Point process learning for non-parametric intensity estimation with focus on Voronoi estimation

Alexander Thorén

UNIVERSITY OF
GOTHENBURG

Department of Mathematical Sciences
University of Gothenburg
Gothenburg, Sweden 2023

Notation

X	a random variable or point process
x	a realization of a random variable or point pattern
$\mathbf{X}$	a vector of values $X_1, \ldots, X_n$
$\mathbf{1}\{x\}$	the indicator function, which is 1 if x is true and 0 if false
$\breve{X}$	a marked point process
$\mathbf{x}^{\mathbf{V}}$, $\mathbf{x}^{\mathbf{T}}$	validation and training sets
$\mathbf{x}^p$	a point pattern thinned with some probability p
$\mathcal{V}_x(\mathbf{x}, W)$	the Voronoi cell of the point x which is in the point pattern $\mathbf{x}$ which is observed in the window W

Contents

Abstract

Point process learning is a new statistical theory that gives us a way to estimate parameters using cross-validation for point processes. By thinning a point pattern we are able to create training and validation sets which are then used in prediction errors. These errors give us a way to measure the discrepancy between two point processes and are used to measure how well the training sets can predict the validation sets. We investigate non-parametric intensity estimation methods with a focus on the resample-smoothing Voronoi estimator. This estimator works by repeatedly thinning a point pattern, finding the Voronoi intensity estimate of the thinned point pattern, and then using the mean as the final intensity estimate. Previously, only a thumb rule was given as to how to choose parameters for the resample-smoothing Voronoi estimator but with the help of point process learning we now have a data-driven method to estimate these parameters.

Acknowledgements

First and foremost, I would like to thank Ottmar Cronie, to whom I am deeply grateful for his continuous support and guidance during this thesis and all his work on both point process learning and the re-sample smoothing Voronoi estimator which enabled this thesis.

I would also like to give my deepest appreciation to Aila Särrkä for her support and invaluable feedback throughout this thesis.

A special thanks to Anna Källsgård who was the opponent for this thesis and also provided important feedback.

The simulations for this thesis were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at the Chalmers Centre for Computational Science and Engineering partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

1 Introduction

A point process is essentially a random collection of points on some space where both the number and locations of points follow some random distribution. The locations of trees in a forest, calls to emergency services, and the positions of bacterial growth in a petri dish can all be seen as point processes.

An important characteristic of all point processes is the intensity function or intensity for short. The intensity $\rho(x)$ governs the expected number of points of a point process X in a region W via

$$\mathbb{E}[X(W)] = \int_W \rho(x)dx$$

where $X(W)$ is the number of points in W. There are many methods to estimate the intensity, both parametric and non-parametric. For a long time, the most popular non-parametric intensity estimation method has been kernel estimation which utilizes some kernel function parameterized by a bandwidth h. The choice of this bandwidth is the main challenge in kernel intensity estimation and there are several popular methods that can be used to make this choice as described by Silverman [1998, p. 43].

Another non-parametric intensity estimation method is the Voronoi estimator. This estimator works by creating a Voronoi tessellation of an observed point pattern and then taking the inverse of the size of each Voronoi cell as the estimated intensity. A recent development using this intensity estimation method that shows great potential is the resample-smoothing Voronoi estimator as seen in Moradi et al. [2018]. This method takes the average of m Voronoi intensity estimates for which each underlying point pattern has been thinned with some retention probability p. To thin a point pattern simply means that each point is independently retained with some probability p. Using the thumb rule as proposed by Moradi et al. [2018] the resample-smoothing Voronoi estimator achieves better performance than the state-of-the-art bandwidth selection method for the kernel estimator.

Another promising recent result is point process learning which is a data-driven parameter selection method as seen in Cronie et al. [2021] and has shown very promising results when used for bandwidth selection. In this thesis, we will investigate how point process learning performs when selecting the number of estimates m and the retention probability p for the resample-smoothing Voronoi estimator and compare its performance to kernel intensity estimation.

In this thesis, we will start by introducing some general point process theory, introduce the specific point process models we will investigate, and then introduce the kernel and resample-smoothing Voronoi estimators. After that, we will move on to the theory behind point process learning. Finally, we will see the results of using point process learning to select parameters for the aforementioned intensity estimators.

2 Preliminaries

2.1 Point processes

We start by laying the foundation for this entire thesis; a point process. A point process X at its most foundational level is a random sequence of points in some space S where each point represents the site of some event. In this thesis, we will only investigate point processes defined on $\mathbb{R}^2$ which are observed on the unit square window $[0,1] \times [0,1]$. An example of a realization of a point process, which is called a point pattern, can be seen in Figure 1.

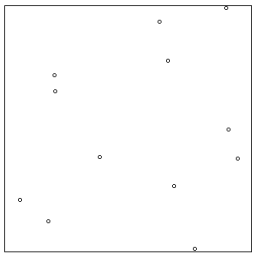


Figure 1: An example of a point process on the unit square.

To begin to formally define a point process we first need to establish some preliminaries, the first of which is a σ-algebra. Given some set Y, a σ-algebra, Σ, of Y, has the following properties [Chiu et al., 2013, p. 28]:

1. $Y \in \Sigma$
2. if $A \in \Sigma$ then $A^C \in \Sigma$
3. if $A_1, A_2, \ldots \in \Sigma$ then $\bigcup_{i=1}^{\infty} A_i \in \Sigma$.

The tuple of Y and the σ-algebra Σ, $[Y, \Sigma]$ is called a measurable space.

Now that we have defined a σ-algebra we can introduce the so-called Borel σ-algebra. A Borel σ-algebra is created by taking the open subsets of some set and then applying the above described σ-algebra operations on them. We will reserve the notation $\subseteq$ for Borel sets, i.e. members of a Borel σ-algebra.

Now we can formally define a point process $X = \{X_i\}_{i=1}^{N}$, on S as a random variable or element in the measurable space $[Y, \Sigma]$. Y is the collection of all point patterns $\mathbf{x} = \{x_1, \ldots, x_n\} \subseteq S$ which are

- locally finite, i.e. for a bounded $A \subseteq S$ we have $\mathbf{x}(A) = \#(\mathbf{x} \cap A) < \infty$,
- simple, i.e. $x_i \neq x_j$ if $i \neq j$ [Chiu et al., 2013, p. 108-109].

$\mathcal{N}$ is the σ-algebra generated by $\mathbf{x} \mapsto \mathbf{x}(A), \mathbf{x} \in \mathcal{X}, A \subseteq S$. The elements of $\mathcal{N}$ are subsets of the collection of point patterns with specific properties, e.g. $\{\mathbf{x} : \mathbf{x}(A) = 0\} \in \mathcal{N}, \mathbf{x} \in \mathcal{X}$ for some specific $A \subseteq S$.

Furthermore, given a probability space $[\Omega, \mathcal{A}, \mathbf{P}]$, a point process X is a measurable mapping into $[\mathcal{X}, \mathcal{N}]$ which generates a distribution P of X on $[\mathcal{X}, \mathcal{N}]$ [Chiu et al., 2013, p. 109]. This distribution can be seen as

$$P(Y) = \mathbf{P}(X \in Y) = \mathbf{P}(\{\omega \in \Omega : X(\omega) \in Y\}) \quad \text{for } Y \in \mathcal{N}. \tag{1}$$

Note that point process is the term for the random collection of points while a realization of the point process is called a point pattern. Henceforth the cardinality of a point pattern $\mathbf{x}$ in some $A \subseteq S$ will be denoted $\mathbf{x}(A)$.

Furthermore, the family of so-called finite-dimensional distributions of a point process X is the collection of joint distributions of $(X(A_1), \ldots, X(A_n))$ for bounded Borel sets $A_i \subseteq S$, $i = 1, \ldots, n$. Given a metric d such that (S, d) is a complete and separable metric space the distribution of X on that space is entirely determined by its finite-dimensional distributions [Van Lieshout, 2000, p. 7]. The most common such metric space, and the only one we will consider in this thesis, is $\mathbb{R}^d$ and the Euclidean distance $d(u, v) = \|u - v\|$ with $u, v \in \mathbb{R}^d$ [Cronie et al., 2021, p. 4].

2.1.1 Intensity function

The intensity function ρ, or intensity for short, of a point process, governs the expected number of points of a point process X in any set A and is as such of interest in all investigations regarding point processes. The intensity can be seen as a heatmap of how likely an event is to occur at any point of A. It can be seen as the equivalent of a probability distribution function of a random variable for a point process with the exception that it does not necessarily integrate to 1.

Formally, it is defined as the function satisfying [Chiu et al., 2013, p. 51]

$$\mathbb{E}[X(A)] = \int_A \rho(x) dx \tag{2}$$

with $\rho(x)$ able to take many forms such as:

- a constant $\rho(x) = \rho$
- a function with respect to the position x, $\rho(x)$
- or even be an element of a random distribution.

Given a point process $X = \{X_i\}_{i=1}^N$, we say that X is stationary if the translated point process $X_t = \{X_i + t\}_{i=1}^N$ has the same distribution for any real-valued t. Furthermore, we say that X is isotropic if the same is true for the rotated point process $X_r = \{\mathbf{r}X_i\}_{i=1}^N$ for any rotation $\mathbf{r}$ about the origin [Chiu et al., 2013, p. 42]. Obviously, if a point process X is stationary it must also have a constant intensity ρ.

The point process we saw in Figure 1 has a constant intensity of $\rho = 15$ meaning that we have

$$\mathbb{E}[X(A)] = \int_A \rho(x)dx = \int_A \rho dx = \rho \int_A dx = \rho|A|$$

and since $A = [0,1]^2$ and thus $|A| = 1$ we get that the expected number of points is simply 15.

2.1.2 Papangelou conditional intensity

The Papangelou conditional intensity $\lambda(\cdot;\cdot)$ may be described as [Van Lieshout, 2000, p. 39]

$$\lambda(u;\mathbf{x})du = P(X(du) = 1|X \cap (du)^c = \mathbf{x} \cap (du)^c). \tag{3}$$

Heuristically, the Papangelou conditional intensity is simply the probability of observing a point in the infinitesimal region du around u conditioned on observing the point pattern $\mathbf{x}$ in the complement to this infinitesimal region, $(du)^c$.

The Papangelou conditional intensity $\lambda(u;X)$ is then a random variable whose mean is [Illian et al., 2008, p. 29]

$$E[\lambda(u;X)] = \rho(u) \tag{4}$$

which is an alternative definition of the intensity function. We remind ourselves that the expected number of points of a point process X in a set W is given by

$$\mathbb{E}[X(W)] = \int_W \rho(u)du$$

and if we let W be an infinitesimally small area, du around some location u, we get

$$\mathbb{E}[X(du)] = 0 \cdot P(X(du) = 0) + 1 \cdot P(X(du) = 1) = P(X(du) = 1) = \rho(u)du$$

since we are only interested in simple point processes that can at most have 1 point at each location. Furthermore, given a density $p(\mathbf{x})$ with respect to a Poisson process, the Papangelou conditional intensity can alternatively be written as [Van Lieshout, 2000, p. 41]

$$\lambda(u;\mathbf{x}) = \frac{p(\mathbf{x} \cup u)}{p(\mathbf{x})}.$$

2.1.3 Product densities

In this section, we will continue developing the concept of intensity by defining the intensity of higher orders. To begin, we introduce the concept of a measure. Previously we noted that the tuple of a set Y and the σ-algebra Σ of Y, $[Y, \Sigma]$ is called a measurable space. A so-called measure on such a tuple is a function $\mu : \Sigma \to [0, \infty]$ with the following properties:

- $\mu(\emptyset) = 0$,
- $\mu\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mu(A_i)$

$\forall A_1, A_2, \ldots \in \Sigma$ such that $A_i \cap A_j = \emptyset$ if $i \neq j$ [Chiu et al., 2013, p. 29]. An important measure that will be useful later on in this section is the Lebesgue measure, ν_d on $[\mathbb{R}^d, \mathcal{B}^d]$, which is defined as

$$\nu_d(Q) = (v_1 - u_1) \cdot \ldots \cdot (v_d - u_d)$$

when $Q = [u_1, v_1] \times \ldots \times [u_d, v_d]$ [Chiu et al., 2013, p. 30]. Here $\mathcal{B}^d$ is the σ-algebra of Borel sets on $\mathbb{R}^d$ [Chiu et al., 2013, p. 28].

Now that we know what a measure is we can move on and introduce the concept of moment measures for point processes. For random variables, the moments are useful tools, particularly the first raw moment and the second central moment which are also known as the mean and variance. An equivalent concept exists for point processes and is called moment measures. Given Borel sets, $B, B_1, \ldots, B_n$, and a Borel σ-algebra, $\mathcal{B}^{nd}$, in $\mathbb{R}^d$, the n^{th} moment measure of a point process X, $\mu^{(n)}$, is defined on $\mathcal{B}^{nd}$ by

$$\int_{\mathbb{R}^{nd}} f(x_1, \ldots, x_n) \mu^{(n)}(d(x_1, \ldots, x_n)) = \mathbb{E}\left[\sum_{x_1, \ldots, x_n \in X} f(x_1, \ldots, x_n)\right]$$

where f is any non-negative measurable function on $\mathbb{R}^{nd}$ [Chiu et al., 2013, p. 120-121]. If

$$f(x_1, \ldots, x_n) = \mathbf{1}\{x_1 \in B_1\} \cdot \ldots \cdot \mathbf{1}\{x_n \in B_n\}$$

we have

$$\mu^{(n)}(B_1 \times \ldots \times B_n) = \mathbb{E}[X(B_1) \cdot \ldots \cdot X(B_n)].$$

Furthermore, if $B_1 = \ldots = B_n = B$ then

$$\mu^{(n)}(B^n) = \mathbb{E}[X(B)^n]$$

and is then the n^{th} moment of the random variable $X(B)$ [Chiu et al., 2013, p. 121]. When $n = 1$ we have

$$\mu^{(1)}(B) = \mathbb{E}[X(B)]$$

which, as we saw in the previous section, is governed by the intensity of X and is the expected number of points of X in B.

Closely related to moment measures is the so-called factorial moment measures, $\alpha^{(n)}$ again defined on $\mathcal{B}^{nd}$ by

$$\int_{\mathbb{R}^{nd}} f(x_1, \ldots, x_n) \alpha^{(n)}(d(x_1, \ldots, x_n)) = \mathbb{E}\left[\sum\nolimits^{\neq}_{x_1, \ldots, x_n \in X} f(x_1, \ldots, x_n)\right]$$

where $\sum^{\neq}$ indicates that we are summing over all n-tuples of distinct points in X. If we have that $B_1, \ldots, B_n$ are pairwise disjoint, then [Chiu et al., 2013, p. 121]

$$\mu^{(n)}(B_1 \times \ldots \times B_n) = \alpha^{(n)}(B_1 \times \ldots \times B_n).$$

Now, we can finally introduce the concept of product densities. If $\alpha^{(n)}$ is locally finite and absolutely continuous with respect to ν_d, then $\alpha^{(n)}$ has an n^{th}-order product density $\varrho^{(n)}$ which is defined by

$$\alpha^{(n)}(B_1 \times \ldots \times B_n) = \int_{B_1} \ldots \int_{B_n} \varrho^{(n)}(x_1, \ldots, x_n) dx_1 \ldots dx_n.$$

Furthermore, for any non-negative bounded measurable function f we also have

$$\mathbb{E}\left[\sum\nolimits^{\neq}_{x_1,\ldots,x_n \in X} f(x_1, \ldots, x_n)\right] = \int \ldots \int f(x_1, \ldots, x_n) \varrho^{(n)}(x_1, \ldots, x_n) dx_1 \ldots dx_n.$$

Heuristically, given pairwise disjoint balls $C_1, \ldots, C_n$ with centers $x_1, \ldots, x_n$ and infinitesimal volumes $dV_1, \ldots, dV_n$ then $\varrho^{(n)}(x_1, \ldots, x_n) dV_1 \ldots dV_n$ is the probability that there is a point of a point process X in each of $C_1, \ldots, C_n$ [Chiu et al., 2013, p. 122].

Alternatively, if we introduce the so-called intensity measure $\Lambda(B)$ which is defined as [Chiu et al., 2013, p. 51]

$$\Lambda(B) = \mathbb{E}[X(B)] = \int_B \rho(x) dx$$

we can define the n^{th}-order product density as

$$\varrho^{(n)}(x_1, \ldots, x_n) = \mathbb{E}\left[\prod_{i=1}^{n} \Lambda(x_i)\right]$$

if $x_1, \ldots, x_n$ are pairwise disjoint [Møller et al., 1998, p. 456].

Given that $n = 1$ we already saw that

$$\mu^{(1)}(B) = \mathbb{E}[X(B)] = \int_B \rho(x) dx$$

where $\rho(x)$ is the intensity of a point process X.

Furthermore, we also obviously have that $\mu^{(1)}(B) = \alpha^{(1)}(B)$ and that

$$\alpha^{(1)}(B) = \int_B \varrho^{(1)}(x) dx.$$

and therefore $\varrho^{(1)}(x) = \rho(x)$.

In particular, we are interested in $\varrho^{(2)}$ which is used to construct the pair correlation function which is defined as

$$g(x_1, x_2) = \frac{\varrho^{(2)}(x_1, x_2)}{\varrho^{(1)}(x_1)\varrho^{(1)}(x_2)} = \frac{\varrho^{(2)}(x_1, x_2)}{\rho(x_1)\rho(x_2)}$$

for any $x_1, x_2 \in \mathbb{R}^d$ [Van Lieshout, 2019, p. 100]. The pair correlation function is useful in quantifying whether or not a point process is regular, in which case $g(x_1, x_2) < 1$, clustered, in which case $g(x_1, x_2) > 1$, or completely spatially random, in which case $g(x_1, x_2) = 1$.

2.1.4 Marked point process

The points of a point process can be seen as events and it is quite common to want to associate some extra characteristic with these events which gives rise to the marked point process. Given some space S and a point process $X = \{X_i\}_{i=1}^N$, which we call the ground process, the marked point process is then the sequence $\breve{X} = \{(X_i, M_i)\}_{i=1}^N$ with the marks M_i belonging to a so-called mark space, $\mathcal{M}$. Some examples are:

- if x_i is the location of a tree, then m_i could be the diameter or radius of that tree,
- if x_i is a particle, m_i is that particle's size,
- if x_i is the location of an earthquake, m_i is the time the earthquake happened [Chiu et al., 2013, p. 116].

2.1.5 Thinned point process

For various reasons, we can sometimes want to thin a point process by some degree. To thin a point process simply means to randomly remove some points of the process according to some rule. Given a point process X on some space S, the thinned process X_t is some subset of $X, X_t \subset X$. The simplest form of thinning is p-thinning, which given a $p \in (0, 1)$ and a point process $X = \{X_i\}_{i=1}^N$ can be seen as the marginal point process Cronie et al. [2021]

$$X_t = \{X_i : (X_i, M_i) \in \breve{X}, M_i = 1\} \tag{5}$$

where $\breve{X}$ is a marked point process

$$\breve{X} = \{(X_i, M_i)\}_{i=1}^N \subseteq S \times \mathcal{M}, \mathcal{M} = \{0, 1\}. \tag{6}$$

In this case, we have that a point is retained with probability p and deleted with probability $1 - p$, i.e. $P(M_i = 1) = p$. Furthermore, all M_i are independent of each other resulting in an independent thinning. In general, if the thinning does not depend on X, i.e. $p(u; X) = p(u), u \in S$, we say that the thinning is independent [Chiu et al., 2013, p.159]. The function $p(u)$ is a generalization of p-thinning where we instead let the retention probability be a function of the location u. Such a thinning process is called $p(u)$-thinning.

In general, if the original point process X has intensity $\rho(x)$ then the point process X_t that has been thinned with some probability $p(x)$ will have intensity $p(x)\rho(x)$ [Chiu et al., 2013, p. 160].

2.2 Random field

A random field or spatial random process $X_s, s \in S$ is a family of random variables and is determined by its joint distributions

$$F(y_1, \ldots, y_n; s_1, \ldots, s_n) = P(X_{s_1} \leq y_1, \ldots, X_{s_n} \leq y_n)$$

for finite n and every collection of $s_1, \ldots, s_n$ of locations in S where X_s is the random variable at location s [Van Lieshout, 2019, p. 9]. A random field is called Gaussian if the above distribution is a normal distribution.

In a previous section, we introduced the intensity as governing the expected number of points of a point process in an observed region W. Later on, we will see a point process model which uses a random field as the intensity. Heuristically, a random field can be seen as a family of random variables indexed by the locations of some region. For example, the temperatures over a map of Sweden.

This process can be characterized by its mean,

$$\mu(s) = \mathbb{E}[X_s], \tag{7}$$

and covariance,

$$C(s,t) = \mathbb{E}[(X_s - \mu(s))(X_t - \mu(t))], \tag{8}$$

functions. The only requirements for a function $C(s,t)$ to be a valid covariance function is that it

- is symmetric, $C(s,t) = C(t,s)$,
- is positive-semidefinite, $\sum_i \sum_j a_i a_j C(s_i, s_j) \geq 0 \; \forall s_1, \ldots, s_n \in S, a \in \mathbb{R}^n$.

A spatial random process is stationary if invariant under translation by a vector $t \in \mathbb{R}^n$, i.e. if

$$F(y_1, \ldots, y_n; s_1, \ldots, s_n) = F(y_1, \ldots, y_n; s_1 + t, \ldots, s_n + t).$$

In this case the covariance function $C(s,t)$ only depends on $s - t$. Furthermore, a stationary random process is called isotropic if it is invariant under rotation. In this case, the covariance function only depends on $|s - t| = r$. The covariance function can then be written as

$$C(s,t) = \sigma^2 c(|s - t|) = \sigma^2 c(r) \tag{9}$$

where $c(r)$ is called the correlation function and σ^2 is the variance. A common correlation function is the exponential correlation function given by [Rudemo, 2020, p.70]

$$c(r, \theta) = \exp(-r/\theta) \tag{10}$$

where θ is a scaling parameter.

2.3 Point process models

Next, we describe some of the point processes that we will be investigating later on.

2.3.1 Poisson process

We start with the Poisson process, one of the most basic point processes. While it is a very basic process, it is still a cornerstone of the field of point processes. Before we define the Poisson process, we first remind ourselves that if a random variable Y, is Poisson distributed and is parameterized by some $\lambda > 0$, $Y \sim \text{Po}(\lambda)$, then Y has probability density function

$$f_Y(k) = P(Y = k) = \frac{\lambda^k e^{-\lambda}}{k!}.$$

Given some window $W \subseteq S$ and a constant intensity ρ, the homogeneous Poisson process, which we will call X, has the following properties:

- the number of points of X in W is $\text{Po}(\rho|W|)$ distributed, i.e.

$$X(W) \sim \text{Po}(\rho|W|), \tag{11}$$

- given disjoint regions $W_1, W_2, \ldots$ the counts $X(W_1), X(W_2), \ldots$ are independent,
- the points of $X \cap W$ are uniformly, independently distributed in W.

We have already seen an example of this point process in Figure 1 with $\rho = 15$.

A Poisson process is completely spatially random, abbreviated CSR. Complete spatial randomness is often used as a hypothesis to determine if a point pattern is Poisson or not [Chiu et al., 2013, p. 56-57]. For Poisson processes in general, we have that the pair correlation function is $g(x_1, x_2) = 1$ since $\varrho^{(1)}(x) = \rho$ and $\varrho^{(2)}(x_1, x_2) = \rho(x_1)\rho(x_2) = \rho^2$.

2.3.2 Inhomogeneous Poisson process

In the previous section, we defined a homogeneous Poisson process where the term homogeneity refers to the fact that the intensity is constant. However, we can instead let the intensity be a function of the location of the space it is defined on. If we are in the same setting as the previous section, the intensity is then a function $\rho(u), u \in S$. In this case the number of points of X in W is then Poisson distributed with

$$X(W) \sim \text{Po}\left(\int_W \rho(u)du\right). \tag{12}$$

If we look at Figure 2 we can see an example of such a point process where the intensity is

$$\rho(u) = \rho(x, y) = 10 + 100x.$$

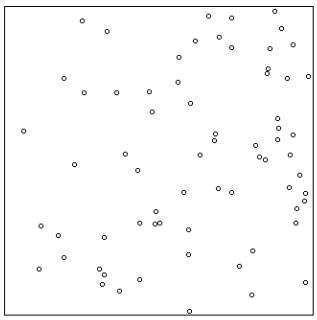


Figure 2: An example of an inhomogeneous Poisson process.

2.3.3 Log-Gaussian Cox process

Now we introduce an interesting model; the Log-Gaussian Cox process. This process is given by a Poisson process where the intensity function is given by a realization of a random function, more specifically, a log-Gaussian random field. To define such a process we start with a Gaussian random field $Z = (Z_s)_{s \in S}$ defined on some space S, as described in Section 2.2, characterized by some mean function $\mu(s), s \in S$ and a covariance function $C(s,t), s,t \in S$ [Van Lieshout, 2019, p. 120-121]. One might think to use a realization of this Gaussian field as the intensity function of a Poisson process but it may be possible for the field to take values less than 0 regardless of mean and covariance, which is not allowed for an intensity function. Instead, we let the intensity be given by a realization of $\exp(Z_s) = P(s)$ where $P(s)$ is the so-called stochastic intensity. The intensity is then given by

$$\rho(u) = \mathbb{E}[P(u)] = \exp\left(\mu(u) + \frac{C(u,u)}{2}\right) = \exp\left(\mu(u) + \frac{\sigma^2}{2}\right). \tag{13}$$

After a stochastic intensity has been generated it is then used as the intensity for an inhomogeneous Poisson process as described in the previous Section.

To see that the intensity is indeed given by (13) we remind ourselves that the moment-generating function of a random variable X is given by

$$M_X[t] = \mathbb{E}[e^{Xt}]$$

and if X is a Gaussian distributed random variable with mean μ and variance σ^2 we have

$$M_X[t] = e^{\mu t} e^{\frac{\sigma^2 t^2}{2}}$$

which is exactly what we see in (13).

If the random field Z_s, $s \in S$ is stationary and isotropic and given pairwise disjoint $s_1, \ldots, s_n \in S$, let $\xi = \sum_{i=1}^{n} \mu(s_i)$ and

$$\kappa = \sum_{i=1}^{n} \sigma^2(s_i) + 2 \sum_{1 \leq i < j \leq n} \sigma(s_i)\sigma(s_j)c(s_i, s_j),$$

We then have that

$$\sum_{i=1}^{n} Z_{s_i} \sim N(\xi, \kappa)$$

We remind ourselves that the covariance function of a random field is given by

$$C(s,t) = \sigma^2 c(s,t)$$

where $c(s,t)$ is the correlation function. We also remind ourselves that

$$\varrho^{(n)}(s_1, \ldots, s_n) = \mathbb{E}\left[\prod_{i=1}^{n} \Lambda(s_i)\right]$$

and for a Log-Gaussian Cox process, we have that $\Lambda(s) = \exp(Z_s)$. This gives us that [Møller et al., 1998, p. 456]

$$\begin{aligned} \varrho^{(n)}(s_1, \ldots, s_n) &= \mathbb{E}\left[\prod_{i=1}^{n} \Lambda(s_i)\right] = \mathbb{E}\left[\prod_{i=1}^{n} \exp(Z_{s_i}))\right] \\ &= \mathbb{E}\left[\exp\left(\sum_{i=1}^{n} Z_{s_i}\right)\right] = \exp(\xi + \frac{\kappa}{2}). \end{aligned}$$

In Figure 3 we can see an example of the point process with

$$\mu(u) = \mu(x, y) = 4.5x$$

and

$$C(s,t) = \sigma^2 \exp(|s-t|/\theta) = 2\exp(|s-t|/0.1).$$

In the left plot, we see a realization of the process, in the middle, we see the stochastic intensity, and to the right, we see the realization of the Gaussian random field which was transformed and used as the stochastic intensity.

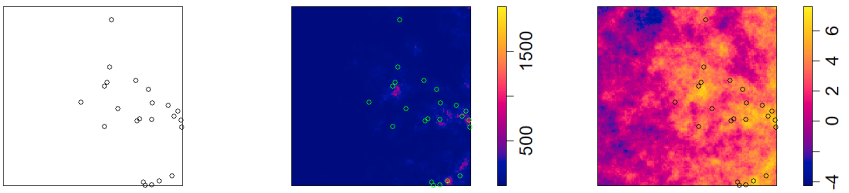


Figure 3: An example of a Log-Gaussian Cox process along with the underlying intensity and the Gaussian field. Left: point pattern, middle: intensity, and right: Gaussian field.

2.3.4 Simple sequential inhibition process

Many natural phenomena that can be modeled as a point process, such as the locations of trees in a forest, exhibit a quality known as inhibition. Inhibition is the property that for all points x_i of a point process $X \in W \subseteq S$ there is a distance $d \in (0, \infty]$ such that the probability of observing another point x_j of X within a d-radius ball of x_i is lower than outside the ball. If this probability is 0 we say that X is a hard-core process and otherwise we say that X is a soft-core process. This distance d is called the soft- or hard-core distance [Chiu et al., 2013, p. 176]. Alternatively, point processes are said to display inhibition if their pair correlation function is $g < 1$ [Van Lieshout, 2019, p. 102].

We will be investigating the so-called simple sequential inhibition process which is an example of a hard-core process. This point process can be simulated by starting with an empty point process $X = \emptyset$ and thereafter iteratively adding new points to X in $W \subseteq S$. We start by adding a point that is uniformly distributed on the entirety of W, thereafter, new points are distributed uniformly on $W \setminus \bigcup_{i=1}^{n} b(x_i, d)$ for $x_i \in X$ where $b(x_i, d)$ is a d-radius disc centered on x_i [Illian et al., 2008, p. 133].

For this thesis, we will only consider point processes of this kind with r such that it is always possible to generate the desired number of points n, i.e. $\mathbb{E}[X(W)] = n$, in which case the process is stationary. Therefore, we have that the intensity is

$$\mathbb{E}[X(W)] = n = \int_W \rho(u)du = \rho|W|$$

and since we are only interested in the unit window and thus $|W| = 1$ we get $\rho = n$.

2.4 Intensity estimation

As we now have seen, the intensity is an important characteristic of point processes and as such the estimation of it is of great interest. We will now describe

some methods that are used to do this.

If we have a constant intensity ρ and observe a point pattern $\mathbf{x}$ in some observation window W a fairly natural estimation method of ρ is simply

$$\hat{\rho} = \frac{\mathbf{x}(W)}{|W|}.$$

In this thesis, we will only look at non-parametric intensity estimation methods. In parametric methods, we assume that the observed data comes from such distribution parameterized by some θ and try to find an estimate $\hat{\theta}$ using the data. In non-parametric methods, we make no assumptions about the distribution which has generated the data.

2.4.1 Kernel density estimation

Before we can describe the first intensity estimation method we have to introduce the general concept of kernel estimation, which is a method for estimating probability density functions. Given an independent and identically distributed sample $x_1, \ldots, x_n$ from some unknown distribution with probability density function $f(x)$ we use a kernel function $k(x)$ which satisfies

$$\int_{-\infty}^{\infty} k(x)dx = 1$$

to find an estimate of $f(x)$ via

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} k_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} k(\frac{x - x_i}{h})$$

where h is the so-called bandwidth. The kernel function is usually a symmetric probability density function [Silverman, 1998, p. 13-15]. One of the main challenges in kernel estimation is choosing an appropriate bandwidth.

2.4.2 Kernel intensity estimation

Kernel estimation has been the most prominent method to estimate the intensity function of a point process for some time. In this scenario, given a point pattern $\mathbf{x}$, the intensity estimate is given by

$$\hat{\rho}_h^K(u) = \sum_{x \in \mathbf{x}} \frac{k_h(u - x)}{w_h(u, x)}$$

where $w_h(u, x)$ is an edge-correction term [Cronie et al., 2021, p. 11].

If we look at Figure 4 we can see an example of how this intensity estimation works. We have a homogeneous Poisson process with $\rho = 5$ and on each point x of X we center a, in this case, Gaussian kernel with bandwidth 0.1. As we can see this bandwidth is far from optimal but it serves well to illustrate the method.

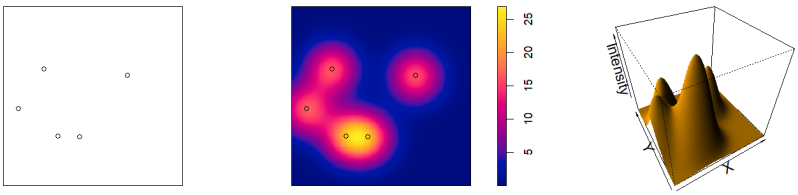


Figure 4: An example of kernel intensity estimation on a homogeneous Poisson process with $\rho = 5$. Left: point pattern, middle: estimated intensity with point pattern overlaid, and right: 3d plot of estimated intensity.

The current state-of-the-art bandwidth selection method was presented by Cronie and Lieshout [2018] which we will now give a brief description of. Given a point process X on $\mathbb{R}^2$ with intensity function $\rho(x)$ the so-called Campbell theorem states that for any non-negative, measurable function $h : \mathbb{R}^d \to \mathbb{R}_+$ we have

$$\mathbb{E}[\sum_{x \in X} h(x)] = \int_{\mathbb{R}^d} h(x)\rho(x)dx$$

where $\mathbb{R}_+ = \{x \mid x \in \mathbb{R}, x \geq 0\}$ [Cronie and Lieshout, 2018, p.456]. Given this theorem, and if we observe X in a window W, we have that

$$\mathbb{E}\left[\sum_{x \in X \cap W} \frac{1}{\rho(x)}\right] = \int_W \frac{1}{\rho(x)}\rho(x)dx = |W|$$

if $\rho(x) > 0$ for any $x \in W$. We have that $\sum_{x \in X \cap W} \frac{1}{\rho(x)}$ is an unbiased estimator of the size of the window W which is what this bandwidth selection method utilizes. Define

$$T_\kappa(h; X, W) = \begin{cases} \sum_{x \in X \cap W} \frac{1}{\hat{\rho}_h(x; X, W)} & \text{if } X \cap W \neq \emptyset \\ |W| & \text{otherwise} \end{cases}$$

and

$$F_\kappa(h; X, W) = (T_\kappa(h; X, W) - |W|)^2$$

where $\hat{\rho}_h(x)$ is the kernel intensity estimate at location x with bandwidth h. The bandwidth h is then chosen by finding the bandwidth that minimizes $F_\kappa(h; X, W)$ [Cronie and Lieshout, 2018, p. 457]. If $X \cap W = \emptyset$ we can make no conclusion about what bandwidth is optimal and so for any h we have that $F_\kappa(h; X, W)$ is 0. Essentially, this approach allows us to choose a bandwidth that optimally estimates the size of the observation window instead of trying to optimally estimate the intensity.

2.4.3 Voronoi tessellations

Now we move on to the next intensity estimation method we will investigate and start by introducing the Voronoi tessellation. Let $\mathbf{x} = \{x_1, \dots, x_n\}$ be a collection of points in some space S. Each point of S then belongs to a Voronoi cell $\mathcal{V}_{x_i}(\mathbf{x})$ for some $x_i \in \mathbf{x}$. A Voronoi cell is defined as [Chiu et al., 2013, p. 347]

$$\mathcal{V}_{x_i}(\mathbf{x}) = \{u \in S : d(u, x_i) \leq d(u, x_j), \forall x_j \in \mathbf{x} \setminus \{x_i\}\} \tag{14}$$

where $d(u, v)$ is the distance metric on S between the points u and v. Since we are only interested in $\mathbb{R}^2$, $d(u, v)$ is simply the Euclidean distance $d(u, v) = \|u - v\|$. If we instead have a point process $\mathbf{x} \in W \subseteq S$, we write

$$\mathcal{V}_{x_i}(\mathbf{x}, W) = \{u \in W : d(u, x_i) \leq d(u, x_j), \forall x_j \in \mathbf{x} \setminus \{x_i\}\}. \tag{15}$$

If we look at Figure 5 we can see an example of a Voronoi tessellation on the same point pattern as in Figure 4.

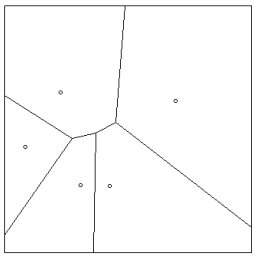


Figure 5: An example of a Voronoi tessellation of a homogeneous Poisson process with $\rho = 5$.

2.4.4 Voronoi intensity estimation

Now that we know what a Voronoi tessellation is we can describe how it can be used to estimate the intensity. Given a point process X in some window $W \subseteq S$ the Voronoi intensity estimator for a point $u \in W$ is given by [Moradi et al., 2018]

$$\hat{\rho}^{\mathcal{V}}(u; X, W) = \sum_{x \in X \cap W} \frac{\mathbf{1}\{u \in \mathcal{V}_x\}}{|\mathcal{V}_x|}. \tag{16}$$

The idea here is that at a point u of W with few nearby points of $\mathbf{x}$ its corresponding Voronoi cell will be larger and as such the estimated intensity $\frac{\mathbf{1}\{u \in \mathcal{V}_x\}}{|\mathcal{V}_x|}$ at u will be smaller.

If we look at Figure 6 we can see an example of this intensity estimation method on an inhomogeneous Poisson process with

$$\rho(u) = \rho(x, y) = 5 + 50x.$$

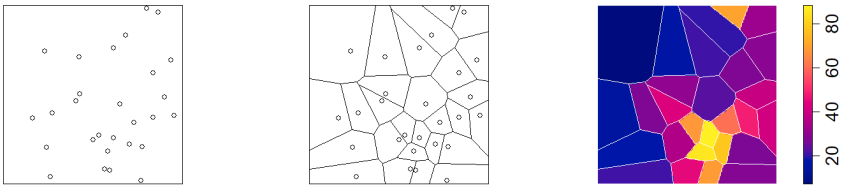


Figure 6: An example of Voronoi intensity estimation on an inhomogeneous Poisson process. Left: point pattern, middle: Voronoi tessellation with point pattern overlaid, and right: estimated intensity with Voronoi tessellation overlaid.

2.4.5 Resample-smoothing of Voronoi intensity estimators

As we saw in the previous section the estimated intensity was not terribly accurate. However, if we instead perform a series of these estimations based on thinned point patterns and use the mean as the estimate we get much better results. Note, that this is not a method exclusive to the Voronoi estimator and works for all types of intensity estimators. The general idea is that if a point process is thinned with a probability $p(x)$ then the intensity of that thinned point process is $\rho_t(x) = p(x)\rho(x)$. We can then estimate the intensity of this thinned point process and use it to find an estimate of the original intensity via

$$\hat{\rho}(x) = \frac{\hat{\rho}_t(x)}{p(x)}.$$

Furthermore, if we thin the point process m times and find m intensity estimates we get

$$\hat{\rho}(x) = \frac{1}{m}\sum_{i=1}^{m}\frac{\hat{\rho}_t(x)}{p(x)} = \frac{1}{mp(x)}\sum_{i=1}^{m}\hat{\rho}_t(x).$$

Given a point process X in some window $W \subseteq S$, a retention probability p_v and a number m of random independent thinnings, the resample-smoothed Voronoi intensity estimator is given by Moradi et al. [2018]

$$\hat{\rho}^{\mathcal{V}}_{p_v,m}(u; X, W) = \frac{1}{mp_v}\sum_{i=1}^{m}\sum_{x\in X_i^{p_v}}\frac{\mathbf{1}\{u \in \mathcal{V}_x(X_i^{p_v}, W)\}}{|\mathcal{V}_x(X_i^{p_v}, W)|} \tag{17}$$

where X_i^p is the ith random thinning.

In short, we perform Voronoi intensity estimation on m copies of the point pattern that each have been thinned with some, but the same for all copies, probability $p_v \in (0, 1)$ and then use the mean as the intensity estimate. Since this intensity estimate is based on thinned point patterns we must also finally divide by p_v to ensure that we are on the right scale.

Moradi et al. [2018] propose as a rule of thumb to let $p_v \leq 0.2$ and $m = 400$. In general, Moradi et al. [2018] found that lower values of p_v performed better and that there was only a small performance improvement to be gained by letting $m \geq 200$.

2.4.6 Conditional intensity estimation

It is important to point out that for both estimators we have seen, given a point pattern $\mathbf{x}$ of some point process X, we have that the intensity estimate is of the form $\hat{\rho}(u; \mathbf{x})$ i.e., the estimator depends on the point pattern which means that it is a conditional intensity. Since we also have

$$\rho(u) = \mathbb{E}[\lambda(u; X)]$$

and that we can assume that the point pattern $\mathbf{x}$ is fairly typical in the distribution of X we also get that

$$\rho(u) = \mathbb{E}[\lambda(u; X)] \approx \lambda(u, \mathbf{x}).$$

Previously in this thesis, we have seen non-parametric intensity estimation methods that depend on some parameter θ. The goal of this thesis is to investigate whether or not point process learning can choose this parameter θ well. If we do choose θ well, we have $\lambda(u, \mathbf{x}) \approx \hat{\rho}_{\hat{\theta}}(u; \mathbf{x})$, i.e. the intensity estimate parameterized by $\hat{\theta}$ is close to the true intensity, and thus we get

$$\rho(u) = \mathbb{E}[\lambda(u; X)] \approx \lambda(u, \mathbf{x}) \approx \hat{\rho}_{\hat{\theta}}(u; \mathbf{x})$$

which we can then use as our estimate for the intensity function.

2.5 Edge effects

A common problem that affects most point process inference is edge effects. Edge effects are caused by the fact that we, in most situations, only observe a part of the point process. However, some of the points that are not observed would still have had an impact on what we are trying to estimate in the observed window. For example, if we consider a kernel intensity estimation using a Gaussian kernel then the kernels centered on points close to our observed window W would still have parts in W. The same is true for the Voronoi estimator, for points in W close to the edge of the window it can be true that there would be a point just outside the edge of the window, resulting in the points corresponding Voronoi cell being larger than it should and thus the intensity will be lower.

If we look at Figure 7 we can see an example of how edge effects look when using Voronoi intensity estimation. We start with a homogeneous Poisson process with $\rho = 50$ on the unit window marked by the dashed line. We then, at first, only observe and estimate the intensity on the window $[0.2, 0.8]^2$ which is seen in the middle plot. As we can see, the intensity around the edges is in this case much larger. Looking at the right plot, where the full point pattern was used to estimate the intensity, we see that the estimate in the sub-window looks much better.

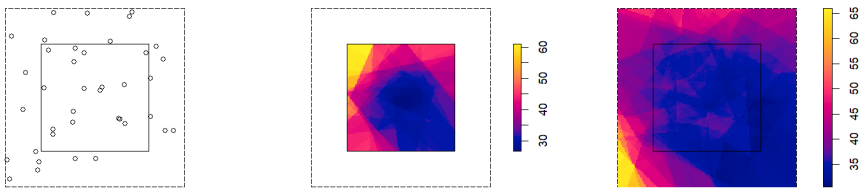


Figure 7: An example of edge effects on Voronoi intensity estimation on a homogeneous Poisson process with $\rho = 50$.

3 Point process learning

Now, we can move on to the main topic of this thesis: point process learning, which is a new statistical theory for point processes. Point process learning utilizes cross-validation to create training and validation sets which are then used in point process prediction errors parameterized by some parameter of interest θ. These errors are then combined and transformed and used as loss functions allowing us to choose an optimal θ.

3.1 Cross-validation

An important concept in the field of statistical and machine learning which we will be using is cross-validation. The term cross-validation within statistical learning refers to the method of splitting the available data into a training and validation set. The training set is used to estimate the parameters of a statistical model after which the validation data is used to evaluate the statistical model given the estimated parameters. There are different methods for constructing cross-validation sets. A cross-validation method for point patterns is to construct independent thinned copies [Cronie et al., 2021, p. 13]. Given a point pattern $\mathbf{x}$ we can construct training and validation sets via random p-thinning of $\mathbf{x}$, for some $p \in (0, 1)$. Given a random p-thinning $\mathbf{z}$, the training

and validation sets are then given by [Cronie et al., 2021, p. 14]

$$\mathbf{x}^T = \mathbf{z}$$
$$\mathbf{x}^V = \mathbf{x} \setminus \mathbf{z}.$$

Often we have that we perform repeated cross-validations which means we instead have $k > 1$ independent thinnings, $\mathbf{z}_1, \ldots, \mathbf{z}_k$. We then get

$$\mathbf{x}_i^T = \mathbf{z}_i$$
$$\mathbf{x}_i^V = \mathbf{x}_i \setminus \mathbf{z}_i$$
$$i = 1, \ldots, k.$$

We will investigate two methods of choosing cross-validation sets; multinomial k-fold and Monte Carlo. Both of these cross-validation methods use independent thinning which is important as in such cases we know that the thinned intensity is $\rho_t(x) = p(x)\rho(x)$ and can therefore be used to estimate the original intensity.

3.1.1 Multinomial k-fold cross-validation

Given some $k \geq 2$ and a point pattern $\mathbf{x}$, we create training and validation sets by generating the marked pattern $\breve{\mathbf{x}} = \{(x_i, m_i)\}_{i=1}^n, m_i \in \{1, \ldots, k\}$ where m_i are from a multinomial distribution with k outcomes and $p_1 = \ldots = p_k = 1/k$. We then create k validation and training sets via

$$\mathbf{x}_i^V = \{x_j \in \mathbf{x} : m_j = i\}$$
$$\mathbf{x}_i^T = \mathbf{x} \setminus \mathbf{x}_i^T = \{x_j \in \mathbf{x} : m_j \neq i\}.$$

The difference between multinomial k-fold cross-validation and the general cross-validation idea as described in Section 3.1 is that $\mathbf{x}_i^V \cap \mathbf{x}_j^V = \emptyset, i \neq j$.

We note that multinomial k-fold is different from the classical k-fold cross-validation. In classical k-fold cross-validation we have some data $\mathbf{y} = \{y_1, \ldots, y_n\}$ which is then randomly shuffled after which $\mathbf{y}$ is split into k-separate subsets. A statistical or machine learning model is then trained using $k-1$ of these subsets with the last being used to evaluate the model. This is done a total of k times to ensure that each subset is used to evaluate a model exactly one time. The reason for not using this approach is that this is not a case of independent thinning and as such the alternative multinomial k-fold cross-validation was introduced by Cronie et al. [2021] for point process learning.

3.1.2 Monte Carlo cross-validation

In the same setting as the previous section along with some probability $p \in (0, 1)$ we create k training and validation sets by creating k p-thinnings $\mathbf{z}_1, \ldots, \mathbf{z}_k$. Using these we create training and validation sets via

$$\mathbf{x}_i^V = \mathbf{z}_i$$
$$\mathbf{x}_i^T = \mathbf{x} \setminus \mathbf{z}_i.$$

Note that in this setting we do not necessarily have $\mathbf{x}_i^V \cap \mathbf{x}_i^V = \emptyset,\ i \neq j$.

3.2 Prediction errors

We will now describe the foundational concept of this entire thesis: point process prediction errors. We start by giving a general description of what an estimator is. If we have a point process X which we observe in some $W \subseteq S$ an estimator assigns some numerical value to a characteristic of X based on the observed realization $\mathbf{x}$. The family of so-called general parameterized estimators can be described as

$$\Xi_\Theta = \{\xi_\theta : \theta \in \Theta\},$$

with

$$\xi_\theta(u; \mathbf{x}) \in \mathbb{R}^d,\ u \in S,\ \theta \in \Theta \subseteq \mathbb{R}^l$$

where $d,\ l \geq 1$. Note that it is not always true that $\xi_\theta(u; \mathbf{x})$ depends on $\mathbf{x}$, for example, if $\xi(u; \mathbf{x})$ is constant with respect to $\mathbf{x}$, we have instead $\xi_\theta(u)$. In our case we have that $\xi(u; \mathbf{x}) = w\rho(u; \mathbf{x})$ with either the Voronoi estimator, $\rho^{\mathcal{V}}$ and $\theta = (p_v, m)$ or the kernel estimator ρ^K with $\theta = h$. We also have a weight w, which is to ensure that the prediction error has an expected value of 0.

Point process prediction errors, or innovations as they have also been referred to, for two point patterns $\mathbf{y}$ and $\mathbf{z}$, are measures of how well Y can predict Z. A bivariate prediction error is defined as [Cronie et al., 2021, p. 9]

$$\mathcal{I}_{\xi_\theta}^{h_\theta}(W; \mathbf{z}, \mathbf{y}) = \sum_{x \in \mathbf{z} \cap W} h_\theta(x; \mathbf{y} \setminus x) - \int_W h_\theta(u; \mathbf{y}) \xi_\theta(u; \mathbf{y}) du. \tag{18}$$

Furthermore, in this thesis we will assume that $h_\theta(u; \mathbf{x})$ is given by

$$h_\theta(u; \mathbf{x}) = f(\xi_\theta(u; \mathbf{x}))$$

for some $f : \mathbb{R} \to \mathbb{R}$ which is called the test function, which gives us

$$\begin{aligned}
\mathcal{I}_{\xi_\theta}^{h_\theta}(W; \mathbf{z}, \mathbf{y}) &= \sum_{x \in \mathbf{z} \cap W} f(\xi_\theta(x; \mathbf{y} \setminus x)) - \int_W f(\xi_\theta(u; \mathbf{y})) \xi_\theta(u; \mathbf{y}) du \\
&= \sum_{x \in \mathbf{z} \cap W} f(w\rho_\theta(x; \mathbf{y} \setminus x)) - \int_W f(w\rho_\theta(u; \mathbf{y})) w\rho_\theta(u; \mathbf{y}) du
\end{aligned}$$

where w is, as previously mentioned, a weight that ensures that

$$\mathbb{E}[\mathcal{I}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^{V,p}, \mathbf{x}_i^{T,p})] = 0$$

where $\mathbf{x}_i^{T,p}$ and $\mathbf{x}_i^{V,p}$ are a pair of training and validation sets that have been thinned by some retention probability p. This weight is dependent on whether or not the process being investigated is attractive or repulsive. Given a point process X which generates a point pattern $\mathbf{x}$ and a $\mathbf{y} \subseteq \mathbf{x}$ we say that X is attractive or repulsive if the conditional intensity $\lambda(\cdot; \mathbf{y})$ is smaller or larger,

respectively, than $\lambda(\cdot; \mathbf{x})$ [Cronie et al., 2021, p. 4]. If Z is an independent p-thinning of X and $Y = X \setminus Z$ then we have that $w \leq p$ if X is repulsive, $w \geq p$ if X is attractive, and $w = p$ if X is a Poisson process [Cronie et al., 2021, p. 10]. For this thesis, we will use $w = p$ as it works for all types of point processes.

A common group of test functions is $f(x) = x^{-\gamma}$ with $\gamma = 0.5$ or $\gamma = 1$ with the choice of $\gamma = 1$ resulting in low variance of estimators [Cronie et al., 2021, p. 8]. In the case of $\gamma = 1$ we get that (18) becomes

$$\begin{aligned}
\mathcal{I}^{h_\theta}_{\xi_\theta}(W; \mathbf{z}, \mathbf{y}) &= \sum_{x \in \mathbf{z} \cap W} h_\theta(x; \mathbf{y} \setminus x) - \int_W h_\theta(u; \mathbf{y}) \xi_\theta(u; \mathbf{y}) du \\
&= \sum_{x \in \mathbf{z} \cap W} \frac{1}{\xi_\theta(x; \mathbf{y} \setminus x)} - \int_W \frac{\xi_\theta(u; \mathbf{y})}{\xi_\theta(u; \mathbf{y})} du \\
&= \sum_{x \in \mathbf{z} \cap W} \frac{1}{\xi_\theta(x; \mathbf{y} \setminus x)} - \int_W 1 du \\
&= \sum_{x \in \mathbf{z} \cap W} \frac{1}{\xi_\theta(x; \mathbf{y} \setminus x)} - |W|.
\end{aligned}$$

If we apply the prediction error to a pair of training and validation sets we get

$$\begin{aligned}
\mathcal{I}^{h_\theta}_{\xi_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T) &= \sum_{x \in \mathbf{x}_i^V} h_\theta(x; \mathbf{x}_i^T) - \int_W h_\theta(u; \mathbf{x}_i^T) \xi_\theta(u; \mathbf{x}_i^T) du \qquad (19) \\
&= \sum_{x \in \mathbf{x}_i^V} f(\xi_\theta(x; \mathbf{x}_i^T)) - \int_W f(\xi_\theta(u; \mathbf{x}_i^T)) \xi_\theta(u; \mathbf{x}_i^T) du \\
&= \sum_{x \in \mathbf{x}_i^V} f(w\rho_\theta(x; \mathbf{x}_i^T)) - \int_W f(w\rho_\theta(u; \mathbf{x}_i^T)) w\rho_\theta(u; \mathbf{x}_i^T) du.
\end{aligned}$$

Heuristically, we have that the actual prediction is done by the summation in (18) while the integral simply ensures that that prediction error is 0 when $\xi_\theta(u) = w\rho(u)$ is the true intensity of $\mathbf{y}$ and $\mathbf{z}$. Simply put, the prediction error is 0 when it is based on the true intensity. If we use an intensity estimate, $\hat{\rho}$ in (19), then the closer the intensity estimate is to the true intensity ρ, the closer the prediction error will be to 0.

3.3 Loss functions

Next, we describe how to combine and transform the prediction errors from the training and validation sets into one loss value. Loss functions are used in optimization problems in order to find an approximate optimal solution. In general, these problems can be described as, assuming that the obtained data, $\mathbf{x}$, comes from one of the models within a family parameterized by θ and that the

loss function is intended to bring us as close to the true parameter as possible, using the data to find an estimate $\hat{\theta} = \hat{\theta}(\mathbf{x})$ of θ, which is a minimizer of the loss function $\mathcal{L}(\theta)$, $\theta \in \Theta$. The closer the estimate is to the true value θ the lower $\mathcal{L}(\hat{\theta})$ will be. Hence, the goal is to find

$$\arg\min_{\theta} \mathcal{L}(\theta)$$

which we choose as our final parameter estimate, i.e. a candidate for the true, unknown, parameter.

In the case of point process learning, we have that [Cronie et al., 2021, p. 16]

$$\mathcal{L}(\theta) = \mathcal{L}(\theta; \{(\mathbf{x}_i^V, \mathbf{x}_i^T)\}_{i=1}^k, p_c, k, \Xi_\Theta, \mathcal{H}_\Theta), \quad \theta \in \Theta \tag{20}$$

for Monte-Carlo and

$$\mathcal{L}(\theta) = \mathcal{L}(\theta; \{(\mathbf{x}_i^V, \mathbf{x}_i^T)\}_{i=1}^k, k, \Xi_\Theta, \mathcal{H}_\Theta), \quad \theta \in \Theta$$

for k-fold cross-validation, and will be based on an estimate of the point prediction error. Here we have a point pattern, $\mathbf{x}$, which is a realization of a point process X which has then been split into the k validation and training sets $\{(\mathbf{x}_i^V, \mathbf{x}_i^T)\}_{i=1}^k$ by some cross-validation method. In the case of Monte Carlo cross-validation, the validation sets are independent p_c-thinnings of $\mathbf{x}$. Furthermore, we have that Ξ_Θ and $\mathcal{H}_\Theta$ are the families of estimators and test functions, respectively, parameterized by $\theta \in \Theta$. In the case of the resample-smoothed Voronoi intensity estimator, we have that $\theta = (m, p_v)$ and that $\Theta = \mathbb{N} \times (0, 1)$ where $\mathbb{N} = \{1, 2, \ldots\}$, and in the case of the kernel intensity estimator, we have that $\theta = h$ and $\Theta = [0, \infty)$.

We should note at this point that it is possible for both cross-validation methods described to produce training and validation sets that are empty. In such a case the summation in (19) would disappear and might affect the estimation. To counteract this we modify the prediction error to instead be

$$\tilde{\mathcal{I}}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T) = \mathbf{1}\{\mathbf{x}_i^V(W) > 0 \wedge \mathbf{x}_i^T(W) > 0\} \mathcal{I}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T).$$

Some common choices of loss functions are

$$\mathcal{L}_1(\theta) = \frac{1}{k} \sum_{i=1}^k |\tilde{\mathcal{I}}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T)|$$

$$\mathcal{L}_2(\theta) = \frac{1}{k} \sum_{i=1}^k \left(\tilde{\mathcal{I}}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T)\right)^2$$

$$\mathcal{L}_3(\theta) = \frac{1}{k} \left(\sum_{i=1}^k \tilde{\mathcal{I}}_{\xi_\theta}^{h_\theta}(W; \mathbf{x}_i^V, \mathbf{x}_i^T)\right)^2.$$

3.4 Evaluation metrics

Now that we have described the method of choosing optimal parameters we also need to introduce some methods of evaluating these choices. Given the true intensity $\rho(u)$ at some location u in an observation window W and an intensity estimate $\hat{\rho}_\theta(u)$ we will use

- Integrated Absolute Bias,

$$\text{IAB} = \int_W |\mathbb{E}[\hat{\rho}_\theta(u;X)] - \rho(u)|du = \int_W |\bar{\rho_\theta}(u;X) - \rho(u)|du$$

- Integrated Variance,

$$\text{IV} = \int_W \text{Var}(\hat{\rho}_\theta(u;X))du$$

- Integrated Square Bias,

$$\text{ISB} = \int_W \left(\mathbb{E}[\hat{\rho}_\theta(u;X)] - \rho(u)\right)^2 du = \int_W (\bar{\rho_\theta}(u;X) - \rho(u))^2 du$$

- Mean Integrated Square Error, MISE = IV + ISB.

4 Simulation study

Now that the relevant preliminaries and theory have been explained we can move on to the actual simulation study. We start by further describing the point processes that will be investigated and the methods that will be used.

4.1 Point process models

The point processes used in this project are the same as the ones used in the paper by Moradi et al. [2018] so that the results obtained here can be compared to previous results. These point processes are:

- A homogeneous Poisson process with $\rho = 60$.
- An inhomogeneous Poisson process with $\rho(x,y) = |10 + 90\sin(16x)|$.
- A simple sequential inhibition (SSI) process with an inhibition distance of 0.03 and a total number of points of 450. With this inhibition distance, it is always possible to place 450 points which means that the intensity is simply $\rho(x,y) = 450$. Next, this process is further thinned with retention probability

$$\begin{aligned} p(x,y) = &\mathbf{1}\{x < 1/3\}|x - 0.02| + \\ &\mathbf{1}\{1/3 \le x < 2/3\}|x - 0.5| + \\ &\mathbf{1}\{x \ge 2/3\}|x - 0.95| \end{aligned}$$

  resulting in an inhomogeneous model with intensity $\rho(x,y) = 450p(x,y)$.

- A Log-Gaussian Cox process with mean function $\mu(x, y) = \log(40|\sin(20x)|)$ and covariance function $C(s, t) = 2\exp\{-\|s - t\|/0.1\}$ for the underlying random field. This results in the intensity

$$\rho(x, y) = \mathbb{E}[P(x, y)] = \exp\{\log(40|\sin(20x)|) + \frac{2}{2}\} = 40|\sin(20x)|e^1.$$

All these point processes are realized on the unit window $W = [0, 1]^2$. When realized on such a window these processes have an expected number of points of 60, 58.6, 68.4, and 53.6 respectively.

Choosing these point processes allows us to test the method on a variety of types of point processes. We have both homogeneous and inhomogeneous Poisson processes, inhibition and inhomogeneity in the thinned SSI process, and clustering and inhomogeneity in the Log-Gaussian Cox process.

4.2 Set-up

Before we start applying point process learning we will discuss some of the parameter and hyperparameter choices as well as give an overview of the general algorithm that we apply.

The primary parameter of interest in this simulation study is the retention probability p_v in the resample-smoothing Voronoi intensity estimator (equation 17). As seen in Tables 1-7 in Moradi et al. [2018] and discussed in sections 4.1-4.4 the optimal choice seems to follow the thumb-rule $p_v \leq 0.2$. Given this, we employ a higher degree of granularity in our choices for $p_v \leq 0.2$ and let

$$p_v \in \{.01, .02, .03, .04, .05, .06, .07, .08, .09, .1, .15, .25, .5, .75, .9\}.$$

Originally, we had even more choices of p_v in the interval $[0.1, 0.2]$ but decided to cut these out due to the computational complexity. The choice of the number of samples in the Voronoi estimator, m, is also an important parameter. Looking again at Tables 1, 3, 5, and 7 in Moradi et al. [2018] we see that it seems like IAB and ISB do not improve much for $m \geq 200$. However, we do see that IV improves noticeably which would result in MISE also improving. As we see that larger values of m improve performance we will investigate $m \in \{250, 500, 1000, 2000\}$. Ideally, we would investigate some values of $m > 2000$ but due to the computational complexity, we limit ourselves to these values.

There are also a number of hyperparameters that are used to create the cross-validation sets: k for both k-fold and MCCV and p_c for MCCV. We choose the common values $k \in \{2, 5, 10\}$ for k-fold. For Monte Carlo cross-validation Cronie et al. [2021] recommend fixing $k \geq 100$ as performance above that does not significantly increase. However, due to the computational complexity of MCCV, we limit ourselves to $k = 50$.

In this setting, we also need to choose a $p_c \in (0, 1)$ which is the retention probability for the p-thinning used to create the training and validation sets. We again take inspiration from Cronie et al. [2021] and investigate $p_c = 0.1, 0.3, 0.5, 0.7, 0.9$.

As previously mentioned, the test function $f(x)$ as described in section 3.2 is chosen to be $f(x) = x^{-\gamma} = x^{-1}$.

4.2.1 Algorithm

Finally, we present an overview of the algorithm used.

- Given a point process X, generate $\mathbf{x}_1, \ldots, \mathbf{x}_{100}$ realizations
- For each $\mathbf{x}_i$, $i = 1, \ldots, 100$
  - find the set $\{\mathcal{L}(\theta; \mathbf{x}_i)\}$ for all parameter candidates $\theta \in \Theta$
  - Choose a parameter estimate $\hat{\theta}_i = \arg\min\{\mathcal{L}(\theta; \mathbf{x}_i)\}$
  - Calculate the intensity estimate on the full point pattern using the estimated parameter, $\hat{\rho}_{\hat{\theta}_i}(\mathbf{x}_i)$
- Finally, find the evaluation metrics based on all intensity estimates $\hat{\rho}_{\hat{\theta}_i}(\mathbf{x}_i)$, $i = 1, \ldots, 100$.

4.3 Voronoi intensity estimation

We start by presenting the result of applying point process learning to choose parameters for the Voronoi intensity estimator. A number of tables will be present where the column "all k" appears. This is the result when treating k as a parameter and not a hyperparameter. More precisely, instead of selecting

$$\hat{\theta} = \arg\min_{\theta}\{\mathcal{L}(\theta)\} = \arg\min\{\mathcal{L}(\theta; \{(\mathbf{x}_i^V, \mathbf{x}_i^T)\}_{i=1}^k, p_c, k, \Xi_\Theta, \mathcal{H}_\Theta)\}$$

with $\theta = (m, p_v)$, we choose

$$\hat{\theta} = \arg\min_{\theta}\{\mathcal{L}(\theta)\} = \arg\min\{\mathcal{L}(\theta; \{(\mathbf{x}_i^V, \mathbf{x}_i^T)\}_{i=1}^k, p_c, \Xi_\Theta, \mathcal{H}_\Theta)\}$$

where $\theta = (m, p_v, k)$.

The values we will see in this subsection will work as a baseline for future comparisons.

4.3.1 Homogeneous Poisson process

In Table 1 we see the evaluation metrics for the homogeneous Poisson process where we can see that $k = 5$ seems to achieve the lowest MISE score of around 225-230. Comparing this to Table 1 in Moradi et al. [2018] we see that the best performance is $p_v = 0.01$ with a MISE of around 85-100 which is better than what our method achieves.

Looking at Figure 8 we see three plots, an intensity estimate, the intensity error which is simply the intensity estimate minus the true intensity, and finally, the true intensity. All of these plots have the point pattern overlaid. As we can

see, the intensity estimate looks quite homogeneous although it is affected by edge effects.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	239.32	234.86	237.31
5	187.64	179.13	179.05
10	205.13	195.44	176.05
all	193.14	190.14	170.17

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	39.90	38.39	38.92
5	45.77	45.81	46.20
10	43.49	46.35	47.28
all	43.55	43.88	47.09

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	5.21	5.17	5.19
5	5.76	5.80	5.81
10	5.60	5.78	5.87
all	5.57	5.64	5.87

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	279.22	273.25	276.23
5	233.42	224.94	225.25
10	248.62	241.78	223.33
all	236.69	234.02	217.26

Table 1: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.160	0.163	0.160
5	0.111	0.109	0.106
10	0.130	0.110	0.100

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	617.5	565.0	617.5
5	602.5	577.5	590.0
10	510.0	480.0	535.0

Table 2: Average parameter values selected for the Voronoi estimator of the homogeneous Poisson process.

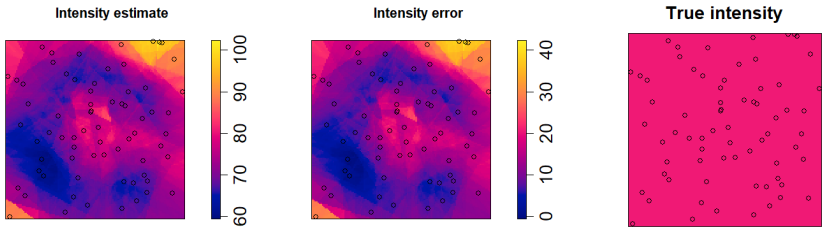


Figure 8: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the Voronoi method on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.3.2 Inhomogeneous Poisson process

Looking at Table 3 we see similar results in terms of MISE for $k = 5, 10$ and that $k = 2$ seems to perform slightly worse. Comparing this to $p_v = 0.01$ in Table 3 in Moradi et al. [2018] we see very similar results in terms of ISB and larger values of IV for our method.

Looking at Figure 9 we see that the intensity estimate looks fairly homogeneous with a fair degree of edge effects. We also note that the periodicity of the true intensity is not captured.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	257.46	243.23	260.38
5	200.33	197.43	200.08
10	200.70	172.06	177.45
all	198.64	183.57	179.01

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	884.71	884.98	883.94
5	888.59	885.91	887.07
10	886.16	891.84	890.21
all	889.48	888.85	889.19

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	25.45	25.48	25.43
5	25.51	25.46	25.49
10	25.49	25.53	25.52
all	25.51	25.52	25.53

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1142.16	1128.21	1144.32
5	1088.92	1083.34	1087.15
10	1086.85	1063.90	1067.65
all	1088.12	1072.42	1068.21

Table 3: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.169	0.166	0.169
5	0.119	0.122	0.118
10	0.124	0.100	0.106

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	680.0	722.5	680.0
5	602.5	517.5	615.0
10	612.5	522.5	575.0

Table 4: Average parameter values selected for the Voronoi estimator of the inhomogeneous Poisson process.

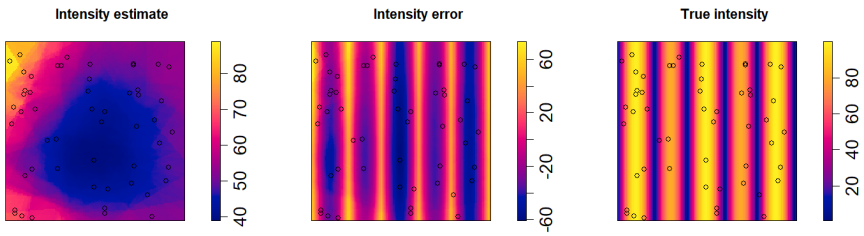


Figure 9: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.3.3 Log-Gaussian Cox process

Looking at Table 5 we can see the evaluation metrics for the Log-Gaussian Cox process. Comparing this to previous point processes we see much larger values here. We see that $k = 10$ has the lowest MISE, with $k = 5$ in second place. We also see that the different loss functions do not give similar results as we have seen previously, noticeably $\mathcal{L}_1$ with $k = 10$ performs best. Looking at Table 5 in Moradi et al. [2018] we see that IV decreases with p_v which is what we see here as well.

Looking at Figure 10 we again see a fair degree of edge effects and that the intensity estimate looks fairly homogeneous and fails to capture the periodicity of the intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	4193.32	4400.82	4152.71
5	2716.68	2921.55	3120.62
10	1750.73	1879.86	3286.17
all	2358.39	2472.93	3198.98

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1177.52	1169.44	1178.11
5	1199.63	1200.23	1192.39
10	1219.94	1212.15	1194.01
all	1210.68	1208.59	1194.10

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	29.87	29.76	29.87
5	30.23	30.21	30.08
10	30.59	30.48	30.18
all	30.40	30.34	30.14

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	5370.84	5570.25	5330.82
5	3916.31	4121.78	4313.01
10	2970.67	3092.01	4480.18
all	3569.06	3681.51	4393.08

Table 5: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.231	0.249	0.230
5	0.164	0.166	0.177
10	0.106	0.114	0.167

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	685.0	595.0	660.0
5	792.5	755.0	725.0
10	615.0	652.5	760.0

Table 6: Average parameter values selected for the Voronoi estimator of the Log-Gaussian Cox process.

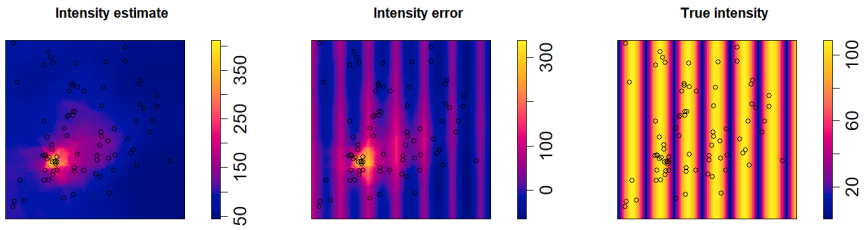


Figure 10: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.3.3.1 Monte Carlo cross-validation

Of all the point processes investigated the Log-Gaussian Cox process had the worst performance in terms of MISE but also displayed a significant amount of variance with respect to k. Due to this, we decided to see how Monte Carlo cross-validation performed on this model the results of which can be seen in Table 7. If we compare this to Table 5 we can see that only $p_c = 0.1$ achieves better results than $k = 10$ in the k-fold setting. In general, we can see that ISB decreases slightly with respect to p_c but that IV drastically increases.

Furthermore, we also treated p_c as a parameter rather than a hyperparameter, choosing p_c by finding the lowest loss value. The results of this approach can be seen in Table 8 which, as we can see, offers no improvement over fixing $p_c = 0.1$.

In Table 60 in the appendix, we can see the number of times each parameter was selected. We can clearly see that p_c is only ever chosen to be 0.1 or 0.3 which is tabulated in Table 61.

(a) IV

p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	1516.11	1523.35	2447.29
0.3	2632.21	2620.55	3044.13
0.5	3645.27	3668.46	3679.32
0.7	4832.23	4956.10	4848.08
0.9	11818.61	13193.57	11799.82

(b) ISB

p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	1215.13	1218.83	1200.51
0.3	1200.38	1199.82	1199.48
0.5	1186.88	1187.70	1184.81
0.7	1181.28	1180.45	1180.96
0.9	1112.36	1107.43	1112.22

(c) IAB

p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	30.40	30.46	30.21
0.3	30.18	30.20	30.15
0.5	30.01	30.04	29.98
0.7	29.89	29.86	29.87
0.9	28.75	28.64	28.75

(d) MISE

p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	2731.24	2742.18	3647.80
0.3	3832.60	3820.38	4243.61
0.5	4832.15	4856.16	4864.13
0.7	6013.51	6136.55	6029.04
0.9	12930.96	14301.00	12912.03

Table 7: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by MCCV.

	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
IV	1867.33	1870.06	2404.49
ISB	1206.67	1206.79	1198.79
IAB	30.32	30.29	30.21
MISE	3074.01	3076.85	3603.28

Table 8: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by MCCV, treating p_c as a parameter.

4.3.4 Simple sequential inhibition process

Looking at Table 9 we see MISE values of around 1550-1560 and that all values of k seem to perform very similarly. Comparing this to $p_v = 0.01$ in Table 7 in Moradi et al. [2018] we see that our method achieves slightly lower ISB but higher IV when looking at $p = 0.01$. As we have seen before, there appears to be no difference between the different loss functions.

Looking at Figure 11 we again see a fair degree of edge effects. Here we note that we see some of the higher intensity regions of the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	179.22	173.32	176.77
5	227.25	182.99	192.08
10	134.19	118.02	163.94
all	193.35	174.42	166.28

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1373.39	1372.32	1371.92
5	1327.31	1375.28	1363.95
10	1430.04	1450.33	1390.86
all	1355.26	1375.91	1391.64

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	31.10	31.09	31.09
5	30.47	31.02	30.94
10	31.68	31.87	31.26
all	30.78	31.02	31.28

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1552.61	1545.64	1548.70
5	1554.56	1558.27	1556.02
10	1564.23	1568.35	1554.80
all	1548.61	1550.32	1557.91

Table 9: Evaluation metrics for the simple sequential inhibition process.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.200	0.201	0.200
5	0.203	0.169	0.179
10	0.147	0.132	0.165

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	630.0	647.5	627.5
5	522.5	460.0	537.5
10	610.0	550.0	662.5

Table 10: Average parameter values selected for the Voronoi estimator of the simple sequential inhibition process.

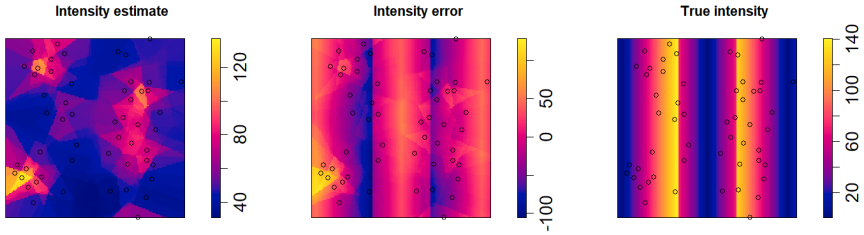


Figure 11: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.4 Kernel intensity estimation

In order to compare the results of the Voronoi estimator we will also perform kernel intensity estimation. To do this we need to choose some bandwidths to

investigate. Given a point pattern, $\mathbf{x} = \{x_i\}_{i=1}^n \subseteq W$, the nearest neighbor distance for a point x_i is

$$\min\{d(x_i, x_j) : x_j \in \mathbf{x}\}$$

where $d(x_i, x_j)$ is the distance from x_i to x_j. The smallest nearest neighbor distance is then

$$\min\{d(x_i, x_j) : x_i, x_j \in \mathbf{x}, x_i \neq x_j\}.$$

Next, we find the diameter of the window W,

$$\text{diam}(W) = \max\{d(u, v) : u, v \in W\}.$$

Knowing these two values we create a geometric sequence, i.e. a logarithmically linear sequence, from the smallest nearest neighbor distance to $\text{diam}(W)/2$. For each value in this sequence, the intensity is estimated using a Gaussian kernel. This method will not always work but the point processes examined here have a large enough mean number of points that it was not a problem. These are the default bandwidths used in the Cronie-van Lieshout bandwidth selection method as implemented in the `R` library `spatstat` [Baddeley].

In Section 2.4.2 we mentioned an edge-correction term. As there are no equivalent edge-correction methods for the resample-smoothing Voronoi estimator we will not use edge-correction for the kernel intensity estimator. We should note that using edge-correction would improve the results we will see in this section.

We now move on to present the results of using point process learning to select a bandwidth for the kernel intensity estimator.

4.4.1 Poisson process

Comparing Table 11 to Table 1 we see that the kernel estimator achieves much worse results than the Voronoi estimator. Although both estimators achieve similar IV, the kernel estimator has much worse performance in terms of bias. In Table 12 we can see the average bandwidth chosen with respect to k. In Figure 12 we can see plots of the intensity estimate. As we can see the kernel performs well but suffers due to edge effects. We note that edge effects for the kernel estimator result in the intensity estimate around the boundary being lower which is the opposite of what we saw for the resample-smoothing Voronoi estimator.

Moradi et al. [2018] found that using the Cronie-van Lieshout bandwidth selection method with so-called uniform edge-correction resulted in a MISE of ca. 690.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	161.95	162.72	161.95
5	232.92	225.52	232.58
10	241.12	224.86	253.02
all	250.69	234.23	253.02

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	348.11	352.76	348.11
5	270.71	278.80	270.12
10	262.60	274.87	251.65
all	255.50	267.68	251.65

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	15.68	15.88	15.68
5	12.86	13.19	12.83
10	12.55	13.01	12.14
all	12.27	12.74	12.14

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	510.06	515.48	510.06
5	503.63	504.32	502.70
10	503.73	499.74	504.68
all	506.19	501.91	504.68

Table 11: Evaluation metrics for the kernel estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation.

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.183	0.186	0.183
5	0.149	0.154	0.149
10	0.145	0.151	0.140
all	0.142	0.148	0.140

Table 12: Average bandwidth chosen for the homogeneous Poisson process by k-fold cross-validation.

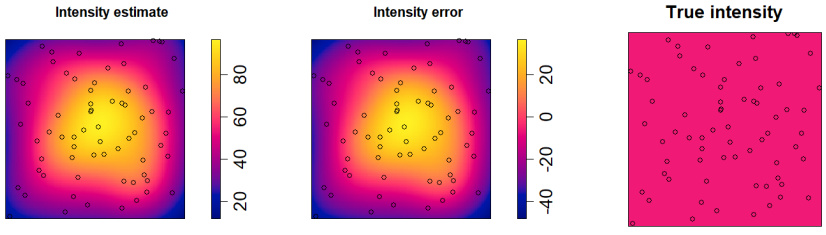


Figure 12: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the kernel method on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.4.2 Inhomogneous Poisson process

Comparing Table 3 to Table 13 we see similar results as in the previous section, namely that the kernel estimator achieves similar values of IV but larger ISB

resulting in a larger MISE. Looking at Figure 13 we can see plots of the intensity estimate. As we can see the average intensity looks very homogeneous and it fails to capture the periodicity of the true intensity. Furthermore, we again see some edge effects affecting the performance of this estimator.

Moradi et al. [2018] found that using the Cronie-van Lieshout bandwidth selection method with uniform edge-correction resulted in a MISE of ca. 1430.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	188.40	182.58	188.40
5	261.14	251.77	266.82
10	278.89	268.55	296.72
all	287.93	275.08	297.96

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1180.08	1191.59	1180.08
5	1076.65	1091.64	1069.92
10	1054.81	1076.36	1042.55
all	1045.25	1063.85	1040.63

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	29.30	29.44	29.30
5	28.09	28.28	28.01
10	27.84	28.11	27.69
all	27.72	27.95	27.67

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1368.48	1374.17	1368.48
5	1337.79	1343.41	1336.74
10	1333.70	1344.90	1339.27
all	1333.17	1338.93	1338.59

Table 13: Evaluation metrics for the kernel estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation.

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.182	0.187	0.182
5	0.146	0.152	0.144
10	0.140	0.149	0.135
all	0.136	0.144	0.135

Table 14: Average bandwidth chosen for the inhomogeneous Poisson process by k-fold cross-validation.

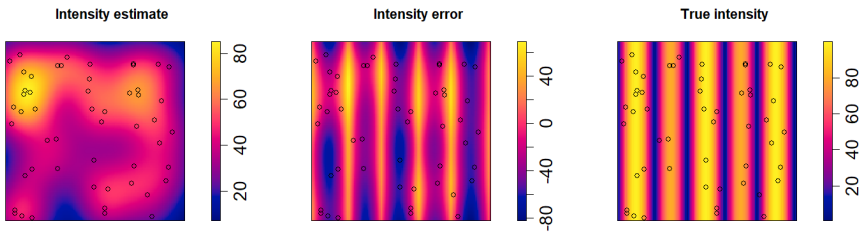


Figure 13: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.4.3 Log-Gaussian Cox process

Comparing Table 15 to Table 5 we see that the kernel estimator offers a very small improvement compared to the Voronoi estimator. We do note that the best result achieved by the Voronoi estimator is the same as for the kernel estimator and that results do not vary much with respect to the loss function. Furthermore, we note that the kernel estimator is worse in terms of ISB and better in terms of IV. Looking at Table 16 we see that the average bandwidth appears to be negatively correlated with k. This is similar to the behavior in Table 6a. Looking at Figure 14 we see the average intensity estimate looks fairly homogeneous and fails to capture the periodicity of the true intensity. We again note a fair degree of edge effects.

Moradi et al. [2018] found that using the Cronie-van Lieshout bandwidth selection method with so-called uniform edge-correction resulted in a MISE of ca. 10980.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1361.43	1339.32	1361.43
5	1682.43	1547.75	1747.49
10	1605.52	1432.21	1799.11
all	1766.48	1549.63	1802.60

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1624.61	1633.10	1624.61
5	1549.01	1589.08	1539.21
10	1524.34	1560.90	1490.33
all	1512.16	1541.86	1488.01

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	35.18	35.26	35.18
5	34.39	34.81	34.29
10	34.17	34.57	33.81
all	34.02	34.36	33.78

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2986.04	2972.42	2986.04
5	3231.44	3136.83	3286.70
10	3129.86	2993.11	3289.44
all	3278.64	3091.49	3290.62

Table 15: Evaluation metrics for the kernel estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.186	0.189	0.186
5	0.165	0.177	0.162
10	0.159	0.169	0.147
all	0.154	0.162	0.146

Table 16: Average bandwidth chosen for the Log-Gaussian Cox process by k-fold cross-validation.

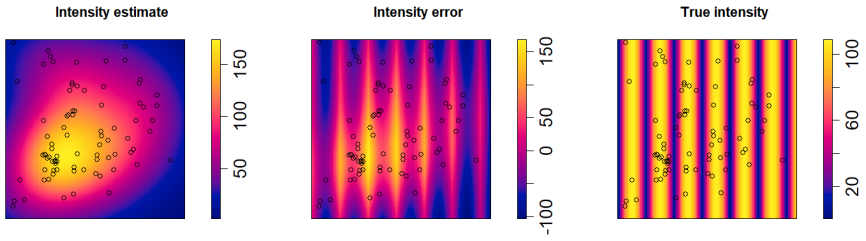


Figure 14: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.4.4 Simple sequential inhibition process

Comparing Table 17 to Table 9 we see that the kernel estimator actually performs quite a lot better than the Voronoi estimator. We note that IV is similar

for both estimators but the ISB is much lower for the kernel estimator. It is important to point out that this result was previously found in Moradi et al. [2018]. Looking at Figure 15 we can see a plot of the average intensity estimate. As we can see the kernel estimator captures the two high intensity regions nicely which the resample-smoothing Voronoi estimator did not.

Moradi et al. [2018] found that using the Cronie-van Lieshout bandwidth selection method with so-called uniform edge-correction resulted in a MISE of ca. 1170.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	173.93	171.19	173.93
5	223.88	216.54	233.59
10	203.75	202.30	242.66
all	226.93	221.39	244.36

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1043.24	1056.04	1043.24
5	899.74	925.14	859.34
10	913.16	931.45	829.50
all	873.47	893.21	826.58

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	24.62	24.79	24.62
5	22.66	23.02	22.05
10	22.83	23.10	21.61
all	22.27	22.56	21.56

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1217.17	1227.22	1217.17
5	1123.62	1141.67	1092.93
10	1116.91	1133.75	1072.17
all	1100.40	1114.60	1070.95

Table 17: Evaluation metrics for the kernel estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.155	0.157	0.155
5	0.135	0.138	0.130
10	0.136	0.139	0.127
all	0.131	0.134	0.126

Table 18: Average bandwidth chosen for the simple sequential inhibition process by k-fold cross-validation.

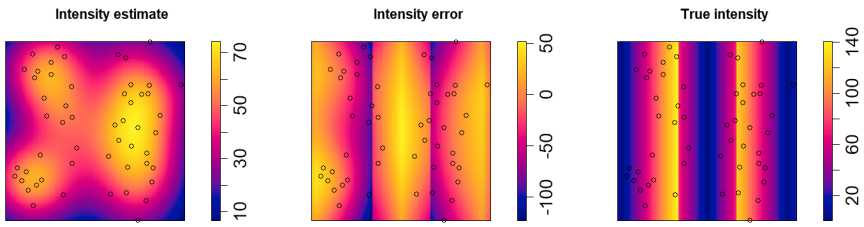


Figure 15: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.5 Anisotropic kernel

We recall that kernel intensity estimators are of the form

$$\hat{\rho}_h(u) = \sum_{x \in \mathbf{x}} \frac{k_h(u - x)}{w_h(u, x)}$$

where $h > 0$ is some bandwidth. This is a special case where h is a matrix. This matrix is the equivalent of the covariance matrix in a multivariate Gaussian distribution. In this case, we construct bandwidth matrices of the form

$$H = h \begin{bmatrix} w_x & w_{xy} \\ w_{xy} & w_y \end{bmatrix}$$

with $w_x, w_y \in \{1, 2, 4\}$ and $w_{xy} \in \{-0.8, -0.2, 0, 0.2, 0.8\}$ and h chosen as in Section 4.4. As we have seen the intensities of all models, except the homogeneous Poisson, are periodic with respect to x and remain constant with respect to y. Our hypothesis was that using a matrix that resulted in a kernel that was stretched out along the y-axis would result in intensity estimates that performed well. In Figure 16 we can see examples of Gaussian kernels with different bandwidth matrices. Top left we have the case where $w_x = w_y = 1, w_{xy} = 0$, top right we have $w_x = 1, w_y = 4, w_{xy} = 0$, bottom left $w_x = 4, w_y = 1, w_{xy} = 0$ and bottom right $w_x = w_y = 1, w_{xy} = 0.8$.

As we will see, the performance of anisotropic kernel intensity estimation performed worse than the isotropic kernel. This is quite interesting as one of the choices of matrices investigated was the same as the isotropic. As the anisotropic variant is more flexible it can also overfit to a higher degree and therefore have lower bias but higher variance which is exactly what we will see in the results presented in this section. The hope for this idea was that the optimal bandwidth matrix would be elliptical, stretching out along the y-axis, as most of the investigated intensities are periodic with respect to x.

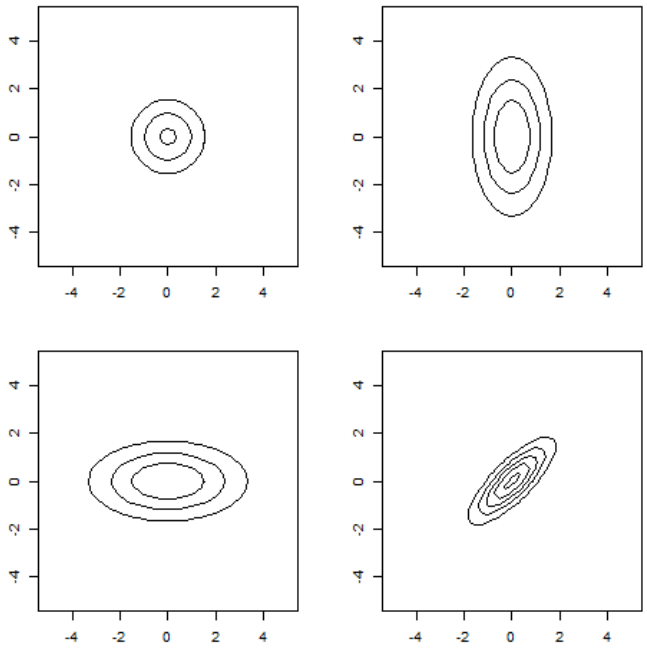


Figure 16: Examples of Gaussian kernels with different bandwidth matrices.

4.5.1 Log-Gaussian Cox process

Looking at Table 19 we see the results of point process learning for the anisotropic kernel estimator. Comparing these results to what we see in Table 15 we see that this estimator performs worse in terms of IV and slightly better in terms of ISB. As this method is more flexible than the isotropic kernel it can overfit to a higher degree which is what we see here. Previously, the best results in terms of MISE came from the resample-smoothing Voronoi estimator with $k = 2$ and using $\mathcal{L}(1)$ with values ca. 2970.

In Figure 17 we can see the intensity estimate which looks very homogeneous and does not capture the periodic characteristic of the true intensity. Furthermore, we can also see a fair degree of edge effects.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1782.48	1596.75	1782.48
5	2011.13	2003.35	2233.83
10	1802.68	1796.89	2160.40
all	2112.52	2074.98	2189.09

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1574.06	1587.11	1574.06
5	1505.23	1529.99	1483.73
10	1491.33	1503.60	1444.02
all	1472.69	1487.92	1440.01

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	34.68	34.81	34.68
5	33.96	34.21	33.72
10	33.85	33.97	33.31
all	33.63	33.78	33.28

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	3356.54	3183.86	3356.54
5	3516.37	3533.34	3717.56
10	3294.02	3300.50	3604.42
all	3585.21	3562.91	3629.10

Table 19: Evaluation metrics for the anisotropic kernel estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

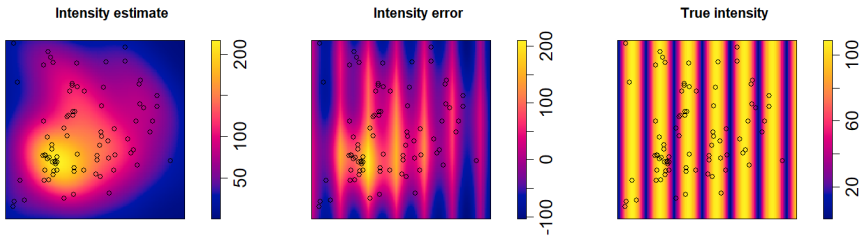


Figure 17: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the anisotropic kernel method on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.5.2 Simple sequential inhibition process

Comparing Table 20 to Table 17 we again see that the anisotropic kernel performs worse than the isotropic one. Interestingly enough, however, we do not see the same results as for the Log-Gaussian Cox process. Here we instead see a decrease in IV and an increase in ISB. Previously, the best results in terms of MISE came from using the isotropic kernel intensity estimator with values ca. 1100.

In Figure 18 we see the intensity estimate. We see that it manages to capture the fact that there are two areas with higher intensity. However, it fails to fully capture the low intensity area in the middle.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	96.34	97.85	96.34
5	100.37	105.73	103.60
10	102.34	100.84	99.57
all	100.07	100.39	99.57

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1287.29	1289.93	1287.29
5	1268.34	1266.42	1265.27
10	1266.51	1273.94	1267.68
all	1266.49	1268.11	1267.68

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	27.71	27.73	27.71
5	27.55	27.52	27.51
10	27.52	27.59	27.55
all	27.53	27.53	27.55

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1383.63	1387.78	1383.63
5	1368.71	1372.16	1368.87
10	1368.85	1374.78	1367.25
all	1366.56	1368.50	1367.25

Table 20: Evaluation metrics for the anisotropic kernel estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

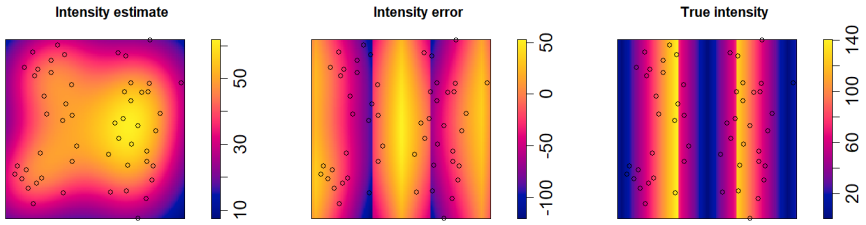


Figure 18: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the anisotropic kernel method on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.6 Localized tessellations

One of the flaws of the Voronoi estimator is that it can fail to capture localized features. For example, if we have an intensity function

$$\rho(x, y) = 0 \cdot \mathbf{1}\{x \geq 0.5 \wedge y \geq 0.5\} + 50 \cdot \mathbf{1}\{x < 0.5 \vee y < 0.5\}$$

of an inhomogeneous Poisson process, X, simulated in $W = [0, 1]^2$, the Voronoi estimator will perform quite poorly in $[0.5, 1]^2$ due to the fact that this sub-window will still belong to tessellations centered on points elsewhere. The kernel estimator, on the other hand, would not as it is the sum of symmetric kernel functions centered on the points of a point pattern. If there are no points in a

sub-window and a sufficiently small enough bandwidth is chosen, the estimated intensity will be approximately 0 there.

What if instead of creating a tessellation of a point pattern $\mathbf{x}$, which is then used to estimate the intensity via

$$\hat{\rho}^{\mathcal{V}}(u; X, W) = \sum_{x \in \mathbf{x}} \frac{\mathbf{1}\{u \in \mathcal{V}_x(\mathbf{x}, W)\}}{|\mathcal{V}_x(\mathbf{x}, W)|},$$

we split W into a number of disjoint sub-windows, $W = \bigcup_{i=1}^{n} W_i$, giving us

$$\hat{\rho}^{\mathcal{V}}(u; \mathbf{x}, \mathbf{W}) = \sum_{i=1}^{n} \sum_{x \in \mathbf{x}} \frac{\mathbf{1}\{u \in \mathcal{V}_x(\mathbf{x} \cap W_i, W_i)\}}{|\mathcal{V}_x(\mathbf{x} \cap W_i, W_i)|}.$$

To investigate this we will create sub-windows by splitting W into quadrats. If we have a square window $W = [a, b] \times [c, d]$ we partition both intervals, $[a, b]$ and $[c, d]$, into n_q subintervals, $a = a_0 < a_1, < \ldots a_{n_q} = b$, $c = c_0 < c_1, < \ldots c_{n_q} = d$. We then let $W_{i,j} = [a_{i-1}, a_i] \times [c_{j-1}, c_j]$. For simplicity's sake we denote a sub-window W_i and let $W = \bigcup_{i=1}^{n_q^2} W_i$.

Due to the increased computational complexity, we again limit ourselves to only investigating the Log-Gaussian Cox and simple sequential inhibition processes.

As we will see, this method failed quite badly. As we have seen in almost all plots of the average intensity estimates, almost no intensity estimation method has managed to capture the periodic nature of either the inhomogeneous Poisson or the Log-Gaussian Cox process. The hope for localized tessellations was that it would then be able to capture the intensity's fluctuation on a smaller scale. However, as we saw the edge effects simply outweighed any benefit they might have had. Perhaps using periodic or some other form of edge-correction could have improved results but overall this method showed no potential.

4.6.1 Log-Gaussian Cox process

In Table 21 we see the results for the Log-Gaussian Cox process. We see that IV, ISB, and MISE all increase with respect to n_q and that this method offers no improvement. Previously, the best results in terms of MISE came from the resample-smoothing Voronoi estimator with values ca. 2970.

Looking at Figure 19 we can see a large degree of edge effects which affects the performance of this method negatively. We note that the edge effects are exacerbated by the fact that each quadrat is affected by its own boundary.

(a) IV

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	5589.2	5379.6	5552.0
2	5	4603.8	4574.3	3851.0
2	10	3541.3	3254.9	4154.5
2	all	4722.8	4637.3	4192.5
4	2	5393.0	5493.9	5372.3
4	5	4179.4	3368.6	4154.2
4	10	3667.3	3009.8	3889.0
4	all	4020.4	3423.6	3964.9
6	2	6946.9	6955.9	6762.7
6	5	5182.4	5366.9	5119.4
6	10	4691.3	4468.9	4618.9
6	all	5019.1	5249.2	4860.9

(b) ISB

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	1177.5	1183.9	1174.6
2	5	1175.3	1180.1	1182.9
2	10	1176.3	1183.7	1176.7
2	all	1178.9	1180.0	1175.2
4	2	1200.7	1199.7	1191.4
4	5	1389.2	1446.4	1387.7
4	10	1453.5	1546.5	1465.8
4	all	1408.2	1409.8	1416.5
6	2	1290.7	1287.8	1302.4
6	5	1578.9	1545.8	1595.6
6	10	1660.6	1699.8	1659.8
6	all	1601.2	1553.0	1604.5

(c) IAB

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	29.8	29.9	29.8
2	5	29.9	30.0	30.0
2	10	29.9	30.1	29.9
2	all	29.9	29.9	29.8
4	2	30.4	30.3	30.2
4	5	32.6	33.3	32.6
4	10	33.4	34.3	33.5
4	all	32.9	33.0	33.0
6	2	31.5	31.5	31.7
6	5	34.7	34.4	34.9
6	10	35.5	35.9	35.5
6	all	35.0	34.5	35.0

(d) MISE

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	6766.7	6563.5	6726.6
2	5	5779.1	5754.4	5033.9
2	10	4717.7	4438.6	5331.2
2	all	5901.7	5817.3	5367.7
4	2	6593.7	6693.5	6563.7
4	5	5568.5	4815.0	5541.9
4	10	5120.8	4556.3	5354.8
4	all	5428.6	4833.4	5381.4
6	2	8237.6	8243.7	8065.1
6	5	6761.3	6912.7	6715.0
6	10	6351.9	6168.7	6278.7
6	all	6620.3	6802.2	6465.4

Table 21: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

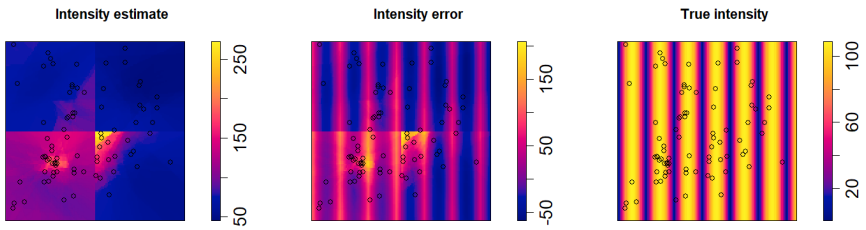


Figure 19: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method on sub-windows on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.6.2 Simple sequential inhibition process

In Table 22 we see the results of the localized Voronoi estimator. For $n_q = 4$ we see some improvement in terms of ISB but this is accompanied by a large increase in variance resulting in a larger MISE. For $n_q = 6$ we see a larger improvement in ISB which is also accompanied by an increase in IV. However, the resulting MISE is still lower than for $n_q = 4$ which might indicate that even larger values of n_q might display better results. For all n_q we see no improvement in terms of MISE when comparing Table 22 to Table 9. Previously, the best results in terms of MISE came from using the isotropic kernel intensity estimator with values ca. 1100.

As we can see in Figure 20 the edge effects are exacerbated by this method which negatively affects the results.

(a) IV

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	281.7	279.5	279.9
2	5	271.2	274.2	260.2
2	10	269.6	255.4	288.3
2	all	264.6	270.5	293.5
4	2	673.5	665.0	674.1
4	5	559.3	537.3	573.2
4	10	536.6	494.5	531.2
4	all	586.4	579.9	548.5
6	2	1427.3	1420.4	1416.0
6	5	1021.5	1021.4	1010.2
6	10	916.6	913.2	922.2
6	all	933.8	977.4	994.1

(b) ISB

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	1423.9	1430.0	1425.3
2	5	1427.8	1427.2	1446.8
2	10	1440.9	1452.0	1408.7
2	all	1423.7	1438.5	1409.4
4	2	1291.0	1291.4	1287.0
4	5	1389.9	1430.8	1389.8
4	10	1425.1	1473.4	1428.9
4	all	1389.9	1397.4	1398.2
6	2	463.0	456.2	458.0
6	5	660.7	680.9	679.4
6	10	790.0	759.1	760.0
6	all	740.5	698.1	722.6

(c) IAB

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	31.8	31.9	31.8
2	5	31.8	31.8	32.1
2	10	32.0	32.1	31.6
2	all	31.8	32.0	31.6
4	2	30.3	30.3	30.2
4	5	30.7	30.9	30.7
4	10	30.8	31.1	30.9
4	all	30.7	30.7	30.7
6	2	18.2	18.1	18.1
6	5	20.7	21.0	21.0
6	10	22.4	22.0	22.1
6	all	21.8	21.2	21.6

(d) MISE

n_q	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2	1705.6	1709.5	1705.2
2	5	1699.0	1701.4	1707.0
2	10	1710.5	1707.5	1697.0
2	all	1688.4	1708.9	1702.9
4	2	1964.5	1956.3	1961.1
4	5	1949.2	1968.1	1963.0
4	10	1961.8	1967.9	1960.1
4	all	1976.3	1977.3	1946.6
6	2	1890.3	1876.6	1873.9
6	5	1682.2	1702.2	1689.6
6	10	1706.6	1672.3	1682.3
6	all	1674.4	1675.4	1716.7

Table 22: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

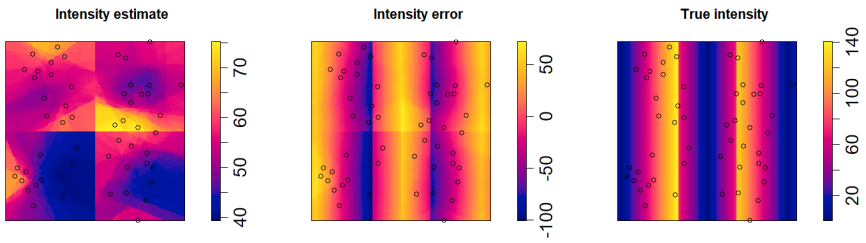


Figure 20: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method on sub-windows on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.7 Parameter re-scaling

As previously mentioned, the goal of point process learning for the Voronoi estimator is to find an optimal probability p_v with which to create thinned point patterns in the resample-smoothing. These thinned patterns are then used in (17) to create an intensity estimate. This p_v will then be used on the full point pattern to construct a new intensity estimate. When finding the optimal p_v the Voronoi estimator uses the training set

$$\hat{\rho}^{\mathcal{V}}_{p_v,m}(u; \mathbf{x}_i^T, W) = \frac{1}{mp_v} \sum_{j=1}^{m} \sum_{x \in \mathbf{x}_i^{T,p_v}} \frac{\mathbf{1}\{u \in \mathcal{V}_x(\mathbf{x}_i^{T,p_v})\}}{|\mathcal{V}_x(\mathbf{x}_i^{T,p_v})|}$$

where $\mathbf{x}_i^{T,p_v}$ is the i^{th} training fold of the point pattern $\mathbf{x}$ that has been thinned with retention probability p_v. If $\mathbf{x}$ contains n points and is split into k folds, the ith training fold, $\mathbf{x}_i^T$, will on average contain $n - \frac{n}{k}$ points. Given an optimal p_v the expected number of points used to create each Voronoi tesselation will be $p_v(n - \frac{n}{k})$ during training. When the intensity is finally estimated on the full point pattern, $\mathbf{x}$, the expected number of points used to create each Voronoi tesselation will instead be $p_v n$ points. For small values of k, this difference can be quite large and might be responsible for the poor performance we saw in Section 4.3. Because of this, we will investigate what happens when p_v is re-scaled to $p_v - \frac{p_v}{k}$ to see if the results are affected.

In the tables that will be presented, there will be an entry with $k =$ all. This is the result of treating the parameter k as a parameter rather than a hyperparameter. In this case, we choose k, as well as m and p_v, as the minimizer of the loss function as described in Section 3.3. This k is then used to rescale p_v in the same manner as described previously.

4.7.1 Poisson process

In Table 23 we see the results of using the re-scaled p_v parameter on the homogeneous Poisson process. Comparing this to using the original p_v parameter in

Table 1 we see varying degrees of improvements in terms of MISE. We see that the largest improvement happens for $k = 2$ and that for $k > 2$ the results are diminishing. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with values ca. 230.

Looking at Table 23 we see that ISB values now seem fairly constant for all k. Comparing this to Table 1 we see that $k = 2$ appears to have a slightly lower ISB.

In Table 23 we see that the variance for all values of k seems to be improved, with the largest improvement for $k = 2$.

Looking at Figure 21 we see that the intensity estimate looks quite homogeneous even more so than what we saw in Figure 8.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	142.22	150.79	154.29
5	160.91	155.22	153.56
10	189.47	172.67	158.29
all	182.58	173.05	157.00

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	46.62	46.28	47.43
5	46.60	46.49	47.83
10	45.28	47.65	48.46
all	44.91	46.50	48.04

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	5.89	5.89	5.93
5	5.88	5.87	5.95
10	5.70	5.89	5.96
all	5.71	5.80	5.95

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	188.85	197.07	201.72
5	207.52	201.71	201.39
10	234.76	220.31	206.76
all	227.48	219.55	205.04

Table 23: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation with re-scaled p_v.

4.7.2 Inhomogeneous Poisson process

Looking at Table 24 we see the results of re-scaling the p_v parameter for the inhomogeneous Poisson process. If we compare this to Table 3 we see similar improvements as when re-scaling the homogeneous Poisson process. In terms of ISB the results here seem fairly constant which we saw in Section 4.3.2 as well. In terms of IV, we see improvements for all k with the largest improvement for $k = 2$. Comparing the MISE values in Table 24 to Table 3 we see some improvements with the most notable decrease for $k = 2$. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with values ca. 1070.

Comparing Figure 22 to Figure 9 we can see that the intensity estimates look quite similar.

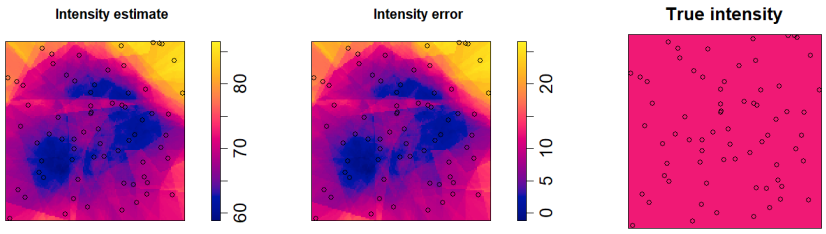


Figure 21: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the Voronoi method with re-scaled p_v parameter on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 5$.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	156.46	155.71	155.56
5	169.59	174.35	168.45
10	192.44	164.63	166.42
all	174.92	167.67	167.82

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	890.12	892.58	891.12
5	890.49	887.92	889.81
10	887.14	892.85	891.76
all	890.95	890.88	891.17

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	25.61	25.62	25.63
5	25.55	25.53	25.55
10	25.49	25.55	25.57
all	25.56	25.56	25.52

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1046.58	1048.28	1046.68
5	1060.09	1062.26	1058.26
10	1079.59	1057.48	1058.18
all	1065.87	1058.55	1058.98

Table 24: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation with re-scaled p_v.

4.7.3 Log-Gaussian Cox process

In Table 25 we see the results using the re-scaled p_v parameter on the Log-Gaussian Cox process. Comparing this to Table 5, we see improvements in MISE for all values of k, with $k = 10$ again achieving the best results. We note that $\mathcal{L}_3$ achieves much worse results than the other loss functions. Comparing ISB values we see that they seem very similar for both methods. For IV values we see that the re-scaled p_v does offer some improvement for all k with the biggest improvement for $k = 2$. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with values ca. 2970.

Looking at Figure 23 we again see that it looks very similar to Figure 10.

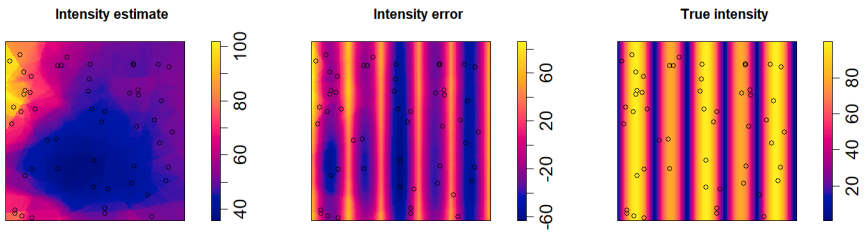


Figure 22: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with re-scaled p_v parameter on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 5$.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2100.57	2191.78	2083.30
5	2188.47	2329.71	2420.88
10	1664.24	1715.95	2953.24
all	1913.51	2122.00	2830.75

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1209.39	1204.76	1209.77
5	1215.00	1210.84	1205.59
10	1225.20	1219.42	1197.95
all	1219.61	1216.59	1199.42

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	30.39	30.35	30.41
5	30.46	30.39	30.30
10	30.64	30.58	30.25
all	30.54	30.47	30.24

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	3309.96	3396.54	3293.07
5	3403.46	3540.54	3626.47
10	2889.45	2935.38	4151.18
all	3133.12	3338.59	4030.17

Table 25: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation with re-scaled p_v.

4.7.4 Simple sequential inhibition process

In Table 26 we see the results of using the re-scaled p_v parameter on the simple sequential inhibition process. If we compare this to Table 9 we see that this method offers no real improvement in terms of MISE. We do note improvements in terms of IV but those are canceled out by increases in ISB. Previously, the best results in terms of MISE came from using the kernel estimator with values ca. 1100.

Looking at Figure 24 we see that the intensity estimate looks very similar to Figure 11.

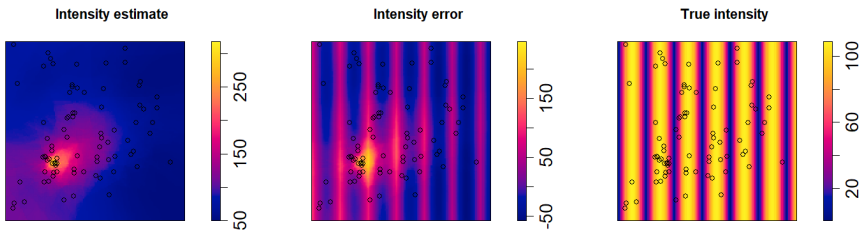


Figure 23: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with re-scaled p_v parameter on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 5$.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	92.01	92.86	92.40
5	173.01	139.26	145.05
10	119.24	100.24	145.01
all	155.56	140.62	143.82

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1486.79	1489.48	1487.05
5	1379.14	1419.35	1408.76
10	1444.61	1462.14	1411.57
all	1394.36	1412.58	1411.07

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	32.28	32.29	32.26
5	31.06	31.49	31.43
10	31.80	31.99	31.47
all	31.22	31.41	31.48

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1578.81	1582.34	1579.45
5	1552.15	1558.60	1553.81
10	1563.86	1562.38	1556.58
all	1549.92	1553.20	1554.89

Table 26: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation with re-scaled p_v.

4.8 Regularization

Regularization is the process of adding some sort of regularization term to a loss function with the goal of obtaining a simpler solution. In our case, since Moradi et al. [2018] showed that lower values of p_v in the Voronoi estimator result in better estimates, it seems natural to add a regularization term in the form of a function of p_v. Previously we chose an optimal $\hat{\theta}$ via

$$\hat{\theta} = \arg\min\{\mathcal{L}(\theta)\},\ \theta = (m, p),$$

which becomes

$$\hat{\theta} = \arg\min\{\mathcal{L}(\theta) + \lambda_r R(\theta)\}$$

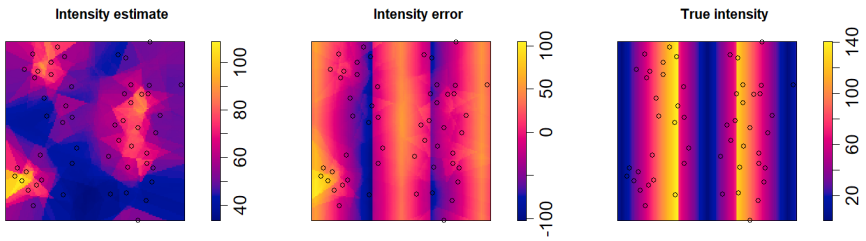


Figure 24: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with re-scaled p_v parameter on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 5$.

where $R(\theta)$ is the regularization term, which is commonly the $\mathcal{L}_1$ or $\mathcal{L}_2$ norm, and $\lambda_r > 0$ is the regularization factor or strength. Note that $\lambda_r = 0$ is simply the same as no regularization, therefore, we restrict $\lambda_r > 0$.

In this Section, we will present the results of choosing optimal $\hat{\theta}$ via point process learning with regularization. We investigate $\lambda_r = 0.5, 1, 5, 10$ and $R_1(\theta) = |p|$ and $R_2(\theta) = p^2$. Note that we restrict our investigation to regularization terms as functions of p_v and not m. There is nothing forcing us to do this and regularizing m could be of interest if we are limited in computational power and would prefer a smaller number of re-samples.

Obviously, this has never been done before and so there is no literature to reference in the choice of λ_r or $R(\theta)$ but $\lambda_r = 0.5, 1, 5, 10$ seem like reasonable choices, and are commonly used in machine learning. Picking λ_r too small means that the regularization term will not have any impact on the choice of $\hat{\theta}$, and picking a λ_r that is too large means that it will eclipse the value of $\mathcal{L}(\theta)$. As such, these choices seem like reasonable choices to investigate.

4.8.1 Poisson process

In Tables 27 and 28 we can see the evaluation metrics of choosing θ with different regularization terms and strengths. We see that MISE seems to be decreasing with respect to λ_r. Looking at IV and ISB we see that IV is decreasing and ISB increasing with respect to λ_r but that the decrease in IV is larger than the increase in ISB. This suggests that regularization decreases the amount of overfitting. We also note that in Table 28 we see a smaller scale of improvement which is to be expected since the regularization term here is p^2. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with re-scaled p_v parameter with values ca. 190.

Comparing this to Table 1 we see improvements for both R, all λ_r, and all k.

Looking at Tables 65 and 66 we see that p_v decreases with respect to λ_r and that $R_1(\theta) = |p|$ decreases faster.

Looking at Figure 25 and comparing it to Figure 8 we see that degree of edge effects has increased by applying regularization. We also note that the intensity estimate looks quite homogeneous.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	208.5	223.9	218.7
0.5	5	160.2	157.9	150.0
0.5	10	144.1	153.6	130.8
0.5	all	148.4	150.1	136.5
1.0	2	190.1	207.7	207.0
1.0	5	147.5	144.3	143.6
1.0	10	146.4	132.9	123.8
1.0	all	141.5	136.8	121.9
5.0	2	152.9	177.6	171.2
5.0	5	123.9	127.3	118.0
5.0	10	103.3	116.6	111.6
5.0	all	113.9	111.1	104.6
10.0	2	147.3	158.6	153.7
10.0	5	109.2	119.5	109.3
10.0	10	103.8	110.0	89.8
10.0	all	113.4	113.3	99.5

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	42.3	40.9	39.6
0.5	5	50.8	50.1	51.4
0.5	10	51.3	51.6	49.6
0.5	all	51.3	50.5	50.6
1.0	2	46.6	43.4	43.5
1.0	5	51.3	50.8	51.3
1.0	10	50.9	50.1	51.3
1.0	all	50.9	50.6	50.9
5.0	2	51.7	46.6	50.6
5.0	5	51.8	51.1	50.4
5.0	10	55.7	52.1	57.0
5.0	all	55.7	52.0	57.4
10.0	2	52.0	50.1	49.6
10.0	5	58.0	52.5	57.1
10.0	10	88.9	62.2	88.6
10.0	all	86.6	61.4	83.8

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	5.5	5.3	5.3
0.5	5	6.1	6.1	6.1
0.5	10	6.1	6.2	6.1
0.5	all	6.1	6.1	6.1
1.0	2	5.8	5.6	5.6
1.0	5	6.2	6.1	6.2
1.0	10	6.1	6.1	6.1
1.0	all	6.1	6.1	6.1
5.0	2	6.2	5.8	6.1
5.0	5	6.0	6.1	5.9
5.0	10	6.0	6.0	6.1
5.0	all	6.1	6.0	6.1
10.0	2	6.2	6.1	6.1
10.0	5	6.1	6.0	6.1
10.0	10	7.7	6.3	7.7
10.0	all	7.6	6.3	7.4

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	250.9	264.8	258.3
0.5	5	211.0	208.0	201.5
0.5	10	195.3	205.3	180.4
0.5	all	199.7	200.6	187.1
1.0	2	236.7	251.1	250.6
1.0	5	198.8	195.0	195.0
1.0	10	197.3	183.0	175.1
1.0	all	192.3	187.4	172.8
5.0	2	204.6	224.2	221.9
5.0	5	175.7	178.4	168.4
5.0	10	159.0	168.7	168.6
5.0	all	169.7	163.1	162.0
10.0	2	199.3	208.8	203.3
10.0	5	167.2	172.0	166.4
10.0	10	192.7	172.2	178.5
10.0	all	200.0	174.7	183.3

Table 27: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation with regularization $|p|$.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	224.1	237.6	238.6
0.5	5	171.0	167.9	166.6
0.5	10	171.0	161.3	159.1
0.5	all	165.5	162.9	155.0
1.0	2	210.3	229.3	230.3
1.0	5	165.5	165.0	161.0
1.0	10	156.4	155.7	150.2
1.0	all	162.5	153.5	147.4
5.0	2	187.0	197.0	201.2
5.0	5	156.6	151.1	147.8
5.0	10	131.8	152.7	132.5
5.0	all	139.5	141.8	127.8
10.0	2	169.3	189.9	190.8
10.0	5	143.4	147.2	142.8
10.0	10	134.8	136.7	121.2
10.0	all	138.8	136.2	121.5

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	39.5	39.4	40.7
0.5	5	47.8	47.8	50.3
0.5	10	49.0	50.4	50.7
0.5	all	48.7	50.0	49.7
1.0	2	41.2	40.8	40.7
1.0	5	50.9	48.2	50.4
1.0	10	50.2	51.0	50.7
1.0	all	48.4	50.2	50.5
5.0	2	46.2	44.0	45.8
5.0	5	51.5	50.6	50.2
5.0	10	50.7	50.8	50.2
5.0	all	50.3	50.9	50.6
10.0	2	50.3	45.5	47.9
10.0	5	50.1	52.3	51.4
10.0	10	51.1	50.4	51.0
10.0	all	50.6	49.9	50.4

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	5.2	5.2	5.3
0.5	5	5.9	5.9	6.1
0.5	10	6.0	6.1	6.1
0.5	all	6.0	6.1	6.0
1.0	2	5.4	5.3	5.3
1.0	5	6.1	6.0	6.1
1.0	10	6.1	6.1	6.1
1.0	all	6.0	6.1	6.1
5.0	2	5.8	5.6	5.7
5.0	5	6.2	6.1	6.1
5.0	10	6.1	6.1	6.1
5.0	all	6.1	6.1	6.1
10.0	2	6.1	5.8	5.8
10.0	5	6.1	6.2	6.2
10.0	10	6.1	6.1	6.1
10.0	all	6.1	6.1	6.0

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	263.6	277.0	279.3
0.5	5	218.8	215.8	216.9
0.5	10	220.0	211.7	209.8
0.5	all	214.1	212.9	204.7
1.0	2	251.5	270.1	271.0
1.0	5	216.3	213.2	211.4
1.0	10	206.6	206.7	200.9
1.0	all	210.9	203.8	197.8
5.0	2	233.2	241.0	247.0
5.0	5	208.1	201.7	197.9
5.0	10	182.5	203.5	182.7
5.0	all	189.8	192.6	178.5
10.0	2	219.7	235.4	238.7
10.0	5	193.6	199.5	194.2
10.0	10	185.9	187.1	172.1
10.0	all	189.4	186.1	171.8

Table 28: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation with regularization p^2.

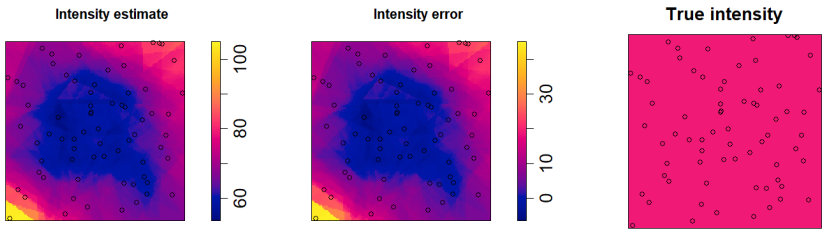


Figure 25: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the Voronoi method with regularization strength $\lambda_r = 1$ and factor $R(\theta) = |p_v|$ on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.8.2 Inhomogeneous Poisson process

Looking at Tables 29 and 30 we see the results for the inhomogeneous Poisson process. We see similar results here as we did in the previous Section, a decrease in IV and an increase in ISB with respect to λ_r resulting in an overall decrease in MISE. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with re-scaled p_v parameter with values ca. 1050.

Looking at Figure **??** and comparing it to Figure 9 we again see an increased amount of edge effects and that the intensity estimate looks more homogeneous.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	214.0	234.2	223.3
0.5	5	161.0	162.6	158.3
0.5	10	145.1	148.6	139.1
0.5	all	157.7	151.7	144.3
1.0	2	200.1	220.5	221.6
1.0	5	160.7	153.0	144.4
1.0	10	136.8	138.3	128.6
1.0	all	141.8	141.5	134.7
5.0	2	154.3	179.3	171.0
5.0	5	124.5	123.1	114.5
5.0	10	106.8	111.7	99.2
5.0	all	124.6	118.3	102.4
10.0	2	138.0	159.8	164.4
10.0	5	113.5	119.1	102.9
10.0	10	104.5	104.1	96.5
10.0	all	104.5	110.2	96.2

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	885.2	886.4	885.5
0.5	5	894.0	893.0	894.4
0.5	10	893.1	892.7	893.5
0.5	all	893.9	893.6	895.0
1.0	2	888.7	886.0	886.8
1.0	5	894.3	892.5	893.7
1.0	10	893.8	893.1	894.1
1.0	all	893.4	892.1	895.5
5.0	2	893.1	891.7	892.8
5.0	5	895.8	895.2	896.9
5.0	10	904.0	897.9	908.3
5.0	all	897.6	897.0	909.3
10.0	2	895.6	893.0	894.9
10.0	5	911.6	900.4	907.3
10.0	10	932.8	912.4	926.9
10.0	all	931.4	910.8	929.6

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	25.5	25.5	25.5
0.5	5	25.6	25.6	25.6
0.5	10	25.6	25.6	25.6
0.5	all	25.6	25.6	25.7
1.0	2	25.5	25.5	25.5
1.0	5	25.6	25.6	25.6
1.0	10	25.6	25.6	25.7
1.0	all	25.6	25.6	25.7
5.0	2	25.6	25.5	25.5
5.0	5	25.9	25.7	25.9
5.0	10	26.1	25.9	26.2
5.0	all	25.9	25.9	26.2
10.0	2	25.7	25.6	25.5
10.0	5	26.3	26.0	26.2
10.0	10	26.7	26.3	26.6
10.0	all	26.6	26.2	26.6

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1099.2	1120.6	1108.8
0.5	5	1055.1	1055.6	1052.7
0.5	10	1038.2	1041.4	1032.6
0.5	all	1051.6	1045.3	1039.3
1.0	2	1088.8	1106.5	1108.4
1.0	5	1055.0	1045.5	1038.1
1.0	10	1030.6	1031.4	1022.7
1.0	all	1035.2	1033.6	1030.2
5.0	2	1047.4	1071.0	1063.8
5.0	5	1020.3	1018.3	1011.4
5.0	10	1010.8	1009.5	1007.5
5.0	all	1022.2	1015.3	1011.8
10.0	2	1033.6	1052.8	1059.3
10.0	5	1025.1	1019.5	1010.2
10.0	10	1037.3	1016.5	1023.4
10.0	all	1035.9	1021.0	1025.8

Table 29: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation with regularization $|p|$.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	236.4	237.1	244.6
0.5	5	173.2	180.4	173.5
0.5	10	170.7	170.9	163.8
0.5	all	170.1	169.5	163.4
1.0	2	226.2	236.5	237.3
1.0	5	168.5	167.1	163.9
1.0	10	165.0	161.5	158.8
1.0	all	158.7	168.1	153.0
5.0	2	202.3	211.5	206.5
5.0	5	155.5	151.3	154.5
5.0	10	143.2	149.6	137.9
5.0	all	143.9	151.0	134.0
10.0	2	172.4	200.9	195.3
10.0	5	145.2	144.6	139.5
10.0	10	129.5	143.0	134.4
10.0	all	141.2	134.8	127.8

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	885.4	885.7	885.8
0.5	5	891.4	890.2	892.4
0.5	10	891.6	892.3	894.5
0.5	all	891.2	892.2	892.4
1.0	2	886.4	887.3	886.9
1.0	5	892.9	893.2	892.4
1.0	10	892.7	893.4	895.2
1.0	all	891.4	893.5	893.7
5.0	2	889.0	887.3	886.0
5.0	5	894.0	894.3	894.6
5.0	10	893.8	894.8	894.8
5.0	all	893.9	894.5	894.5
10.0	2	892.7	888.6	888.0
10.0	5	894.3	893.4	894.5
10.0	10	893.8	894.5	894.9
10.0	all	894.0	893.0	893.5

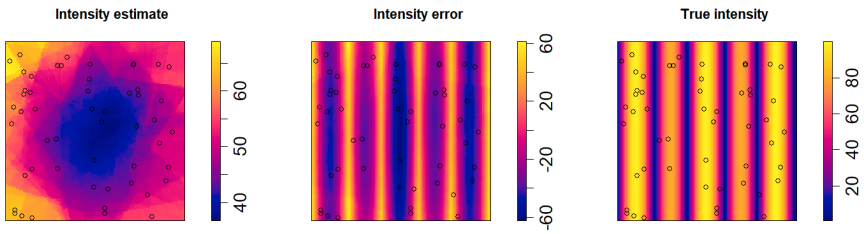
(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	25.4	25.5	25.5
0.5	5	25.5	25.6	25.5
0.5	10	25.5	25.5	25.6
0.5	all	25.5	25.5	25.5
1.0	2	25.4	25.5	25.5
1.0	5	25.6	25.6	25.5
1.0	10	25.6	25.6	25.6
1.0	all	25.5	25.6	25.6
5.0	2	25.5	25.5	25.5
5.0	5	25.6	25.6	25.6
5.0	10	25.7	25.6	25.7
5.0	all	25.6	25.6	25.6
10.0	2	25.5	25.5	25.5
10.0	5	25.6	25.6	25.6
10.0	10	25.7	25.6	25.7
10.0	all	25.6	25.7	25.7

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1121.8	1122.7	1130.4
0.5	5	1064.6	1070.6	1065.9
0.5	10	1062.2	1063.2	1058.3
0.5	all	1061.3	1061.7	1055.8
1.0	2	1112.6	1123.8	1124.2
1.0	5	1061.3	1060.3	1056.4
1.0	10	1057.7	1054.9	1053.9
1.0	all	1050.2	1061.6	1046.7
5.0	2	1091.3	1098.9	1092.5
5.0	5	1049.5	1045.6	1049.1
5.0	10	1037.0	1044.3	1032.7
5.0	all	1037.8	1045.6	1028.5
10.0	2	1065.1	1089.5	1083.3
10.0	5	1039.5	1038.0	1034.0
10.0	10	1023.2	1037.5	1029.2
10.0	all	1035.2	1027.8	1021.3

Table 30: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation with regularization p^2.



Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with regularization strength $\lambda_r = 1$ and factor $R(\theta) = |p_v|$ on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.8.3 Log-Gaussian Cox process

Looking at Tables 31 and 32 we see the results for the Log-Gaussian Cox process. The results here seem to follow the same trend that we saw for the two Poisson processes. We see that IV decreases with λ_r and that ISB increases. The increase in ISB however, is quite small compared to the decrease in IV leading to improvements in MISE. We again observe that the improvements when using R_2 are smaller than when using R_1. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with re-scaled p_v parameter with values ca. 2890.

Looking at Figure 26 we see the same trend as previously in this section, an increased amount of edge effects, and a more homogeneous average intensity estimate.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	3497.4	3755.7	3630.1
0.5	5	1903.0	1932.2	1669.3
0.5	10	1419.3	1503.1	1386.7
0.5	all	1655.1	1715.2	1390.3
1.0	2	2724.6	3659.4	3349.6
1.0	5	1634.1	1790.2	1473.3
1.0	10	1324.8	1414.7	1346.4
1.0	all	1551.1	1589.0	1331.1
5.0	2	1771.2	2707.1	2261.4
5.0	5	1311.5	1285.3	1164.6
5.0	10	1129.7	1113.1	1067.5
5.0	all	1260.9	1140.0	1080.9
10.0	2	1599.3	1869.0	1867.5
10.0	5	1136.1	1168.2	1113.8
10.0	10	1021.5	1069.7	978.9
10.0	all	1129.3	1159.0	1038.7

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1194.5	1183.4	1186.7
0.5	5	1216.5	1224.0	1224.3
0.5	10	1228.6	1227.4	1245.6
0.5	all	1227.5	1226.3	1246.3
1.0	2	1202.0	1184.4	1195.6
1.0	5	1222.5	1225.9	1234.9
1.0	10	1240.0	1236.1	1254.6
1.0	all	1234.0	1233.1	1259.3
5.0	2	1218.5	1211.3	1213.7
5.0	5	1256.9	1260.1	1271.0
5.0	10	1297.3	1286.9	1315.8
5.0	all	1274.0	1288.4	1314.2
10.0	2	1228.7	1218.5	1217.6
10.0	5	1294.2	1288.3	1297.2
10.0	10	1361.8	1320.0	1358.0
10.0	all	1326.8	1300.9	1340.1

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	30.1	29.9	30.0
0.5	5	30.5	30.5	30.6
0.5	10	30.7	30.7	30.9
0.5	all	30.6	30.6	31.0
1.0	2	30.2	30.0	30.1
1.0	5	30.6	30.6	30.8
1.0	10	30.9	30.8	31.1
1.0	all	30.7	30.7	31.1
5.0	2	30.4	30.3	30.3
5.0	5	31.1	31.2	31.3
5.0	10	31.7	31.5	31.9
5.0	all	31.3	31.5	31.9
10.0	2	30.6	30.5	30.4
10.0	5	31.6	31.5	31.7
10.0	10	32.5	32.0	32.4
10.0	all	32.0	31.7	32.2

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	4691.8	4939.2	4816.8
0.5	5	3119.6	3156.2	2893.6
0.5	10	2648.0	2730.5	2632.3
0.5	all	2882.6	2941.5	2636.6
1.0	2	3926.6	4843.8	4545.2
1.0	5	2856.7	3016.1	2708.2
1.0	10	2564.9	2650.8	2600.9
1.0	all	2785.1	2822.1	2590.4
5.0	2	2989.6	3918.5	3475.1
5.0	5	2568.4	2545.4	2435.6
5.0	10	2426.9	2400.0	2383.4
5.0	all	2534.9	2428.5	2395.1
10.0	2	2828.0	3087.5	3085.0
10.0	5	2430.2	2456.6	2411.0
10.0	10	2383.3	2389.7	2336.9
10.0	all	2456.1	2460.0	2378.8

Table 31: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation with regularization $|p|$.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	3812.7	3781.7	3998.3
0.5	5	2290.0	2270.7	2178.1
0.5	10	1692.4	1655.0	1643.8
0.5	all	2070.4	2051.9	1613.0
1.0	2	3477.2	3777.2	3800.6
1.0	5	2189.4	2074.2	1871.8
1.0	10	1504.1	1632.2	1494.1
1.0	all	1872.6	1886.9	1455.8
5.0	2	2417.6	3272.0	3014.7
5.0	5	1704.6	1773.4	1561.8
5.0	10	1355.4	1419.5	1376.5
5.0	all	1610.3	1646.5	1352.4
10.0	2	2303.9	3127.9	2468.7
10.0	5	1595.0	1513.7	1468.0
10.0	10	1327.5	1366.6	1254.1
10.0	all	1514.8	1473.6	1228.8

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1183.5	1184.1	1182.5
0.5	5	1212.5	1214.9	1217.5
0.5	10	1225.8	1223.2	1226.7
0.5	all	1212.5	1217.5	1230.0
1.0	2	1191.0	1182.1	1186.0
1.0	5	1212.6	1216.6	1217.0
1.0	10	1229.6	1226.7	1234.9
1.0	all	1221.0	1219.9	1233.2
5.0	2	1205.0	1193.7	1197.7
5.0	5	1227.1	1218.9	1228.5
5.0	10	1233.9	1237.7	1245.9
5.0	all	1230.4	1228.0	1245.9
10.0	2	1208.8	1199.6	1208.2
10.0	5	1231.0	1230.3	1234.6
10.0	10	1244.2	1241.9	1259.9
10.0	all	1234.7	1237.7	1259.8

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	29.9	30.0	29.9
0.5	5	30.4	30.4	30.4
0.5	10	30.6	30.6	30.6
0.5	all	30.4	30.4	30.7
1.0	2	30.1	29.9	30.0
1.0	5	30.4	30.4	30.4
1.0	10	30.7	30.6	30.8
1.0	all	30.5	30.5	30.7
5.0	2	30.2	30.1	30.1
5.0	5	30.6	30.5	30.6
5.0	10	30.8	30.8	30.9
5.0	all	30.7	30.6	30.9
10.0	2	30.3	30.1	30.3
10.0	5	30.7	30.7	30.7
10.0	10	30.9	30.8	31.1
10.0	all	30.8	30.8	31.1

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	4996.2	4965.8	5180.8
0.5	5	3502.5	3485.6	3395.6
0.5	10	2918.3	2878.2	2870.5
0.5	all	3282.9	3269.4	2843.0
1.0	2	4668.2	4959.3	4986.6
1.0	5	3402.0	3290.8	3088.8
1.0	10	2733.7	2858.9	2729.0
1.0	all	3093.5	3106.8	2689.0
5.0	2	3622.6	4465.7	4212.4
5.0	5	2931.7	2992.3	2790.3
5.0	10	2589.3	2657.2	2622.4
5.0	all	2840.7	2874.5	2598.3
10.0	2	3512.6	4327.4	3676.9
10.0	5	2826.0	2744.0	2702.6
10.0	10	2571.7	2608.5	2514.0
10.0	all	2749.5	2711.3	2488.6

Table 32: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation with regularization p^2.

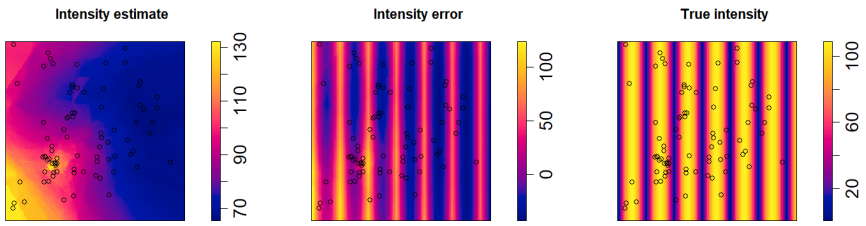


Figure 26: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with regularization strength $\lambda_r = 1$ and factor $R(\theta) = |p_v|$ on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.8.4 Simple sequential inhibition process

In Tables 33 and 34 we see the results for the simple sequential inhibition process and again note similar trends as in the previous two Sections. However, for this process, we see no real improvement in terms of MISE when comparing results to Table 9. In fact, we see that this method seems to perform slightly worse even. We see noticeable improvements in terms of IV that are outweighed by increases in terms of ISB. Looking at Table 34 we see slightly better results than in Table 33 which would suggest that a smaller regularization, and thereby larger p_v, achieve better results. Previously, the best results in terms of MISE came from using the kernel estimator with values ca. 1100.

Looking at Figure 27 we again see an increased amount of edge effects and that the intensity estimate looks more homogeneous. We also note that the two regions of high intensity of the true intensity were not captured by the average intensity estimate.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	123.4	144.5	136.4
0.5	5	84.4	88.2	80.0
0.5	10	74.1	76.1	70.7
0.5	all	78.9	85.1	69.9
1.0	2	111.6	131.2	125.4
1.0	5	77.9	76.9	73.4
1.0	10	73.5	69.7	65.0
1.0	all	74.8	76.4	64.8
5.0	2	83.3	98.4	96.5
5.0	5	58.7	62.6	56.0
5.0	10	54.5	59.4	52.0
5.0	all	54.4	61.4	50.0
10.0	2	73.2	84.0	85.6
10.0	5	53.8	58.1	54.1
10.0	10	49.8	54.0	47.1
10.0	all	45.9	49.1	52.8

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1454.7	1418.0	1427.4
0.5	5	1509.2	1502.0	1507.9
0.5	10	1512.8	1512.6	1514.7
0.5	all	1505.0	1505.3	1512.9
1.0	2	1464.3	1436.9	1449.4
1.0	5	1509.1	1510.9	1509.6
1.0	10	1515.7	1512.8	1520.1
1.0	all	1509.2	1509.9	1515.2
5.0	2	1507.7	1486.8	1495.8
5.0	5	1518.1	1514.0	1518.2
5.0	10	1521.7	1516.8	1526.9
5.0	all	1518.2	1517.8	1526.5
10.0	2	1517.9	1505.6	1506.5
10.0	5	1526.3	1521.4	1526.2
10.0	10	1547.9	1526.3	1549.3
10.0	all	1536.8	1524.2	1544.7

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	32.0	31.6	31.7
0.5	5	32.6	32.5	32.5
0.5	10	32.5	32.5	32.5
0.5	all	32.5	32.5	32.5
1.0	2	32.1	31.8	32.0
1.0	5	32.5	32.6	32.5
1.0	10	32.5	32.5	32.5
1.0	all	32.5	32.5	32.5
5.0	2	32.5	32.4	32.4
5.0	5	32.4	32.4	32.4
5.0	10	32.3	32.4	32.3
5.0	all	32.3	32.4	32.2
10.0	2	32.6	32.5	32.5
10.0	5	32.2	32.3	32.3
10.0	10	32.2	32.2	32.2
10.0	all	32.2	32.2	32.2

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1578.1	1562.5	1563.8
0.5	5	1593.6	1590.3	1587.8
0.5	10	1586.9	1588.8	1585.4
0.5	all	1583.9	1590.4	1582.8
1.0	2	1575.9	1568.2	1574.8
1.0	5	1587.0	1587.8	1582.9
1.0	10	1589.2	1582.5	1585.2
1.0	all	1584.1	1586.3	1580.0
5.0	2	1591.0	1585.2	1592.3
5.0	5	1576.9	1576.6	1574.2
5.0	10	1576.2	1576.2	1578.9
5.0	all	1572.7	1579.2	1576.5
10.0	2	1591.1	1589.6	1592.1
10.0	5	1580.1	1579.5	1580.3
10.0	10	1597.7	1580.3	1596.4
10.0	all	1582.7	1573.3	1597.5

Table 33: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation with regularization $|p|$.

(a) IV

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	143.5	149.7	143.3
0.5	5	101.5	100.3	90.8
0.5	10	86.4	87.5	86.9
0.5	all	90.3	91.5	81.4
1.0	2	128.8	145.3	145.5
1.0	5	97.1	90.5	88.4
1.0	10	78.8	84.4	79.1
1.0	all	83.5	86.5	79.3
5.0	2	109.9	120.3	117.9
5.0	5	79.4	76.7	79.4
5.0	10	74.9	72.7	65.4
5.0	all	77.5	75.6	62.9
10.0	2	90.7	115.0	109.3
10.0	5	73.7	74.0	69.6
10.0	10	65.8	74.1	65.4
10.0	all	69.6	68.9	64.3

(b) ISB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1422.6	1406.9	1418.4
0.5	5	1483.5	1489.0	1497.2
0.5	10	1506.4	1504.4	1507.1
0.5	all	1493.6	1496.6	1505.9
1.0	2	1440.4	1415.2	1419.9
1.0	5	1492.1	1496.0	1501.8
1.0	10	1509.6	1504.1	1509.8
1.0	all	1501.1	1501.0	1510.1
5.0	2	1477.1	1452.9	1462.5
5.0	5	1512.2	1508.8	1513.7
5.0	10	1515.0	1511.9	1515.6
5.0	all	1511.5	1509.6	1516.4
10.0	2	1500.0	1464.6	1471.2
10.0	5	1512.1	1510.7	1513.8
10.0	10	1514.8	1514.2	1514.0
10.0	all	1515.4	1513.6	1517.5

(c) IAB

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	31.7	31.5	31.6
0.5	5	32.3	32.4	32.4
0.5	10	32.5	32.5	32.5
0.5	all	32.4	32.4	32.5
1.0	2	31.9	31.6	31.7
1.0	5	32.4	32.5	32.5
1.0	10	32.5	32.5	32.5
1.0	all	32.5	32.5	32.5
5.0	2	32.3	32.0	32.1
5.0	5	32.5	32.5	32.6
5.0	10	32.6	32.5	32.5
5.0	all	32.5	32.5	32.5
10.0	2	32.5	32.1	32.2
10.0	5	32.5	32.5	32.5
10.0	10	32.5	32.5	32.4
10.0	all	32.5	32.5	32.5

(d) MISE

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	1566.1	1556.6	1561.7
0.5	5	1585.0	1589.3	1588.0
0.5	10	1592.7	1591.9	1594.0
0.5	all	1583.9	1588.1	1587.2
1.0	2	1569.2	1560.5	1565.4
1.0	5	1589.2	1586.5	1590.3
1.0	10	1588.4	1588.5	1588.9
1.0	all	1584.7	1587.5	1589.5
5.0	2	1587.0	1573.2	1580.3
5.0	5	1591.7	1585.5	1593.1
5.0	10	1589.9	1584.6	1581.0
5.0	all	1589.0	1585.2	1579.4
10.0	2	1590.7	1579.6	1580.5
10.0	5	1585.8	1584.6	1583.4
10.0	10	1580.6	1588.4	1579.5
10.0	all	1585.0	1582.6	1581.8

Table 34: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation with regularization p^2.

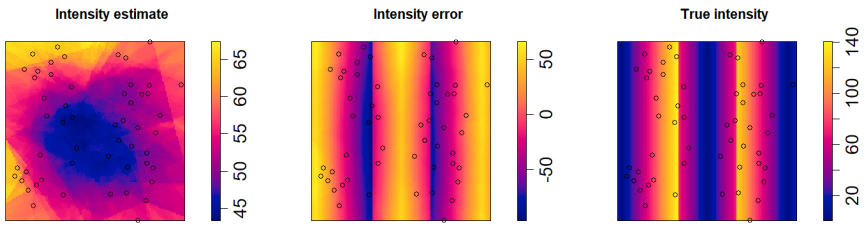


Figure 27: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with regularization strength $\lambda_r = 1$ and factor $R(\theta) = |p_v|$ on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.9 Fixed thinning size

In Equation (17) we create thinnings of a point pattern with retention probability p_v by independently marking each point x_i of the point pattern $\mathbf{x}$ with a mark of 1 with probability p_v. The result is that the number of points in the thinned pattern is $\text{Bin}(n, p_v)$ distributed where n is the number of points of $\mathbf{x}$. An alternative approach to this is to draw a random sample of a fixed size of points without replacement from $\mathbf{x}$ where the size is $\lfloor p_v n \rfloor$. The difference here is that we avoid scenarios where the outcome of $\text{Bin}(n, p_v)$ is 0 or 1 in which case the estimated intensity is 0 or $1/|W|$. These scenarios are quite common for low values of p_v which Moradi et al. [2018] showed had better performance.

In Moradi et al. [2018] it was shown that lower values of p_v result in better intensity estimates with $p = 0.01$ generally having the best performance. We know that we need at least 2 or more points in the thinned pattern for the Voronoi estimator to actually result in something that is not 0 or $1/|W|$. At low values of p_v the probability that 2 or more points are retained is quite low meaning that we might observe interesting results by instead looking at fixed sizes of $2, 3, \ldots, 10$. Note that this means we choose p_v such that $p_v n = 2, 3, \ldots, 10$.

Due to the computational complexity, we limit this investigation to the Log-Gaussian Cox and simple sequential inhibition processes as they have displayed the worst results so far.

4.9.1 Log-Gaussian Cox process

We can see the results for the Log-Gaussian Cox process in Table 35. If we compare this to what we see in Table 5 we see very nice improvements for all k. The results in terms of ISB seem very similar to the random thinning size but the improvement in IV is very noticeable. In Table 5 we see that variance is very large for lower values of k which is something we see here as well but on

a much smaller scale. Comparing the results here to Table 5 in Moradi et al. [2018] we see that this method achieves almost the same results as random size thinnings with $p = 0.01$. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with regularization with values ca. 2400.

In Table 36 we see the average parameter values selected. We see that the selected size seems to be around 5 points for all loss functions except $\mathcal{L}_3$ where we see some variance.

In Figure 28 we can see plots of the intensity estimate. As we can see the intensity estimate looks quite homogeneous.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1396.08	1409.27	1364.07
5	1234.58	1288.54	1711.04
10	1175.46	1133.25	1379.24
all	1185.65	1202.72	1458.28

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1217.82	1212.57	1211.02
5	1213.52	1201.16	1209.60
10	1201.40	1203.52	1218.56
all	1204.25	1205.71	1214.59

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	30.67	30.57	30.56
5	30.59	30.29	30.45
10	30.28	30.31	30.49
all	30.41	30.33	30.45

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2613.91	2621.84	2575.09
5	2448.09	2489.70	2920.65
10	2376.85	2336.77	2597.80
all	2389.90	2408.43	2672.87

Table 35: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation with fixed size thinnings.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	5.010	5.160	5.020
5	5.280	5.780	7.190
10	5.120	5.110	6.750

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	552.5	515.0	557.5
5	515.0	542.5	540.0
10	532.5	545.0	577.5

Table 36: Average parameter values selected for the Voronoi estimator of the Log-Gaussian Cox process.

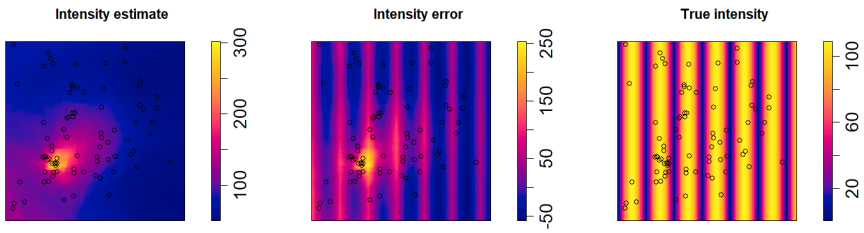


Figure 28: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with fixed thinning size on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.9.2 Simple sequential inhibition process

The results for the simple sequential inhibition process can be seen in Table 37. Comparing these results to Table 9 we see that this method achieves worse performance in terms of ISB and MISE and small improvements in terms of IV. Previously, the best results in terms of MISE came from using the kernel estimator with values ca. 1100.

If we compare Figure 29 to 11 we can see that the intensity estimate with fixed thinning size looks more homogeneous with an increased amount of edge effects.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	150.67	152.33	150.37
5	164.87	165.35	159.83
10	147.79	169.14	163.21
all	167.23	166.73	162.59

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1496.56	1495.93	1493.89
5	1491.02	1488.50	1492.30
10	1500.85	1501.91	1500.32
all	1495.35	1492.62	1500.38

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	32.19	32.19	32.16
5	32.15	32.14	32.16
10	32.31	32.27	32.24
all	32.20	32.18	32.24

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1647.23	1648.26	1644.26
5	1655.89	1653.85	1652.13
10	1648.63	1671.05	1663.53
all	1662.58	1659.35	1662.97

Table 37: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation with fixed size thinnings.

(a) p_v

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2.430	2.540	2.430
5	3.050	3.210	2.740
10	3.730	3.540	3.080

(b) m

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	487.5	500.0	487.5
5	505.0	532.5	575.0
10	520.0	567.5	570.0

Table 38: Average parameter values selected for the Voronoi estimator of the simple sequential inhibition process.

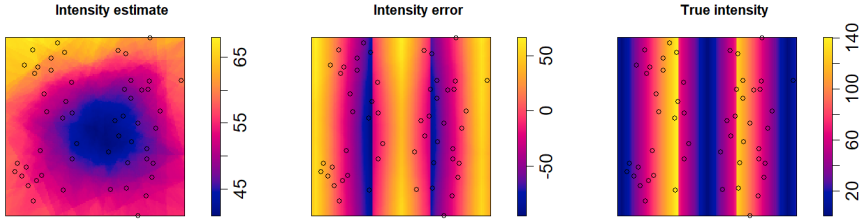


Figure 29: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with fixed thinning size on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.10 Periodic edge-correction

As we have seen in the plots of the average estimated intensities, the Voronoi estimator suffers from edge effects. For the kernel estimator, there are a number of edge-correction methods available but this is not the case for the Voronoi estimator. One edge-correction method that would work on the Voronoi estimator is periodic edge-correction. This method takes copies of the point pattern of interest and creates a 3×3 grid, i.e. surrounding the original point pattern with 8 copies of itself, estimates the intensity on the full 3×3 grid, and then uses the central subwindow as the final intensity estimate. An example of this method can be seen in Figure 30.

While there are other edge-correction methods available for the kernel estimator we limit our investigation to periodic edge-correction so that results are comparable.

The investigation of this edge-correction method comes with an increase in computational complexity and as such, we restrict the parameter m to only be $m = 250, 500$.

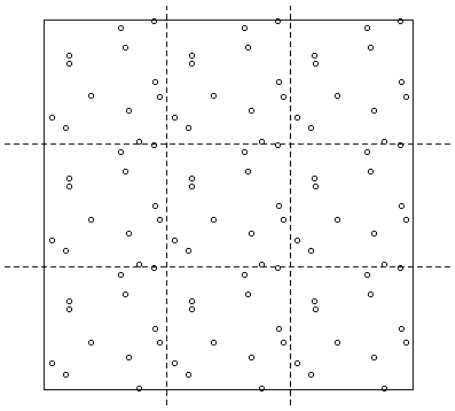


Figure 30: A realization of a homogeneous Poisson process with $\rho = 15$, with periodic edge-correction.

4.10.1 Voronoi intensity estimation

We start by presenting the results of using periodic edge-correction on the Voronoi estimator.

4.10.1.1 Poisson process If we compare Table 39 to 1 we see that this edge-correction method eliminates almost all bias and also decreases the variance. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with regularization with values ca. 165.

Looking at Figure 31 we can see that the intensity estimate looks quite homogeneous and is close to the true intensity. Comparing this Figure to 8 we can see that the edge effects have been eliminated.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	146.68	146.21	151.36
5	107.79	111.41	108.25
10	112.37	100.82	97.17
all	105.49	108.23	100.66

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1.65	1.68	1.96
5	0.73	0.61	1.10
10	0.74	0.59	0.69
all	0.91	1.05	0.93

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1.06	1.07	1.17
5	0.66	0.62	0.86
10	0.70	0.62	0.69
all	0.77	0.85	0.83

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	148.34	147.88	153.32
5	108.53	112.02	109.35
10	113.12	101.41	97.86
all	106.40	109.28	101.60

Table 39: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation.

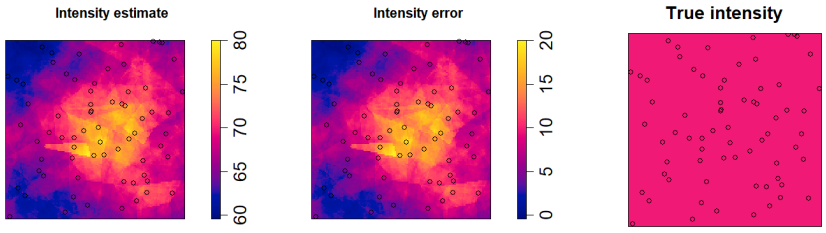


Figure 31: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the Voronoi method with periodic edge-correction on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.1.2 Inhomogeneous Poisson process Looking at Table 40 and comparing that to Table 3 we see a clear improvement in IV and a smaller improvement in ISB. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with regularization with values ca. 1010.

Furthermore, looking at Figure 32 we see that the intensity estimate has less edge effects and looks quite homogeneous compared to Figure 9. However, it fails to capture the periodicity of the true intensity which is what we see in Figure 9 as well.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	147.30	139.93	147.99
5	126.65	119.47	113.50
10	129.07	107.15	97.40
all	119.08	109.01	100.72

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	864.38	864.63	864.78
5	867.00	865.81	865.74
10	862.57	864.27	865.23
all	866.64	866.90	865.97

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	25.25	25.28	25.28
5	25.28	25.28	25.26
10	25.22	25.25	25.27
all	25.28	25.27	25.28

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1011.67	1004.57	1012.77
5	993.65	985.28	979.24
10	991.64	971.42	962.63
all	985.72	975.91	966.69

Table 40: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation.

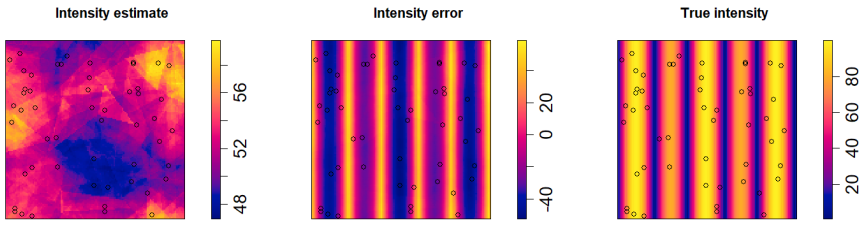


Figure 32: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.1.3 Log-Gaussian Cox process Comparing Table 41 to Table 5 we see an increasing degree of improvement with respect to k. For $k = 2$ the improvement is quite small while noticeably larger for $k = 10$. We again see that the main improvement comes from the decrease in IV alongside a smaller improvement in ISB. Previously, the best results in terms of MISE came from using the resample-smoothing Voronoi estimator with fixed thinning size with values ca. 2350.

Looking at Figure 33 we see that it resembles Figure 10 however, using this method we see that the interval of intensity estimate is smaller. We again see that it fails to capture the periodicity of the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	4097.86	3948.94	4094.32
5	2202.32	2279.19	3113.36
10	1253.04	1275.35	3908.04
all	2058.49	1980.64	3079.98

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1108.39	1110.36	1107.68
5	1122.50	1125.18	1112.44
10	1130.73	1133.19	1114.51
all	1124.57	1124.00	1114.18

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	29.19	29.23	29.17
5	29.44	29.48	29.25
10	29.58	29.62	29.25
all	29.47	29.46	29.29

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	5206.25	5059.30	5202.00
5	3324.83	3404.37	4225.81
10	2383.78	2408.54	5022.55
all	3183.05	3104.64	4194.16

Table 41: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

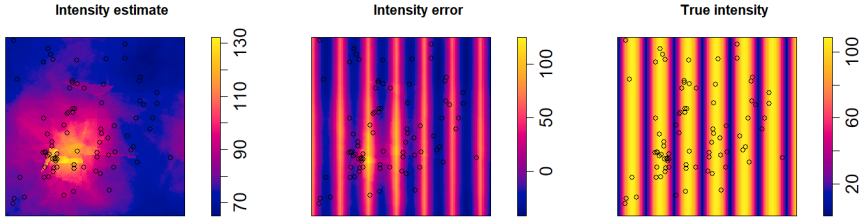


Figure 33: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.10.1.4 Simple sequential inhibition process Comparing Tables 42 and 9 we see similar results as in previous Sections, a larger decrease in IV and a smaller decrease in ISB. Previously, the best results in terms of MISE came from using the kernel estimator with values ca. 1100.

Looking at Figure 34 we can see that the edge effects are decreased.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	124.89	120.79	124.28
5	125.65	147.43	169.54
10	129.29	112.06	171.24
all	132.39	140.93	166.53

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1292.38	1295.68	1292.49
5	1304.47	1267.81	1219.66
10	1300.05	1328.23	1232.95
all	1296.72	1274.53	1233.61

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	30.20	30.21	30.20
5	30.28	29.86	29.22
10	30.22	30.60	29.40
all	30.19	29.94	29.39

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1417.27	1416.47	1416.77
5	1430.13	1415.24	1389.20
10	1429.35	1440.29	1404.19
all	1429.11	1415.46	1400.13

Table 42: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

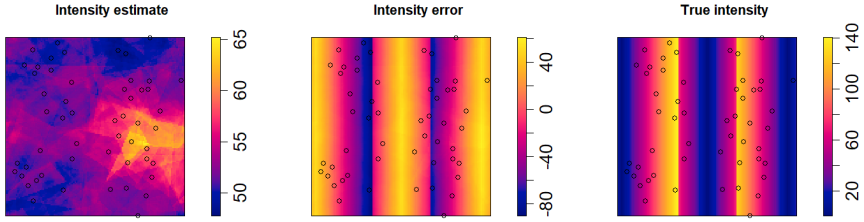


Figure 34: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.10.2 Kernel intensity estimation

We now move on to presenting the results of using periodic edge-correction on the kernel estimator.

4.10.2.1 Poisson process Comparing Table 43 to Table 11 we see that this edge-correction method offers a very large improvement. As we can see the integrated square bias is very close to 0 for all k and L.

Comparing Figure 35 to Figure 12 we can see that the edge effects have been removed by this method.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	83.81	83.81	83.81
5	91.42	87.42	85.11
10	122.49	113.53	87.17
all	115.12	100.90	87.17

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.62	0.62	0.62
5	0.94	0.86	0.71
10	0.81	0.89	0.65
all	0.89	0.81	0.65

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.76	0.76	0.76
5	0.81	0.81	0.78
10	0.77	0.81	0.77
all	0.79	0.78	0.77

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	84.43	84.43	84.43
5	92.35	88.28	85.82
10	123.30	114.41	87.83
all	116.01	101.72	87.83

Table 43: Evaluation metrics for the kernel estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation.

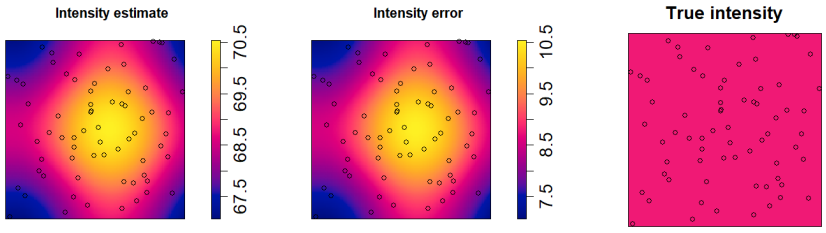


Figure 35: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the kernel method with periodic edge-correction on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.2.2 Inhomogeneous Poisson process Comparing Table 44 to Table 13 we see that that this edge-correction method results in a large decrease in both IV and ISB.

Comparing Figure 36 to 13 we can see that this method eliminates edge effects. We also note that the intensity estimate looks very homogeneous and fails to capture the periodicity of the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	94.43	94.21	94.43
5	93.78	95.45	91.04
10	99.94	92.62	88.26
all	91.87	91.10	88.26

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	865.30	865.31	865.30
5	865.58	865.58	865.61
10	863.67	864.25	865.81
all	865.13	865.04	865.81

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	25.25	25.25	25.25
5	25.25	25.25	25.25
10	25.23	25.24	25.25
all	25.24	25.25	25.25

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	959.73	959.52	959.73
5	959.37	961.04	956.65
10	963.61	956.86	954.07
all	957.00	956.14	954.07

Table 44: Evaluation metrics for the kernel estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation.

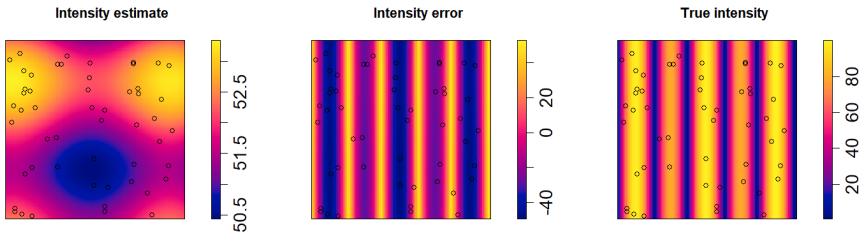


Figure 36: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method with periodic edge-correction on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.2.3 Log-Gaussian Cox process Comparing Table 45 to Table 15 we see the same results as we have for the previous processes, a decrease in both IV and ISB.

Furthermore, comparing Figure 37 to 14 we again see no noticeable edge effects. We again note that the intensity estimate fails to capture the periodicity of the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1289.41	1280.86	1289.41
5	1282.29	1354.28	1544.84
10	1123.27	1066.45	1285.28
all	1248.55	1286.42	1523.19

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1138.60	1139.19	1138.60
5	1138.46	1138.05	1137.42
10	1138.72	1138.86	1139.13
all	1138.78	1140.26	1139.71

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	29.71	29.72	29.71
5	29.72	29.71	29.70
10	29.70	29.70	29.69
all	29.71	29.72	29.72

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2428.01	2420.06	2428.01
5	2420.75	2492.33	2682.27
10	2261.99	2205.30	2424.42
all	2387.34	2426.68	2662.89

Table 45: Evaluation metrics for the v estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

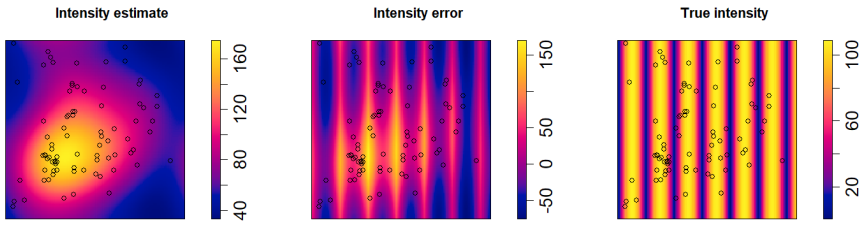


Figure 37: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method with periodic edge-correction on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.10.2.4 Simple sequential inhibition process If we compare Table 46 to Table 17 we see a decrease in IV but an increase in ISB resulting in a higher MISE.

Comparing Figure 38 to Figure 15 we can see that the edge effects have disappeared. We note that the intensity estimate manages to capture some of the two higher intensity regions.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	133.83	127.22	134.57
5	155.71	156.70	173.99
10	127.32	152.52	207.43
all	162.66	169.58	211.11

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1158.23	1168.91	1155.18
5	1133.45	1135.46	1066.37
10	1158.50	1115.48	969.60
all	1099.59	1099.28	965.88

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	28.31	28.46	28.27
5	27.97	28.00	27.04
10	28.32	27.72	25.62
all	27.51	27.49	25.56

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1292.06	1296.13	1289.75
5	1289.16	1292.16	1240.36
10	1285.82	1268.00	1177.03
all	1262.25	1268.86	1176.99

Table 46: Evaluation metrics for the kernel estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

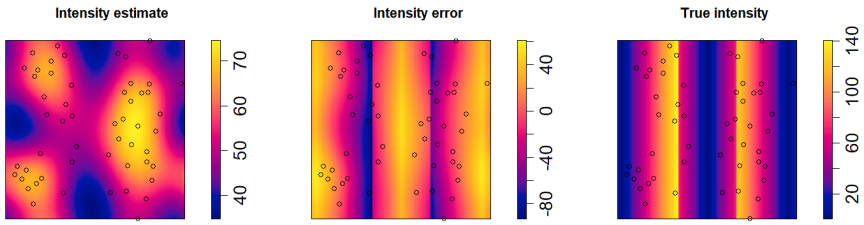


Figure 38: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the kernel method with periodic edge-correction on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

4.10.3 Fixed-size Voronoi intensity estimation

As we have seen the periodic edge-correction performs quite well, with a noticeable increase in performance. Before this, the best results were seen when using thinnings of fixed size in the Voronoi estimator. We will now investigate what happens if we apply periodic edge-correction to this method. Note that previously we investigated fixed-size thinnings with $2, 3, \ldots, 10$ points. Since we now surround the original point pattern with 8 copies of itself we instead investigate $9 \cdot 2, 9 \cdot 3, \ldots, 9 \cdot 10$.

4.10.3.1 Poisson process Looking at Table 47 we see MISE scores of around 90-95. Comparing this to Table 39 this is a noticeable improvement.

However, if we compare it to Table 43 we see that the kernel estimator performs better at $k = 2$.

Looking at Figure 39 we can see that the intensity estimate looks quite homogeneous and is close to the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	91.35	90.17	91.78
5	92.52	87.18	89.23
10	94.71	90.15	92.92
all	95.08	93.40	90.41

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.52	0.49	0.52
5	0.51	0.44	0.64
10	1.01	0.79	0.69
all	0.88	0.80	0.96

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.60	0.63	0.64
5	0.55	0.58	0.65
10	0.82	0.75	0.70
all	0.78	0.72	0.82

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	91.87	90.66	92.30
5	93.03	87.62	89.87
10	95.72	90.95	93.61
all	95.96	94.20	91.38

Table 47: Evaluation metrics for the Voronoi estimated intensity of the homogeneous Poisson process with parameters selected by k-fold cross-validation.

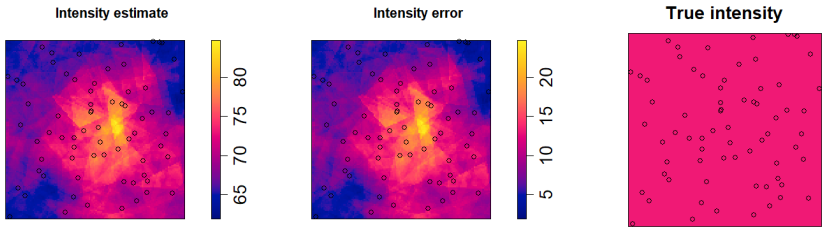


Figure 39: Left: intensity estimate, middle: intensity error, and right: true intensity, in this case, a constant 60. This estimate was produced by the Voronoi method with periodic edge-correction and fixed thinning size on the homogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.3.2 Inhomogeneous Poisson process If we compare Table 48 to Table 40 we see that using fixed size thinnings offers some improvement. Comparing this method to the kernel intensity estimator in Table 44 we again see that the kernel estimator performs just a tad better by having a slightly lower IV.

If we look at Figure 40 we can see a plot of the intensity estimate. We again see that this estimator fails to capture the periodic nature of the true intensity.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	100.65	100.75	99.48
5	101.46	100.48	102.00
10	104.06	97.13	102.40
all	100.12	96.74	104.49

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	866.06	865.11	866.55
5	865.89	865.97	865.02
10	864.67	864.79	865.77
all	864.60	865.56	864.89

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	25.29	25.27	25.27
5	25.25	25.29	25.26
10	25.25	25.26	25.25
all	25.25	25.28	25.25

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	966.71	965.87	966.03
5	967.35	966.45	967.02
10	968.74	961.92	968.17
all	964.72	962.30	969.38

Table 48: Evaluation metrics for the Voronoi estimated intensity of the inhomogeneous Poisson process with parameters selected by k-fold cross-validation.

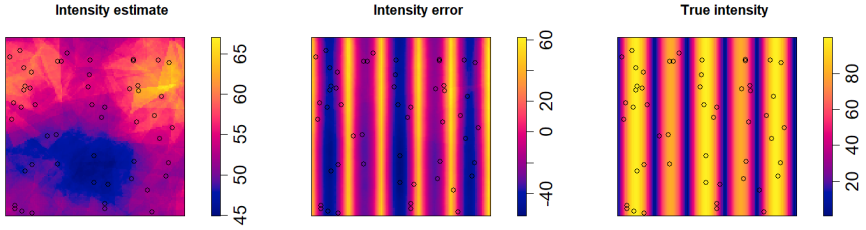


Figure 40: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction and fixed thinning size on the inhomogeneous Poisson process by $\mathcal{L}_1$ and $k = 10$.

4.10.3.3 Log-Gaussian Cox process Comparing Table 49 to Table 41 we can see that using fixed size thinnings offers a large improvement in IV resulting in a MISE of around 2065 which is the lowest score this model has achieved yet. Furthermore, looking at Table 45 we can see that Voronoi estimator performs better than the kernel estimator for this model.

Looking at Figure 41 we see that it looks quite similar to Figure 37.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1196.76	1038.35	1197.97
5	1027.81	1024.31	1169.44
10	930.46	935.66	990.38
all	983.07	1016.51	1066.18

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1130.41	1131.47	1128.00
5	1135.21	1132.32	1131.72
10	1132.82	1134.19	1135.66
all	1131.13	1131.03	1133.33

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	29.61	29.62	29.56
5	29.68	29.64	29.60
10	29.64	29.65	29.66
all	29.62	29.61	29.63

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	2327.16	2169.82	2325.97
5	2163.02	2156.63	2301.16
10	2063.28	2069.85	2126.04
all	2114.20	2147.54	2199.51

Table 49: Evaluation metrics for the Voronoi estimated intensity of the Log-Gaussian Cox process with parameters selected by k-fold cross-validation.

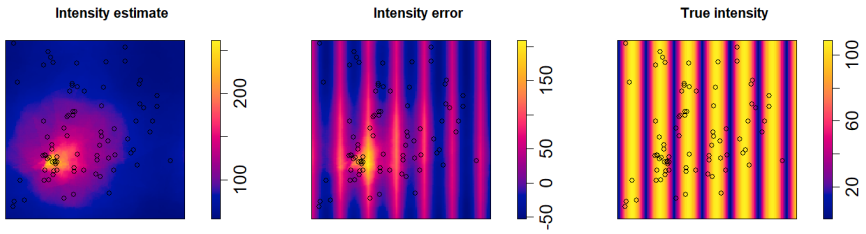


Figure 41: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction and fixed thinning size on the Log-Gaussian Cox process by $\mathcal{L}_1$ and $k = 10$.

4.10.3.4 Simple sequential inhibition process Comparing Table 50 to Table 42 we see that fixed size thinnings actually performs slightly worse. This behavior is also seen when comparing Tables 9 and 37.

Looking at Figure 42 we see that the intensity estimate looks worse than in Figure 38 with no real separation of the two higher intensity regions.

(a) IV

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	47.12	46.28	46.82
5	51.96	56.86	52.88
10	53.15	52.18	51.38
all	52.73	55.99	53.29

(b) ISB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1411.50	1408.26	1409.87
5	1402.56	1395.27	1400.31
10	1400.80	1402.74	1404.18
all	1403.20	1394.47	1401.53

(c) IAB

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	31.57	31.53	31.56
5	31.47	31.41	31.47
10	31.45	31.49	31.49
all	31.48	31.38	31.48

(d) MISE

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	1458.63	1454.54	1456.69
5	1454.52	1452.13	1453.19
10	1453.95	1454.92	1455.56
all	1455.93	1450.46	1454.82

Table 50: Evaluation metrics for the Voronoi estimated intensity of the simple sequential inhibition process with parameters selected by k-fold cross-validation.

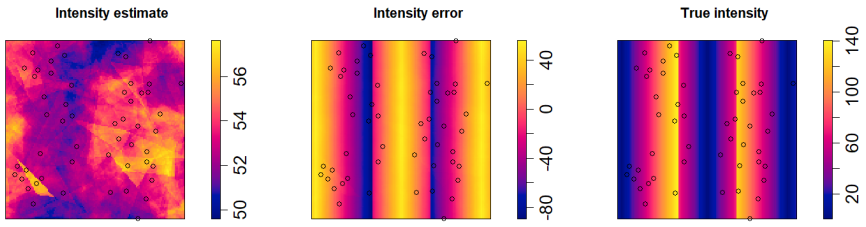


Figure 42: Left: intensity estimate, middle: intensity error, and right: true intensity. This estimate was produced by the Voronoi method with periodic edge-correction and fixed thinning size on the simple sequential inhibition process by $\mathcal{L}_1$ and $k = 10$.

5 Discussion

As we have now seen, neither estimator managed to capture the periodic nature of the inhomogeneous Poisson and Log-Gaussian Cox process. The kernel estimator managed to capture it in the simple sequential inhibition process as did the resample-smoothing Voronoi estimator at first. In the latter results, where most often techniques were used to try to achieve a lower p_v, the Voronoi estimator failed to capture the periodicity. This is also reflected in the evaluation metrics as we saw the best mean integrated squared error for the simple sequential inhibition process in Subsection 4.3. This suggests that for some point process models using a lower p_v is not always better.

While both estimators have their advantages and disadvantages, the clear winner when it comes to performance is the kernel estimator. Not only does it only have one parameter to select but it also does not use resample-smoothing which makes it significantly faster. It should be noted that I experimented briefly with applying resample-smoothing to the kernel estimator and found that the results were much slower than for the resample-smoothing Voronoi estimator.

5.1 Cross-validation

At the start of this project, we had planned on using both k-fold and Monte Carlo cross-validation but we quickly realized that MCCV was incredibly computationally expensive. The problem already scales in time with respect to k and in the MCCV setting, we have both large values of k and a separate hyperparameter to investigate resulting in unfeasible run times. We did investigate MCCV on the Log-Gaussian Cox process but as we saw the improvements, if any, were quite small and thus decided to not investigate it further. In the k-fold setting we have to compute a total of

$$\sum_{k\in\mathbf{k}}\sum_{m_j\in\mathbf{m}}\sum_{p_v\in\mathbf{p}}\sum_{i=1}^{k}\sum_{m=1}^{m_j}1$$

Voronoi tessellations. Comparatively, in the MCCV setting, we have

$$\sum_{p_c\in\mathbf{p}_c}\sum_{k\in\mathbf{k}}\sum_{m_j\in\mathbf{m}}\sum_{p_v\in\mathbf{p}}\sum_{i=1}^{k}\sum_{m=1}^{m_j}1$$

an additional sum over the retention probabilities used to create the training and validation sets. The potential small performance improvement of MCCV is outweighed by the large increase in computational complexity and is in most scenarios not worth it.

5.2 Regularization

As we saw, applying regularization to the loss functions results in noticeable improvements. However, I am hesitant in interpreting these results. On one hand, it is a valid machine learning technique, and on the other, maybe it simply forces p_v to be lower artificially. When fitting a linear model

$$Y = X\beta$$

a regularization term is added when estimating

$$\arg\min_{\beta}\|X\beta - Y\| + \lambda\|\beta\|$$

which essentially penalizes each β_i that is non-zero to prevent overfitting. In this setting, we add regularization to avoid including features that do not actually

affect the outcome resulting in a simpler model. However, in our case we add regularization to prefer a lower p_v but what is to say that a lower p_v is a simpler model? In the end, we know from Moradi et al. [2018] that a lower p_v is better and, as we will see, regularization does improve results so this method must at least have some merit.

Throughout this thesis, we have been comparing Voronoi and kernel estimators and to have a fair comparison we should also have included a regularized kernel estimator. However, how would one go about regularizing the kernel estimator? For the Voronoi estimator, we have a fairly obvious choice of making the regularization term a function of p_v but for the kernel estimator, there is no such clear choice. One idea is perhaps to try and formulate a regularization term as a function of the variance of the estimated intensity, to prefer a smoother intensity. Another idea that was suggested by Moradi et al. [2018] is to use resample-smoothing on the kernel estimator as well, in which case we could apply regularization on the thinning probability, assuming that results improve with a lower retention probability.

5.3 Re-scaling parameters

As seen in the Tables of the average parameter values in Section 4.3 it is clear that p_v decreases with respect to k. In the k-fold setting, for a given k and a point pattern with n points, we have that the training set contains $n - \frac{n}{k}$ points. The fact that we see a decrease in p_v as k increases suggests that the optimal p_v is dependent on the number of points n. For that reason, something of interest is to investigate how the Voronoi estimator performs on point processes with a larger range in the number of points. In this thesis we investigated four processes: homogeneous and inhomogeneous Poisson, simple sequential inhibition, and Log-Gaussian Cox processes with an expected number of points 60, 58.6, 68.4, and 53.6 respectively. All these intensities are quite close to each other and it would be interesting to see how the Voronoi estimator compares to the kernel estimator on point processes with an expected number of points of around 10 and past 100. Furthermore, as we saw in Section 4.7, re-scaling the parameter p_v to address this dependence improved results.

Looking at the Tables that display average bandwidth in Section 4.4 we see that the average bandwidth decreases as k increases. This is, to me, somewhat easier to understand why, as for low values of k we have fewer points in the training set, and is thus sparser. Therefore, to ensure that at least the majority of the observation window has non-zero intensity, the bandwidth has to be larger. For this reason, it might be of interest to investigate what happens when the bandwidth is re-scaled.

5.4 Edge effects

As we have seen in plots of the average estimated intensities of the Voronoi estimator, edge effects are a large problem regardless of p_v. In Section 4.10 we suggested an edge-correction method that did improve results noticeably but it

is accompanied by a further increase in computational complexity. In general, the time complexity of creating a Voronoi tessellation scales linearly with the number of points and as such, using this edge-correction method increases the time complexity 9-fold. By default, the intensity estimates are represented as 128×128 matrices. To ensure that the final intensity estimate has the same dimensions, the intensity estimate on the periodic edge-corrected pattern has to use 384×384 matrices which further increases the computational complexity. Another issue with this edge-correction method is that it requires that the observation window W is rectangular as we otherwise would not be able to surround the original point pattern with copies of itself. Furthermore, it also assumes that the intensity is symmetric about the center of the observation window which is not always the case. Nonetheless, this edge-correction method produced intensity estimates that were clearly better than the intensity estimates using no edge-correction.

5.5 Thinning

One of the main insights about the Voronoi estimator from this thesis is that fixed-size thinnings perform much better than randomly, independently deleting points. Note that it is not necessarily that the thinnings are all of the same size but rather that we avoid ever having thinned point patterns that contain no or only 1 point. It would be of interest to investigate an alternative sampling method, i.e. use some distribution that does not assume the value 0 or 1 to determine the number of points in the thinned pattern rather than keep the number of points constant.

Throughout this thesis, we have used a fixed probability to create thinned point patterns to be used in the Voronoi estimator. However, as described in Section 2.1.5, the thinning probability can also be a function of the location, i.e. instead of a constant p, we have a function $p(u)$ for $u \in W$, when W is the observation window of a point process. This function could be determined via point process learning by fixing some function and then gradually changing the coefficients. In such a case the resample-smoothing Voronoi intensity estimator would look like

$$\hat{\rho}^{\mathcal{V}}_{p_v,m}(u; X, W) = \frac{1}{m p_v(u)} \sum_{i=1}^{m} \sum_{x \in X_i^{p_v}} \frac{\mathbf{1}\{u \in \mathcal{V}_x(X_i^{p_v}, W)\}}{|\mathcal{V}_x(X_i^{p_v}, W)|}.$$

Another area of interest is to investigate how using gradient descent or another optimization algorithm would affect the results of using point process learning. That is, rather than fixing some parameter values to investigate, we start with some initial parameter value and then change it by some small value to estimate the gradient of the loss function. We then move in the opposite direction of this estimated gradient and repeat until we, hopefully, find the global minimum of the loss function.

6 Conclusion

Our working hypothesis as this thesis began was that the Voronoi estimator was simply better than the kernel estimator but as we have shown this is not the case. The results of this thesis have shown that both estimators have their advantages and disadvantages. Further investigations into variations of both estimators are needed to come to a conclusion regarding this. As we have seen the Voronoi estimator is vastly improved by using resample-smoothing and as such, it would be of great interest to investigate how resample-smoothing affects the kernel estimator.

In this thesis we have applied point process learning on a number of estimators with success, further demonstrating the potential of this framework.

7 Appendix

7.1 Voronoi intensity estimation

7.1.1 Poisson process

(a) $k = 2$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.09	250	7	7	7
0.09	500	2	2	2
0.1	250	11	12	11
0.1	500	2	1	2
0.1	1000	2	1	2
0.1	2000	0	1	0
0.15	250	17	18	17
0.15	500	17	17	17
0.15	1000	10	9	10
0.15	2000	9	6	9
0.25	250	10	13	10
0.25	500	8	8	8
0.25	1000	4	4	4
0.25	2000	1	1	1

(b) $k = 5$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	1	3	2
0.06	500	1	1	1
0.07	250	7	4	3
0.07	500	1	1	2
0.08	250	11	7	9
0.08	500	1	3	3
0.08	1000	2	3	2
0.08	2000	0	1	0
0.09	250	10	15	15
0.09	500	8	7	7
0.09	1000	1	1	1
0.09	2000	2	2	3
0.1	250	10	11	8
0.1	500	7	8	9
0.1	1000	10	7	9
0.1	2000	2	1	4
0.15	250	7	5	9
0.15	500	6	7	6
0.15	1000	3	4	2
0.15	2000	4	4	2
0.25	250	1	2	0
0.25	500	3	3	3
0.25	1000	1	0	0
0.25	2000	1	0	0

Table 51: Number of times each parameter was selected for the homogeneous Poisson process.

(a) $k = 10$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.05	250	2	1	1
0.06	250	1	2	1
0.06	500	0	2	0
0.06	1000	2	2	1
0.07	250	7	6	4
0.07	500	4	4	5
0.07	1000	1	1	0
0.07	2000	1	0	0
0.08	250	7	8	10
0.08	500	4	4	7
0.08	1000	3	3	4
0.08	2000	0	1	0
0.09	250	8	9	10
0.09	500	5	7	11
0.09	1000	1	5	3
0.09	2000	0	0	4
0.1	250	13	11	9
0.1	500	11	10	8
0.1	1000	3	1	4
0.1	2000	0	0	1
0.15	250	6	8	8
0.15	500	5	6	5
0.15	1000	2	2	2
0.15	2000	2	0	0
0.25	250	3	3	1
0.25	500	1	2	1
0.25	1000	3	0	0
0.25	2000	1	1	0
0.5	250	1	0	0
0.5	500	1	1	0
0.75	250	2	0	0

(b) all k

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.05	250	1	0	1
0.06	250	2	3	1
0.06	500	1	2	0
0.06	1000	1	0	1
0.07	250	4	5	4
0.07	500	3	3	5
0.07	1000	0	1	0
0.08	250	11	6	10
0.08	500	2	5	7
0.08	1000	2	2	4
0.08	2000	0	1	0
0.09	250	8	8	10
0.09	500	4	7	11
0.09	1000	1	4	3
0.09	2000	2	2	4
0.1	250	17	12	9
0.1	500	9	8	8
0.1	1000	5	5	4
0.1	2000	1	1	1
0.15	250	9	6	8
0.15	500	2	6	5
0.15	1000	1	2	2
0.15	2000	3	3	0
0.25	250	3	4	1
0.25	500	3	2	1
0.25	1000	3	0	0
0.25	2000	0	1	0
0.5	250	1	0	0
0.5	500	0	1	0
0.75	250	1	0	0

Table 52: Number of times each parameter was selected for the homogeneous Poisson process.

(a) $k = 2, 5$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
5	0.06	250	1	3	0
5	0.06	500	1	1	0
5	0.07	250	3	3	0
5	0.07	500	1	1	0
5	0.08	250	4	3	0
5	0.08	500	0	3	0
5	0.08	1000	1	2	0
5	0.09	250	4	4	0
5	0.09	500	2	3	0
5	0.09	2000	2	2	0
5	0.1	250	7	6	0
5	0.1	500	4	3	0
5	0.1	1000	3	4	0
5	0.1	2000	1	1	0
5	0.15	250	3	2	0
5	0.15	500	0	2	0
5	0.15	1000	1	2	0
5	0.15	2000	2	3	0
5	0.25	250	1	1	0
5	0.25	500	2	1	0

(b) $k = 10$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
10	0.05	250	1	0	1
10	0.06	250	1	0	1
10	0.06	500	0	1	0
10	0.06	1000	1	0	1
10	0.07	250	1	2	4
10	0.07	500	2	2	5
10	0.07	1000	0	1	0
10	0.08	250	7	3	10
10	0.08	500	2	2	7
10	0.08	1000	1	0	4
10	0.08	2000	0	1	0
10	0.09	250	4	4	10
10	0.09	500	2	4	11
10	0.09	1000	1	4	3
10	0.09	2000	0	0	4
10	0.1	250	10	6	9
10	0.1	500	5	5	8
10	0.1	1000	2	1	4
10	0.1	2000	0	0	1
10	0.15	250	6	4	8
10	0.15	500	2	4	5
10	0.15	1000	0	0	2
10	0.15	2000	1	0	0
10	0.25	250	2	3	1
10	0.25	500	1	1	1
10	0.25	1000	3	0	0
10	0.25	2000	0	1	0
10	0.5	250	1	0	0
10	0.5	500	0	1	0
10	0.75	250	1	0	0

Table 53: Number of times each parameter, including k, was selected for the homogeneous Poisson process.

7.1.2 Inhomogeneous Poisson process

(a) $k = 2$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.08	250	1	1	1
0.08	500	0	1	0
0.09	250	5	5	5
0.09	500	1	0	1
0.1	250	5	5	5
0.1	500	4	5	4
0.1	1000	1	1	1
0.1	2000	2	2	2
0.15	250	25	21	25
0.15	500	14	15	14
0.15	1000	4	8	4
0.15	2000	9	9	9
0.25	250	12	9	12
0.25	500	7	7	7
0.25	1000	4	5	4
0.25	2000	6	6	6

(b) $k = 5$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	0	1	1
0.07	250	2	2	1
0.07	500	4	1	2
0.07	1000	2	1	1
0.08	250	10	11	9
0.08	500	4	3	3
0.08	1000	3	2	2
0.08	2000	0	1	0
0.09	250	9	7	6
0.09	500	4	5	5
0.09	1000	2	3	2
0.09	2000	4	1	4
0.1	250	9	10	16
0.1	500	8	9	8
0.1	1000	5	4	4
0.1	2000	0	0	2
0.15	250	9	13	10
0.15	500	9	11	7
0.15	1000	4	5	8
0.15	2000	2	1	2
0.25	250	4	5	3
0.25	500	2	2	1
0.25	1000	2	1	2
0.25	2000	2	1	1

Table 54: Number of times each parameter was selected for the inhomogeneous Poisson process.

(a) $k = 10$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	4	5	2
0.06	500	1	2	0
0.07	250	2	6	3
0.07	500	3	2	1
0.07	1000	2	0	2
0.08	250	4	5	5
0.08	500	2	3	3
0.08	1000	0	2	2
0.08	2000	1	0	0
0.09	250	10	14	12
0.09	500	7	5	8
0.09	1000	3	5	3
0.09	2000	1	0	1
0.1	250	8	11	9
0.1	500	8	13	11
0.1	1000	7	9	10
0.1	2000	4	2	2
0.15	250	12	6	15
0.15	500	6	4	5
0.15	1000	7	3	3
0.15	2000	0	1	3
0.25	250	1	0	0
0.25	500	3	2	0
0.25	1000	1	0	0
0.25	2000	1	0	0
0.5	1000	1	0	0
0.75	1000	1	0	0

(b) all k

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	1	2	2
0.06	500	1	2	0
0.07	250	3	6	3
0.07	500	2	0	1
0.07	1000	1	0	2
0.08	250	8	8	5
0.08	500	3	3	3
0.08	1000	1	0	2
0.08	2000	0	1	0
0.09	250	10	7	12
0.09	500	4	6	8
0.09	1000	2	3	3
0.09	2000	4	0	1
0.1	250	9	15	9
0.1	500	6	8	11
0.1	1000	8	9	10
0.1	2000	3	2	2
0.15	250	14	9	15
0.15	500	7	11	5
0.15	1000	6	3	3
0.15	2000	0	1	3
0.25	250	1	1	0
0.25	500	3	2	0
0.25	2000	2	1	0
0.5	1000	1	0	0

Table 55: Number of times each parameter was selected for the inhomogeneous Poisson process.

(a) $k = 2, 5$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
5	0.07	250	1	2	0
5	0.07	500	2	0	0
5	0.07	1000	1	0	0
5	0.08	250	7	6	0
5	0.08	500	2	3	0
5	0.08	1000	1	0	0
5	0.08	2000	0	1	0
5	0.09	250	7	3	0
5	0.09	500	2	3	0
5	0.09	1000	1	2	0
5	0.09	2000	3	0	0
5	0.1	250	6	7	0
5	0.1	500	5	5	0
5	0.1	1000	3	2	0
5	0.15	250	6	6	0
5	0.15	500	5	9	0
5	0.15	1000	2	2	0
5	0.25	250	1	1	0
5	0.25	500	1	1	0
5	0.25	2000	1	1	0

(b) $k = 10$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
10	0.06	250	1	2	2
10	0.06	500	1	2	0
10	0.07	250	2	4	3
10	0.07	500	0	0	1
10	0.07	1000	0	0	2
10	0.08	250	1	2	5
10	0.08	500	1	0	3
10	0.08	1000	0	0	2
10	0.09	250	3	4	12
10	0.09	500	2	3	8
10	0.09	1000	1	1	3
10	0.09	2000	1	0	1
10	0.1	250	3	8	9
10	0.1	500	1	3	11
10	0.1	1000	5	7	10
10	0.1	2000	3	2	2
10	0.15	250	8	3	15
10	0.15	500	2	2	5
10	0.15	1000	4	1	3
10	0.15	2000	0	1	3
10	0.25	500	2	1	0
10	0.25	2000	1	0	0
10	0.5	1000	1	0	0

Table 56: Number of times each parameter, including k, was selected for the inhomogeneous Poisson process.

7.1.3 Log-Gaussian Cox process

(a) $k = 2$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	1	1	1
0.07	250	1	0	0
0.07	500	0	1	1
0.08	250	2	3	2
0.08	500	1	1	1
0.09	250	3	2	3
0.09	500	0	1	0
0.09	1000	0	1	0
0.1	250	2	6	2
0.1	1000	1	0	1
0.15	250	17	17	17
0.15	500	6	6	7
0.15	1000	5	4	5
0.15	2000	3	0	3
0.25	250	12	10	11
0.25	500	15	14	16
0.25	1000	11	8	11
0.25	2000	10	8	9
0.5	250	4	5	4
0.5	500	3	6	4
0.5	1000	0	2	0
0.5	2000	1	1	0
0.75	250	2	2	2
0.9	500	0	1	0

(b) $k = 5$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.02	250	1	0	1
0.03	500	1	1	0
0.03	2000	1	1	1
0.04	250	1	0	0
0.04	500	0	1	0
0.04	1000	2	2	0
0.05	250	2	0	1
0.05	500	1	1	0
0.05	1000	1	1	0
0.05	2000	2	0	0
0.06	250	0	2	0
0.06	500	1	2	0
0.06	2000	1	0	0
0.07	250	1	3	1
0.07	500	0	1	1
0.07	1000	0	1	1
0.08	250	5	5	4
0.08	500	2	1	1
0.08	1000	1	1	3
0.08	2000	0	1	0
0.09	250	3	6	5
0.09	500	3	2	4
0.09	1000	4	1	2
0.1	250	5	5	2
0.1	500	1	1	2
0.1	1000	1	3	0
0.1	2000	3	3	0
0.15	250	10	9	9
0.15	500	8	5	11
0.15	1000	6	5	6
0.15	2000	7	8	8
0.25	250	3	4	9
0.25	500	7	6	11
0.25	1000	7	8	10
0.25	2000	1	2	3
0.5	250	2	4	2
0.5	500	2	3	0
0.5	1000	2	0	1
0.5	2000	2	1	1

Table 57: Number of times each parameter was selected for the Log-Gaussian Cox process.

(a) $k = 10$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.02	250	0	0	1
0.02	500	3	1	1
0.02	1000	0	1	0
0.03	250	2	2	1
0.03	500	2	2	1
0.03	2000	1	0	0
0.04	250	0	1	2
0.04	500	2	2	2
0.04	1000	0	0	1
0.04	2000	1	0	2
0.05	250	0	1	0
0.05	500	1	2	0
0.05	1000	1	2	1
0.05	2000	3	1	1
0.06	250	6	5	0
0.06	500	4	2	1
0.06	2000	0	0	1
0.07	250	6	9	2
0.07	500	2	1	1
0.07	1000	1	1	0
0.08	250	5	4	2
0.08	500	6	2	0
0.08	1000	0	2	0
0.08	2000	0	1	1
0.09	250	9	5	5
0.09	500	1	2	1
0.09	1000	2	1	0
0.09	2000	1	2	1
0.1	250	5	4	7
0.1	500	4	4	1
0.1	1000	1	2	3
0.1	2000	0	1	2
0.15	250	8	12	12
0.15	500	6	3	8
0.15	1000	4	4	3
0.15	2000	2	4	5
0.25	250	5	6	7
0.25	500	1	1	7
0.25	1000	1	3	8
0.25	2000	4	4	3
0.5	250	0	0	3
0.5	2000	0	0	2
0.75	2000	0	0	1

(b) all k

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.02	250	1	0	1
0.02	500	1	0	0
0.03	250	0	0	1
0.03	500	1	2	1
0.03	2000	1	1	1
0.04	250	1	0	1
0.04	500	1	1	1
0.04	1000	1	2	0
0.04	2000	0	0	2
0.05	250	2	0	0
0.05	500	1	1	0
0.05	1000	1	1	1
0.05	2000	2	0	1
0.06	250	4	5	0
0.06	500	1	1	1
0.06	2000	1	0	1
0.07	250	2	3	3
0.07	500	2	1	1
0.07	1000	0	1	1
0.08	250	4	5	2
0.08	500	6	2	0
0.08	1000	1	1	1
0.08	2000	0	0	1
0.09	250	6	8	5
0.09	500	3	1	1
0.09	1000	2	1	0
0.09	2000	0	2	1
0.1	250	6	6	7
0.1	500	0	2	1
0.1	1000	0	2	3
0.1	2000	1	2	2
0.15	250	10	8	12
0.15	500	6	4	8
0.15	1000	4	6	2
0.15	2000	8	8	7
0.25	250	3	5	7
0.25	500	5	5	8
0.25	1000	4	5	8
0.25	2000	2	3	2
0.5	250	1	2	3
0.5	500	2	2	0
0.5	1000	1	0	0
0.5	2000	2	1	1
0.75	2000	0	0	1

Table 58: Number of times each parameter was selected for the Log-Gaussian Cox process.

(a) $k = 2, 5$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	0.08	500	0	1	0
5	0.02	250	1	0	1
5	0.03	500	0	1	0
5	0.03	2000	1	1	1
5	0.04	250	1	0	0
5	0.04	1000	1	2	0
5	0.05	250	2	0	0
5	0.05	500	1	1	0
5	0.05	1000	1	1	0
5	0.05	2000	1	0	0
5	0.06	250	0	2	0
5	0.06	500	0	1	0
5	0.06	2000	1	0	0
5	0.07	250	1	2	1
5	0.07	1000	0	1	1
5	0.08	250	4	4	0
5	0.08	500	2	0	0
5	0.08	1000	1	1	1
5	0.09	250	2	5	0
5	0.09	500	3	1	0
5	0.09	1000	2	0	0
5	0.1	250	4	4	0
5	0.1	500	0	1	0
5	0.1	1000	0	1	0
5	0.1	2000	1	1	0
5	0.15	250	5	5	0
5	0.15	500	4	3	0
5	0.15	1000	3	3	0
5	0.15	2000	7	6	2
5	0.25	250	1	3	1
5	0.25	500	4	5	1
5	0.25	1000	4	4	0
5	0.25	2000	1	2	0
5	0.5	250	1	2	1
5	0.5	500	2	2	0
5	0.5	1000	1	0	0
5	0.5	2000	2	1	0

(b) $k = 10$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
10	0.02	500	1	0	0
10	0.03	250	0	0	1
10	0.03	500	1	1	1
10	0.04	250	0	0	1
10	0.04	500	1	1	1
10	0.04	2000	0	0	2
10	0.05	1000	0	0	1
10	0.05	2000	1	0	1
10	0.06	250	4	3	0
10	0.06	500	1	0	1
10	0.06	2000	0	0	1
10	0.07	250	1	1	2
10	0.07	500	2	1	1
10	0.08	250	0	1	2
10	0.08	500	4	1	0
10	0.08	2000	0	0	1
10	0.09	250	4	3	5
10	0.09	500	0	0	1
10	0.09	1000	0	1	0
10	0.09	2000	0	2	1
10	0.1	250	2	2	7
10	0.1	500	0	1	1
10	0.1	1000	0	1	3
10	0.1	2000	0	1	2
10	0.15	250	5	3	12
10	0.15	500	2	1	8
10	0.15	1000	1	3	2
10	0.15	2000	1	2	5
10	0.25	250	2	2	6
10	0.25	500	1	0	7
10	0.25	1000	0	1	8
10	0.25	2000	1	1	2
10	0.5	250	0	0	2
10	0.5	2000	0	0	1
10	0.75	2000	0	0	1

Table 59: Number of times each parameter, including k, was selected for the Log-Gaussian Cox process.

(a) $p < 0.1$

p_v	m	p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.03	250	0.1	1	1	0
0.03	1000	0.1	0	0	2
0.03	2000	0.1	1	1	0
0.04	250	0.3	0	0	1
0.04	500	0.3	0	0	1
0.04	1000	0.1	1	0	0
0.04	1000	0.3	0	0	1
0.05	250	0.1	1	1	1
0.05	500	0.1	2	0	1
0.05	1000	0.1	0	0	1
0.05	2000	0.3	0	0	1
0.06	250	0.1	3	3	2
0.06	250	0.3	0	1	0
0.06	500	0.1	0	1	0
0.06	2000	0.3	1	1	1
0.07	250	0.1	3	4	1
0.07	250	0.3	0	1	0
0.07	500	0.1	1	0	0
0.07	2000	0.1	1	1	1
0.08	250	0.1	6	2	0
0.08	500	0.1	3	2	0
0.08	500	0.3	0	1	1
0.08	1000	0.1	0	1	0
0.08	1000	0.3	1	0	0
0.08	2000	0.1	0	0	1
0.08	2000	0.3	1	1	0
0.09	250	0.1	0	3	0
0.09	250	0.3	0	1	1
0.09	500	0.1	2	2	1
0.09	1000	0.1	7	4	3
0.09	1000	0.3	1	0	0
0.09	2000	0.1	1	0	1

(b) $p \geq 0.1$

p_v	m	p_c	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	250	0.1	4	5	4
0.1	250	0.3	1	1	1
0.1	500	0.1	1	2	1
0.1	500	0.3	2	2	1
0.1	1000	0.1	3	4	3
0.1	1000	0.3	0	0	2
0.1	2000	0.1	1	3	2
0.1	2000	0.3	1	1	0
0.15	250	0.1	6	10	5
0.15	250	0.3	2	2	0
0.15	500	0.1	6	8	5
0.15	500	0.3	3	1	2
0.15	1000	0.1	10	5	5
0.15	1000	0.3	1	2	2
0.15	2000	0.1	4	3	8
0.15	2000	0.3	3	1	3
0.25	250	0.1	3	3	7
0.25	250	0.3	1	0	0
0.25	500	0.1	2	4	7
0.25	500	0.3	0	0	1
0.25	1000	0.1	1	2	8
0.25	1000	0.3	2	3	0
0.25	2000	0.1	4	2	5
0.25	2000	0.3	0	1	1
0.5	250	0.1	1	0	1
0.5	250	0.3	0	0	1
0.5	500	0.1	0	1	1
0.5	500	0.3	0	1	0
0.5	2000	0.1	0	1	1
0.5	2000	0.3	1	0	0
0.75	500	0.1	0	0	1

Table 60: Number of times each parameter was selected for the Log-Gaussian Cox process.

p_v	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.1	79	75	100
0.3	21	25	0

Table 61: Number of times each p_c was chosen for the Log-Gaussian Cox process.

7.1.3.1 Monte-Carlo cross-validation

7.1.4 Simple sequential inhibition process

(a) $k = 2$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.08	250	1	1	1
0.09	250	3	3	3
0.1	250	4	5	4
0.1	500	1	0	1
0.1	2000	1	1	1
0.15	250	19	20	19
0.15	500	19	16	19
0.15	1000	9	8	9
0.15	2000	7	7	7
0.25	250	12	14	13
0.25	500	9	7	8
0.25	1000	5	7	5
0.25	2000	3	4	3
0.5	250	4	4	4
0.5	500	1	1	1
0.5	1000	1	2	1
0.75	250	1	0	1

(b) $k = 5$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.07	250	2	3	1
0.08	250	3	3	0
0.08	500	2	2	2
0.08	1000	1	2	1
0.08	2000	0	1	0
0.09	250	7	6	9
0.09	500	1	3	3
0.09	1000	2	2	1
0.1	250	17	22	13
0.1	500	8	9	9
0.1	1000	0	1	3
0.1	2000	1	1	1
0.15	250	18	17	22
0.15	500	8	8	11
0.15	1000	2	2	3
0.15	2000	2	0	3
0.25	250	5	2	2
0.25	500	1	5	2
0.25	1000	2	0	1
0.25	2000	2	0	0
0.5	250	4	0	2
0.5	500	4	6	5
0.5	1000	2	2	2
0.5	2000	2	1	3
0.75	250	1	1	0
0.75	500	0	0	1
0.75	1000	2	1	0
0.9	1000	1	0	0

Table 62: Number of times each parameter was selected for the simple sequential inhibition process.

(a) $k = 10$

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	2	2	0
0.07	250	4	5	3
0.07	500	0	1	1
0.07	1000	1	2	0
0.07	2000	1	1	0
0.08	250	9	13	7
0.08	500	7	7	1
0.08	1000	3	1	0
0.08	2000	0	0	1
0.09	250	11	7	10
0.09	500	9	12	11
0.09	1000	3	3	2
0.09	2000	2	3	3
0.1	250	5	7	11
0.1	500	9	6	6
0.1	1000	3	1	2
0.1	2000	2	1	6
0.15	250	6	9	8
0.15	500	2	2	4
0.15	1000	2	4	3
0.15	2000	3	2	3
0.25	250	4	2	1
0.25	500	0	2	2
0.25	1000	1	1	1
0.25	2000	2	0	0
0.5	250	5	3	5
0.5	500	1	2	5
0.5	1000	2	0	2
0.5	2000	0	0	2
0.75	500	1	0	0
0.75	1000	0	1	0

(b) all k

p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.06	250	1	2	0
0.07	250	4	3	3
0.07	500	0	1	1
0.07	2000	1	1	0
0.08	250	7	6	7
0.08	500	2	4	1
0.08	1000	1	1	0
0.08	2000	0	1	1
0.09	250	7	5	11
0.09	500	4	7	10
0.09	1000	2	3	2
0.09	2000	0	1	3
0.1	250	15	16	11
0.1	500	6	7	6
0.1	1000	0	0	2
0.1	2000	1	0	6
0.15	250	15	16	8
0.15	500	6	4	4
0.15	1000	2	2	3
0.15	2000	3	1	3
0.25	250	3	1	1
0.25	500	0	4	2
0.25	1000	2	1	1
0.25	2000	2	0	0
0.5	250	6	3	5
0.5	500	3	5	5
0.5	1000	4	2	2
0.5	2000	2	1	2
0.75	250	1	1	0
0.75	1000	0	1	0

Table 63: Number of times each parameter was selected for the simple sequential inhibition process.

(a) $k = 2, 5$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
5	0.07	250	1	1	0
5	0.08	250	3	1	0
5	0.08	1000	0	1	0
5	0.08	2000	0	1	0
5	0.09	250	4	3	1
5	0.09	500	0	1	0
5	0.09	1000	1	1	0
5	0.1	250	12	12	0
5	0.1	500	4	5	0
5	0.1	2000	1	0	0
5	0.15	250	13	11	0
5	0.15	500	6	4	0
5	0.15	1000	1	1	0
5	0.15	2000	2	0	0
5	0.25	250	3	1	0
5	0.25	500	0	4	0
5	0.25	1000	2	0	0
5	0.25	2000	2	0	0
5	0.5	250	2	0	0
5	0.5	500	3	5	0
5	0.5	1000	2	2	0
5	0.5	2000	2	1	0
5	0.75	250	1	1	0

(b) $k = 10$

k	p_v	m	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
10	0.06	250	1	2	0
10	0.07	250	3	2	3
10	0.07	500	0	1	1
10	0.07	2000	1	1	0
10	0.08	250	4	5	7
10	0.08	500	2	4	1
10	0.08	1000	1	0	0
10	0.08	2000	0	0	1
10	0.09	250	3	2	10
10	0.09	500	4	6	10
10	0.09	1000	1	2	2
10	0.09	2000	0	1	3
10	0.1	250	3	4	11
10	0.1	500	2	2	6
10	0.1	1000	0	0	2
10	0.1	2000	0	0	6
10	0.15	250	2	5	8
10	0.15	500	0	0	4
10	0.15	1000	1	1	3
10	0.15	2000	1	1	3
10	0.25	250	0	0	1
10	0.25	500	0	0	2
10	0.25	1000	0	1	1
10	0.5	250	4	3	5
10	0.5	500	0	0	5
10	0.5	1000	2	0	2
10	0.5	2000	0	0	2
10	0.75	1000	0	1	0

Table 64: Number of times each parameter, including k, was selected for the simple sequential inhibition process.

7.2 Regularization

7.2.1 Poisson process

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.132	0.149	0.148
0.5	5	0.086	0.088	0.083
0.5	10	0.074	0.080	0.068
1.0	2	0.121	0.135	0.133
1.0	5	0.078	0.081	0.074
1.0	10	0.065	0.070	0.060
5.0	2	0.081	0.105	0.101
5.0	5	0.050	0.057	0.051
5.0	10	0.043	0.050	0.041
10.0	2	0.066	0.089	0.085
10.0	5	0.041	0.046	0.043
10.0	10	0.033	0.040	0.033

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	655.0	642.5	667.5
0.5	5	527.5	530.0	627.5
0.5	10	565.0	515.0	605.0
1.0	2	620.0	597.5	660.0
1.0	5	600.0	542.5	552.5
1.0	10	605.0	632.5	677.5
5.0	2	505.0	582.5	547.5
5.0	5	657.5	627.5	600.0
5.0	10	707.5	725.0	815.0
10.0	2	460.0	495.0	527.5
10.0	5	700.0	610.0	707.5
10.0	10	802.5	830.0	937.5

Table 65: Average parameter values selected for the Voronoi estimator of the homogeneous Poisson process with regularization $|p|$.

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.152	0.158	0.155
0.5	5	0.097	0.097	0.094
0.5	10	0.093	0.089	0.087
1.0	2	0.139	0.153	0.154
1.0	5	0.091	0.094	0.091
1.0	10	0.087	0.086	0.080
5.0	2	0.114	0.129	0.127
5.0	5	0.080	0.082	0.078
5.0	10	0.069	0.073	0.064
10.0	2	0.097	0.121	0.120
10.0	5	0.071	0.077	0.069
10.0	10	0.064	0.067	0.058

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	677.5	632.5	685.0
0.5	5	567.5	550.0	540.0
0.5	10	450.0	485.0	557.5
1.0	2	670.0	640.0	677.5
1.0	5	527.5	555.0	540.0
1.0	10	532.5	507.5	577.5
5.0	2	582.5	617.5	662.5
5.0	5	622.5	565.0	635.0
5.0	10	605.0	587.5	677.5
10.0	2	562.5	565.0	592.5
10.0	5	597.5	572.5	567.5
10.0	10	672.5	660.0	715.0

Table 66: Average parameter values selected for the Voronoi estimator of the homogeneous Poisson process with regularization p^2.

7.2.2 Inhomogeneous Poisson process

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.136	0.151	0.150
0.5	5	0.085	0.089	0.082
0.5	10	0.077	0.080	0.069
1.0	2	0.122	0.141	0.138
1.0	5	0.079	0.080	0.074
1.0	10	0.067	0.071	0.061
5.0	2	0.084	0.106	0.100
5.0	5	0.051	0.058	0.052
5.0	10	0.044	0.049	0.041
10.0	2	0.067	0.091	0.087
10.0	5	0.041	0.046	0.043
10.0	10	0.034	0.040	0.035

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	622.5	632.5	652.5
0.5	5	485.0	542.5	587.5
0.5	10	492.5	502.5	532.5
1.0	2	580.0	655.0	620.0
1.0	5	542.5	457.5	577.5
1.0	10	545.0	520.0	655.0
5.0	2	452.5	567.5	540.0
5.0	5	670.0	680.0	670.0
5.0	10	872.5	822.5	935.0
10.0	2	492.5	495.0	482.5
10.0	5	757.5	720.0	730.0
10.0	10	720.0	825.0	790.0

Table 67: Average parameter values selected for the Voronoi estimator of the inhomogeneous Poisson process with regularization $|p|$.

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.154	0.157	0.162
0.5	5	0.099	0.105	0.098
0.5	10	0.094	0.092	0.089
1.0	2	0.142	0.155	0.157
1.0	5	0.096	0.099	0.094
1.0	10	0.088	0.088	0.083
5.0	2	0.118	0.134	0.130
5.0	5	0.080	0.082	0.078
5.0	10	0.071	0.076	0.066
10.0	2	0.100	0.123	0.121
10.0	5	0.073	0.076	0.071
10.0	10	0.064	0.067	0.059

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	657.5	715.0	667.5
0.5	5	570.0	535.0	615.0
0.5	10	575.0	545.0	595.0
1.0	2	667.5	712.5	650.0
1.0	5	565.0	545.0	607.5
1.0	10	567.5	537.5	572.5
5.0	2	610.0	692.5	627.5
5.0	5	535.0	515.0	600.0
5.0	10	535.0	500.0	575.0
10.0	2	532.5	612.5	602.5
10.0	5	527.5	542.5	592.5
10.0	10	577.5	580.0	737.5

Table 68: Average parameter values selected for the Voronoi estimator of the inhomogeneous Poisson process with regularization p^2.

7.2.3 Log-Gaussian Cox process

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.169	0.192	0.182
0.5	5	0.094	0.100	0.087
0.5	10	0.073	0.080	0.065
1.0	2	0.141	0.177	0.164
1.0	5	0.075	0.082	0.072
1.0	10	0.064	0.070	0.056
5.0	2	0.087	0.118	0.111
5.0	5	0.050	0.054	0.048
5.0	10	0.043	0.048	0.040
10.0	2	0.069	0.096	0.093
10.0	5	0.041	0.045	0.041
10.0	10	0.033	0.041	0.034

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	707.5	640.0	685.0
0.5	5	702.5	695.0	750.0
0.5	10	595.0	580.0	697.5
1.0	2	582.5	705.0	680.0
1.0	5	577.5	662.5	730.0
1.0	10	637.5	640.0	675.0
5.0	2	595.0	692.5	687.5
5.0	5	627.5	655.0	737.5
5.0	10	715.0	745.0	710.0
10.0	2	510.0	567.5	632.5
10.0	5	667.5	707.5	687.5
10.0	10	812.5	777.5	805.0

Table 69: Average parameter values selected for the Voronoi estimator of the Log-Gaussian Cox process with regularization $|p|$.

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.192	0.204	0.202
0.5	5	0.125	0.128	0.118
0.5	10	0.093	0.098	0.086
1.0	2	0.179	0.192	0.190
1.0	5	0.110	0.116	0.101
1.0	10	0.082	0.090	0.078
5.0	2	0.125	0.153	0.145
5.0	5	0.077	0.084	0.076
5.0	10	0.068	0.072	0.061
10.0	2	0.110	0.132	0.127
10.0	5	0.070	0.072	0.067
10.0	10	0.060	0.065	0.054

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	717.5	637.5	680.0
0.5	5	812.5	782.5	745.0
0.5	10	635.0	615.0	642.5
1.0	2	707.5	655.0	682.5
1.0	5	822.5	752.5	682.5
1.0	10	617.5	590.0	647.5
5.0	2	705.0	652.5	637.5
5.0	5	602.5	707.5	737.5
5.0	10	615.0	592.5	755.0
10.0	2	722.5	670.0	690.0
10.0	5	562.5	652.5	667.5
10.0	10	640.0	635.0	712.5

Table 70: Average parameter values selected for the Voronoi estimator of the Log-Gaussian Cox process with regularization p^2.

7.2.4 Simple sequential inhibition process

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.148	0.173	0.167
0.5	5	0.092	0.095	0.088
0.5	10	0.080	0.082	0.075
1.0	2	0.136	0.159	0.152
1.0	5	0.082	0.086	0.081
1.0	10	0.072	0.077	0.064
5.0	2	0.086	0.115	0.107
5.0	5	0.055	0.063	0.056
5.0	10	0.047	0.054	0.045
10.0	2	0.074	0.093	0.091
10.0	5	0.044	0.051	0.046
10.0	10	0.037	0.044	0.038

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	667.5	700.0	627.5
0.5	5	547.5	500.0	540.0
0.5	10	482.5	597.5	592.5
1.0	2	652.5	715.0	667.5
1.0	5	615.0	527.5	592.5
1.0	10	527.5	597.5	602.5
5.0	2	465.0	630.0	575.0
5.0	5	585.0	580.0	600.0
5.0	10	765.0	630.0	802.5
10.0	2	512.5	522.5	472.5
10.0	5	667.5	710.0	672.5
10.0	10	807.5	670.0	835.0

Table 71: Average parameter values selected for the Voronoi estimator of the simple sequential inhibition process with regularization $|p|$.

(a) p_v

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	0.172	0.178	0.172
0.5	5	0.113	0.113	0.104
0.5	10	0.093	0.095	0.092
1.0	2	0.156	0.175	0.172
1.0	5	0.105	0.104	0.098
1.0	10	0.087	0.092	0.087
5.0	2	0.128	0.146	0.142
5.0	5	0.084	0.088	0.083
5.0	10	0.076	0.078	0.070
10.0	2	0.101	0.137	0.134
10.0	5	0.076	0.079	0.074
10.0	10	0.067	0.073	0.061

(b) m

λ_r	k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
0.5	2	637.5	692.5	635.0
0.5	5	452.5	500.0	525.0
0.5	10	607.5	575.0	692.5
1.0	2	665.0	695.0	635.0
1.0	5	457.5	555.0	482.5
1.0	10	575.0	587.5	682.5
5.0	2	647.5	735.0	697.5
5.0	5	605.0	530.0	577.5
5.0	10	510.0	612.5	632.5
10.0	2	572.5	692.5	675.0
10.0	5	610.0	520.0	655.0
10.0	10	595.0	627.5	620.0

Table 72: Average parameter values selected for the Voronoi estimator of the simple sequential inhibition process with regularization p^2.

7.3 Anisotropic kernel

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	$\begin{bmatrix} 3.81\text{e}{-2} & 4.68\text{e}{-4} \\ 4.68\text{e}{-4} & 3.19\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.96\text{e}{-2} & 9.61\text{e}{-4} \\ 9.61\text{e}{-4} & 3.44\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.81\text{e}{-2} & 4.68\text{e}{-4} \\ 4.68\text{e}{-4} & 3.19\text{e}{-2} \end{bmatrix}$
5	$\begin{bmatrix} 2.89\text{e}{-2} & 6.03\text{e}{-5} \\ 6.03\text{e}{-5} & 2.72\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.20\text{e}{-2} & 2.69\text{e}{-5} \\ 2.69\text{e}{-5} & 2.92\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 2.66\text{e}{-2} & -3.88\text{e}{-5} \\ -3.88\text{e}{-5} & 2.43\text{e}{-2} \end{bmatrix}$
10	$\begin{bmatrix} 2.59\text{e}{-2} & 3.41\text{e}{-4} \\ 3.41\text{e}{-4} & 2.64\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 2.96\text{e}{-2} & 1.39\text{e}{-5} \\ 1.39\text{e}{-5} & 2.67\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 2.10\text{e}{-2} & -1.51\text{e}{-4} \\ -1.51\text{e}{-4} & 2.17\text{e}{-2} \end{bmatrix}$
all	$\begin{bmatrix} 2.50\text{e}{-2} & -2.12\text{e}{-4} \\ -2.12\text{e}{-4} & 2.46\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 2.78\text{e}{-2} & 6.57\text{e}{-5} \\ 6.57\text{e}{-5} & 2.35\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 2.10\text{e}{-2} & -1.61\text{e}{-4} \\ -1.61\text{e}{-4} & 2.15\text{e}{-2} \end{bmatrix}$

Table 73: Average bandwidth matrix for the Log-Gaussian Cox process.

k	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$
2	$\begin{bmatrix} 3.34\text{e}{-2} & 9.59\text{e}{-4} \\ 9.59\text{e}{-4} & 3.88\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.38\text{e}{-2} & 1.20\text{e}{-3} \\ 1.20\text{e}{-3} & 3.94\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.34\text{e}{-2} & 9.59\text{e}{-4} \\ 9.59\text{e}{-4} & 3.88\text{e}{-2} \end{bmatrix}$
5	$\begin{bmatrix} 3.20\text{e}{-2} & 1.29\text{e}{-3} \\ 1.29\text{e}{-3} & 3.27\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.22\text{e}{-2} & 2.27\text{e}{-3} \\ 2.27\text{e}{-3} & 3.30\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.20\text{e}{-2} & 1.66\text{e}{-3} \\ 1.66\text{e}{-3} & 3.24\text{e}{-2} \end{bmatrix}$
10	$\begin{bmatrix} 3.20\text{e}{-2} & -5.80\text{e}{-4} \\ -5.80\text{e}{-4} & 3.33\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.27\text{e}{-2} & -1.57\text{e}{-3} \\ -1.57\text{e}{-3} & 3.53\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.20\text{e}{-2} & -5.28\text{e}{-5} \\ -5.28\text{e}{-5} & 3.21\text{e}{-2} \end{bmatrix}$
all	$\begin{bmatrix} 3.20\text{e}{-2} & -2.69\text{e}{-4} \\ -2.69\text{e}{-4} & 3.24\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.20\text{e}{-2} & 1.72\text{e}{-4} \\ 1.72\text{e}{-4} & 3.39\text{e}{-2} \end{bmatrix}$	$\begin{bmatrix} 3.20\text{e}{-2} & -5.28\text{e}{-5} \\ -5.28\text{e}{-5} & 3.21\text{e}{-2} \end{bmatrix}$

Table 74: Average bandwidth matrix for the simple sequential inhibition process.

References

B. W. Silverman. *Density estimation for statistics and data analysis*. Number 26 in Monographs on statistics and applied probability. Chapman & Hall/CRC, Boca Raton, 1998. ISBN 9780412246203.

M. Mehdi Moradi, Ottmar Cronie, Ege Rubak, Raphael Lachieze-Rey, Jorge Mateu, and Adrian Baddeley. Resample-smoothing of Voronoi intensity estimators. *arXiv:1807.02590 [stat]*, July 2018. URL `http://arxiv.org/abs/1807.02590`. arXiv: 1807.02590.

Ottmar Cronie, Mehdi Moradi, and Christophe A. N. Biscio. Statistical learning and cross-validation for point processes. *arXiv:2103.01356 [math, stat]*, March 2021. URL `http://arxiv.org/abs/2103.01356`. arXiv: 2103.01356.

Sung Nok Chiu, Dietrich Stoyan, W. S Kendall, and Joseph Mecke. *Stochastic geometry and its applications.* John Wiley & Sons, Ltd, 2013. ISBN 9781118658239 9781118658253 9781118658246 9781118658222 9780470664810 9781299940840. URL `http://public.eblib.com/choice/publicfullrecord.aspx?p=1246379_0`. OCLC: 843455184.

M. N. M. Van Lieshout. *Markov point processes and their applications*. Imperial College Press, London, 2000. ISBN 9781860940712. OCLC: ocm45101421.

Janine Illian, Antti Penttinen, Helga Stoyan, and Dietrich Stoyan. *Statistical analysis and modelling of spatial point patterns*. Statistics in practice. John Wiley & Sons, Ltd, Chichester, England ; Hoboken, NJ, 2008. ISBN 9780470014912. OCLC: ocn180852093.

Jesper Møller, Anne Randi Syversveen, and Rasmus Plenge Waagepetersen. Log gaussian cox processes. *Scandinavian Journal of Statistics*, 25(3):451–482, 1998. ISSN 03036898, 14679469. URL `http://www.jstor.org/stable/4616515`.

M. N. M. Van Lieshout. *Theory of spatial statistics: a concise introduction*. CRC Press, Taylor & Francis Group, is an imprint of Taylor & Francis Group, an informa business, Chapman & Hall, Boca Raton, 2019. ISBN 9780429052866 9780429627033.

Mats Rudemo. Statistics of Imaging. February 2020.

Ottmar Cronie and Marie-Colette Lieshout. A non-model-based approach to bandwidth selection for kernel estimators of spatial intensity functions. *Biometrika*, 105(2):455–462, 2018. ISSN 0006-3444.

Adrian Baddeley. spatstat.core. `https://github.com/spatstat/spatstat.core/blob/6c80ceb9572d03f9046bc95c02d0ad53b6ff7f70/R/bw.CvL.R`. URL `https://github.com/spatstat/spatstat.core/blob/6c80ceb9572d03f9046bc95c02d0ad53b6ff7f70/R/bw.CvL.R`. original-date: 2017-01-02T04:54:22Z.