Estimating intensity function of Poisson Point Processes: Review and Illustrations

Md. Anees Parwez (BS 1704)   **Bhaskar Ray** (BS 1717)   **Souhardya Sengupta** (BS 1726)

Indian Statistical Institute, Kolkata

Abstract

In this article, we review techniques for estimating intensity function of Poisson Point Processes. We review parametric as well as non-parametric methods for the same and also consider extension of parametric methods to include the effect of spatially varying covariates on the intensity function. A detailed discussion of approaches to the problem of bandwidth election for kernel estimators is presented. Finally, we apply these methods on simulated and real-life datasets.

1 Introduction

Spatial statistics refer to the sub-domain of statistical analysis concerned with analysis of spatial data. The data in this case has a spatial relevance, and we are concerned with all the general questions, we address in usual statistical analysis. A special case of the same is the problem of estimation of the intensity of a Poisson Point Process. A Poisson Point process is a special case of point process, where occurrences satisfy complete randomness and the number of points occurring on a set follows Poisson with mean equal to the measure of that set.
We, in this article, will be concerned with estimation of intensity of Poisson point process, over mostly, 2 dimensional Euclidean space. However, the same would be applicable for higher dimensions with minimal changes. Poisson processes find application in varying fields including environment, medicine, image processing, etc.

We assume that we have data points

$$\boldsymbol{X}_1, \boldsymbol{X}_2, \dots \text{ from } PPP(\Lambda) \text{ on } \mathbb{R}^2,$$

where

$$d\Lambda(x,y) = \lambda(x,y)dxdy,$$

and

$$\lambda : \mathbb{R}^2 \longmapsto [0, \infty)$$

is the intensity function. We are concerned with the estimation of this intensity function. We take up this problem next.

As is the case with most of the domains of statistics, estimation here as well, can be broadly divided into two classes - the parametric and the non-parametric methods.

The parametric methods assume that the intensity belongs to a known class of parametric functions (Λ_θ), indexed by an unknown parameter $\theta \in \Theta$, except which, the form of the intensity function is known. The problem essentially is to estimate this θ, based on the observed data.
The non-parametric class of methods doesn't make any such assumptions, and hence is more flexible as compared to its parametric counterparts. These methods are however, *lazy*, in the sense that they do not require any training phase, but use the entire data set again and again to make predictions, making them more expensive computationally as compared to the former class. Also, hypothesis testing is quite involved in most of these cases.
Another class of methods try to explain the intensity using some covariates. The reader might find such a situation similar to the case of Generalized Linear Models. The core assumption is that, there lies a set of predictors that affect the intensity function, which in turn affects the observables, that is the occurrences of the PPP.
Thus to sum up, we discuss the following methods:

- Parametric Methods
- Non-Parametric Methods
- Methods involving covariates

The article is organised as follows. In the next section, we consider some parametric techniques of inference. In Section 3, we consider the non-parametric counterparts. We consider estimation based on covariates in Section 4. The following Section 5 provides examples illustrating the application of the methods developed. We briefly discuss edge effects in Section 6. All our implementations have been done using the `spatstat`[1] package in `R`. An introduction to the same and some brief details of our implementations are available in Appendix A.

2 Parametric Methods

Assume that we have iid observables $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{R}$ from $PPP(\lambda)$ on a set $\mathcal{S}$, where $\mathcal{R} \subset \mathcal{S}$, and λ refers to the intensity function. For the sake of simplicity, we will assume $\mathcal{S} \subset \mathbb{R}^p$, and also that we are interested in estimation of the intensity function in the set $\mathcal{R}$ only.
Moreover, assume that the intensity function takes the following parametric form:

$$\lambda(s) = \lambda(s; \theta), s \in \mathcal{S}, \theta \in \Theta. \tag{1}$$

We will try to estimate the unknown parameter θ.

There can be primarily the following two methods to collect data:

- The exact coordinates of the occurrence of the points
- We can, alternatively, assume that the set $\mathcal{R}$ can be divided into disjoint bins:

  $$\mathcal{R} = \bigcup_{i=1}^{k} \mathcal{R}_\rangle,$$

  and for each i, we can define

  $$N_i = \sum_{j=1}^{n} I(\boldsymbol{x}_j \in \mathcal{R}_i),$$

to be the number of points in the i^{th} bin. Another possible dataset can be $(N_1, \ldots, N_k)$.
We will resort to maximum likelihood estimation in both these cases, for estimation of θ.

To begin with, consider $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \overset{iid}{\sim} PPP(\lambda = \lambda(.;\theta)$, then the likelihood takes the form

$$\begin{aligned}
p(x_1, \ldots, x_n, N = n) =& p(x_1, \ldots, x_n | N = n) P(N = n) \\
=& \left(\prod_{i=1}^{n} \frac{\lambda(x_i)}{\int_S \lambda(s) ds} \right) \times exp\left(- \int_S \lambda(s) ds \right) \frac{\left(\int_S \lambda(s) ds \right)^n}{n!} \\
\propto& exp\left(- \int_S \lambda(s) ds \right) \times \prod_{i=1}^{n} \lambda(x_i).
\end{aligned}$$

, and thus, the log-likelihood (except the terms not depending on θ) has the form:

$$L(\theta) = - \int_{\mathcal{R}} \lambda(s; \theta) ds + \sum_{i=1}^{n} \log \lambda(x_i; \theta), \tag{2}$$

and hence, the ML estimate is giving by:

$$\widehat{\theta}_{ML} = \arg\max_{\theta} L(\theta). \tag{3}$$

Now, the maximization in Equation 3 can be done by setting score function (that is the derivative of the likelihood)to 0, that is $\nabla_\theta L(\theta) = 0$. This gives:

$$\sum_{i=1}^{n} \frac{1}{\lambda(\boldsymbol{x}_i; \theta)} \nabla_\theta \lambda(\boldsymbol{x}_i; \theta) = \int_{\mathcal{R}} \nabla_\theta \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} \tag{4}$$

Equation 4 can be then solved for θ, to obtain the ML estimate. Obviously we need to know the parametric form to be able to solve the equation, as a closed form solution for an arbitrary form is not available.

Now, let's consider the other case when we have data in the form of counts in the bins $(N_1, \ldots, N_k)$, in accordance to the notation we introduced earlier.
Note that for the i^{th} bin, the counts satisfy:

$$N_i \sim Poi\left(\int_{\mathcal{R}_i} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} \right)$$

Then the log-likelihood in this case takes the form:

$$L_{bin}(\theta) = - \sum_{i=1}^{k} \int_{\mathcal{R}_i} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} + \sum_{i=1}^{k} N_i \log \left(\int_{\mathcal{R}_i} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} \right),$$

which can be further simplified to

$$L_{bin}(\theta) = - \int_{\mathcal{R}} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} + \sum_{i=1}^{k} N_i \log \left(\int_{\mathcal{R}_i} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} \right) \tag{5}$$

Taking gradient gives us the equation:

$$\sum_{i=1}^{k} N_i \frac{\int_{\mathcal{R}_i} \nabla_\theta \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s}}{\int_{\mathcal{R}_i} \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s}} = \int_{\mathcal{R}_i} \nabla_\theta \lambda(\boldsymbol{s}; \theta) d\boldsymbol{s} \tag{6}$$

The solution to Equation 6, gives us the required estimate. Depending on the data at hand, any one of the Equations 4 or 6 can be used to obtain the ML estimate of θ.

2.1 GLM based methods

The similarity of spatial intensity estimation and Poisson regression is not a surprise. The same framework of estimation in GLMs can also be employed for intensity estimation of spatial Poisson point processes. The likelihood is exactly same as Equation 2. In addition, we assume that the intensity function is linked with the systematic component, which in turn is a linear combination of the spatial coordinates or a function of the spatial coordinates. That is, we write

$$g(\lambda(\boldsymbol{x}, \boldsymbol{\alpha})) = \boldsymbol{\alpha}^T \boldsymbol{b}(\boldsymbol{x}), \tag{7}$$

where g is a link function, and

$$\boldsymbol{b}(\boldsymbol{x}) = (b_1(\boldsymbol{x}), \ldots, b_k(\boldsymbol{x})),$$

is a collection of basis functions, we expand the coordinates into.
The rest of the fitting process is exactly the same as of GLM. Here as well, the log-link happens to be the canonical link. We can use IRLS/NR iterations for fitting the model.

This method holds a definite advantage over the method discussed in Section 2, in terms of interpretability. Besides, these methods hold a capability of a extension, where we might be able to include covariates in the regression other than the spatial coordinates. These covariate based-techniques are discussed in Section 4. For all these cases, we can use the usual GLM based tests for establishing the significance of coefficients as well as the goodness of fit of the model (X^2 or G^2).

Despite all these parametric frameworks, these methods suffer from the same disadvantages as any other parametric method does. We assume a parametric form for the intensity which unless in a very restrictive setting, is hard to predict. Thus, even the best fitted model in a parametric setting might not produce adequate fit. This calls for more flexible methods of analysis. In the next section, we turn into non-parametric alternatives of the methods discussed in this section.

3 Non-parametric Methods

The non parametric model allow more flexibility and do not require to predict a parametric structure. We discuss three commonly used non parametric methods: Splines, kernels and nearest neighbours. Of these, kernels are most commonly used and we explore this method in more details. One of the major hurdle in many non-parametric is tuning of hyperparamters, which is also discuss. In the next subsection, we begin with consideration of spline-based methods. In Section 3.2, we take up Kernel based estimation, and in Section 3.3, we consider k nearest neighbors based methods.

3.1 Splines

We consider the simple case of $PPP(\lambda)$ in $1-D$. This is an $IPP(\lambda)$ where rate $\lambda = \lambda(t)$ varies over time. The cumulative intensity function, $\Lambda(S) = \int_0^S \lambda(t)dt$. The intensity function, $\lambda(t)$, for a $IPP(\lambda)$ is assumed to be positive for all $t \in (0, S]$ and is continuous for almost every $t \in (0, S]$. The cumulative intensity function is to be estimated from k realizations of the $IPP(\lambda)$ on $(0, S]$, where S is a known constant.

Let $\{n_i\}_1^k$ be the number of observations in the i^{th}realization, $n = \sum_i n_i$ and let $t_{(l)}, t_{(2)}, .., t_{(n)}$ be the order statistics of the superposition of the k realizations. Define $t_{(o)} = 0$ and $t_{(n+l)} = S$.

Setting $\widehat{\Lambda_K(S)} = n/k$ yields a process where the expected number of events by time S is n/k. We fit a piecewise-linear estimator of the cumulative intensity function between the time values in the superposition:

$$\lambda(t_{(0)}) = 0$$
$$\lambda(t_{(n+1)}) = n/k$$
$$\lambda(t_{(i)}) = \frac{i}{n+1} \times \frac{n}{k}$$

for $i = 1, 2, \dots n$.
As shown in Figure 1 , we can interpolate to get the intensity as:

$$\widehat{\Lambda_k(t)} = \left[\frac{i}{n+1}\frac{n}{k}\right] + \frac{t - t_{(i)}}{t_{(i+1)} - t_{(i)}} \times \frac{n}{(n+1)k} \qquad t_{(i)} < t \leq t_{(i+1)} \tag{8}$$

for $i = 1, 2, \dots n$. In fact, writing this $\widehat{\Lambda_k(t)}$ as a step function estimator it can be shown that a

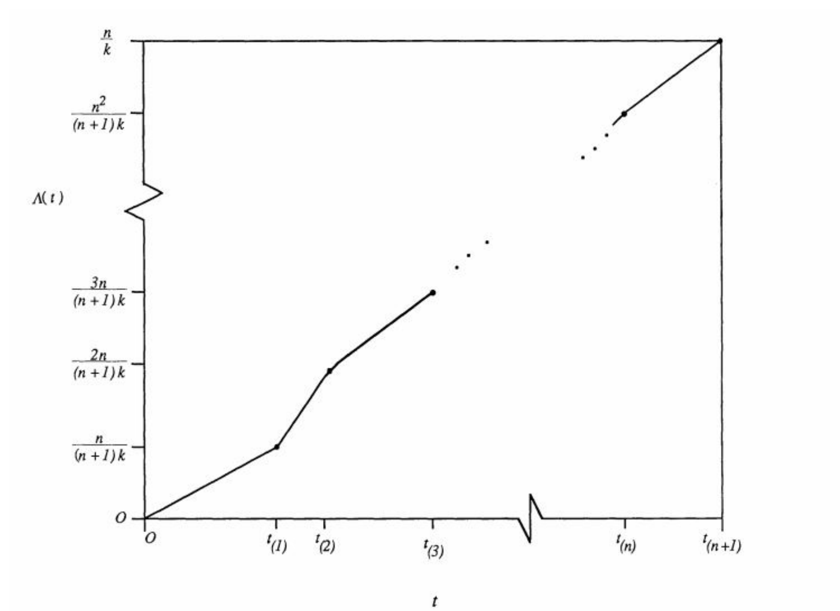


Figure 1: Piece wise Linear Estimate. Illustration borrowed from [7]

strong consistency result holds [7], i.e., $\lim_{k\to\infty} \widehat{\Lambda_k(t)} = \Lambda(t) \quad w.p.\ 1$. The consistency result in important for performing statistical inference on $\Lambda(t)$ based on $\Lambda_k(t)$.

In this method, for a given $t \in (0, S]$ we area able to find a i for which $t_i < t \leq t_{i+1}$. However, the concept of *betweeness* in higher dimension is not very clear. Hence, the idea of extending this method to spatial point process is not very immediate to us. We do not purse it any further as in most application kernels are used.

3.2 Kernel Based Estimates

Kernel-based estimates are one of the most popular methods of intensity estimation. The method is very similar to estimation of densities using kernels. However, one needs to make minute corrections depending on the problem at hand.

We will be considering kernels as weighing functions for different observations. To motivate the usage we start with the first ever proposed estimate of intensity, due to Berman and Diggle.

3.2.1 Motivation: The Berman-Diggle estimators and Beyond

Berman and Diggle, proposed a very intuitive estimator for the intensity of a Point process. This method, however, is not restricted to estimation of Poisson intensity only, and would apply to any general point process.

Suppose we have, $\boldsymbol{X}$, a spatial point process with an intensity $\lambda(\boldsymbol{x})$ defined in $\mathbb{R}^2$ and $(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N)$ a realization of X observed on a bounded region $W \subset \mathbb{R}^2$. Consider a small ball of radius h about a point $\boldsymbol{x} \subset \mathbb{R}^2$. We might assume that the intensity is more or less uniform around that point. Then the expected number of points in that ball is given by

$$\begin{aligned} M(B(\boldsymbol{x};h)) = & \int_{B(\boldsymbol{x};h)} \lambda(\boldsymbol{z}) d\boldsymbol{z} \\ \approx & \lambda(\boldsymbol{x}) \int_{B(\boldsymbol{x};h)} d\boldsymbol{z} \\ = & \lambda(\boldsymbol{x}) vol(B(\boldsymbol{x};h)) \end{aligned}$$

Note that the observed number of points $N(B(\boldsymbol{x};h))$, is an unbiased estimate of $M(B(\boldsymbol{x};h))$. Thus, a natural estimate of the intensity at that point could be the average count of observations falling in the area, i.e.

$$\hat{\lambda}(x) = \frac{N(B(x,h))}{vol(B(x,h))} = \frac{N(B(x,h))}{\pi h^2} \tag{9}$$

Obviously, this intensity estimator suffers lots of disadvantages, the prominent of them being the fact that, with a relatively sparse pattern, most of the balls $B(\boldsymbol{z};h)$ with small h, wouldn't contain any of the occurrence points, while with large h, the assumption of uniform intensity won't hold good. To improvise over this, we shall consider an equivalent representation of Equation 9 :

$$\hat{\lambda}(x) = \sum_{i=1}^{N} k_h(x - X_i) \tag{10}$$

where, $k(z) = \mathbb{I}(z \in B(0,h))/vol(B(0,h))$.
A natural extension, to include influence of points outside the ball $B(\boldsymbol{x};h)$ can be to allow arbitrary kernels k in Equation 10 above.

3.2.2 Kernel intensity estimation for 1-D

In [4], Diggle shows that the kernel estimate is slightly non standard in sense that we divide the estimate in Equation 10 by an edge correction term, $\rho_h(x) = \int_W k_h(x-u)\ du$. We will discuss more about edge correction later in Section 6. Note that taking $\rho_h(x) = 1$ results in an estimator having form similar to that of the kernel density estimator. Thus, in a 1-D, he first suggested the kernel estimate of $\lambda(.)$ as:

$$\hat{\lambda}_h(y) = \frac{1}{p_h(y)} \sum_{i=1}^{n} h^{-1}\ k(h^{-1}(y - S_i)) \tag{11}$$

where $S_1, \ldots S_n$ are points from 1-D Poisson process. A successful implementation of this method dwells upon choosing an appropriate h parameter, the choice of kernel is of secondary importance.

In general, however, no simple recipe for the choice of bandwidth exists. We consider this for a stationary Cox process, with rate process $\{\Lambda(x) : x \in \mathbb{R}^2\}$ which has a realization $\lambda(x)$. The point process is thus, an inhomogeneous Poisson process with rate function $\lambda(x)$. We would choose that h which minimizes $MSE(h) = \mathbb{E}\left[\hat{\lambda}_h(y) - \Lambda(y)\right]$.

Let $\mathbb{E}(\Lambda(x)) = \mu$, $\mathbb{E}(\Lambda(x)\ \Lambda(y)) = \nu(|x-y|)$ and let $K(t) = \mu^{-2}\int_0^t \nu(u)du$ be the corresponding reduced second moment measure. Then, it can be shown that there is a closed form expression $MSE(h)$. Specifically,

$$MSE(h) = \nu(0) + \mu\{1 - 2\mu K(h)\}/(2h) + \{\mu/(2h)\}^2 \int_0^{2h} K(y)dy$$

In order to estimate the value of h which minimizes $MSE(h)$ we require estimates of μ and $K(h)$. The natural estimator for μ is n/T. An estimator for $K(t)$ is the one dimensional analogue of Ripley's estimator [10]. With these estimates we can numerically find h which minimizes $MSE(h)$. The implementation in R can be done using `bw.diggle` in `spatstat` package[1].

Unfortunately, this method doesn't work for a general kernel because a closed form expression of $MSE(h)$ couldn't be calculated. Though for Gaussian kernel we may use Silverman's thumb rule [11]. This applies to general kernel estimation procedure with gaussian kernel. An approximation of the optimal h which $AMISE$, results in choosing $h_{opt} = 1.06\sigma \times n^{-1/5}$. The sample standard deviation is a consistent estimate of σ and can be used to find optimal h. A more robust estimate is given by

$$h_{opt} = 0.9 \times min\left(\hat{\sigma}, \frac{IQR}{1.34}\right) \times n^{-1/5} \tag{12}$$

where IQR=Inter Quantile Range. It is a usual practice, to take $h_{opt} = n^{-1/5}$

3.2.3 Kernel intensity estimation for higher dimensions

For spatial point processes, kernel intensity estimation has been addressed assuming a scalar bandwidth parameter. If $\boldsymbol{x} \subset \mathbb{R}^2$, by change in density, $k_h(x) = h^{-d}k(h^{-d}x)$ Thus, the estimate in 11 is extended as:

$$\hat{\lambda}(\boldsymbol{x}) = \frac{1}{\rho_h(\boldsymbol{x})}\sum_{i=1}^{n} h^{-2}\ k\left(\frac{x - X_i}{h}\right) \tag{13}$$

Under certain regularity conditions, it can be shown that the first two moments of 13 exists. So one way to estimate bandwidth parameter is through minimizing $MISE(h)$:

$$\begin{aligned} MISE(H) &= \mathbb{E}\left[\int_W \ (\lambda_{0,H}(x) - \lambda_o(x))^2 \cdot dx\right] \\ &= \int_W bias(x,H)\cdot dx + \int_W var(x,H)\cdot dx \end{aligned} \tag{14}$$

However since this expression implicitly depends on the unknown intensity, it has limited applications.

Unlike the one dimension case, here even for uniform kernel the expression for evaluating $MSE(h)$ is very challenging though there is few literature to help minimize instead an equivalent quantity $M(t)$ [2].

Another commonly used procedure is 'Likelihood cross-validation' [3] which is to numerically maximizing he leave-one-out cross-validation log likelihood, $PPK_k(h)$:

$$PPL_k(h \mid X, W) = \sum_{i=1}^{N} log(\hat{\lambda}(X_i \mid h, X \setminus X_i)) - \int_W \hat{\lambda}(u \mid h, X) \cdot du \tag{15}$$

With $\rho(u) = h^{-2} \int_W k(\frac{x-u}{h})du$, and uniform kernel,, we can simplify this equation,

$$PPL_k(h \mid X, W) = \sum_{i=1}^{N} \log\left(\frac{N(B(X_i, h) \cap W) - 1}{vol(B(X_i, h) \cap W)}\right) - \int_W \frac{N(B(u, h) \cap W)}{vol(B(u, h) \cap W)} du$$

The optimal h is taken as the one which minimizes $PPL(h \mid X, W)$. In implementation this is done numerically. This method is commonly used in R, using the command `bw.ppl` in the R-package `spatstat`.

3.2.4 Kernel intensity estimation with general bandwidth matrix

The assumption of scalar bandwidth parameter can be quite restrictive specially for anisotropic and highly inhomogeneous point processes. A more general approach dwelling on the philosophy of bivariate kernel intensity estimation, is to allow any symmetric and positive-definite bandwidth matrix, H. Thus the kernel intensity estimate is:

$$\begin{aligned} \hat{\lambda}_H(x) &= \frac{1}{p_H(x)} \sum_{i=1}^{N} k_H(x - X_i) \\ &= p_H(x)^{-1} |H|^{-1/2} \sum_{i=1}^{N} k\left(H^{-1/2}(x - X_i)\right) \end{aligned} \tag{16}$$

Observe that $H = h^2 \times I_2$ results in 13. The estimation of optimal H is not very immediate. A method based on bootstrap bandwidth selection for the consistent kernel intensity is proposed in [5] We discuss this method briefly.

The idea justifying the use of bandwidth selection procesure is based upon consistency arguments. Denote $m = \int_W \lambda(x)dx$. Given that the number of events of an inhomogeneous Poisson point process, $N \sim Poi(m)$, we can establish:

$$f(x) = \frac{\lambda(x)}{\int_W \lambda(x) \cdot dx} = \frac{\lambda(x)}{m}$$

where, $f(x)$ denote bivariate density of the event locations. Denote $\lambda_o(x) = f(x)$, then a kernel density estimate of $\lambda_o(x)$ is:

$$\begin{aligned} \hat{\lambda}_{o,H}(x) &= \frac{\hat{\lambda}_H(x)}{N} \mathbb{I}(N \neq 0) \\ &= [p_H(x)N]^{-1} |H|^{-1/2} \sum_{i=1}^{N} k\left(H^{-1/2}(x - X_i)\right) \mathbb{I}(N \neq 0) \end{aligned} \tag{17}$$

We are now interested in estimating the intensity $\lambda_o(x)$. It can be shown that 17 is infact a consistent estimator of bivariate density $\lambda_o(x)$ [6]. Consistency requires an asymptotic framework which can be of two types: increasing-domain and increasing-intensity. Since in increasing-domain changes occur only on boundary of W, we work with the latter, so that asymptotically $m \to \infty$. Apart from this we assume that $n \to \infty$, and all entries of H tend to 0, such that $n|H|^{1/2} \to \infty$

To obtain H, we thus need to choose H that minimizes $MISE(H)$ between $\lambda_o(x)$ and $\hat{\lambda}_{o,H}$.In this case too, the $MISE(H)$ doesn't have a closed form expression, but Asymtotically, the dominant term ($AMISE(H)$) can be has closed form.

$$MISE(H) = AMISE(H) + o(A(m)\,|H|^{-1/2} + tr^2(H))$$

here $A(m) = \mathbb{E}\left[\frac{1}{N}\mathbb{I}(N > 0)\right] \to 0 \text{ as } m \to \infty.$

But $AMISE(H)$ needs to be estimated since it depends on unknown intensity $\lambda_o(x)$. Given the points $(X_1, \ldots X_n)$, a kernel intensity estimator (17) $\lambda_{0,G}(x)$ with pilot bandwidth matrix G is obtained. Defining 'Bootstrap kernel intensity estimator' as $\lambda^*_{o,H}(x)$,

$$\lambda^*_{o,H}(x) = \frac{\hat{\lambda}^*_H(x)}{N^*}\,\mathbb{I}(N^* \neq 0) \tag{18}$$

Like the previous case, the consistency of $\lambda^*_{o,H}(x)$ for $\lambda_{0,G}(x)$ can also be established [5]. Further, when certain consistency requirements are placed on G, a more powerful result that can be proved is that $AMISE^*(H)$ in the bootstrap case, is a consistent estimator of $AMISE(H)$. Thus, it is sufficient to minimize $AMISE^*(H)$, to obtain an the optimal bandwidth matrix H. $AMISE^*(H)$ involves no unknown term other than H, so we proceed numerically using Newton- Rapson to minimize $AMISE^*(H)$ and choose $arg \min\limits_H \; AMISE^*(H)$ as our optimal bandwidth matrix.

Simulation of methodologies discussed are presented in Section 5 using Gaussian kernel. Also, using data on bombing in World War 2, the comparison between non-parametric and parametric method are explored.

3.3 K Nearest Neighbour based estimation

The idea behind nearest neighbour estimators is very similar to the notion introduced in 3.2.1. Let $(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N)$ is realization of $X \sim PPP(\lambda(\boldsymbol{x}))$ observed on a bounded region $W \subset \mathbb{R}^2$. For a point $\boldsymbol{x} \in \mathbb{R}^2$, let $d_k(x)$ be distance of the k^{th} nearest neighbour of $\boldsymbol{x}$. k-NN intensity estimation assumes intensity in the ball $B(\boldsymbol{x}, d_k(x))$ is constant. Thus the density at point $\boldsymbol{x}$ is:

$$\frac{k}{\pi \times (d_k(x))^2} \tag{19}$$

To tune the hyper parameter k, we can try to maximize leave-one-out log likelihood as we did for kernel case in Section 3.2.3. This we suggest as leave one out approach is fairly standard procedure in any knn based approach.

Another standard procedure for $k - nn$ is to use the rule of thumb:

$$k_{opt} = \sqrt{n}$$

This is commonly used in classification domain, can might be extended here. By default the `nndensity` command in R-package `spatstat` uses this thumb rule.

The major advantage of nearest neighbour estimator is its quick computation and ease of parameter tuning. However this advantage is under weighed , by the breakdown of uniform intensity in highly inhomogenous process for large k. The simulations are presented in 5. Observe that knn gives a almost 0 intensity to large chunk of region. The reason might be the intensity at a point $\boldsymbol{x} \in \mathbb{R}^2$, it completely determined by the distance of the $k-th$ nearest neighbour (as numerator is k for all points).

4 Methods involving covariates

Let us assume $\boldsymbol{X}$ is a generic spatial $PPP(\lambda)$ on a set $\mathcal{S} \subseteq \mathbb{R}^p$. As of yet, we have assumed parametric and non-parametric forms for the intensity function, λ based on the point of observation. Now, we shall extend it further to include features of the points in the model. This is especially useful in real-life datasets where occurrences in a natural environment depend on various biological or geographical features etc. Before proceeding further, let us introduce certain notations and definitions that we shall follow throughout.
Let $\mathcal{R} \subseteq \mathcal{S}$ be a bounded observation window and $\boldsymbol{x} = \{x_1, \ldots, x_n\} \in \mathcal{R}$ be a realization of $\boldsymbol{X}$ on the set $\mathcal{R}$.
For a subset $B \subseteq \mathcal{S}$, denote the number of points observed in B by

$$N(B) = |\boldsymbol{X} \cap B|$$

Now, as $\boldsymbol{X}$ is a spatial $PPP(\lambda)$,

$$N(B) \sim \text{Poisson}(\Lambda(B))$$

where

$$\Lambda(B) = \int_B \lambda(u)du$$

where we may interpret $\lambda(u)du$ to be the probability that exactly one point falls in an infinitesimally small region containing the point u and of area du.

4.1 Modelling the covariates

For a point $u \in \mathcal{S}$, let

$$\boldsymbol{z}(u) = (z_1(u), \ldots, z_k(u))^\top$$

denote the vector of covariates, measured at u.
We want to model the intensity function, $\lambda(u)$ with respect to $\boldsymbol{z}(u)$ for each $u \in \mathcal{S}$. We shall consider the **Poisson log-linear model** for the purpose. Let

$$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_k)^\top$$

denote the vector of unknown regression parameters. We assume that the locations of events are realized from a modulated Poisson process. Parametrizing λ on u and $\boldsymbol{\beta}$, the model is

$$\log(\lambda(u; \boldsymbol{\beta})) = \boldsymbol{\beta}^\top \boldsymbol{z}(u); \quad u \in \mathcal{S} \tag{20}$$

which is similar to that proposed in Equation 1.

4.2 Estimation of model parameters

Similar to Equation 2, we derive the kernel of the log-likelihood as

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{n} \log(\lambda(x_i; \boldsymbol{\beta})) - \int_{\mathcal{R}} \lambda(u; \boldsymbol{\beta}) du \tag{21}$$

$$= \boldsymbol{\beta}^\top \sum_{i=1}^{n} \boldsymbol{z}(x_i) - \int_{\mathcal{R}} \exp(\boldsymbol{\beta}^\top \boldsymbol{z}(u)) du \tag{22}$$

The likelihood equations are given by

$$\boldsymbol{\psi}_{\mathcal{R}}(\boldsymbol{\beta}) = \mathbf{0}$$

where

$$\boldsymbol{\psi}_{\mathcal{R}}(\boldsymbol{\beta}) = \frac{1}{|\mathcal{R}|} \left[\sum_{i=1}^{n} \boldsymbol{z}(x_i) - \int_{\mathcal{R}} \boldsymbol{z}(u) \exp(\boldsymbol{\beta}^\top \boldsymbol{z}(u)) du \right] \tag{23}$$

The maximum likelihood estimator, $\hat{\boldsymbol{\beta}}$ is obtained by solving 23. Rathbun, Shiffman and Gwaltney (2007)[9] proposed to divide the likelihood equations by the area $|\mathcal{R}|$ of the study region for technical reasons related to investigations of the large-sample properties of the proposed estimator, which we shall not delve into any further.

It was demonstrated by Rathbun and Cressie (1994)[8], under increasing domain asymptotics, that $\hat{\boldsymbol{\beta}}$ is consistent, asymptotically efficient and asymptotically Gaussian with asymptotic variance-covariance matrix,

$$\mathrm{var}(\hat{\boldsymbol{\beta}}) = (\mathcal{I}(\beta))^{-1}$$

where

$$\mathcal{I}(\beta) = \int_{\mathcal{R}} \boldsymbol{z}(u) \boldsymbol{z}^\top(u) \exp(\boldsymbol{\beta}^\top \boldsymbol{z}(u)) du \tag{24}$$

is the Fisher information matrix.

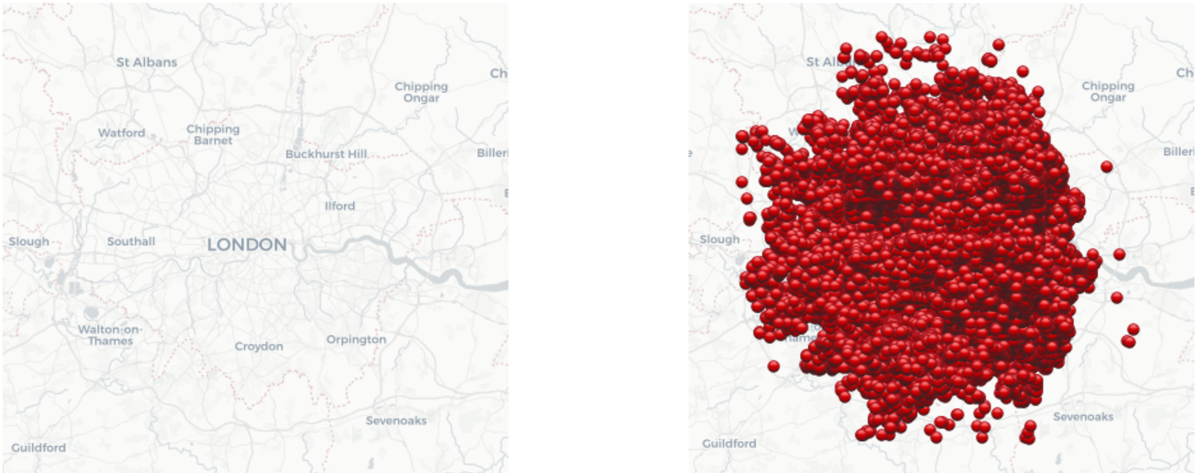
5 Illustrative Examples

In this section, we will present some examples illustrating the methods developed above.

5.1 Example 1: The London Blitz

The Blitz was a war campaign by Germany during the World war II against Britain. The Germans conducted air-raids in London for 56 continuous days dropping a total of 9000 bombs and claiming many lives. Look into Figure 2, for a plot of the bombing points in London. The red balls mark the bombing locations. The dataset has been collected from [12].

We will assume that the bombs fall according to a spatial Poisson Point Process and will try to estimate it's intensity function. For our purposes, we have taken the bombing locations in a window. We have also centered the data so as to make the mean zero in both the coordinates. A plot of the same is available in Figure 3. This furnishes an example where we can guess a form of the intensity function. We can see from Figure 3 that, the intensity is quite high at the center and falls down as we go away radially. This is also intuitive from the context, as we would be expecting more bombs to fall in the heart of London and as we go away, the intensity should decrease. This, inspires us



(a) Without bombs (b) With bombs

Figure 2: The map of London with and without the Blitz Bombings

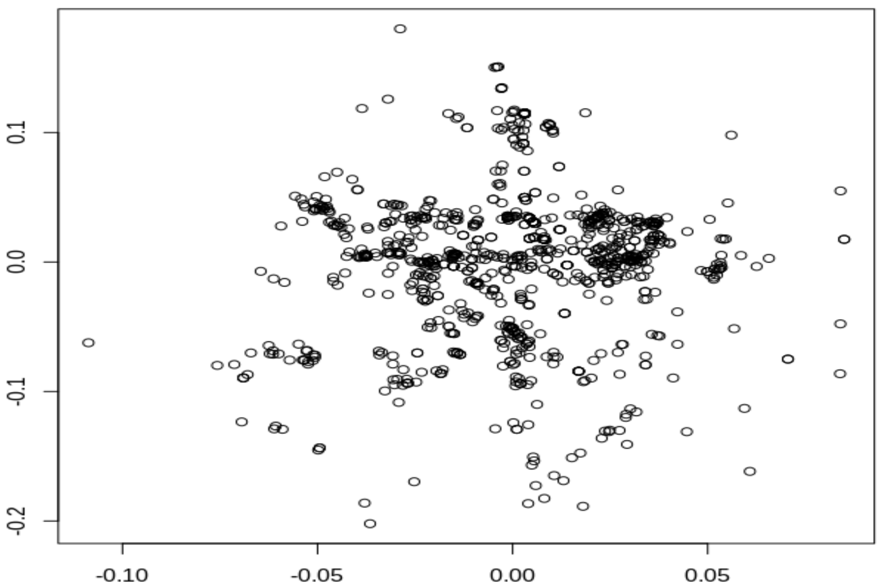


Figure 3: Scatter-plot of the bombing locations (centered)

to assume radially decreasing forms of the intensity function.
Let's assume the following general second degree independence form for the intensity function:

$$\log \lambda(x, y; \boldsymbol{\beta}) = \beta_0 + \beta_1 x^2 + \beta_2 y^2 + \beta_3 x + \beta_4 y,$$

where we have used the canonical log-link function (in accordance with Section 2.1). Ideally, the coefficients of β_1 and β_2 must be negative, but well let ML estimation take care of that. The estimated coefficients, and hence, the form of the intensity function turn out to be (after fitting using Fisher Scoring),

$$\lambda(x, y) = exp\left(11.426 - 627.159x^2 - 152.157y^2 + 0.149x + 0.04y\right). \tag{25}$$

Next, we can use the usual Wald tests to assess the significance of the coefficients. The values of these Z-statistics are listed in Table 1.

Coefficients	Z-values
$\boldsymbol{\beta}_0$	243.493^*
$\boldsymbol{\beta}_1$	-20.750^*
$\boldsymbol{\beta}_2$	-20.14^*
$\boldsymbol{\beta}_3$	0.128
$\boldsymbol{\beta}_4$	0.07

Table 1: The z-values of the coefficients. The coefficients significant at 5% level are marked $*$

We see that the effects of x and y are insignificant, implying that the part inside the kernel of the exponential is purely quadratic. So we drop the term, to get the intensity function:

$$\lambda(x,y) = exp\left(11.42573 - 627.25862x^2 - 152.16394y^2\right). \tag{26}$$

A heat map of the estimated intensity is available in Figure 4. As we can see, the heat map matches

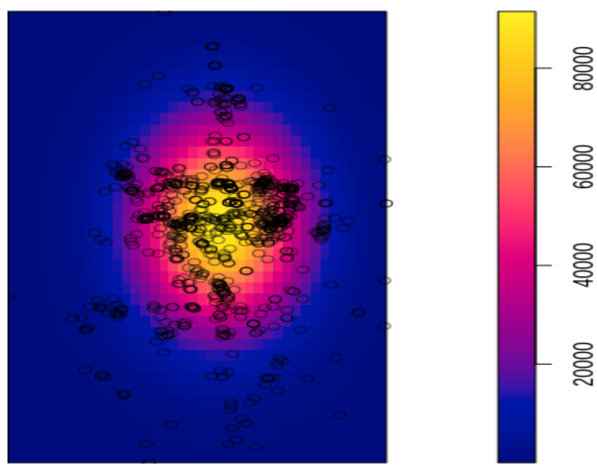


Figure 4: Heat map of the estimated intensity function of Equation 26

with our intuition of bombing patterns.

Spatial Poisson intensity estimation bears lots of similarity with log-linear models for contingency tables. The row/column effects of contingency tables translates to the corresponding coordinates effects, as a continuous analogue. We can address questions like independence of the coordinates (for example, whether the contours are spherical), etc.

Non-parametric estimation: Let's assess the performance of kernel estimator with Gaussian kernel in this case. The plot of the intensity function is available in Figure 5. The bandwidth has been chosen using the rule of thumb. Clearly the intensity estimated by the kernel method matches our parametric guessed form. This, suggests towards the flexibility of non-parametric estimation.

5.2 Example 2: Simulation Experiments

Though parametric methods are useful and more accurate, when we know the form of the intensity function, we seldom have this liberty. Consider the following intensity for a 2 dimensional spatial

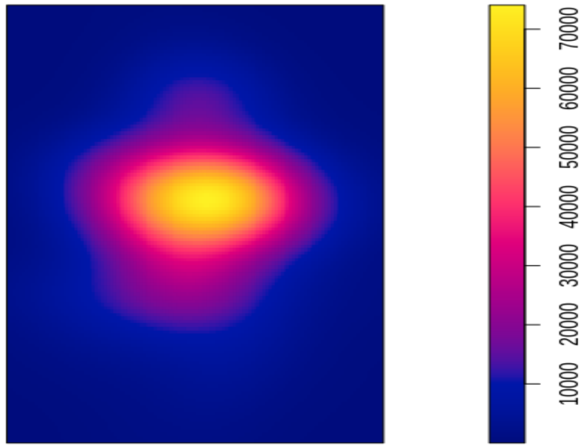
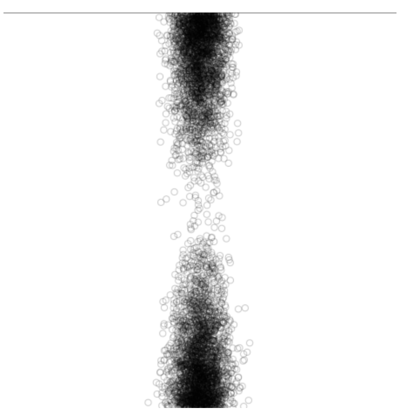


Figure 5: Intensity function for the kernel estimator for Example 1

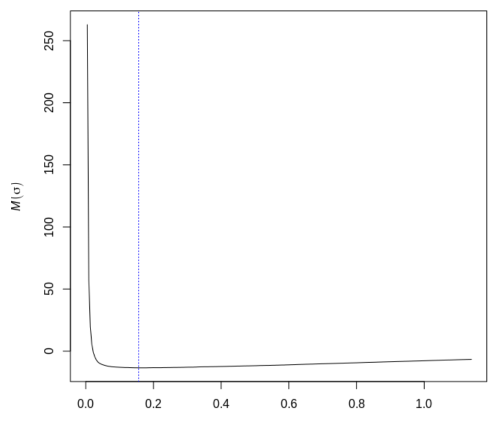
Poisson Process on $\mathbb{R}^2$:

$$\lambda(x, y) = 5e^{-x^2/2}(1 + y^2). \tag{27}$$

Figure 6a gives a plot of the same in the window $[-10, 10]^2$. The reader might try to guess the form



(a) Occurrences in $[-10, 10]^2$

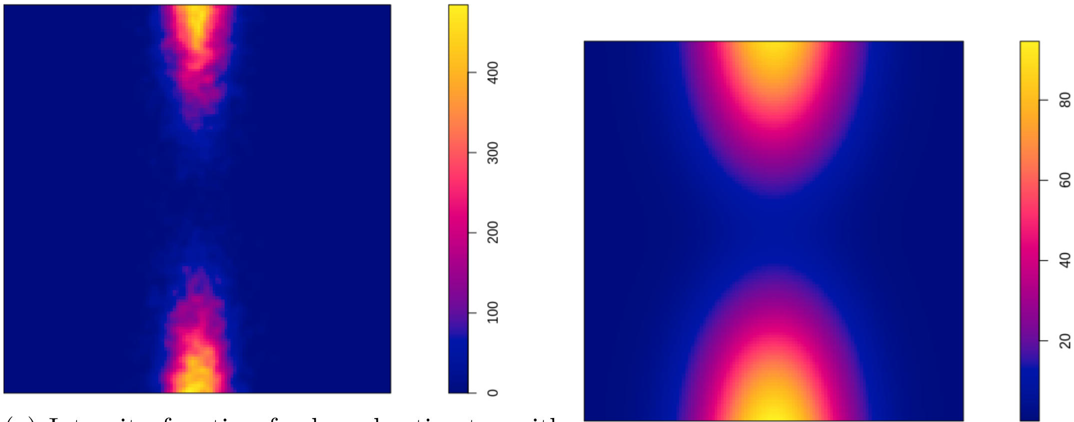


(b) CV MSE values. The blue line denotes the optimal. ($\sigma = h/2$)

Figure 6: Dataset for simulation 1

of intensity function (27), just by looking into the figure, to be convinced that guessing forms are not easy. These situations are apt examples of cases where non-parametric methods outperform parametric methods by a large margin. We will demonstrate the estimated intensity for both, the kernel estimator as well as the nearest neighbor estimator.

- **The Kernel Estimator:** Just to demonstrate the process of finding bandwidth using the MSE minimization method of Section 3.2.2, we give a plot of the MSE values against h in Figure. 6b We use the Gaussian Kernel. The optimal value obtained is $h = 0.3128$ for MSE minimization. The smoothed out intensity plot is given in Figure 7a. The intensity plot where the bandwidth has been chosen using rule of thumb is given in Figure 7b. The fit looks



(a) Intensity function for kernel estimator with bandwidth selected using MSE value minimization.

(b) Intensity function for the kernel estimator

Figure 7: Results of the kernel intensity estimation, bandwidth selected using rule of thumb

reasonable for both the cases. However, it is smoother for the rule of thumb while coarser for the former. This, is noticeable difference, and might suggest overfitting in the former. Now, we assess the performance of the k-NN intensity estimation (Section 3.3) on the same data on the same window.

- **kNN intensity estimation:** We used cross validation to determine the optimal value of k. It turned out to be $\boldsymbol{k} = 78$.
  The heat map of the estimated intensity is available in Figure 8. The fit here as well looks

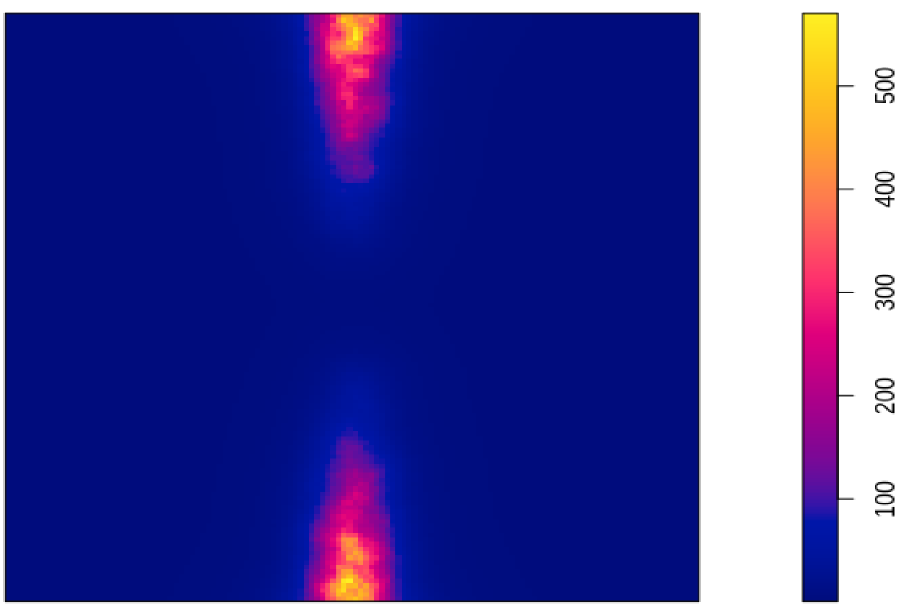


Figure 8: Heat map of estimated intensity using kNN estimation

reasonable. However, there is an evident difference between the fitted intensities in Figure 7b and that of 8. The intensity fitted using the kernel method is smoother as compared to that fitted using the kNN method. This is obvious, a kNN estimation is bit more discrete than the kernel estimator.

5.3 Example 3: The `bei` Dataset

The `bei` dataset was obtained from the `spatstat` package in `R`. The data consists of *locations of 3605 trees* of the species *Beilschmiedia pendula* (Lauraceae) in a 1000 by 500 metre rectangular sampling region in a tropical rain forest, accompanied by covariate data giving the *elevation* (altitude) and *slope* of elevation in the study region. The scatter-plot of the location of the trees is given in Figure 9 and heat maps representing elevation and slope over the region are provided in Figure 10.

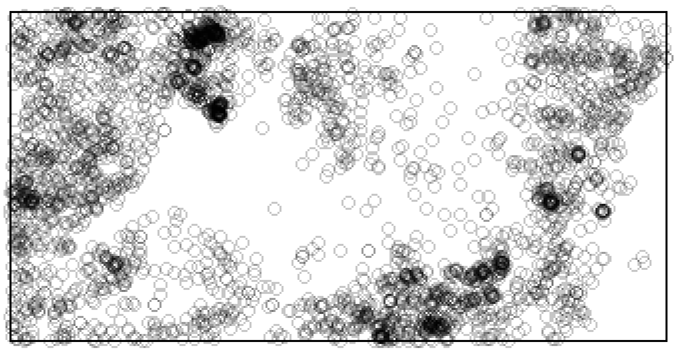


Figure 9: Scatter-plot of the location of the trees

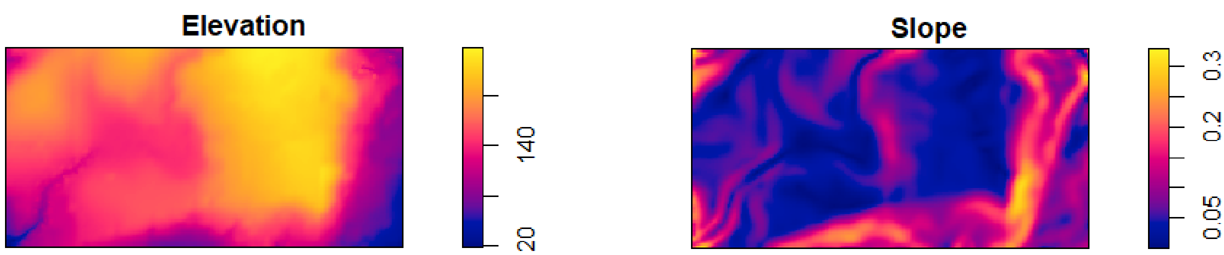


Figure 10: Heat maps of covariates

Let us assume that the trees were born at locations according to a spatial Poisson Point Process and that the intensity function of the process is affected by the covariates, elevation and slope at each point.
We assume the Poisson log-linear model for modelling $\lambda(u)$ on $\boldsymbol{z}(u) = (z_{elev}(u), z_{slope}()u)$ as discussed in Section 4. The model is

$$\log(\lambda(u)) = \alpha + \beta_{elev} z_{elev}(u) + \beta_{slope} z_{slope}(u)$$

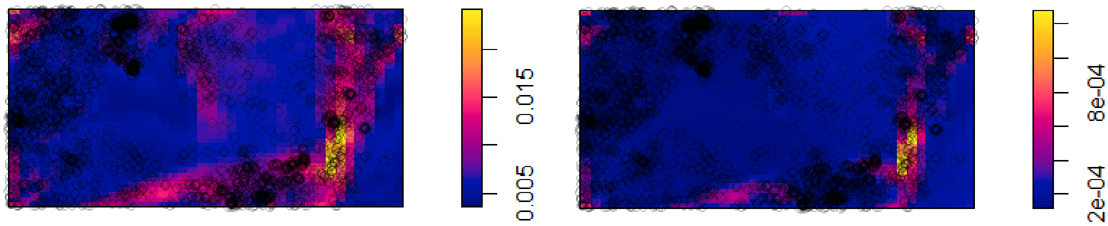
Fitting the model, the estimates and the corresponding Wald Z-statistic are summarized in Table 2. Hence, our fitted model is

$$\log(\lambda(u)) = -8.5635522 + 0.02143995 z_{elev}(u) + 5.8464668 z_{slope}(u)$$

It is to be noted that each of the coefficients is significant at 0.1% level of significance. The fitted values of the intensity function, $\lambda(u)$ and the estimated standard errors at different locations, u, are represented by heat maps in Figure 11.

Coefficients	Estimates	SE	Z-values
$\boldsymbol{\beta}_0$ (Intercept)	-8.56355220	0.341113849	-25.104675
$\boldsymbol{\beta}_{elev}$ (Elevation)	0.02143995	0.002287866	9.371155
$\boldsymbol{\beta}_{slope}$ (Slope)	5.84646680	0.255781018	22.857313

Table 2: The estimates and Z-values of the coefficients.



(a) Fitted values of Intensity function

(b) Estimated SE of Intensity function

Figure 11

For testing significance of the model, we use the G^2 test statistic whose value is obtained as

$$G^2 = 472.8126 \text{ on 2 d.f.}$$

The p-value is thus obtained as $\approx 2.138229 \times 10^{-103}$ which indicates towards the significance of the model.

6 Edge Effects

As we already discussed earlier, for example, say in Section 2, that the region where we want to carry out estimation, is smaller than the region where the PPP occurs. In such cases, it happens that the effects of points lying close to the boundary are under-estimated. This phenomena is illustrated in Figure 12. In the figure, the red dots represent the occurrence of PPP and the set S is the set within which we want to do inference. The boundary has been denoted by ∂S. Now,

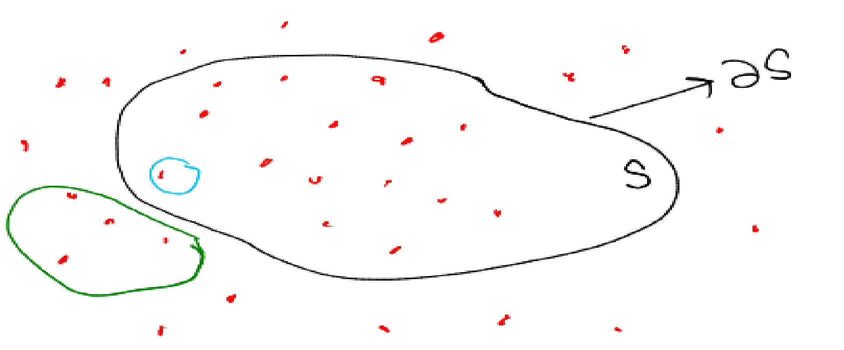


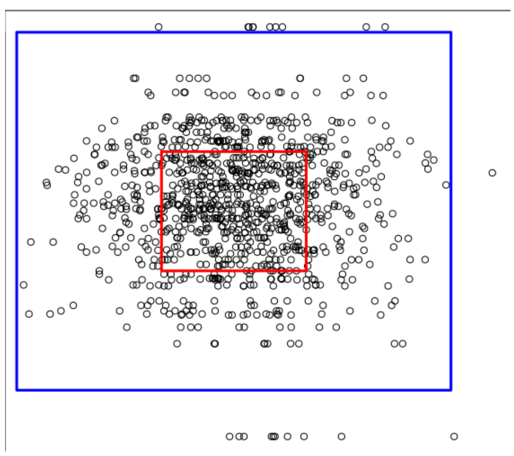
Figure 12: Demonstration of edge effect

consider the point encircled in cyan. If we were to solely restrict on points falling within S, then a clear conclusion would have been that the intensity near the point is quite low. However, note that the process occurred in the entire space and not just the set S. Thus, if we were to consider the nearest neighbors of the marked point, leaving aside the boundary, then due to the points encircled in green we would have concluded that the intensity at cyan-circled point is much higher than what we concluded before.

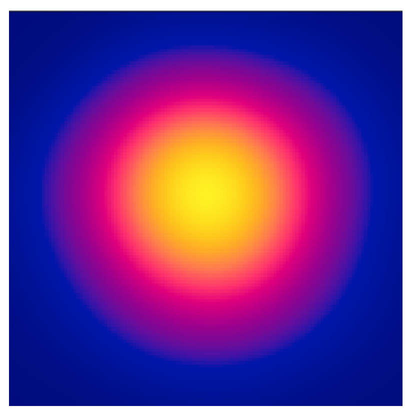
Thus doing estimation only on the basis of points falling inside the set leads to misleading results. In fact, it produces an under-bias. This is due to the fact that we are completely ignoring the effect off the outside of the set on the inside which usually casts on the edge. That's why the name *edge effects*.

We now demonstrate the consequences of ignoring edge effects through an experiment. Consider Figure 13a for scatterplot of the data. We have two windows, the blue one is $[-3,3]^3$ and the red one correspond to $[-1,1]^2$. The intensity estimated using kernel estimation for $[-3,3]^2$ window is available in Figure 13b.

Now, consider the estimated intensities without and with edge effect corrections for the $[-1,1]^2$



(a) Scatter plot of the data



(b) Estimated intensity in the $[-3,3]^2$

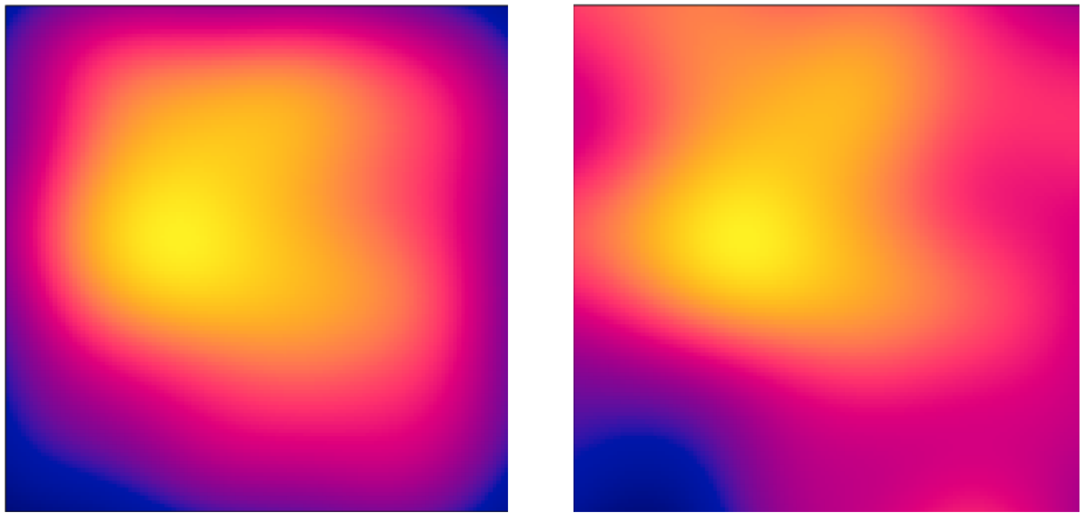
Figure 13: Data for demonstration of edge effect

window. They are available in Figure 14.

Notice the difference in the boundaries of Figure 14a and 14b. For the former, we see the upper boundaries depicted using cooler shades while for the later, we observe warmer shades. We definitely should be expecting warmer shades as from Figure 13 it is clear that the intensity in the window $[-1,1]^2$ is quite high. Without edge correction, we completely disregard the points outside the boundary hence, we get an underbiased estimate at the edges.

7 Conclusions

In this article, we presented an introductory review on estimation of intensity of Poisson Point processes. Though, we used the two and one dimensional Euclidean space as our setting for presenting the methods, it should be notes that these methods are readily generalizable to higher dimensional Euclidean spaces as well. We considered estimation in broadly two categories - Parametric and



(a) Uncorrected intensity plot

(b) Corrected intensity plot

Figure 14: Intensity heat maps in $[-1,1]^2$ window.

Non-parametric.

For parametric techniques, we reviewed the general likelihood based parametric estimation for the cases where we have data in the form of exact coordinate locations and also when the data is available as bin counts. We then considered a bit restrictive setting of using GLM for estimation of intensity functions. The same setting of Poisson regression could be used here. We considered the London Blitz bombing coordinates dataset to demonstrate parametric estimation.

For the non-parametric methods, we reviewed splines for one dimensional case and kernel methods, in general. We reviewed the Berman-Diggle estimates and their essential extensions. We reviewed various methods for bandwidth selection including MSE minimization, te Silverman's rule of thub and the more flexible likelihood cross validation. We also considered general bandwidth matrix for intensity estimation in higher dimensions, which covers the anisotropic cases as well, and presented outline of a bootstrap method of minimising the AMISE for tuning of the same. We also considered Cucala's event density in this regard. Besides, we also considered kNN method for intensity estimation. We illustrated non-parametric estimation using a simulated example of our own and also the bombing data considered before. We also illustrated the MSE minimization method of choosing bandwidth.

Lastly, we considered inclusion of other covariates in estimation of intensity. We specifically restricted to the GLM setup and used the `bei` dataset for illustration of the methods. Lastly, we considered problem of having edge effects and demonstrated with an example that non correcting for it might result in erroneous results.

References

[1] Adrian Baddeley and Rolf Turner. spatstat: An R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42, 2005.

[2] Mark Berman and Peter Diggle. Estimating weighted integrals of the second-order intensity of a spatial point process. *Journal of the Royal Statistical Society. Series B (Methodological)*, 51(1):81–92, 1989.

[3] Ottmar Cronie and M. Lieshout. Bandwidth selection for kernel estimators of the spatial intensity function. 11 2016.

[4] Peter Diggle. A Kernel Method for Smoothing Point Process Data. *Journal of the Royal Statistical Society Series C*, 34(2):138–147, June 1985.

[5] Isabel Fuentes-Santos, Wenceslao González-Manteiga, and Jorge Mateu. Consistent smooth bootstrap kernel intensity estimation for inhomogeneous spatial poisson point processes. *Scandinavian Journal of Statistics*, 43(2):416–435, 2016.

[6] Cucala L. Espacements bidimensionnels et donnÉesentachÉes d'erreurs dans l'analyse desprocessus ponctuels spatiaux. *(HAL)*, 2006.

[7] Lawrence M. Leemis. Nonparametric estimation of the cumulative intensity function for a nonhomogeneous poisson process. *Management Science*, 37(7):886–900, 1991.

[8] Stephen Rathbun and Noel Cressie. Asymptotic properties of estimators for the parameters of spatial inhomogeneous poisson point processes. *Advances in Applied Probability*, 26:122–154, 03 1994.

[9] Stephen L. Rathbun, Saul Shiffman, and Chad J. Gwaltney. Modelling the effects of partially observed covariates on poisson process intensity. *Biometrika*, 94(1):153–165, 2007.

[10] Brian D. Ripley. *Spatial statistics*. Wiley, 1981.

[11] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, London, 1986.

[12] in collaboration with the National Archives University of Portsmouth and funded by JISC. Mapping the london blitz. *Bomb Sight*.

A The `spatstat` Package

The `spatstat` package in R provides implementations of various spatial statistics methods. All the experiments done in this article made use of this package. In this section we give a brief introduction to the methods available in the package and used in this article.

- **Creating `ppp` objects:** Most of the functions in the package relating to spatial Poisson process makes use of the object type `ppp` standing for Poisson Point Patterns. A $2 \times n$ data frame can be converted to an object of this type using the `ppp()` function. For that we need to set a window where we want to do estimation. The following command can set a $[a, b] \times [c, d]$ rectangular window:

  ```
  W = owin(xrange = c(a,b), yrange = c(c,d))
  ```

  See the function `owin()`'s documentation for options of specifying non-rectangular windows. A data frame `dat` can be converted into `ppp` type in the above window, using the following codes:

```
dat.pp = ppp(dat, win = W) \#can use this object from here on
```

- **Simulating a Poisson Point Process:** For simulation of Poisson point processes, the function `rpois.pp()`can be used. It returns an object of `ppp` type. As an example, consider the following line of codes for simulation of the data in Section 5.2:

```
lamb<-function(x,y){
return(5*exp(-x^2/2)*(1+y^2))
}
ex2pattern = rpois.pp(lambda = lamb, nsim = 1) #nsim is the number of
    simulations
plot(ex2pattern) #for plotting the pattern
```

- **GLM fitting:** The following code can be used to fit the intensity of the form in Section 5.1:

```
fit = ppm(dat.pp,~I(x^2)+I(y^2)+x+y)
summary(fit)
plot(fit)
#a sequence of plots including pattern, intensity and SE
```

- **Non-parametric techniques:** Use the following codes for nonparametric estimations.

```
mse = bw.diggle(dat.pp) #for bandwidth selection using MSE
    minimisation
plot(mse) #plot of bandwidth vs mse values
den = density.ppp(dat.pp) #fits the intensity using gaussian kernel
    and bandwidth by rule of thumb
plot(den)
den.bw = density.ppp(dat.pp,sigma = bw.diggle) #mse minimization for
    bandwidth
#see documentation for other kernels
plot(den.bw)
den.nn = nndensity(pat.pp) #knn density with k equal to sqrt of n
plot(den.nn)
```

- **Modelling with covariates:** The following code can be use to model the `bei` dataset and other datasets for which covariate values are available over the entire region of study.

```
#Visualization

plot(bei) #bei gives a ppp class variable cotaining coordinates of
    the trees
plot(bei.extra$elev) #bei.extra$elev gives image class variables for
    elevation
plot(bei.extra$grad) #bei.extra$elev gives image class variables for
    slope

#Model fitting and analysis of fit

fit <- ppm(bei~bei.extra$elev+bei.extra$grad) #log-linear model
summary(fit)
```

```
plot(fit) #plots fitted intensity function and estimated SE one-by-
    one

fit_null <- ppm(bei~1) #null model with constant intensity function
ll1 <- logLik(fit) #log-likelihood & df for our model
ll0 <- logLik(fit_null) #log-likelihood &df for null model
G.sq = -2*as.numeric(ll0-ll1) #G^2 test statistic
pval = pchisq(G.sq, df = 2, lower.tail = F) #p-value
```