



Losing the sensation of touch: Mathematical modeling of diabetic neuropathy using spatial point processes

Att tappa känseln: Matematisk modellering av
diabetesneuropati med hjälp av spatiala
punktprocesser

Examensarbete för kandidatexamen i matematisk statistik vid Göteborgs universitet

Kandidatarbete inom civilingenjörsutbildningen vid Chalmers

Sirada Kaewchino
Maximilian Nylander
Jadd Ujam

Losing the sensation of touch:
Mathematical modeling of diabetic
neuropathy using spatial point processes

Examensarbete för kandidatexamen i matematisk statistik vid Göteborgs universitet
Jadd Ujam Maximilian Nylander

Kandidatarbete i matematik inom civilingenjörsprogrammet Bioteknik vid Chalmers
Sirada Kaewchino

Handledare: Konstantinos Konstantinou

Institutionen för Matematiska vetenskaper
CHALMERS TEKNISKA HÖGSKOLA
GÖTEBORGS UNIVERSITET
Göteborg, Sverige 2022

Preface

The bachelor's thesis has been carried out at the Department of Mathematical Sciences at Chalmers University of Technology. First and foremost, we would like to thank our supervisor Konstantinos Konstantinou for his dedicated commitment and encouraging guidance during the work. We also thank the examiners, Ulla Dinger and Maria Roginskaya. Last but not least, we thank everyone who contributed with comments below tutoring sessions or in reading exchanges.

To begin with, the group believes that all members have contributed equally to the project. All team members' achievements have been recorded in a logbook. This logbook is not included here but below are the group members' individual contributions to each section of the text in the report, as well as the primary division of responsibilities during which the work has been carried out.

§	Section	Main author
-	Popular science presentation	Sirada
-	Abstract and Sammanfattning	Sirada
1	Introduction	Sirada
1.1	Diabetes	Sirada
1.2	Statistical scope	Jadd
1.3	Purpose	Sirada
1.4	Problem	Sirada
1.5	Data	Sirada
1.6	Delimitation	Sirada
2.1	Bootstrapping	Sirada
2.2	Gaussian distribution	Sirada
2.3	Euclidean distance	Maximilian
2.4	Homogeneous Poisson Processes	Maximilian
2.5	Thinning	Maximilian
2.6	Ripley's K-function and L-function	Maximilian
2.7	Isotropic edge correction	Maximilian
2.8	Global envelope test	Jadd
2.9	Convex Hull	Jadd
3.1	Modelling process	Jadd
3.1.1	Model 1: Independent random thinning model	Sirada
3.1.2	Model 2: Deterministic thinning model	Maximilian
3.1.3	Model 3: Gaussian thinning model	Jadd
3.2	Sustainable and ethical aspects	Maximilian
4.1	Pre-Investigation of the data	Maximilian
4.2	Non-spatial results	Maximilian
4.2.1	Model 1	Maximilian
4.2.2	Model 2	Maximilian
4.2.3	Model 3	Maximilian
4.3	Spatial results	Jadd
4.4	L-function for the models	Jadd
4.4.1	Model 1	Jadd
4.4.2	Model 2	Jadd
4.4.3	Model 3	Jadd
5	Concluding discussion	Jadd

Popular science presentation

The number of diabetes cases increases every year. It has been estimated that 700 million people will have diabetes by 2045 [1]. Diabetic neuropathy is a nerve disease that half of all diabetic patients develop. The nerve disease causes nerve damage or dysfunction, and as a result, patients suffer from pain and loss of sensation. The cause of diabetic neuropathy is primarily thought to be correlated with prolonged uncontrolled high blood sugar. However, the specific reason for the degeneration of nerve cells is yet unknown. This project aims to determine if the nerve deterioration follows specific patterns of healthy individuals with those of patients diagnosed with mild diabetes.

Neuropathy indicates a problem within the peripheral nervous system: the network of nerves outside of the brain and spinal cord. Peripheral neuropathy is caused by the damage or dysfunction of one or more peripheral nerves. This can lead to weakness, numbness and pain. Usually, the first incidents of neuropathy occurs in the hands and feet but can later affect other areas and body functions. The nerves send electrochemical signals all over the body and allow us to see, hear, smell and feel. Neuropathy may therefore disrupt the communication between the neurons and the brain. One substantial difference between nerve cells and other cells in the body is that nerve cells have threadlike outgrowths that lead signals to and from the nerve cells via an axon. More extensive axons are surrounded by an electric isolation layer of myelin, a fat substance. Diabetic neuropathy is when the axon worsens, and the myelin layer is damaged, as seen in Figure 1. Early diagnosis of diabetic neuropathy gives patients the best chance of effective treatment. However, accurate diagnosis is vital to ensure appropriate treatment since not all feet or limb pain is due to diabetic neuropathy [2]. Early discovery of diabetic neuropathy can reduce the risk of complications but unfortunately there is currently no method to detect neuropathy before the symptoms occur.

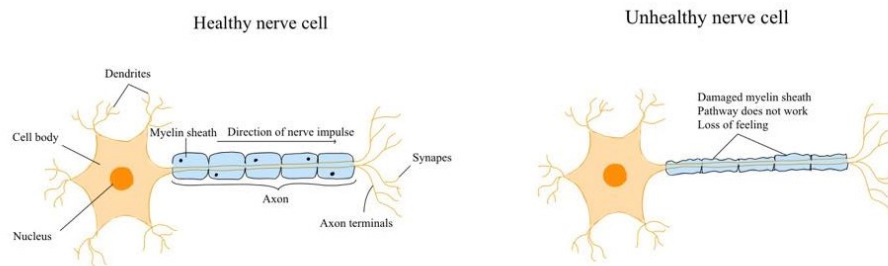


Figure 1: An illustration of a healthy nerve cell and an unhealthy nerve cell.

The provided data consists of nerve point locations from healthy volunteers and mildly diabetic patients. Three models that emulate the nerve fibres' morphological changes were developed. The nerve patterns obtained using the models will be compared to the empirical data to see how well they match up. The first model is an independent random thinning model. The main objective of the model is to test if the nerve thinning happens randomly and independently of the other points. The results show that the nerve thinning is not completely random. Therefore, model 2 was developed to be conditional on the location of the base points but without any aspect of randomness. This model showed better promise than the first one, however for a model to be realistic, it needs an aspect of randomness. This means that model 2 is not a viable option to emulate the neuropathy. Model 3 was then created to incorporate randomness into an otherwise deterministic model. The results of model 3 were satisfactory with regards to the statistics is also a more realistic model if applied to a larger data set. However, it is still not perfect as these models depend solely on the thinning of the nerves. Creating a more sophisticated model that implements more spatial variables might answer the question of how the deterioration of nerve cells occurs at the early stages of neuropathy. If we can solve this problem, we get one step closer to solving the puzzle of neuropathy with this answer.

Sammanfattning

Diabetes har lett till en epidemi av komplikationer i samband med denna sjukdom. Diabetisk neuropati, som orsakar smärta och förlust av känsel på grund av förtvinning av nervfibrer, är en av de vanligaste komplikationerna av diabetes. Konfokalmikroskopi har gjort det möjligt att observera nervändarna i den yttre huden hos sjuka patienter. De tenderar att vara mer samlade än hos friska försökspersoner. Därför är det nödvändigt att förstå processen med förtvinning och den spatiala förtunningen av nervfibrer för att upptäcka diabetisk neuropati i ett tidigt skede.

De två huvudhypoteserna som testades är om nervförtvinningen sker slumpmässigt och oberoende av andra punkter och om nervförtvinningen är betingad av andra punkter. Tre matematiska modeller har utvecklats baserade på spatial förtunning. Den första är en oberoende slumpmässig förtvinningsmodell, den andra är en deterministisk förtvinningsmodell och den tredje är en Gaussisk förtvinningsmodell. Den andra och tredje modellen förtvinar punkter beroende på avståndet från baspunkten. När vi utvärderade spatiala statistiken använde vi den centererade L -funktionen som en sammanfattningsfunktion när vi genomförde ett globalt envelopptest med $N = 500$ simuleringar, där vi testar hypotesen under den empiriska datan från patienterna med mild diabetes. Vi utvärderade även de olika modellerna baserat på icke-spatial statistik som kommer att jämföras med datan från patienterna med mild diabetes.

De spatiala resultaten från den första modellen visade att nervförtunning inte sker slumpmässigt och oberoende ($p = 0,01$), således förkastades nollhypotesen för signifikansnivån $\alpha = 0,05$. Nollhypotesen för signifikansnivån $\alpha = 0,05$ kunde inte förkastas utifrån resultatet av den andra modellen ($p = 0,624$) vilket även gäller för den tredje modellen ($p = 0,056$) för signifikansnivån $\alpha = 0,05$. De icke-spatiala resultaten visade att den första modellen är acceptabel om det önskade resultatet är att enbart observera icke-spatiala egenskaper hos datan medan den andra och tredje modellen inte lämpade sig lika väl för de icke-spatiala egenskaperna. En trolig förklaring till varför den andra och tredje modellen presterade sämre i detta avseende kan vara att spatial förtunning inte är en tillräcklig förklaring bakom de underliggande mekanismerna.

Abstract

Diabetes has led to an epidemic of complications associated with this disease. Diabetic neuropathy, which causes pain and loss of sensation due to degeneration of nerve fibers, is one of the most common complications of diabetes. Confocal microscopy made it possible to observe that the nerve endings in the outer skin of sick patients tend to be more clustered than in healthy subjects. Therefore, it is imperative to understand the process of degeneration and the spatial thinning of nerve fibers to detect diabetic neuropathy at an early stage.

The two main hypotheses were tested are whether the nerve thinning occurs randomly and independently of other points and whether the nerve thinning is conditional on the other points. Three mathematical models were developed based on spatial thinning. The first is an independent random thinning model, the second is a deterministic thinning model and the third is a Gaussian thinning model. The second and third models are conditional on the location of the base point and the distance from it. When evaluating the spatial statistics, we used the centered L -function as a summary function when conducting a global envelope test with $N = 500$ simulations, where we tested the hypothesis under the empirical mild data. We also evaluated the different models based on non-spatial statistics which were compared to the mild data.

The spatial results from the first model showed that nerve thinning does not occur randomly and independently ($p = 0.01$), thus rejected the null-hypothesis for significance level $\alpha = 0.05$. The second model could not be rejected under the null-hypothesis ($p = 0.624$) as well as the third model ($p = 0.056$) for significance level $\alpha = 0.05$. The non-spatial results showed that the first model sufficed if the desired outcome is to observe just non-spatial characteristics of the data whilst the second and third model lacked in this area. A likely explanation as to why the second and third models performed worse in the non-spatial regard, may be that spatial thinning isn't a sufficient explanation behind the underlying mechanisms.

Contents

1	Introduction	1
1.1	Diabetes	1
1.2	Statistical scope	2
1.3	Purpose	2
1.4	Problem	2
1.5	Data	3
1.6	Delimitation	3
2	Theory	4
2.1	Bootstrapping	4
2.2	Gaussian distribution	4
2.3	Euclidean distance	4
2.4	Homogeneous Poisson Processes	4
2.5	Thinning	5
2.6	Ripley's K-function & L-function	5
2.7	Isotropic edge correction	6
2.8	Global envelope test	6
2.9	Convex Hull	6
3	Method	7
3.1	Modeling process	7
3.1.1	Model 1: Independent random thinning model	7
3.1.2	Model 2: Deterministic thinning model	7
3.1.3	Model 3: Gaussian thinning model	8
3.2	Sustainable and ethical aspects	9
4	Results	10
4.1	Pre-Investigation of the data	10
4.2	Non-Spatial results	13
4.2.1	Model 1	13
4.2.2	Model 2	13
4.2.3	Model 3	13
4.3	Spatial results	13
4.3.1	L-function for the models	14
4.3.2	Model 1	15
4.3.3	Model 2	16
4.3.4	Model 3	17
5	Concluding discussion	18
	References	20
	Appendix	21

1 Introduction

Epidermal nerve fibers are thin sensory nerve fibers found in the epidermis, the outer layer of the skin. Diabetic neuropathy is a nerve disease that develops in a diabetic patient and can cause loss of sensation. It has been observed that the morphology of the nerve fibres in the epidermis of patients with diabetic neuropathy appears to be more clustered than that of healthy patients. The endpoints and the structure of the nerve fibres are sensors (i.e for heat and pain). As nerve fibres form, they eventually pierce through the epidermis. These are so-called base points, shown in Figure 2. The nerve fibres then extend into the epidermis and branch until they terminate [3].

Spatial point processes are stochastic processes defined in the spatial domain and can be categorized based on their distribution [3]. Therefore, spatial point processes are a suitable mathematical model to describe the spatial structure of the nerve fibers. Processes, where nerve points tend to be arranged in groups, are called clusters, while processes, where the points end up uniformly scattered, are called completely spatially random.

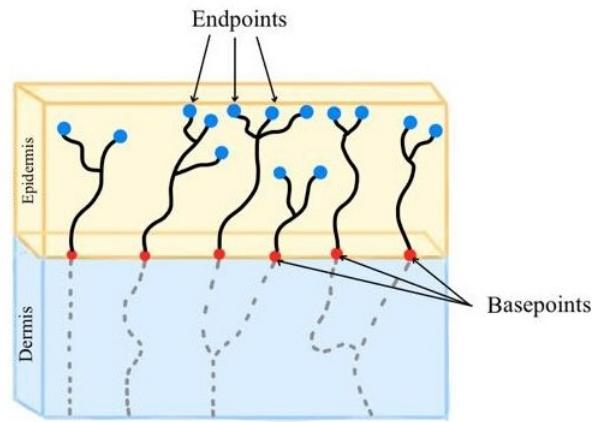


Figure 2: An illustration of the end points and base points in the epidermis.

1.1 Diabetes

Diabetes is a chronic disease that develops either from insufficient insulin production by the pancreas or the body's inability to use the insulin produced by the body properly and effectively. Insulin is a blood sugar level regulating hormone that is produced in the pancreas. Diabetes, if uncontrolled, causes hyperglycemia or high blood sugar and can potentially damage many-body systems, especially the nerves and blood vessels. The two existing types of diabetes (type 1 and type 2) share several characteristics. Overall, their primary difference is in the occurrence and the treatment of the disease. [1].

Diabetes is one of the biggest public health concerns on a global level, with a significant adverse impact on the health of the general population and the socio-economic development of nations. Although the prevalence of diabetes has decreased in some countries, most developed and developing countries have experienced an increase in recent decades. The global diabetes distribution is estimated to be 10.2%, (about 578 million people), and by 2030 it is predicted to increase by 0.7% to 700 million people by 2045. The prevalence is higher in urban (10.8%) than rural (7.2%) areas and in high-income (10.4%) than low-income countries (4.0%). Due to the significant increase in the disease, it has been classified as a global epidemic by the World Health Organization [1]. An early diagnosis of neuropathy can be crucial for the treatment process.

1.2 Statistical scope

Statistics is today regarded by many academics as a tool for which one can analyze and interpret data. To accurately extract relevant information from data is not only important but also provides the main empirical and quantitative “proof” for the conclusions of a study. However, the field of traditional statistics is not defined narrowly enough, which limits the subject in many ways for different reasons. This is the reason why there are different divisions within statistics for different applications. One of these divisions is *spatial statistics* or *spatial analysis*, which is the subdiscipline within statistics that deals with spatial data. Within spatial statistics, many models exist when dealing with spatial data and different kinds of processes to imitate reality better. These different models have specific methods with regard to validation to assess the accuracy of the models.

There are many applicable fields for spatial statistics, which include: spatial economics, image processing, earth science, ecology, geography, epidemiology as well as biology, as we will see in this study. Anything that produces complex location-oriented problems can potentially be analyzed using spatial statistical methods, and by using the lens of spatial analysis, we can assess spatial data to find patterns and trends [4]. In our study, these patterns were analyzed using spatial point pattern methods as the data will take the shape of different points in a rectangular observation window.

Spatial point patterns are frequently occurring in medicinal and biological data and are specifically defined as a data set that contains the location information of events or things. These events or things are represented by a point that can have different attributes. The point can vary in size, colour or shape depending on what sort of data we have. In our spatial point patterns for this study, we have been looking at points that represent a nerve base or nerve ending in the tissues of the patients.

1.3 Purpose

The aim of the thesis is to model the physiological changes caused as diabetic neuropathy advances in the structure of the epidermal nerve fibers. This can be explained in a more digestible manner as the depreciation of nerve fibers in the top layer of diabetic patients’ skin. Our purpose is to find a mathematical model that represents this process. This thesis is a small part of a larger study on diabetic neuropathy, and therefore the results of our study will be used there.

1.4 Problem

The primary objective of this thesis is to improve our understanding of the underlying biological mechanisms that lead to changes in the structure of nerve fibers in diabetic patients with neuropathy. Through a better understanding of the underlying process, more efficient techniques can be developed to detect the disease early. In accomplish our objective, the two-dimensional structure of the nerve patterns has been studied, and models based on spatial thinning have been developed.

There is a significant difference in the number of base and end points in a healthy person compared to a diabetic person. This can be observed in Table 1, where the intensities are presented. The hypothesis that nerve removal occurs at random was tested. Two common patterns are shown when comparing the nerve patterns. One is that there are fewer nerves, and the other is that the nerves are more clustered, as seen in Figure 8, where the results are shown. Therefore, another hypothesis is that there is a connection between the thinning of the nerves and the clustering.

1.5 Data

Data from healthy volunteers and mild diabetic patients comprise the epidermal nerve fiber dataset since the primary goal of this thesis is to investigate neuropathy at the earliest stages. The data includes 28 samples from 8 mild diabetic subjects and 112 samples from 32 healthy controls. We focused on skin samples obtained from the patient's feet since research has shown that the early changes in the physiology of the epidermal nerve fibers occur at an early stage in the distant body regions [3].

The data is treated as realizations of stationary and isotropic point processes in a two-dimensional box. There are two types of points in each point pattern: base and endpoints. The base points are clustered because the nerve fibers may branch into deeper skin layers. Since heat and pain are felt at the endpoints, their spatial structure is critical.

A spatial point process is a collection of random points, and an outcome of such a process is called a point pattern. A spatial point process can create point patterns expressed in $n=1,2,\dots$ dimensions. This project worked with point patterns in \mathbb{R}^2 , as seen in Figure 3.

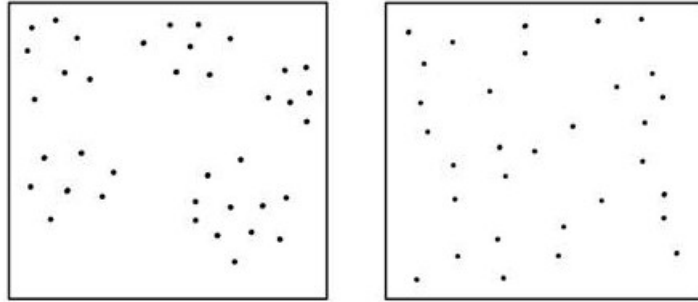


Figure 3: Examples of two-dimensional point patterns

1.6 Delimitation

Producing a mathematical model is a relatively difficult problem, especially when working with a fixed amount of data points. For example, the data we worked with contains a total of 140 samples, with 28 samples from diabetic patients and the rest from healthy volunteers. In general, models trained on a few observations tend to overfit and produce inaccurate results.

The dataset contains non-spatial covariates such as age, BMI and gender, which are not accounted for in the spatial analyses. Due to the small data set, the result might be inaccurate if the covariates were used.

2 Theory

In this section, the prominent mathematical concepts used in the process of creating the models will be listed and explained.

2.1 Bootstrapping

Bootstrapping is the process of resampling from a data set by sampling randomly with replacement. Bootstrapping can be used to derive standard errors, ensure the data is tested efficiently, and mitigate overfitting risks. In algorithmic terms, the Bootstrapping method consists of choosing a sample size B from a population N and sampling randomly m times to get the sample estimates [5].

There are two types of bootstrapping methods applicable in statistics and Machine Learning; the parametric Bootstrap method and the non-parametric bootstrap method. The parametric bootstrap method assumes a parametric distribution for the parameters, while the non-parametric method does not. Our model uses a non-parametric bootstrap based on resampling from the empirical data and acquiring the L-function's statistic(s) of interest [5].

2.2 Gaussian distribution

The Gaussian distribution, also known as the normal distribution, is one of the most important distributions in statistics. This continuous probability distribution describes the distribution of a population centred around its mean, giving it a bell-shaped curve. The distribution is widespread due to the central limit theorem, which states that the average of a large number of independent and identically distributed random variables is approximately Gaussian.

The probability density function $f(x)$ with mean μ and variance σ^2 is given in the formula below [6].

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \text{ for } x \in \mathbb{R} \quad (1)$$

The half-normal distribution is a special case of the folded normal distribution in probability theory and statistics. An ordinary, normal distribution with a mean zero has a half-normal distribution with a fold at the mean.

2.3 Euclidean distance

The Euclidean distance is the length of a line segment between two points in the Euclidean space, which is the fundamental space of classical geometry. The length of the line segment can be calculated using the Pythagorean theorem $a^2 + b^2 = c^2$ where a and b are two sides of a right triangle and c is the hypotenuse. Since we are dealing with line segments in two dimensions, let point p have the coordinates (p_1, p_2) and point q have the coordinates (q_1, q_2) . The following formula is used to calculate the distance between point p and point q and is used in the second and third model that we will present [7].

$$d(p, q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \quad (2)$$

2.4 Homogeneous Poisson Processes

Modelling using Poisson point processes works well when the data points are randomly scattered. If a Poisson point process is termed as homogeneous, it means that the data points are uniformly distributed across the sample space, meaning that the points are equally probable to appear in any arbitrary place in the space and are independent of the location of the data point. The space assumed to be bounded can be denoted as W , and its area denoted as $v(W)$. Let ds be a small region by the point s , ds be the area of the region and N a random variable that represents the

number of points of the process in an arbitrary region. Then the intensity $\lambda(s)$ can be defined as

$$\lambda(s) = \lim_{ds \rightarrow 0} \frac{\mathbf{E}(N(ds))}{ds} \quad (3)$$

In a homogeneous Poisson process, $\lambda(s) \equiv \lambda > 0$. Let λ be the intensity of a Poisson distribution and is the random number of points of the process N in the space W follows a Poisson distribution with the mean $\lambda \cdot v(W) = \mathbf{E}(N(W))$. The homogeneous Poisson point process is often used as a reference to compare and determine whether point patterns are random, regular or clustered [5].

2.5 Thinning

Thinning is an operation you can do on a spatial data set in order to thin out points. The Independent random thinning model is the simplest form of thinning, and it omits points with a probability of $1 - p$ independently of other points. Let N_d be a point pattern of our spatial data set. After the thinning operation, it will yield $N_t \subset N_d$ where N_t is our thinned point pattern [5].

Spatial dependent thinning is an operation which is dependent on the other points in the process. The probability of omitting points will be $1 - p(x)$ where $p(x) = p(x|X)$ where X is the condition on all other points or the entire process [5].

2.6 Ripley's K-function & L-function

Ripley's K-function is used to determine the fit of the model as it finds the average number of points within distance r from a certain point without counting the reference points. It also assumes that the underlying process is stationary (translation invariant) and isotropic (rotation variant). The results from Ripley's K-function are rather difficult to interpret and therefore, according to [5], modern point process statistics rarely use the K-function and instead utilize the L-function. Both functions represent the same information, but there are graphical and statistical benefits to using the L-function. The reason for this is the functions' proportional properties, since $K(r) \propto r^d$ and $L(r) \propto r$ in \mathbb{R}^d .

For stationary and isotropic point processes, Ripley's K-function is defined as

$$K(r) = E_o [N(b(o, r) \setminus \{o\})] / \lambda \quad (4)$$

where $N(b(o, r) \setminus \{o\})$ is the number of further points of N , within a distance $r \geq 0$, from the origin o where λ is the intensity and $E_o(\cdot)$ is the conditional expectation given there is a point of the process in the origin. The L-function in \mathbb{R}^2 , which is a normalized version of the K-function making its expected value linear, is defined as

$$L(r) = \sqrt{\frac{K(r)}{\pi}} \quad (5)$$

One can further normalize the L-function by subtracting r from both sides of the equation, and make it easier to graphically interpret the results, thus letting

$$L(r) - r = \sqrt{\frac{K(r)}{\pi}} - r \equiv 0 \quad (6)$$

which will be used in this thesis when we analyze our results. The value of $L(r)$ for regular processes tends to lie in the interval $[0, r]$ and $L(r) > r$ for clustered processes. Thus $L(r) - r \leq 0$ for regular processes and $L(r) - r \geq 0$ for clustered processes [5].

2.7 Isotropic edge correction

When calculating the functions mentioned in the section above, discs of radius r centered at each point are used. Thus the problem of points being located by the edge of a window arises as section of the circle constructed around the point by the edge is not used which will lead to points being missed. Therefore, one has to use edge correction methods in order to make the estimator for the K -function unbiased. There are plenty of edge correction methods, but the most popular is the isotropic edge correction. An unbiased estimate for Ripley's K -function is given by

$$\hat{K}(r) = \frac{1}{\hat{\lambda}} \frac{1}{n} \sum_{i=1}^n \sum_{j \neq i}^n w(x_i, x_j) \mathbb{1} \{x_i - x_j \in b(o, r)\} \quad (7)$$

The weights are calculated by

$$w_{1,2} = \frac{1}{w(x_1, x_2)} \quad (8)$$

$$w(x_1, x_2) = \frac{v_1(\partial b(x_1, \|x_1 - x_2\|) \cap W)}{2\pi \|x_1 - x_2\|} \quad (9)$$

where the numerator is the length of the circle centered at x_1 with radius $\|x_1 - x_2\|$ that lies within the window W . The weight factor is then divided by the circle perimeter length $2\pi \|x_1 - x_2\|$, see Figure 4 [5].

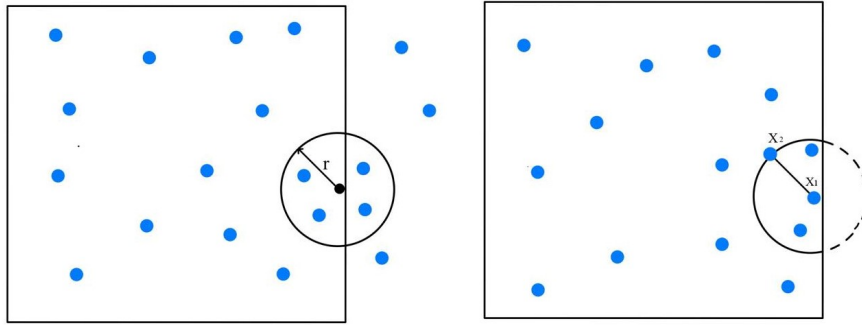


Figure 4: A picture displaying how one can use isotropic edge correction to account for points close to the edges of a window.

2.8 Global envelope test

The global envelope tests are non-parametric statistical tests recently developed for comparing functional or multivariate statistics obtained from the data and under the null-model. The method initially computes the simulated statistics from the model and creates envelopes based on the extreme ranks of the curves for a pre-selected significance level α . There are different methods of ranking the curves, and the method that we used is the extreme rank length method. The constructed envelope has the following interpretation, if the summary function (in our case $L(r) - r$) falls outside the envelopes constructed under the null-model, the null hypothesis that the functional statistics are the same is rejected for significance level α [8].

2.9 Convex Hull

A convex hull or convex envelope of a set X of points in Euclidean space or the Euclidean plane is the smallest convex set that contains X . When X is a bounded subset of the plane. The convex hull may be visualized as the shape formed when a rubber band is stretched around X . The convex hull can be defined as the intersection of all convex sets containing X or as the set of all convex combinations of points in X . By extending this definition to arbitrary real vector spaces, convex hulls can also be extended to oriented matroids.

3 Method

To better understand the problem and formulate the different tasks during the modelling process, we must first use the appropriate tools. In this thesis, we worked with the programming language R, which is designed for dealing with statistical analysis but has many more applications. Within the programming language, there is a package called *spatstat* that includes many prebuilt spatial statistical tools that allowed us to better deal with and interpret spatial data [9].

The next step was to conduct an *exploratory data analysis*, or EDA for short. EDA is the act of transforming and visualizing data to obtain information that is useful in an iterative way by asking more and more questions about the data while searching for answers [10]. *How are the nerves oriented in the tissues of the mild diabetic patients? Are the points clustered? How many clusters on average? How big is the difference between the neuropathic and the healthy patient?* By asking questions like these, we can quickly find interesting and fundamental information about the spatial data effortlessly, which will narrow our path in modeling the thinning process.

The first thinning model that we used is independent random thinning. This thinning method uses a fixed probability or a stochastic process that models how the thinning procedure occurs. This is a basic thinning model that we examined as we delve deeper into the more advanced and complex models.

When dealing with spatial data, it is sometimes useful to calculate the *Ripley's K-function* or alternatively the *L-function* to detect deviations from complete spatial randomness [4][5]. Not only can we statistically test and assess the significance, but we can also garner other useful descriptive statistics, such as the scale of clustering or dispersion of the spatial data, which can be utilized in the final model. After establishing a few thinning models, the fit was evaluated using the global envelope test with 500 simulations and other non-spatial summary statistics.

3.1 Modeling process

In this section, the three models created for the thesis will be explained along with corresponding justifications.

3.1.1 Model 1: Independent random thinning model

An independent random thinning model is tested as our first model. The reason for building this model is to test the hypothesis that there is no underlying mechanism for nerve mortality, and hence the nerves are removed at random. A non-deterministic model was created to test if the nerve cluster deteriorates randomly. It works by randomly thinning endpoints with a probability $1 - P$. Comparing the two data sets, one could discern that the diabetic data had (roughly) 30.766% fewer data points than the healthy ones with $\hat{\lambda}_{healthy} = 0.521 \cdot 10^{-3}$ and $\hat{\lambda}_{mild} = 0.360 \cdot 10^{-3}$. Therefore the probability is chosen as $\hat{P} = 0.692$. The model is then used on data sets from healthy patients to verify if endpoints disappear. The following equation determines the probability of retaining points in the thinning process

$$\hat{P} = \frac{\hat{\lambda}_{mild}}{\hat{\lambda}_{healthy}} \quad (10)$$

3.1.2 Model 2: Deterministic thinning model

The second model that we created is a deterministic model which utilizes the Euclidean distance to determine which points will be omitted or not. This model was built in order to test the hypothesis of end points further away from the base point are weaker than those located in close proximity to the base point. It works by setting the radius parameter manually, which will be used to create a disc around the base point for each cluster. Subsequently, all points which are inside the circle will remain, and points outside the circle will be omitted, which can be seen in Figure 5. This works by using a subsetting function with the logical expression of the radius and the Euclidean

distance, thus excluding the points further away from the base point than the radius of the circle. As it is a deterministic model, bootstrapping was introduced in the model to create randomness, assess the variance of the statistic and correct the sampling bias.

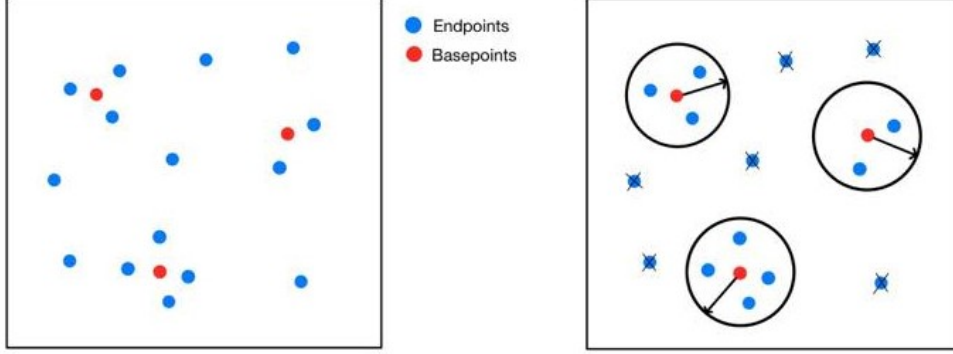


Figure 5: Illustration of the thinning mechanism used in Model 2.

3.1.3 Model 3: Gaussian thinning model

Due to the deterministic nature of the second model, we developed a more sophisticated thinning model that incorporates the half-normal distribution for the retention probability. This way, the summary statistics can be estimated without the need of any re-sampling, however to more accurately reflect the sample of the diabetic patients, bootstrapping was used in the same way as the second model.

Euclidean distance was used in the half-normal distribution, in other words, the further away the point is from the projection of the base point onto the endpoints, the lower the probability of retention. The retention probability around the center of the base point will however be constant at $\hat{P} = 1$ as shown in Figure 6. The justification behind the constant retention probability around the center comes from the fact that the most central points are more significant with regards to the amount of clustering. The parameters for the distribution and the distance at which the retention is constant are tuned to match the peak of the centered L -function with that of the diabetic patients.

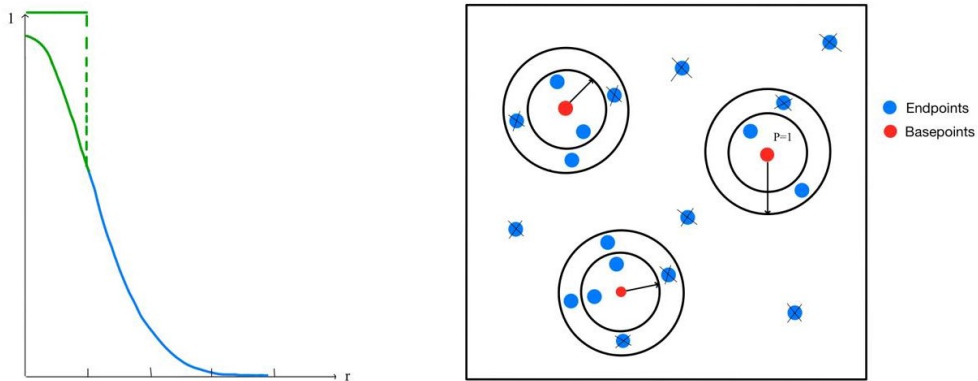


Figure 6: The left figure shows the unnormalized probability density as a function of the radius for the retention probability. The right figure illustrates the thinning mechanism used in model 3.

3.2 Sustainable and ethical aspects

The sustainable and ethical aspects of the project are beneficial for society, research and the public in multiple ways. The results of this project will not negatively affect any of these three areas. It also serves as a great teaching tool for students to acknowledge their work's ethical and sustainable perspective. This allows us to take a step back from the project and contemplate how we contribute value to society.

Our project will benefit society by enabling researchers to understand diabetic neuropathy better. Therefore, researchers can use this knowledge to make advancements in that field and eventually discover the disease earlier than they currently can. By discovering diabetic neuropathy earlier, doctors can start treatment earlier and decrease the intensity of increasing symptoms. In addition, earlier treatment will prolong the time spent between stages of diabetes. This is also in line with goal 3 of the United Nations 17 Sustainable Development Goals, which is "Good health and Well-being." Important to note that since our data set is small, we will make no conclusion about the disease but rather treat the results as indications that require further study.

An increase in diabetes has taken place and will continually increase. Therefore, not only will research advancement be crucial, but it will also spread the necessary awareness of diabetes and its consequences to the public and society. Furthermore, the results of this thesis could also lead to an increased demand for other research areas, such as neuronal differentiation. If this modeling method using spatial statistics proves to be fruitful, this could benefit society by being applied in research for other diseases.

Since the data we have been working with is collected from real-life patients, either diagnosed with mild diabetes or healthy, integrity is a major factor. The data set only provides us with anonymous subject IDs, and therefore no intrusion of integrity has been made. All patients have given their full consent to participate in the study and let the researcher collect data from them. We can not think of any malicious acts that our study can be used for.

4 Results

This section presents the results of the three models and the pre-investigation of the data. The results of all models will be presented by a table summarizing its non-spatial statistics, a graph displaying its L-function, and a graph displaying the results of the global envelope test.

4.1 Pre-Investigation of the data

The pre-investigation of the ENF-data contains both non-spatial and spatial statistics to better compare with the modelled data.

	Healthy	Mild
Average λ per subject (σ) (10^{-3})	0.521 (0.256)	0.360 (0.212)
Average cluster size (σ) (10^{-3})	2.638 (1.705)	2.398 (1.420)
Average area per cluster (10^{-6} m^2)	88.058	63.946
Average λ of clusters (10^{-3})	0.197	0.150

Table 1: Non-spatial statistics obtained from the original ENF-data.

The non-spatial statistics are presented in Table 1. Here, we can clearly see the difference in clustering between the healthy patients and the diabetic patients, which suggests diabetic neuropathy. There is also evidence of a reduction in cluster size, indicating that the thinning isn't exclusive to the base points.

If we look at the spatial statistics of the ENF-data, it is interesting to test for complete spatial randomness (CSR). To test for CSR, we conduct a global envelope test with the diabetic and healthy data along with a Poisson point process demonstrated in Figure 7. As we can see, the empirical function falls completely outside the envelopes created by the Poisson point process. We also note that the p -value of the test is $p \approx 0.01$ thus we reject the null hypothesis that the data are CSR.

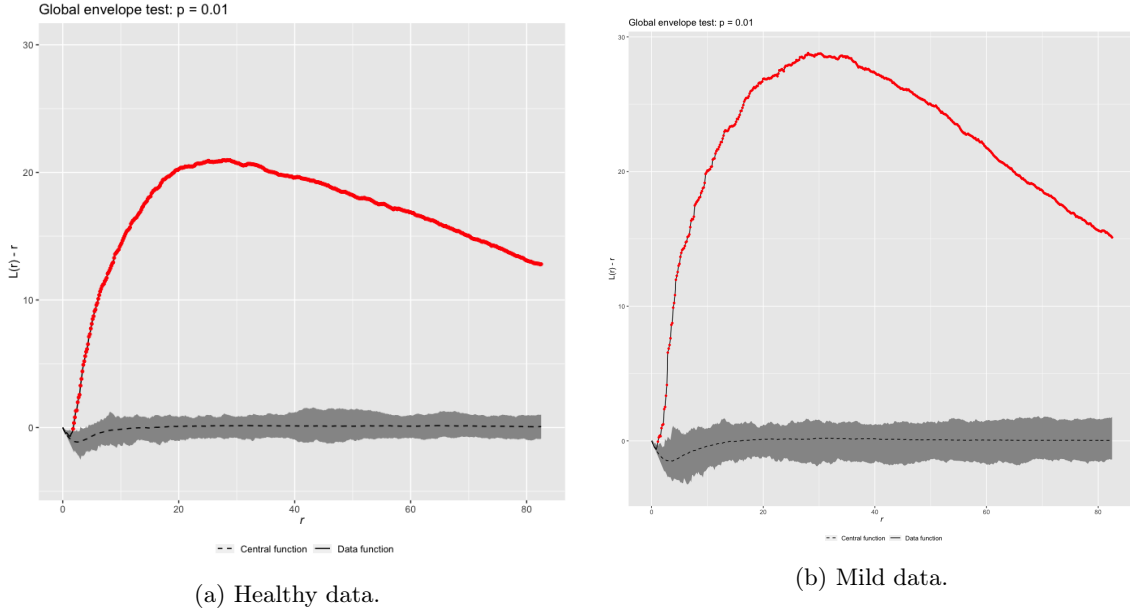


Figure 7: Global envelope tests to test the assumption of CSR on the mild and healthy data

If we instead look at Figure 8, we observe the centered L-function of the healthy and diabetic patients. The centered L-function indicates that the end points of the diabetic patients have become more clustered compared to the healthy patients. The peak of the function for the diabetic patients is reached at around $r \approx 30 \cdot 10^{-6}$ and for the healthy patients, it is reached at around $r \approx 28 \cdot 10^{-6}$. The peak of the centered L-function of the diabetic patients, is at around $L(r) - r \approx 27$. The results of this section will be referenced and compared to when evaluating the different models.

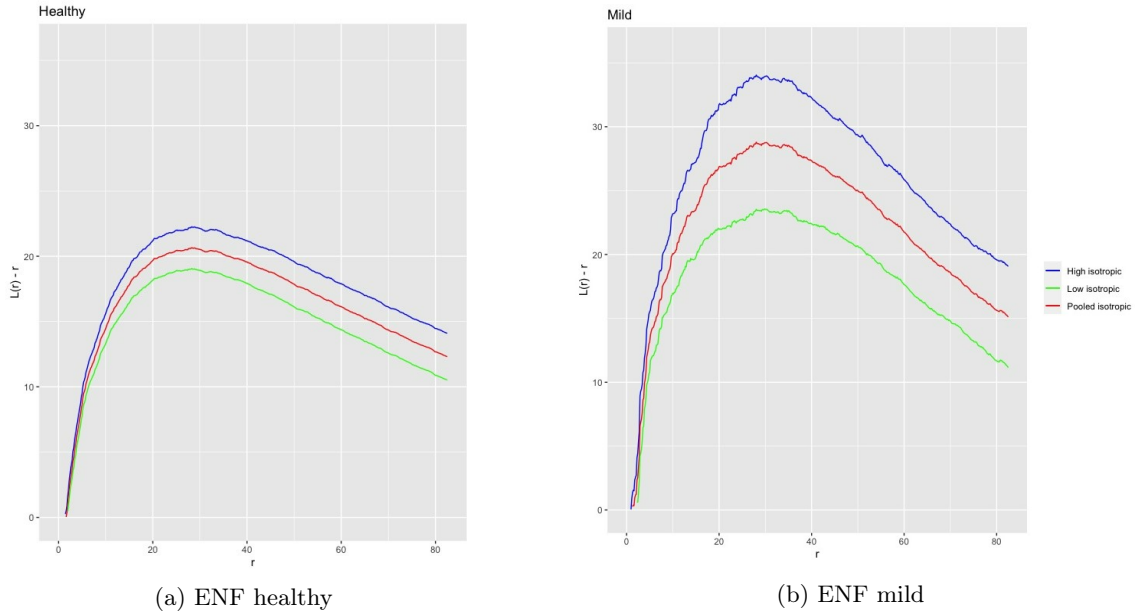
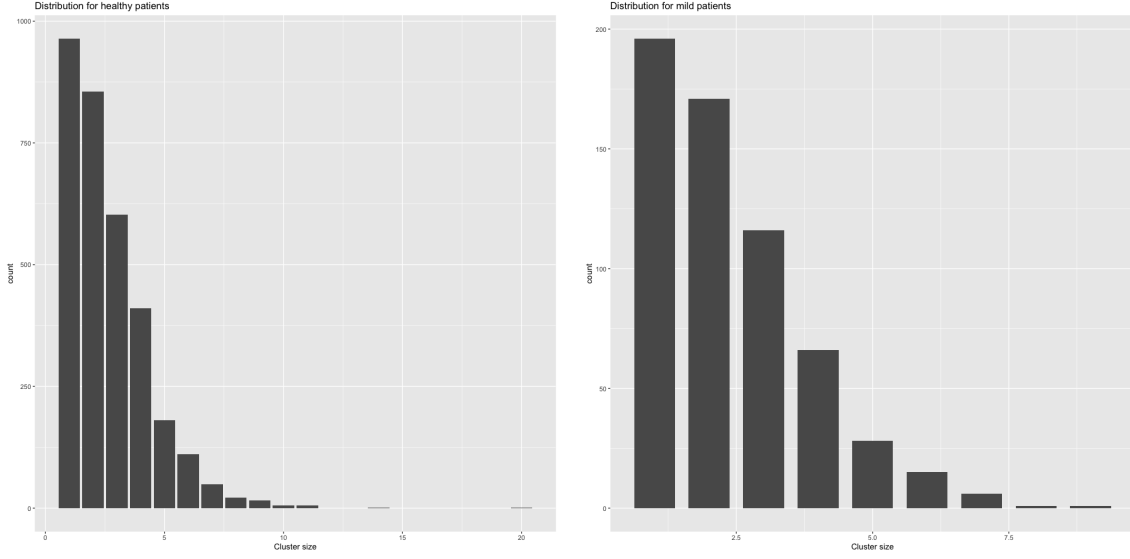


Figure 8: Spatial results of the original ENF-data.

In Figure 9a and in Figure 9b, one can see the distribution of the average cluster size for the healthy data, respectively the mild data.



(a) Distribution of the average cluster size for the healthy data. (b) Distribution of the average cluster size for the mild data.

Figure 9: Distribution of cluster sizes for healthy and mild data.

The distributions are fairly different with the healthy data having on average an additional 0.24 points per cluster. There are a couple outliers in the healthy data set which could offset this statistic. Another important note to add is that the sample sizes differ a lot, which could lead to a comparatively wider distribution to that of the diabetic patients.

In Figure 10, one can see that the intensity of the mild and healthy data differ. The median $\hat{\lambda}_{healthy}$ is 29.25% lower than the median $\hat{\lambda}_{mild}$. The whiskers on top of the box plot for the healthy data is longer than its counterpart for the mild data, thus implying a higher variance for $\hat{\lambda}_{healthy}$.

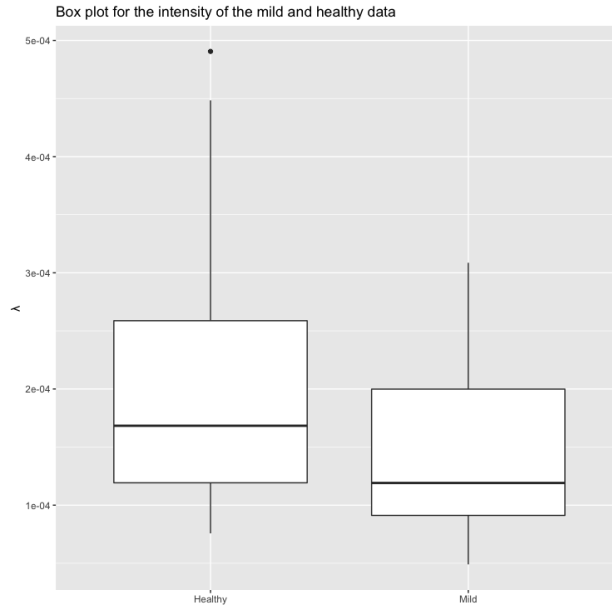


Figure 10: Box plot showing the quartiles, minimum and maximum value of the intensities for mild and healthy.

4.2 Non-Spatial results

In this section we will present the non-spatial summary statistics is summarized in table 2. The code used for this section will be presented in the appendix.

	Model 1	Model 2	Model 3	Mild
Average λ per subject (σ) (10^{-3})	0.362 (0.184)	0.354 (0.196)	0.322 (0.189)	0.360 (0.212)
Average cluster size (σ) (10^{-3})	2.110 (0.450)	2.041 (0.403)	1.856 (0.494)	2.398 (1.420)
Average area per cluster (10^{-6} m^2)	59.352	26.422	33.495	63.946

Table 2: Non-spatial statistics of the data obtained from simulated data using models.

4.2.1 Model 1

Table 2 demonstrates how \hat{P} from model 1 has affected the intensity $\lambda_{healthy}$ of the healthy patients end points, to match the intensity λ_{mild} of the diabetic patients. The average cluster size has decreased by around half a point compared to the healthy data and has 12% less average cluster size than the mild data. This is in contrast to the prior observations, a big difference from the desired outcome as seen in the cluster size distribution of the diabetic patients in Table 2. Average area per cluster from the data obtained by model 1 is close to the mild data, thus performing better than model 2 and 3 in this statistic.

4.2.2 Model 2

Comparing the results obtained from model 2 and the pre-investigation of the data, one can see that the model closely match the average λ and σ_λ , in the mild data set. The average cluster size is on average about 15% lower in the model results but with 71% lower standard deviation. The average area is about $20.3 \cdot 10^{-6} \text{ m}^2$ less than the average area of the mild data. The large area difference between the model results and the mild data is due to the area being calculated by constructing a polygon between the spatial points for each clusters. Thus, clusters that are reduced down to having 1 or 2 points, the area will be 0. As the average cluster size is 2.638 points for the healthy data, one could expect this problem to occur frequently, hence decreasing the validity of this statistic.

4.2.3 Model 3

For model 3, the results were relatively similar, only slightly lower in each statistic, than the results obtained from model 2. The average λ and σ_λ were both about 10% lower than the mild data. Model 3 yields the lowest average cluster size out of all the models, with about a half a point per cluster less than the mild data and with a standard deviation of 0.494. The low average cluster size will also affect the average area per cluster in the same way as mentioned in model 2 which is shown by the average area per cluster being $18 \cdot 10^{-6} \text{ m}^2$.

4.3 Spatial results

In the following section, the three different models will be evaluated using methods within spatial statistics and more specifically, the centered L-function and the global envelope test. For the global envelope test 500 simulations were made.

4.3.1 L-function for the models

The results of the L-function for all three models are presented in Figure 11. The middle line of each envelope is the pooled isotropic and the top line is the high isotropic and the low line is the low isotropic. As one can see, model 1 has a lower curve than the other models with the highest point of its pooled isotropic at $L(r) - r = 21$ while model 2 and model 3 has their peaks at $L(r) - r \approx 28.8$ and $L(r) - r \approx 27.5$ respectively. Thus the L -functions of model 2 and model 3, closely resembles the L -function of the mild data with its peak at $L(r) - r \approx 27$.

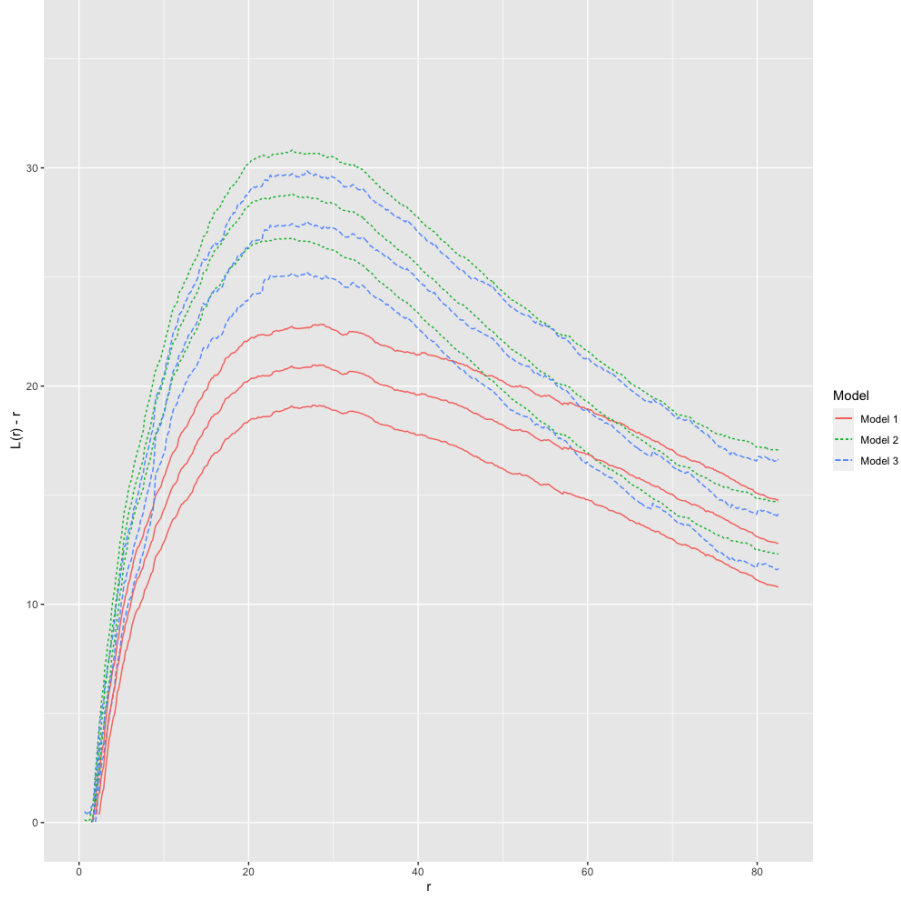
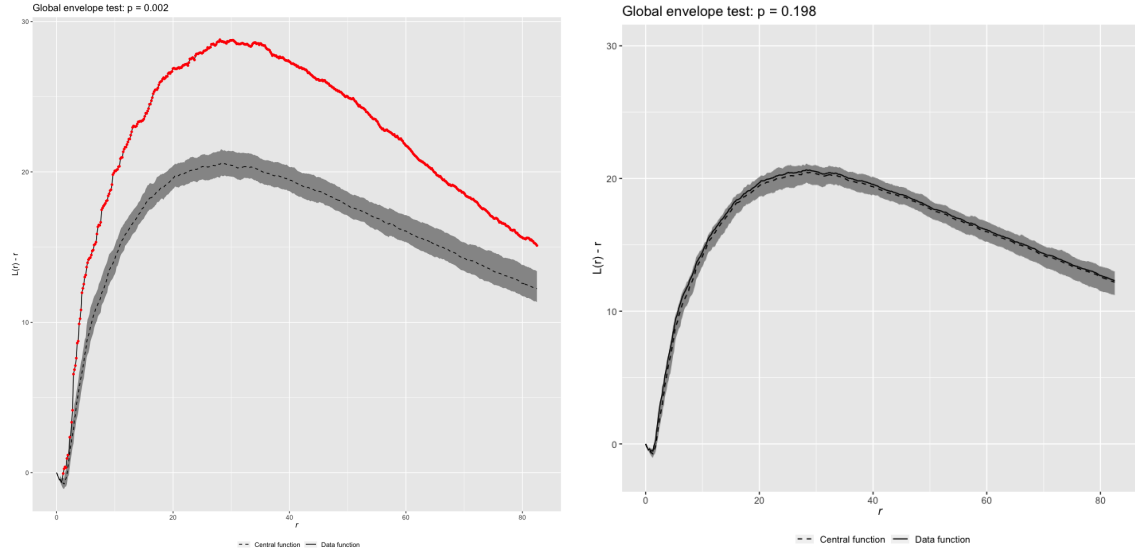


Figure 11: The $L(r) - r$ function graph for all models with isotropic edge correction.

4.3.2 Model 1

From what could be observed in the centered L-function of the diabetic and the healthy patients, it becomes clear that the clustering of the nerve points is more intense in that of the diabetic patients, which is also supported by prior literature and shown in Figure 12. A fundamental issue with Model 1 is that it does not account for the clustering as it is only randomly thinning with an estimated parameter \hat{P} . From this, the centered L-function is no different from the centered L-function of the healthy patients. The only thing affected in model 1 is as expected, only the actual intensity of the points and their distribution. As one can see in Figure 12, the data function obtained from model 1 is outside the bounds of the data function for all values of r hence its low p -value of $p = 0.002$.



(a) Global envelope test for model 1 using the mild data as the empirical data ($\alpha = 0.05$). (b) Global envelope test for model 1 using the healthy data as the empirical data ($\alpha = 0.05$).

Figure 12: The $L(r) - r$ function for the end point patterns with 95% global envelopes constructed from simulations from model 3.

Figure 12b illustrates how the L -function is invariant to the thinning mechanism of model 1. This is expected as clustering is independent of intensity and therefore the null hypothesis of the healthy data under the simulated data is not rejected for significance level $\alpha = 0.05$.

4.3.3 Model 2

As one can see in Figure 13, the L -function closely resembles the L -function for the mild data. The peak of the graph is slightly skewed to the right for the model 2 as it peaks around cluster radius $r \approx 28.8$ compared to the mild data, as it peaks at around $r \approx 30$. The narrower interval given by the low and high isotropic function shows that the results which model 2 yields has less variance compared to the mild data. For the global envelope test, summarized in Figure 13, one can see that the data function is inside the borders for all r , but peaks slightly after the central function. It has a high p-value of $p = 0.624$ thus making it our model with by far the highest p-value. Further tuning of parameters, such as the radius in the model could potentially move the peak of the data function closer to the central function.

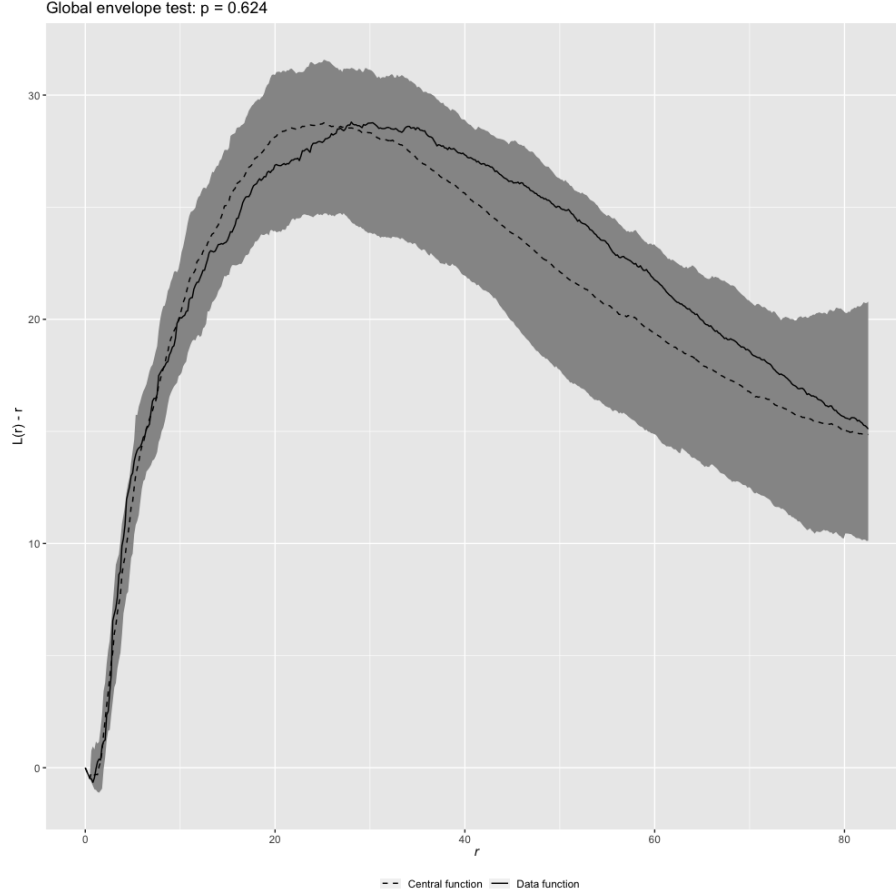


Figure 13: The $L(r) - r$ function for the end point patterns with 95% global envelopes constructed from simulations from model 3.

4.3.4 Model 3

When examining the resulting centered L-function after model 3, it yields some interesting results when comparing to the centered L-function of the diabetic patients. Here, the graph reaches the peak $L(r) - r \approx 27 \cdot 10^{-6}$ for the pooled isotropic curve at cluster radius $r \approx 30 \cdot 10^{-6}$, which is close to results of the diabetic patients.

For the global envelope test, shown in Figure 14, we see that the empirical function is within the envelopes for significance level $\alpha = 0.05$ however we can note that the p-value is a mere 0.056 indicating that the null-hypothesis would be rejected for $\alpha = 0.10$. When solely observing the graph, the most likely explanation could be that the beginning of the curve is what affects the performance the most. Here we see the empirical function being very close to the α most extreme curve, indicating some error. Perhaps this is due to the retention rate being $P_{retention} = 1.00$ when within a certain distance of the underlying base point, which either should have been tuned better or maybe used a different mechanism all together for points close to the center.

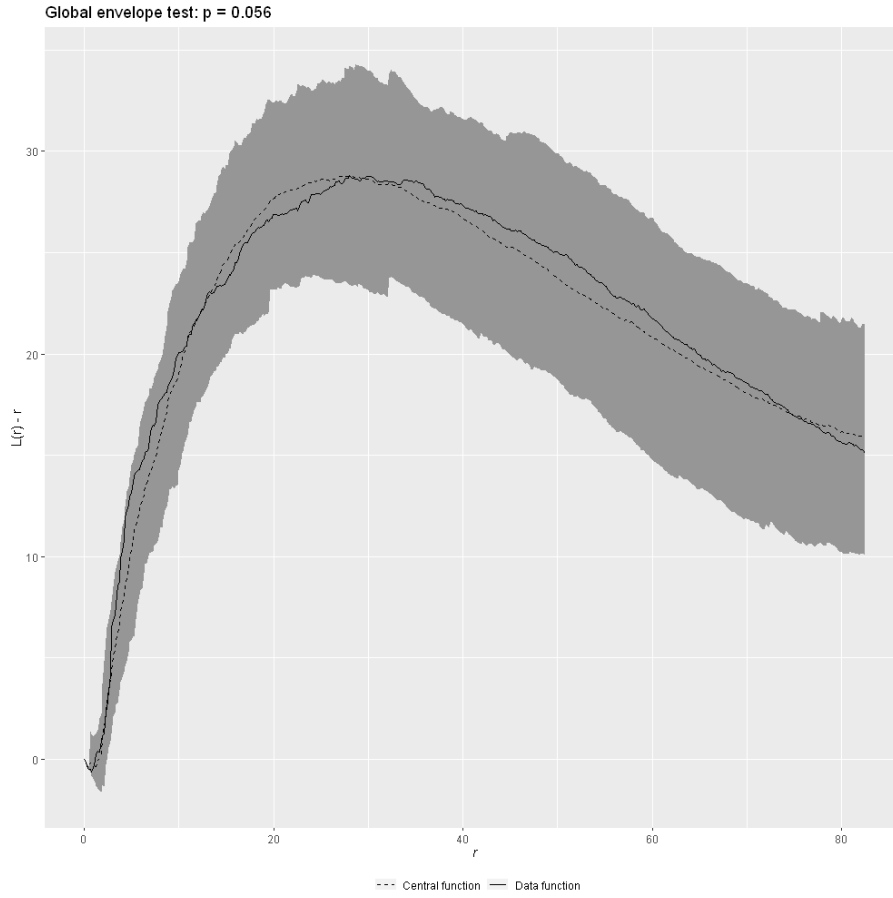


Figure 14: The $L(r) - r$ function for the end point patterns with 95% global envelopes constructed from simulations from model 3.

5 Concluding discussion

This section will discuss the findings that we have made with our models and comment on various aspects of the models. Evidently, the first model did not work well as seen in section 4.3.1 which is highlighted in the spatial results. The reason for this is that neuropathy clearly is not a random independent thinning process, thus spatially dependent hence the construction of the two latter more complicated models. As we used the λ_{mild} and $\lambda_{healthy}$, obtained in the pre-investigation of the data, as a way to estimate the probability P for the first thinning model, the summary statistics were fairly close to the mild results.

The first idea to increase the amount of clustering by thinning on the data of the healthy patients, was to thin on the basis of distance beyond the base point. This model was therefore completely determined by the distance and therefore deterministic. After tuning the distance, the second model yielded interesting results comparable to that of the diabetic patients as demonstrated in 4.3.3 and 4.3.4. The problem is however that this is not a realistic way to model the nerve thinning as it is completely deterministic and needs an aspect that is subject to randomness. The third model intended to answer this question by introducing a probability based thinning process dependent on the Gaussian distribution. This way, the model could be evaluated more realistically as it could simulate multiple samples to be evaluated. The non-spatial statistics of the third model did worse than the second model, however the L-function of the third model seemed to better follow the L-function of the diabetic patients. The same could be said for the global envelope test, however with differing p -values between the two tests. The second model displayed a much higher p -value compared to the very low p -value of the third model. This could be a result of the second model being deterministic and naturally overfitting as the distance parameter has been tuned for this specific dataset. Another possible explanation is that the third model's retention probability is constant beyond a certain distance from the base point and thus did poorly in the simulations if examining the global envelope test at the beginning. Here the central function was much closer to the α most extreme envelopes, which could give rise to a significantly lower p -value.

As one can see in the results, the areas of the clusters in the results yielded by the models are greatly reduced compared to the mild data set. This is due to the fact that most of the patients have a cluster size of around 2.6 points, as illustrated in the pre-investigation of the data. As the algorithm for calculating the convex hull is based upon having at least three corners for the polygon which it creates between data points, clusters which are made up of one or two end points, will have 0 area. We mitigated some of this problem by including the base points thus creating a polygon by combining endpoints and base points, giving the clusters with two endpoints a non-zero area. However, the same downside as mentioned above still exists yet will affect the results less. However since our models thin the healthy data, in combination with the data having its average cluster size being 2.6, means that a lot of clusters will still be counted as zero area, thus impacting the validity of this metric. This problem could be worked around if we had a data set with higher average cluster size.

Throughout this thesis and modeling process, we have had the struggle of having a small data set to model upon. We tried to combat this by introducing bootstrap sampling in order to make less biased measurements and thus, less biased inferences. As one can see in the pre-investigation of the data, there is high variances in the data set, with certain patients having hundreds of points with clusters up to a size of 20 points. In the graphs for the L-functions, the low variances for the models could be a result of the limitation of our data as we did 500 sampling simulations for them which potentially could be too low. Due to our low computational power, 500 simulations took around 1-1.5 hours thus limiting our ability to explore the effects large simulation numbers could have on the results.

Further, one could develop additional models building upon the ones that we have constructed for this thesis. For example, one can use other distributions than the half Gaussian distribution which we used for the third model. Most importantly would be to obtain a larger data set for the mildly diabetic patients as the current sample size is inadequate.

Our models provide an insight into the thinning process and provide a basic method for how this process might occur outside of simulation. The emphasis on the last two models were distance beyond the center which was naturally inferred as clustering was the focus but other covariates such as cluster sizes or areas could have been used. However, the interesting question is if thinning alone is enough to answer what the natural mechanisms induced in diabetic neuropathy. For a basic simulation to acquire interesting results in spatial predictive purposes, model 2 or 3 might suffice. If the focus is solely on non-spatial results, model 1 would be an efficient alternative. For an even more realistic model, interaction type point processes could be a viable option as it is unlikely that the only mechanism behind diabetic neuropathy is thinning.

References

- [1] World Health Organization et al. Diabetes action now: an initiative of the world health organization and the international diabetes federation. 2004.
 - [2] Jorge L Gross, Mirela J De Azevedo, Sandra P Silveiro, Luís Henrique Canani, Maria Luiza Caramori, and Themis Zelmanovitz. Diabetic nephropathy: diagnosis, prevention, and treatment. *Diabetes care*, 28(1):164–176, 2005.
 - [3] Claes Andersson, Peter Guttorp, and Aila Särkkä. Discovering early diabetic neuropathy from epidermal nerve fiber patterns. *Statistics in Medicine*, 35(24):4427–4442, 2016.
 - [4] Adrian Baddeley, Ege Rubak, and Rolf Turner. *Spatial point patterns: methodology and applications with R*. CRC press, 2015.
 - [5] Janine Illian, Antti Penttinen, Helga Stoyan, and Dietrich Stoyan. *Statistical analysis and modelling of spatial point patterns*, volume 70. John Wiley & Sons, 2008.
 - [6] Nathaniel R Goodman. Statistical analysis based on a certain multivariate complex gaussian distribution (an introduction). *The Annals of mathematical statistics*, 34(1):152–177, 1963.
 - [7] David Cohen. Precalculus: A problems-oriented approach , cengage learning. Technical report, ISBN 978-0-534-40212-9, 2004.
 - [8] Mari Myllymäki, Tomáš Mrkvička, Pavel Grabarnik, Henri Seijo, and Ute Hahn. Global envelope tests for spatial processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(2):381–404, 2017.
 - [9] Adrian Baddeley and Rolf Turner. Package ‘spatstat’. 2021.
 - [10] Matthieu Komorowski, Dominic C Marshall, Justin D Saliccioli, and Yves Crutain. Exploratory data analysis. *Secondary analysis of electronic health records*, pages 185–203, 2016.
- [heading=none]

Appendix - code

```
library(spatstat)
library(tidyverse)
library(sp)
library(fdrtool)
library(GET)
library(patchwork)

setwd("~/Desktop/B.Sc Matematisk Statistik/Kandidatarbete GU")
base <- readRDS("BASEPOINTS_2D")
end <- readRDS("ENDPOINTS_2D")

subject_ID <- unique(base$SID)

# pooled L-function of mild and healthy patients, works by changing
#the loop from 1:28 to 29:140 as the first 28 samples are for mild

L_diabetic <- c()
t <- 0
for(i in 1:28){

  t <- t+1
  L_diabetic[[t]] <- Lest(end$ppp[[i]], correction="isotropic")

}

pooled_L_diabetic <- pool.anylist(L_diabetic)

plot(pooled_L_diabetic, ~r~r)

#First model

L_healthy_GET <- c()
for (e in 1:100){

L_healthy <- c()
t <- 0

for(i in 29:140){

  t <- t+1
  L_healthy[[t]] <- Lest(rthin(end$ppp[[i]], P=1-(1-0.692)), correction="
    isotropic") #Remove Lest for ppp object

}

L_healthy_GET[[e]] <- pool.anylist(L_healthy)

}

test <- c()
t <- 0
for(i in 29:140){
  t <- t+1
  test[[t]] <- Lest(end$ppp[[i]], correction = "isotropic")
}
```



```

    ]]
    $y,
    Xc=Xc,
    Yc=Yc,
    sd=16)

)
thinned_clusters <- unique(thinned_clusters) # removes duplicate
points
}
thinned_L[[j]] <- Lest(thinned_clusters, correction="isotropic")
thinned_ppp[[j]] <- thinned_clusters
}
thinned_L_subset_3[[e]] <- thinned_L[samp]
thinned_ppp_subset_3[[e]] <- thinned_ppp[samp]
thinned_L_pooled_3[[e]] <- pool.anylist(thinned_L_subset_3[[e]])
}

#GET for 3rd model

simulations <- vapply(thinned_L_pooled_3, function(x) x$pooliso-x$r, rep
(0,513))
c_test <- create_curve_set(list(obs= pooled_L_diabetic$pooliso-
pooled_L_diabetic$r, r = pooled_L_diabetic$r, sim_m=simulations))

plot(global_envelope_test(c_test, alpha = 0.05)) + ylab("L(r) - r")

#thinning function for 3rd model

thin_fun <- function(x,y,Xc,Yc,sd){

  d <- sqrt((x-Xc)^2+(y-Yc)^2)

  thinprob <- phalfnorm(d, sd2theta(sd))
  if (thinprob<=0.25)
    thinprob <- 0

  prob <- 1-thinprob

}

# Gaussian thinning model

thinned_L <- c()
thinned_ppp <- c()

for(j in 29:140){
  thinned_clusters <- subset(end$ppp[[j]], subset=FALSE)
  for(i in 1:npoints(base$ppp[[j]])){
    Xc <- base$ppp[[j]]$x[i]
    Yc <- base$ppp[[j]]$y[i]
    thinned_clusters <- superimpose(thinned_clusters,
                                     rthin(end$ppp[[j]],
                                             P = thin_fun(
                                               x=end$ppp[[j]]$x,
                                               y=end$ppp[[j]]$y,
                                               Xc=Xc,
                                               Yc=Yc,
                                               sd=16
                                             )
    )
  }
}

```

```

    )
  }
  thinned_clusters <- unique(thinned_clusters) # removes duplicate points
}
thinned_L[[j]] <- Lest(thinned_clusters, correction="isotropic")
thinned_ppp[[j]] <- thinned_clusters
}

thinned_L_subset <- thinned_L[29:140]
thinned_ppp_subset <- thinned_ppp[29:140]
thinned_L_pooled_3 <- pool.anylist(thinned_L_subset)

#GET for 2nd model with bootstrap
dist <- 20
thinned_L_subset_mod2 <- c()
thinned_ppp_subset_mod2 <- c()
thinned_L_pooled_mod2 <- c()

for(e in 1:500){
  samp <- sample(29:140,28)
  thinned_L <- c()
  thinned_ppp <- c()
  for(j in samp){
    thinned_clusters <- subset(end$ppp[[j]], subset=FALSE)
    for(i in 1:npoints(base$ppp[[j]])){
      thinned_clusters <- superimpose(thinned_clusters,
                                      subset(end$ppp[[j]],
                                              subset=dist>sqrt((x-base$ppp[[j]]
                                                                    [[x[i]]^2+(y-base$ppp[[j]]
                                                                    [y[i]]^2)))
      thinned_clusters <- unique(thinned_clusters) # removes duplicate
        points
    }
    thinned_L[[j]] <- Lest(thinned_clusters, correction="isotropic")
    thinned_ppp[[j]] <- thinned_clusters
  }

  thinned_L_subset_mod2[[e]] <- thinned_L[samp]
  thinned_ppp_subset_mod2[[e]] <- thinned_ppp[samp]
  thinned_L_pooled_mod2[[e]] <- pool.anylist(thinned_L_subset_mod2[[e]])
}

#GET test for 2nd model
simulations <- vapply(thinned_L_pooled_mod2, function(x) x$pooliso-x$r, rep
  (0,513))
c_test <- create_curve_set(list(obs= pooled_L_diabetic$pooliso-
  pooled_L_diabetic$r, r = pooled_L_diabetic$r, sim_m=simulations))

plot(global_envelope_test(c_test, alpha = 0.05))+ ylab("L(r) - r")

poolisomodel2 <- lapply(thinned_L_pooled, function(x) x$pooliso)

#Model 1 L-function
ggplot()+geom_line(aes(x=L_healthy_pooled$r, y=L_healthy_pooled$hiiso-
  L_healthy_pooled$r, color="High isotropic"))+
  geom_line(aes(x=L_healthy_pooled$r, y=L_healthy_pooled$pooliso-

```



```

      L_healthy_pooled$r, color="Pooled isotropic"))+
geom_line(aes(x=L_healthy_pooled$r, y=L_healthy_pooled$loiso-
      L_healthy_pooled$r, color="Low isotropic")) +
ylim(0,36)+
labs(x = "r",
      y = "L(r) - r",
      color = "") +
scale_color_manual(values = colors)

#Model 2 L-function
ggplot()+geom_line(aes(x=thinned_L_pooled_2$r, y=thinned_L_pooled_2$hiiso-
      thinned_L_pooled_2$r, color="High isotropic"))+
geom_line(aes(x=thinned_L_pooled_2$r, y=thinned_L_pooled_2$pooliso-
      thinned_L_pooled_2$r, color="Pooled isotropic"))+
geom_line(aes(x=thinned_L_pooled_2$r, y=thinned_L_pooled_2$loiso-
      thinned_L_pooled_2$r, color="Low isotropic")) +
ylim(0,36)+
labs(x = "r",
      y = "L(r) - r",
      color = "") +
scale_color_manual(values = colors)

#Model 3 L-function

colors <- c("High isotropic" = "blue", "Pooled isotropic" = "red", "Low
      isotropic" = "green")

ggplot()+geom_line(aes(x=thinned_L_pooled_3$r, y=thinned_L_pooled_3$hiiso-
      thinned_L_pooled_3$r, color="High isotropic"))+
geom_line(aes(x=thinned_L_pooled_3$r, y=thinned_L_pooled_3$pooliso-
      thinned_L_pooled_3$r, color="Pooled isotropic"))+
geom_line(aes(x=thinned_L_pooled_3$r, y=thinned_L_pooled_3$loiso-
      thinned_L_pooled_3$r, color="Low isotropic")) +
ylim(0,36)+
labs(x = "r",
      y = "L(r) - r",
      color = "") +
scale_color_manual(values = colors)

#Dataset for plotting combined

df_mod3 <- tibble(r=thinned_L_pooled_3$r, high=thinned_L_pooled_3$hiiso, low=
      thinned_L_pooled_3$loiso, pooled=thinned_L_pooled_3$pooliso, Model=
      mod3_factor)
df_mod2 <- tibble(r=thinned_L_pooled_2$r, high=thinned_L_pooled_2$hiiso, low=
      thinned_L_pooled_2$loiso, pooled=thinned_L_pooled_2$pooliso, Model=
      mod2_factor)
df_mod1 <- tibble(r=L_healthy_pooled$r, high=L_healthy_pooled$hiiso, low=
      L_healthy_pooled$loiso, pooled=L_healthy_pooled$pooliso, Model=mod1_factor)

#Creating factors to distinguish the models
mod3_factor <- as_factor(rep("Model 3", nrow(df_mod3)))
mod2_factor <- as_factor(rep("Model 2", nrow(df_mod2)))
mod1_factor <- as_factor(rep("Model 1", nrow(df_mod1)))

df_L_func <- rbind(rbind(df_mod1, df_mod2), df_mod3)

test_df <- df_L_func %>% mutate(High_iso=high-r) %>% mutate(Pooled_iso=

```

```

    pooled-r) %>% mutate(Low_iso=low-r)
test_df %>% group_by(Model) %>% summarise(max(pooled-r))#max value of pooled
-r

#All graphs combined

ggplot(data=test_df, aes(x=r, color=Model, linetype=Model))+geom_line(aes(y=
  High_iso))+
  geom_line(aes(y=Low_iso))+geom_line(aes(y=Pooled_iso))+
  ylim(0,36)+
  labs(x = "r",
        y = "L(r) - r")

#### Non spatial statistics

#Histogram cluster size distribution mild

dist_mild <- c()
for(i in 1:8){
  temp <- 0
  for(j in 1:length(subjects_b[[i]]$SID)){
    temp[j] <- marks(subjects_b[[i]]$ppp[[j]]) %>% select(size)
  }
  dist_mild[[i]] <- temp
}
#SD of cluster size for mild
sd(dist_mild_vec)

#SD of cluster size for healthy
sd(dist_healthy_vec)

#Histogram cluster size distribution healthy
dist_mild_vec<- unlist(dist_mild)
ggplot()+geom_bar(aes(dist_mild_vec), width=0.75)+labs(title="Distribution
  for mild patients")+xlab("Cluster size")

dist_healthy <- c()
for(i in 9:40){
  temp <- 0
  for(j in 1:length(subjects_b[[i]]$SID)){
    temp[j] <- marks(subjects_b[[i]]$ppp[[j]]) %>% select(size)
  }
  dist_healthy[[i]] <- temp
}

dist_healthy_vec<- unlist(dist_healthy)
ggplot()+geom_bar(aes(dist_healthy_vec))+labs(title="Distribution for
  healthy patients")+xlab("Cluster size")

#Averages cluster size mild and healthy
sum(dist_healthy_vec)/length(dist_healthy_vec)
sum(dist_mild_vec)/length(dist_mild_vec)

#Average cluster size per subject on thinned data

ave_clust_size <- c()

```

```

for(i in 1:length(thinned_ppp_subset)){

  ave_clust_size[i] <- length(thinned_ppp_subset[[i]]$marks$Tree)/length(
    unique(thinned_ppp_subset[[i]]$marks$Tree))
}
sum(ave_clust_size)/length(ave_clust_size)

#Average amount of clusters on thinned data
ave_amount_clust <- c()
for(i in 1:length(thinned_ppp_subset)){
  ave_amount_clust[i] <- length(unique(thinned_ppp_subset[[i]]$marks$Tree))
}

sum(ave_amount_clust)/length(ave_amount_clust)

#Total points on thinned data
total_points <- c()
for(i in 1:length(thinned_ppp_subset)){
  total_points[i] <- thinned_ppp_subset[[i]]$n
}

#Average points per subject
sum(total_points)/length(total_points)

#Cluster size distribution
clust_size_dis <- list()
for(i in 1:length(thinned_ppp_subset)){
  clust_size_dis[[i]] <- tabulate(thinned_ppp_subset[[i]]$marks$Tree)
}

clust_size_dis_vec <- unlist(clust_size_dis)
clust_size_dis_vec <- clust_size_dis_vec[!clust_size_dis_vec %in% 0]

ggplot()+geom_bar(aes(clust_size_dis_vec))+labs(title="Distribution for
  thinned model")+xlab("Cluster size")

#Area calculations with base point for mild, healthy and all models. Just
  switch the dataset to get the answer for
#the different models and mild etc.
poly_list <- list()

superimpose_list <- list()
size_rem <- base

for(i in 1:140){
  size_rem$ppp[[i]]$marks <- size_rem$ppp[[i]]$marks[c(1,2)]
  superimpose_list[[i]] <- superimpose(end$ppp[[i]],size_rem$ppp[[i]])
}

for(j in 1:112){

  test_df <- tibble(tree=superimpose_list[[j]]$marks$Tree,x=superimpose_list
    [[j]]$x,y=superimpose_list[[j]]$y)

  poly_create <- list()

```

```

area_vec <- c()
for(i in unique(test_df$tree)){

  test_1 <- test_df %>% filter(tree==i) %>% slice_head()
  poly_df <- test_df %>% filter(tree==i) %>% rbind(test_1) %>% select(-
    tree)
  poly_create[[i]] <- Polygon(poly_df)
  area_vec[i] <- poly_create[[i]]@area
}
poly_list[[j]] <- area_vec
}

poly_list

all_area_vec <- unlist(poly_list)[!is.na(unlist(poly_list))]
sum(all_area_vec)/length(all_area_vec)

#29:140 ger area p 88.05824 med base points
#1:28 ger area p 63.94629 med base points
#59.35185 f r arean p f r sta modellen med base points
#26.42194 f r arean p andra modellen med base points
#33.49481 f r arean p den tredje modellen med base points
#Calculating area of models

poly_list <- list()

superimpose_list <- list()

size_rem <- base[29:140]

#This is for calculating area of the clusters for the models, just switch
  the thinned_ppp_subset
#To the models respective ppp_subset and then run the code above
for(i in 1:112){
  size_rem$ppp[[i]]$marks <- size_rem$ppp[[i]]$marks[c(1,2)]
  superimpose_list[[i]] <- superimpose(thinned_ppp_subset[[i]],size_rem$ppp
    [[i]])
}

#Code below is used to calculate average amount of clusters

ave_amount_clust <- c()
t <- 0
for(i in 1:28){
  t <- t+1
  ave_amount_clust[t] <- length(unique(end$ppp[[i]]$marks$Tree))
}
sum(ave_amount_clust)/length(ave_amount_clust)
#Average amount of clusters on model 1 is 23,5625
#Average amount of clusters on model 2 is 24,65179
#Average amount of clusters on model 3 is 23,74107
#Average amount of clusters on healthy data is 28,21429
#Average amount of clusters on mild data is 21,42857

#Box plots for intensity
intensities_healthy <- c()
for(i in 29:length(base$ppp)){
  intensities_healthy[i] <- intensity(base$ppp[[i]])
}

```

```

}

intensities_mild <- c()
for(i in 1:28){
  intensities_mild[i] <- intensity(base$ppp[[i]])
}

median(intensities_healthy[!is.na(intensities_healthy)]) #mean of intensity
  for healthy 0.0001975019
median(intensities_mild[!is.na(intensities_mild)]) #mean of intensity for
  mild 0.0001502455
1-(0.0001191112/0.0001683502) #29.25% lower

all_intensities <- c(intensities_mild,intensities_healthy[29:140])

facts <- c(rep("Mild",28),rep("Healthy",112))
boxplots <- tibble("vals" = all_intensities, "facts" = facts)
boxplots_2 <- boxplots %>% mutate("facts" = as.factor(facts))

boxplots_2 %>% ggplot() + # basepoints boxplots.
  geom_boxplot(aes(x=facts,y=vals))+labs(title="Box plot for the intensity
    of the mild and healthy data")+xlab("")+ylab("")

#L-function graph for non modelled data
#This is for mild
L_non_modelled_m <- c()

for(i in 1:28){
  L_non_modelled_m[[i]] <- Lest(end$ppp[[i]],correction="isotropic")
}

pooled_non_mod_m <- pool.anylist(L_non_modelled_m)
pooled_non_mod_m

colors <- c("High isotropic" = "blue", "Pooled isotropic" = "red", "Low
  isotropic" = "green")

non_mod_plot_m <- ggplot()+geom_line(aes(x=pooled_non_mod_m$r,y=
  pooled_non_mod_m$hiiso-pooled_non_mod_m$r,color="High isotropic"))+
  geom_line(aes(x=pooled_non_mod_m$r,y=pooled_non_mod_m$pooliso-
  pooled_non_mod_m$r,color="Pooled isotropic"))+
  geom_line(aes(x=pooled_non_mod_m$r,y=pooled_non_mod_m$loiso-
  pooled_non_mod_m$r,color="Low isotropic")) +
  ylim(0,36)+
  labs(x = "r",
    y = "L(r) - r",
    color = "", title="Mild") +
  scale_color_manual(values = colors)
max(pooled_non_mod_m$pooliso-pooled_non_mod_m$r)#Highest peak of the L-
  function for the mild data

#This is the L-function graph for healthy

L_non_modelled_h <- c()
t <- 0
for(i in 29:140){
  t <- t+1

```

```

L_non_modelled_h[[t]] <- Lest(end$ppp[[i]], correction="isotropic")
}

pooled_non_mod_h <- pool.anylist(L_non_modelled_h)
pooled_non_mod_h

non_mod_plot_h <- ggplot()+geom_line(aes(x=pooled_non_mod_h$r,y=
  pooled_non_mod_h$hiiso-pooled_non_mod_h$r,color="High isotropic"))+
  geom_line(aes(x=pooled_non_mod_h$r,y=pooled_non_mod_h$pooliso-
  pooled_non_mod_h$r,color="Pooled isotropic"))+
  geom_line(aes(x=pooled_non_mod_h$r,y=pooled_non_mod_h$loiso-
  pooled_non_mod_h$r,color="Low isotropic")) +
  ylim(0,36)+
  labs(x = "r",
       y = "L(r) - r",
       color = "",title = "Healthy") +
  scale_color_manual(values = colors)

#plot using the package patchwork which combines the plots
non_mod_plot_h+non_mod_plot_m

#CSR, simulating different poisson processes in the same windows as either
the mild
#or healthy data. From that we calculate the respective L-functions and
conduct a GET.

lambda_mild <- 0.0003603452
lambda_healthy <- 0.000521
pois_list <- c()
L_pois <- c()
for(j in 1:100){
  for(i in 1:28){
    window <- subset(end$ppp[[i]], subset=FALSE)$window
    pois_list[[i]] <- Lest(rpoispp(lambda = lambda_healthy, win=window),
      correction="isotropic")
  }
  L_pois[[j]] <- pool.anylist(pois_list)
}
L_healthy_pooled

simulations <- vapply(L_pois,function(x) x$pooliso-x$r,rep(0,513))
c_test <- create_curve_set(list(obs= L_healthy_pooled$pooliso-
  L_healthy_pooled$r, r =L_healthy_pooled$r, sim_m=simulations))

plot(global_envelope_test(c_test,alpha = 0.05)) +
  ylab("L(r) - r")+ylim(c(-4,30))

```