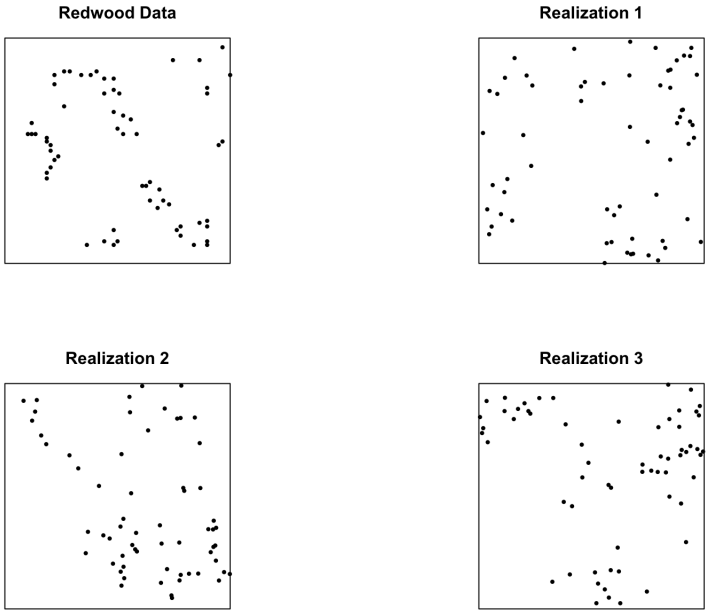


Figure 13: Realizations of simulations ran with $\lambda = 0.05$, $\mu = 6$, and $\sigma = 0.1$.

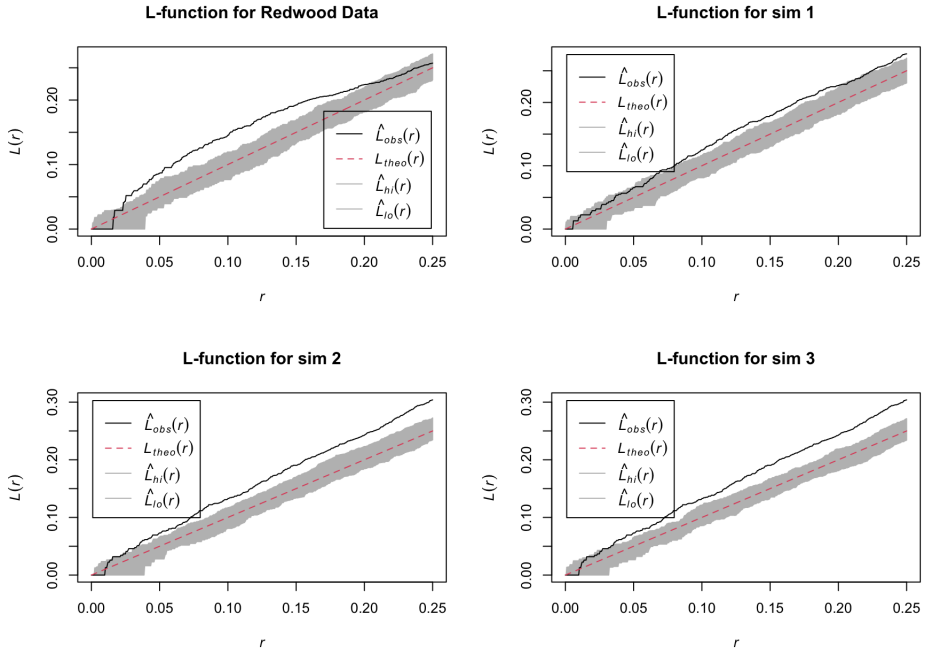


Figure 14: L-functions of simulations ran with $\lambda = 0.05$, $\mu = 6$, and $\sigma = 0.1$.

The results for the L-functions look pretty good, indicating that these are reasonable values for the parameters. Additonally, the realizations seem also to mimic the original dataset in the way it clusters.

Problem 4

a)

The Strauss process is particularly suited for patterns exhibiting repulsion or uniform spred of data points. The interaction between points in a Strauss process is characterized by a pair potential function, influencing the likelihood of finding another point at a given distance from it. It is given that our potential function is given as:

$$\phi(r) = \begin{cases} \beta & \text{if } r \leq r_0, \\ 0 & \text{if } r > r_0. \end{cases}$$

Here, some of the model parameters are given. We now look at all the model parameters of a Strauss-process:

1. **Intensity** (λ): The average density of points within the observation window. Higher values indicate a denser average point distribution.
2. **Interaction parameter** (β): Quantifies the interaction strength between points. $\beta < 1$ indicates repulsion between points within a certain distance. $\beta = 1$ indicates no interaction, reducing the process to a Poisson process. $\beta > 1$ would indicate attraction. The bigger the absolute value, the bigger the repulsion/attraction.
3. **Interaction radius** (r_0): The distance within which the interaction between points is significant. Points further apart than r_0 do not directly influence each other's occurrence. The bigger the value, the bigger the spread between points.

This implementation, as the one discussed in the previous problem, also has issues with a bounded observation window $W \subset \mathbb{R}^2$. Here are some possible issues:

- Points near the boundary have part of their interaction neighborhood outside the observable window.
- This scenario can bias estimates of interaction effects and intensity.

We now want to create some realizations of the process. To do so, we must specify the model parameters discussed above.

Looking at the data in Figure 1, we can try to guesstimate the different model parameters.

- **Interaction radius** (r_0): In our cell data plot, the data has clear gaps, indicating repulsion. If we divide the unit square into smaller sections, it appears that r_0 could be about 0.1 to 0.2, given that points tend to avoid getting closer than this distance. We set r_0 to 0.1
- **Intensity** (λ): This is the average number of points per unit area. To estimate λ, count the number of points and divide by the area. In this plot, there are roughly 40 points, then $\lambda = 40$.
- **Interaction parameter** (β): If points are forbidden to be within the distance r_0 of each other, β should be set to a very low value close to 0 (as $\beta = e^{-\beta}$ in the Strauss model, and we want this to approach 0 for "forbidden distances"). Based on the plot, where some points are close but not too close, we guess $\beta = 2$.

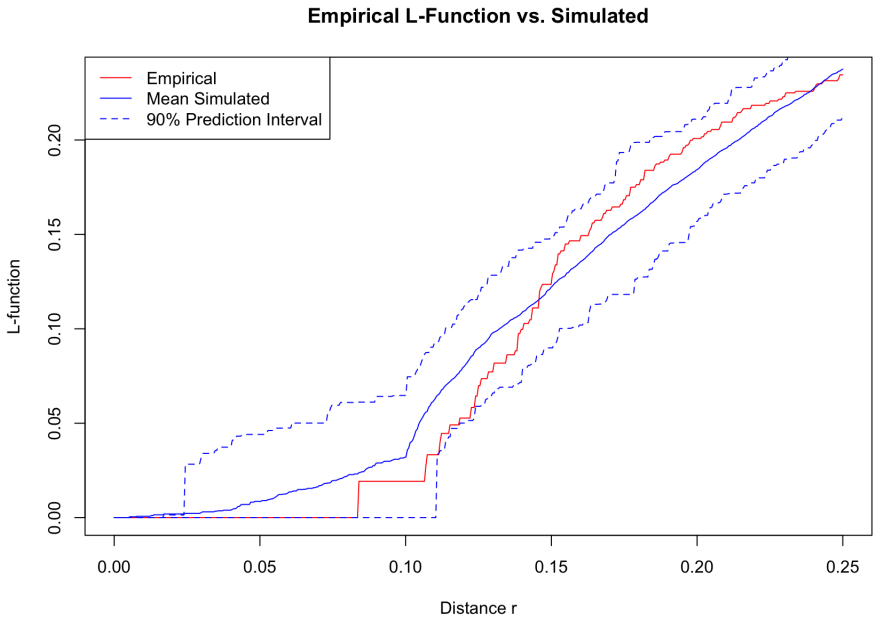


Figure 15: L-functions of simulations ran with $\lambda = 40$, $r_0 = 0.1$, and $\beta = 2$.

We see that the emperical and simulated L-functions line up pretty well, and this suggests that the guesstimated values for the parameters are not fully off.