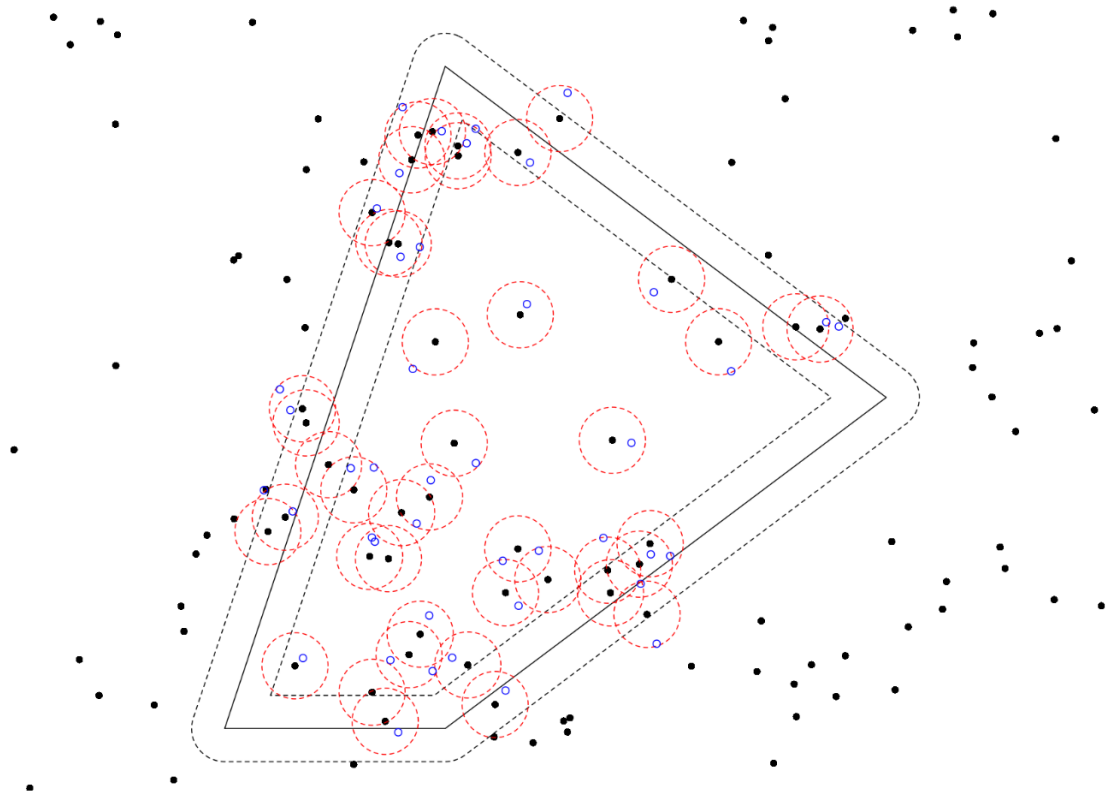


Master Thesis

Data Augmentation for Point Process Learning

Mathematics

MATHIS ROST

DIVISION OF APPLIED MATHEMATICS AND STATISTICS

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Degree project report 2024

Data Augmentation for Point Process Learning

Mathis Rost



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Applied Mathematics
Chalmers University of Technology,
University of Gothenburg
Gothenburg, Sweden 2024

The development and application of two different data augmentation methods to fit artificially generate point process data.
Mathis Rost

© Mathis Rost, 2024.

Supervisor: Ottmar Cronie, Julia Jansson
Examiner: Moritz Schauer

Master Thesis 2024
Department of Applied Mathematics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone +49 15228071401

Cover: Area interaction process shifted according to a uniform distribution.

Typeset in LaTeX
Gothenburg, Sweden 2024

Inference of point processes with help of data augmentation
Mathis Rost
Department of Applied Mathematics and Statistics
Chalmers University of Technology, University of Gothenburg

Abstract

This thesis introduces and evaluates ideas for the use of data augmentation in the area of Point Process Learning. Motivated by the regularizing effect of training with augmented data sets, we create a follow-up work to the paper "A cross-validation-based statistical theory for point processes" by Cronie et al. [2023]. We develop methods for applying data augmentation to point process data. We discuss the possibilities of augmenting the existing process with additional data points generated by a noise process or by moving the already existing points in space. The developed methods are applied to common point process models, like the hard-core process and the area interaction process. The augmented data is then used for inference. The performed simulation study, where different discussed options are applied, shows promising results. The regularizing effect of data augmentation can be observed and thus motivates further investigation in to this topic.

Keywords: Point Process Learning, Data Augmentation, Gibbs Processes, Papangelou Conditional Intensity, Hard-Core Process, Area Interaction Process.

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Ottmar Cronie and Julia Jansson, for their unwavering support, and insightful feedback throughout the entire process of researching and writing this master thesis. Their expertise, dedication, and encouragement have been instrumental in shaping the direction of my work and enhancing the overall quality of this project.
I am also thankful to my co-supervisor, Volker Schmidt from Ulm, for his valuable contributions, support and expertise that enriched the depth and breadth of my research.
Furthermore, I am grateful for the opportunity to write my thesis via the ERASMUS program, which provided me with a unique and enriching academic experience. The cultural exchange and diverse perspectives encountered during this period have undoubtedly contributed to the overall development of my research skills and academic understanding.

Mathis Rost, Gothenburg, February 2024

Contents

1 Introduction **3**
1.1 Motivation . . . 5
1.1.1 Ridge Regression . . . 5
1.1.2 Neural Network . . . 7
1.2 Bias Variance Trade-off . . . 9

2 Point Processes Theory **13**
2.1 General Theory about Point Processes . . . 13
2.1.1 Definition of a Point Process . . . 13
2.1.1.1 A Point Process as a random measure . . . 13
2.1.1.2 A Point Process as a Point Configuration . . . 15
2.1.2 General Characteristics about Point Processes . . . 16
2.1.3 Poisson Processes . . . 18
2.1.4 Gibbs Processes . . . 19
2.1.4.1 Energy Function . . . 19
2.1.4.2 Exponential representation . . . 20
2.1.5 Intensities of Gibbs Processes and the GNZ Formula . . . 21
2.1.6 Examples of Gibbs Processes . . . 27
2.1.6.1 Strauss Process . . . 27
2.1.6.2 Hard-Core Process . . . 31
2.1.6.3 Area Interaction Process . . . 32
2.2 Inference for Gibbs Processes . . . 35
2.2.1 Takacs-Fiksel estimator . . . 36
2.2.1.1 Maximum Pseudo-Likelihood Estimation . . . 36
2.3 Point Process Learning (PPL) . . . 38
2.3.1 Method for Point Process Learning . . . 40
2.3.1.1 Cross-Validation for Point Process Learning . . . 41
2.4 Data Augmentation in Point Process Learning . . . 42

3 Adding Noise to Point Processes **47**
3.1 Properties of Superpositions of Point Processes . . . 48
3.1.1 Conditional Intensities of Superposition . . . 49
3.1.2 Repulsive and Attractive Superpositions . . . 53
3.2 Adding Noise to Validation and Training Data . . . 55
3.2.1 Definition of the Method . . . 56
3.2.2 Remarks on the Method . . . 56

3.2.2.1 Estimator Family for the noised Process 56
3.2.2.2 Analysis of the Estimator 58
3.3 Noising the Test Function 61
3.3.1 Interpretation 62
3.3.2 Definition of the Method 63
3.4 Other Methods for Inserting Noise 63
3.4.1 Noise in the Training Set 64
3.4.2 Noise in the Validation Set 65
3.5 Summary 66

4 Simulation Study for the Additional Noise Method 67
4.1 Experimental Setup 68
4.1.1 Hard-Core Process 69
4.1.2 Area Interaction Process 73
4.2 Results 78
4.2.1 Analysis of the Hard Core Process 78
4.2.2 Analysis of the Area Interaction Process 80

5 Noising Data with Shifting 83
5.1 Motivation 83
5.1.1 The Distribution of the Noise 84
5.2 Method Description 86
5.3 Statistical Properties 87
5.3.1 Interpretation 93

6 Simulation Study for the Shifting Method 97
6.1 Experiments 97
6.1.1 Hard-Core Process 99
6.1.2 Area Interaction Process 103
6.2 Results 107
6.2.1 Analysis of the Hard-Core Process 107
6.2.2 Analysis of the Area Interaction Process 108

7 Conclusion 109
7.1 Future Work 110

1
Introduction

In the realm of statistics, a point process represents a departure from the traditional random sample, encompassing a diverse array of random variables, not necessarily identically independently distributed. Point patterns, emanating from these processes, serve as valuable tools for modeling diverse phenomena and behaviors. This includes modeling events like fires [Aragó et al., 2016], earthquakes [Ogata, 1998], tree locations [Picard et al., 2009] and disease cases [Kelly et al., 2019]. In one-dimensional space, these processes normally capture temporal events, while in higher dimensions, they unveil intricate spatial patterns and correlations.
Understanding the underlying distribution of a point process and how they behave, is a key challenge. Often, we are limited to a handful or even only one specific point pattern, and the availability of a likelihood function is rare, especially for complex point processes [Lieshout, 2019]. Therefore, unlike traditional statistical methods, including classical independent and identically distributed (iid) statistics, the estimation of the distribution with maximum likelihood is often not feasible [Lieshout, 2019].
Hence, in order to understand the data-generating phenomenon, we typically have to resort to other statistical methods when inferring the distributions of the processes. This can then potentially be used in combating pandemics, preventing forest fires, model tree distributions and predicting earthquakes.
The literature already contains several approaches on how distributions of point processes can be inferred with. Recently, a promising method called "Point Process Learning" was introduced by Cronie et al. [2023], which is based on a novel approach to cross-validation for point processes. This method avoids overfitting and leads to better results. Yet, despite its potential, this approach remains relatively unexplored and warrants further investigation. A rises the interesting question: Are there other typical idea or methods from machine learning, which we can combine with Point Process Learning from Cronie et al. [2023] and can we create regularizing methods for improving the method of Cronie et al. [2023] even further?
A natural and interesting method that can be associated with Point Process Learning is data augmentation. Data augmentation is typically used in the field of deep learning, where it is used, for example, for training with additional, slightly modified image data. The regularizing effect of this idea is clearly visible and the results, regarding the performance, are very promising [Shorten and Khoshgoftaar, 2019]. In the field of point process statistics, however, this technique is new and unexplored, which is another motivation behind this work.
On the theoretical front, properties resulting from data augmentation in point processes need exploration. Key properties, such as changes in point process intensity

with augmented points, or shifts in variance and bias of an estimator, depend on the nature of the manipulated or added data.
As far as we know, there is currently no work that uses data augmentation for learning the distribution of point processes. Therefore, there are no approaches yet on how to train with augmented data. However, there is work that investigates the sampling of artificial data. These methods are based on thinning a base process and were introduced by Alie et al. [2023] and in prior work by Adams et al. [2009]. For our work, we will also use other methods to augment our data.
In this thesis, we will first give a general overview of the field of point processes with a focus on Point Process Learning in Chapter 2. We will then discuss in Chapter 3 and 5 ways in which data augmentation can be applied to point processes. These possibilities will be explored theoretically. At the end of each of those chapters in Chapter 4 and Chapter 6, we will conduct experiments for learning with augmented data with help of a simulation study.

1.1 Motivation

In the realm of data science, data augmentation has emerged as a game-changer, revolutionizing machine learning across diverse disciplines [Shorten and Khoshgoftaar, 2019, Hernández-García and König, 2020, van Dyk and Meng, 2001]. This technique involves strategically enriching datasets through variations, fostering improved model performance and adaptability.
Data augmentation's key advantage lies in its ability to amplify dataset diversity, enabling models to better handle real-world complexities [Hernández-García and König, 2020]. In computer vision, it enhances image recognition by introducing variations like rotations and flips, ensuring models grasp diverse scenarios. In natural language processing, textual data augmentation refines language models' understanding of nuances, leading to more accurate results in tasks such as sentiment analysis [Shorten and Khoshgoftaar, 2019].
Data augmentation does give us additional training samples, it however does not directly improve the quality of the dataset, since we can not directly sample from the underlying distribution of the dataset. We might even disturb the underlying distribution by contributing new generated data points. However, we might be able to reduce overfitting by increasing the variety of the dataset. Typically, the augmented data is generated based on certain heuristics proposed in order to ensure resemblance with the original data. This technique is also a form of regularization and can also be achieved by inserting pure noise into the dataset, as we see in the upcoming examples.

The two following examples show different use cases of data augmentation. These will be useful later on, when we develop methods using data augmentation for point process interference.

1.1.1 Ridge Regression

We will focus on a simple method. More precisely, our starting point is a linear regression, where we have data $(X, Y) = \{x_i, y_i\}_{i=1}^n$, with $x_i \in \mathbb{R}^{\mathrm{d}}$ sampled i.i.d.. We thus have the classical linear model relationship

$$Y = w^T X + U, \tag{1.1}$$

where $U = \{u_1, \cdots, u_n\}$ is an error term with $\mathbb{E}[u_i] = 0$. For estimating the coefficients w we can minimize the L_2-loss, i.e.

$$\hat{w} = \arg\min_{w \in \mathbb{R}^{\mathrm{d}}} \|Y - w^T X\|^2. \tag{1.2}$$

By optimizing (1.2) we minimize the mean squared error (MSE). The proposed method can however be sensitive to outliers and tends to overfit in certain situations [Chatterjee and Hadi, 1986]. A common way to tackle this problem is using a loss function with a regularization term included. A popular choice is so-called ridge regression, where the optimal coefficient vector is obtained according to

$$\hat{w} = \arg\min_{w \in \mathbb{R}^{\mathrm{d}}} \|Y - w^T X\|^2 + \lambda \|w\|^2,$$

where λ is a hyperparameter which adjusts the penalty term. A well selected λ can prevent the method from overfitting and lead to better model performance in some cases.

Remark. The idea of a penalty term is not only applicable in linear regression but can also be used, for example, in kernel regression or neural networks to regulate the weights between neurons.

We now show that adding noise to our data has the same effect as ridge regression for our model. Consider the vectors

$$\epsilon = \{\epsilon_1, \cdots, \epsilon_n\} \in \mathbb{R}^{d \times n},$$

where each ϵ_i is a randpm vector, sampled i.i.d. from a distribution with zero mean and a fixed variance, i.e.

$$\mathbb{E}[\epsilon_i] = 0 \text{ and } \mathbb{E}[(\epsilon_i^{(j)})^2] = \sigma^2 \quad , i \in \{1, \cdots, n\}, \quad j \in \{1, \cdots, d\}. \tag{1.3}$$

The covariance matrix for each ϵ_i is therefore given by $\sigma \mathcal{I}_d$. We now noise the data X by adding the noise vector and the data vector, i.e.

$$X^* = X + \epsilon.$$

By applying the L_2-loss to the noised data we get,

$$\begin{aligned} \|Y - w^T X^*\|^2 &= \|Y - w^T(X + \epsilon)\|^2 \\ &= \|Y - w^T X + w^t \epsilon\|^2 \\ &= \|Y - w^T X\|^2 + 2\langle Y - w^T X, w^T \epsilon \rangle + \|\epsilon w\|^2. \end{aligned}$$

We can therefore approximate ridge regression by repeatedly adding noise terms, $\epsilon^{(1)}, \cdots, \epsilon^{(K)}$ and use the fact that for large K it holds that

$$\frac{2}{K} \sum_{i=1}^{K} \langle Y - w^T X, w^T \epsilon^{(i)} \rangle = 2\langle Y - w^T X, w^T \frac{1}{K} \sum_{i=1}^{K} \epsilon^{(i)} \rangle \approx 2\langle Y - w^T X, 0 \rangle = 0.$$

Applying Equation (1.3) to the derived equation we get

$$\mathbb{E}\left[\|Y - w^T X^*\|^2\right] = \|Y - w^T X\|^2 + \sigma^2 \|w\|^2,$$

which means that the error term of the L_2-regression with noise is distributed around the error term of the ridge regression with $\lambda = \sigma^2$ and will therefore have a similar effect, when considering the balanced loss

$$\hat{w} = \arg\min_{w \in \mathbb{R}^d} \frac{1}{K} \sum_{i=1}^{K} \|Y - w^T X_i^*\|^2 \approx \|Y - w^T X\|^2 + \sigma \|w\|^2.$$

An illustration of the described effect can be seen in Figure 1.1.

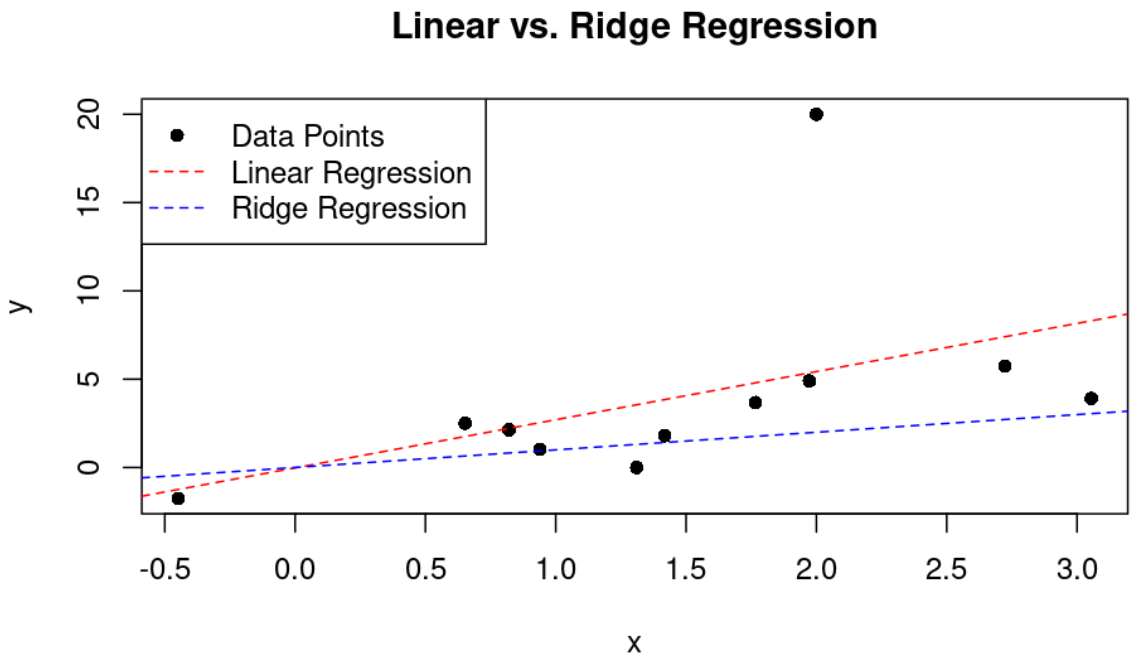


Figure 1.1: Approximated ridge regression with average weight w for minimizing MSE of 20 different noised data sets $X_1^*, \cdots, X_{20}^*$. The data contains an outlier, which affects the weight vector resulting from the MSE minimization (red line). When noising the data the optimal solution is not so sensitive to outliers. Due to the penalty term, the norm of w is lower, which leads to a weaker increase in the regression line (blue line).

1.1.2 Neural Network

The by far most common area where data augmentation is used is in the field of deep learning [Shorten and Khoshgoftaar, 2019]. As an example of how beneficial data augmentation can be, we will train a neural network for image classification with and without augmented data and compare the results.
First, we look at some examples of how images can be augmented. For this experiment we make use of three different techniques. We can rotate the image, we can also adjust the brightness of an image and as a third option we also allow the image to be compressed along the x- and y- axis.
A new augmented data sample can now be created from a combination of the three described adjustments.



Figure 1.2: Images taken from tensorflow data sets by Abadi et al. [2015]. The first image was rotated with an angle of 30 degree. The second has an adjusted brightness and the third image is compressed along the x axis. The new augmented data points can be used for training.

For our example, we use a neural network with an architecture based on convolutional layers combined with max pooling layer, which is a typical setup for image classification. A more detailed description of the used neural network would go beyond the scope of this thesis. The network is first trained with the original data set of 1024 images. The dataset consists of images of dogs and cats and thus has two classes, which we try to predict.
Then the same network is trained with additional augmented data. In the end, we compare the performance of the models. The performance of the different setups can be observed in Figure 1.3.

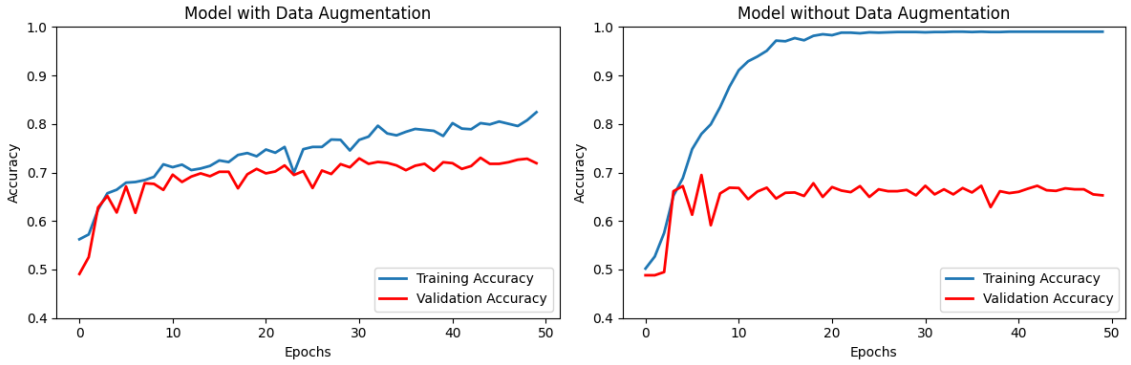


Figure 1.3: On the left we have training and validation accuracy with respect to the amount of trained epochs of the model trained with augmented data. The training accuracy is not too high, which indicates that the model is not overfitting. The right picture shows the accuracy of the same model but trained without augmented data. The model overfits and this leads to a worse performance than the accuracy of the model trained with augmented data.

In Figure 1.3 we illustrate that data augmentation can lead to better performance in deep learning. This was especially benefiting for a small training data set, since without data augmentation we quickly reached the point, where the model already knew all the training data points by heart.

1.2 Bias Variance Trade-off

In the previous section we talked about the regularizing effect of data augmentation. Regulation means that the variance of our model will be reduced, which means that our training algorithm is not too sensitive to the training data and will therefore not overfit the data. An example for a lower variance of a model is observable in Figure 1.1, where the resulting regression line will not be as sensitive to outliers, if we add a regularization term. The consequence is however that the bias of the model might increase, since it holds that

$$\|y - \hat{w}_{reg}^T X\|^2 \geq \|y - \hat{w}^T X\|^2,$$

with

$$\hat{w}_{reg} = \arg\min_{w \in \mathbb{R}^d} \|y - w^T X\| + \lambda \|w\|^2 \quad \text{and} \quad \hat{w} = \arg\min_{w \in \mathbb{R}^d} \|y - w^T X\|. \tag{1.4}$$

The regression line resulting from the L_2 regression will not be that steep, i.e.,

$$\text{Var}\left[\arg\min_{w\in\mathbb{R}^d}\|y - w^T X\| + \lambda\|w\|^2\right] \leq \text{Var}\left[\arg\min_{w\in\mathbb{R}^d}\|y - w^T X\|\right], \quad \lambda > 0.$$

This effect is called the bias variance trade-off. The effect plays a crucial role in machine learning [Belkin et al., 2019]. If the variance is too high, the model is overfitted, whereas too high a bias results in underfitting. It is therefore important to find a trade-off that balances the terms so that we get a good performance [Belkin et al., 2019].
In general, it's hard to express this trade off of bias and variance. In case of a MSE as a loss function we can however decompose it in terms of variance and bias.

Theorem 1.2.1. *Let X and Y be to metric spaces and let f describe a relation between X and Y with*

$$y = f(x) + \epsilon \text{ with } y \in Y \text{ and } x \in X,$$

where ϵ is a random variable with mean zero and variance σ^2. Let f^* be a function resulting from an algorithm trained with the data D. Then we can decompose the MSE into,

$$\mathbb{E}[(y - f^*(x; D))^2] = \sigma^2 + \text{Var}[f^*(x; D)] + \text{Bias}[f^*(x; D)]^2, \tag{1.5}$$

where σ is an error term, which can not be removed resulting from the variance of ϵ.

Proof. The proof is based on the classical decomposition of the second moment, i.e.,

$$\mathbb{E}[X^2] = \text{Var}[X] + \mathbb{E}[X]^2.$$

We first decompose the MSE into

$$\begin{aligned}\mathbb{E}[(y - f^*(x))^2] &= \mathbb{E}[y^2 + f^*(x)^2 - 2yf^*(x)] \\ &= \mathbb{E}[y^2] + \mathbb{E}[f^*(x)^2] - 2\mathbb{E}[yf^*(x)] \\ &= \text{Var}[y] + \mathbb{E}[y]^2 + \text{Var}[f^*(x)] + \mathbb{E}[f^*(x)]^2 - \mathbb{E}[2yf^*(x)].\end{aligned}$$

It holds that $\mathbb{E}[y] = \mathbb{E}[f(x) + \epsilon] = \mathbb{E}[f(x)] + 0 = f(x)$. We can again rearrange the equation to

$$\begin{aligned}\mathbb{E}[(y - f^*(x))^2] &= \text{Var}[y] + \text{Var}[f^*(x)] + (f(x) - \mathbb{E}[f^*(x)])^2 \\ &= \text{Var}[y] + \text{Var}[f^*(x)] + \mathbb{E}[f(x) - f^*(x)]^2.\end{aligned}$$

Substituting the definition of bias and knowing that $\text{Var}[y] = \sigma^2$, we get

$$\mathbb{E}[(y - f^*(x))^2] = \sigma^2 + \text{Var}[f^*(x)] + \text{Bias}[f^*(x)]^2.$$

□

Note that the statement of Theorem 1.2.1 is only based on the decomposition of the second moment.
In an optimal case, we use data augmentation to get closer to the sweet spot between underfitting and overfitting by balancing the trade-off between variance and bias. By using the regularizing effect of data augmentation we can reduce the variance of the model at the cost of a potential higher bias. An illustration can be found in figure. 1.4 Note that also restricting the function class $\mathcal{F}$ can reduce the variance of the model. A more limited choice of parameters can therefore also have a regularizing effect. This is often used in deep learning, where a model with fewer parameters is preferred over a more complex model [Belkin et al., 2019].

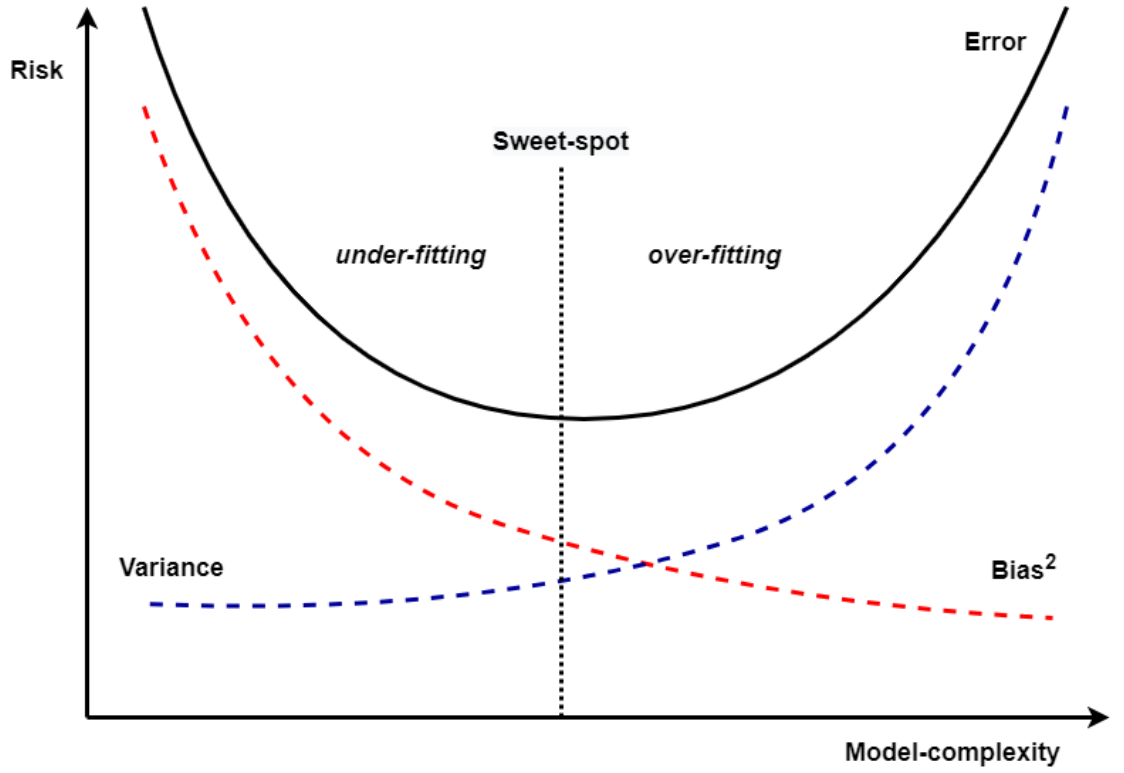


Figure 1.4: Trade-off between bias and variance. In machine learning we often try to find the sweet spot for the best model performance.

2

Point Processes Theory

This chapter summarizes the most important definitions and properties about point processes, with the goal of giving a brief overview of the theory of point processes. The main focus is "Point Process Learning" introduced in the paper "A cross-validation-based statistical theory for point processes" by Cronie et al. [2023], which is the theoretical base of the work of this thesis.

2.1 General Theory about Point Processes

For the general theory of point processes we mostly take the definitions from the book "Theory of spatial statistics: A concise introduction" by Lieshout [2019]. We will however also shed light into aspects which are not in the book. We therefore also use the chapter "Introduction to the Theory of Gibbs Point Processes" by Dereudre [2019] and the book "An introduction to the theory of point processes: volume I: elementary theory and methods" and "An introduction to the theory of point processes: volume II: general theory and structure" by Daley and Vere-Jones [2008], Daley et al. [2003].

2.1.1 Definition of a Point Process

Let S be a (complete separable metric) space with metric $d(\cdot,\cdot)$ and minimal σ-algebra $\mathcal{S}$, which is equipped with a suitable (σ-finite and locally finite) reference measure $A \mapsto |A| = \int_A \, dx$, $A \subset S$. For convenience one may think of $(S, \mathcal{S})$ as a Euclidean space with the associated Borel sigma algebra, i.e. $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$. We denote the ball with radius r and center $x \in S$, by $B_r(x) = \{y \in S | d(x,y) < r\}$.

In the literature, there are two common ways to define a point process [Grandell, 1977]. One focuses on a point process defined as a random measure, while the other one describes a point process as a random variable in a space of point patterns. For completeness, we present both definitions in our theory part. There is, however, a one-to-one correspondence between these two definitions.

2.1.1.1 A Point Process as a random measure

Given a measurable space $(\Omega, \mathcal{F})$, to define a point process we first need the definition of a transition kernel.

Definition 2.1.1 (Transition Kernel)**.**
A function $\kappa : \Omega \times \mathcal{S} \to [0, \infty]$ is called a transition kernel, or simply kernel, from Ω to S, if the following conditions hold:

- *For any $A \in \mathcal{S}$ the mapping*

$$\omega \to \kappa(\omega, A), \quad \omega \in \Omega$$

  is a measurable function.
- *For every fixed $\omega \in \Omega$ the mapping*

$$A \to \kappa(\omega, A), \quad A \in \mathcal{S}$$

  is a measure on $(S, \mathcal{S})$.

The definition of transition kernel is the basis for the definition of a random measure, which is just a transition kernel where the origin space is a probability space.

Definition 2.1.2 (Random Measure)**.**
When $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space we call a kernel $X : \Omega \times \mathcal{S} \to [0, \infty]$ with

$$X(\cdot) = \kappa(\cdot, \cdot) \tag{2.1}$$

a random measure on $(S, \mathcal{S})$.

For the definition of a point process we want to restrict the type of measures, since a point pattern restricted to an area always has integer valued amount of points.

Definition 2.1.3 (Properties of Measures)**.**
A (non-negative) measure μ on S is called

- *locally finite if $\mu(A) < \infty$ for all bounded sets $A \in \mathcal{S}$,*
- *a counting measure if $\mu(A) \in \mathbb{N} \cup \{0, \infty\}$ for all $A \in \mathcal{S}$,*
- *simple if $\mu(\{x\}) \in \{0, 1\}$ for all $x \in S$.*

The last property is only important to define so-called simple point processes. A simple point process is a process where each value is unique almost surely. In this thesis we only consider locally finite simple point processes. Locally finite means that we only expect a finite amount of points in a bounded area almost surely.
We denote the space of simple counting measures on S by

$$\mathcal{M}_S = \{\mu : \mathcal{S} \to \mathbb{N}_0 \cup \{\infty\} | \mu(\{x\}) \in \{0, 1\} \ \forall x \in S, \mu(A) < \infty \ \forall \text{ bounded } A \in \mathcal{S}\}.$$

Note that this space only contains measures with the defined properties in Definition 2.1.3.

Definition 2.1.4 (Simple (locally finite) Point Process)**.**
If X is a random measure, where the induced measure is locally finite and a counting measure, we call X a point process. If the induced measure is also simple we call X a simple point process.
Therefore a simple (locally finite) point process is a measurable function from $(\Omega, \mathcal{F}, \mathbb{P}) \to (\mathcal{M}_S, \mathcal{F}_{\mathcal{M}})$, where $\mathcal{F}_{\mathcal{M}}$ is a minimal chosen σ-algebra such that

$$\kappa(\cdot, A) : \mathcal{M}_S \to \mathbb{N}_0 \text{ with } \kappa(\mu, A) = \mu(A), \quad A \in \mathcal{S}.$$

We will denote by $X(A)$ the random variable which results from fixing $A \in \mathcal{S}$ and making draws $\omega \in \Omega$.
By the subspace $\mathcal{M}_S^W \subset \mathcal{M}_S$ we refer to the space of simple counting measures, where the support of an element is at most W i.e.,

$$\mathcal{M}_S^W = \{\mu \in \mathcal{M}_S |\ \mu(S \setminus W) = 0\}.$$

This is especially important if the process X only exists on an observation window W.

In case the simple and locally finite process X is only given in an observable window W then $X(\cdot)$ is a measurable function from Ω to $\mathcal{M}_S^W$. We then know that X has finitely many elements almost surely. We then can write $X = \{x_1, \ldots, x_n\}$ as a sum of dirac measures with,

$$X(A) = \sum_{i=1}^{n} \delta_A(x_i), \quad A \in \mathcal{S}.$$

2.1.1.2 A Point Process as a Point Configuration

Another but equivalent way to define a point process is by considering the space of all possible point configurations.

Definition 2.1.5 (Locally finite Point Configuration)**.**
The space $\mathcal{X}_S$ of locally finite point configurations or patterns consists of all $\aleph \subset S$ that place finitely many points in all bounded sets $W \subset S$.

Given $W \in \mathcal{S}$ bounded, from time to time we will consider

$$\mathcal{X}_S(W) = \{\aleph \subset W : \aleph \text{ is a finite set}\}$$

We now can describe a point process as an random variable in the previously defined space.

Definition 2.1.6 (Point Process)**.**
A point process is a measurable function from a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to the space $(\mathcal{X}_S, \mathcal{F}_\mathcal{X})$, where $\mathcal{F}_\mathcal{X}$ is a σ-field generated by the counting function

$$X(W) : \aleph \to |\aleph \cap W|.$$

We again call the process X simple if all values of X are unique almost surely. Therefore a point process X with realizations in $\mathcal{X}_S$ is a random locally finite configuration of points such that for all bounded sets $A \in \mathcal{S}$ the number of points of X that fall in A is almost surely a finite random variable which we shall denote by $X(A)$.

Remark. Note that the two definitions given for point processes are equivalent, since when taking an element $\mu \in \mathcal{M}_\mathcal{S}^W$, then it holds for the support of μ that

$$\text{supp}(\mu) \in \mathcal{X}_S(W).$$

Vice versa, the function $\nu : W \times \mathcal{X}_S \to \mathbb{N}_0$, defined as

$$\nu(A, \aleph) = \sum_{x \in X} \delta_A(x) \quad A \subset W,$$

is a measure in $\mathcal{M}_S^W$ for a fixed $\aleph$. therefore we sometimes switch between the two definitions depending on the purpose.

2.1.2 General Characteristics about Point Processes

After we have given the definition of a point process we want to define some general characteristics or attributes of point processes.

Definition 2.1.7 (Stationarity)**.**
A point process X on S, where $S = \mathbb{R}^d$, is said to be stationary if, given any $k \geq 1$, for all bounded sets $W_1, \ldots, W_k \in \mathcal{S}$, all $n_1, \ldots, n_k \in \mathbb{N}_0$ and all $t \in S$, it holds that

$$\mathbb{P}[X(W_1) = n_1, \ldots, X(W_k) = n_k] = \mathbb{P}[X(W_1 + t) = n_1, \ldots, X(W_k + t) = n_k]$$

Remark. Note that Definition 2.1.7 is only valid if S is a Euclidean space. For other spaces this condition could be violated by this translation.
The distribution for a stationary point process X does therefore not depend on the location in space, i.e. it means that the distribution is invariant under translations.

Definition 2.1.8 (Void Probability)**.**
The measure $v_X : \mathcal{S} \to [0, 1]$ with

$$v_X(A) = P[X(A) = 0], \quad A \in \mathcal{S},$$

is called the void probability of X.

The void probability describes how likely it is that a set does not contain any points. This is indeed a strong characteristic since, when known, the distribution of a simple point process X is completely determined by its void probability $v_X(\cdot)$. For the proof see Lieshout [2019].
For additional information of a point process we can mark processes. The marks represent a label of a specific point in the pattern. A marked process is it self again a point process [Daley et al., 2003].

Definition 2.1.9 (Marked Point Process)**.**
A marked point process with locations in S and marks in the mark-space M is a point process X on the product space $S \times M$ with the additional property that the ground process $X_g(\cdot) = X(\cdot \times M)$ is a well-defined point process with $X_g(A) < \infty$ for a bounded $A \in \mathcal{S}$ almost surely.

In general, marks can be all sort of things. A mark can for example indicate a label, a boolean value or a value in S describing a feature.
A marked point process is called a multivariate (or multitype) point process if the mark space is a finite set $\{1, \ldots, K\}$ $K \in \mathbb{N}$.
In the case of a multivariate point process it is clear that the ground process is finite since

$$X_g(\cdot) = X(\cdot \times \{1, \ldots, K\}) = \sum_{i=1}^{K} X_i(\cdot) < \infty.$$

An illustration of a multivariate process can be found in Figure 2.1.

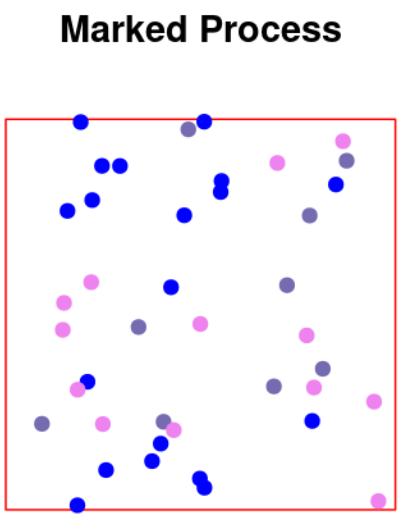


Figure 2.1: A marked process with three different labels indicated by the different colors. The process is generated such that points with the same labels tend to be closer to each other.

2.1.3 Poisson Processes

The Poisson process is a basic reference model in the field of point processes [Lieshout, 2019]. For a Poisson process X on S we need an intensity function $\rho : S \to \mathbb{R}_+$, which indicates the density of a set in $\mathcal{S}$. We then demand that the amount of points in a bounded set $A \in \mathcal{S}$ is Poisson distributed, i.e.

$$X(A) \sim Pois\left(\int_A \rho(x)dx\right). \tag{2.2}$$

Definition 2.1.10 (Poisson Process)**.**
We call X a Poisson process with intensity function $\rho : S \to \mathbb{R}$ if Equation (2.2) holds for every k disjoint and bounded sets $A_1, \ldots, A_k \in \mathcal{S}, k \geq 1$ the random variables $X(A_1), \ldots, X(A_k)$ are independent.

If ρ is constant we call the Poisson process homogeneous and otherwise the process is called inhomogeneous. Note that in the case of a homogeneous process, $X(A)$ is Poisson distributed with mean $|A|\rho$. An example of homogeneous Poisson processes with different intensities can be observed in Figure 2.2.

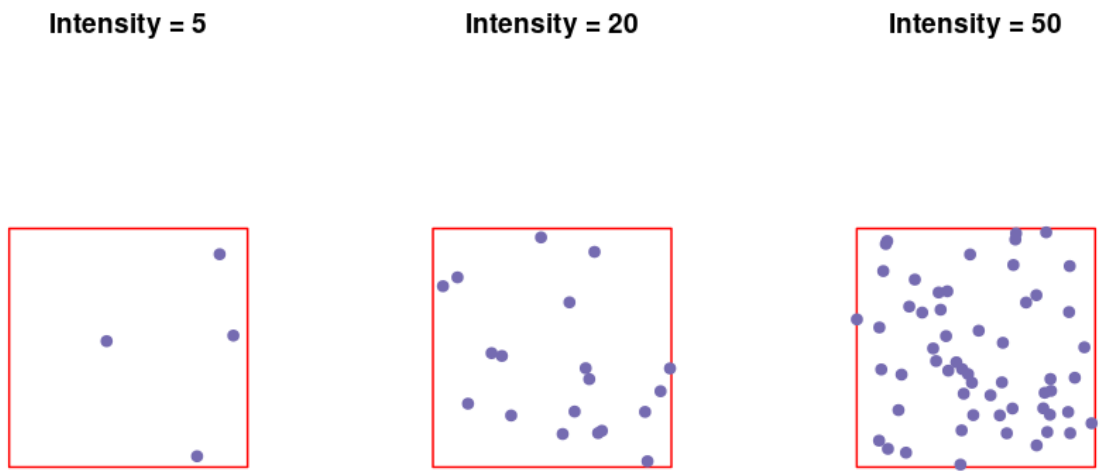


Figure 2.2: Three different realization of a Poisson process with different constant intensity functions. The observation window is the unit square.

2.1.4 Gibbs Processes

Intuitively speaking, the class of Gibbs processes describe a large class of point processes which allow interaction between points of the processes [Dereudre, 2019], where the interaction can lead to a clustered pattern or to a repulsive pattern. The by far most popular Gibbs process is the previously defined Poisson process [Dereudre, 2019], which has no interaction between points. One way of obtaining a proper mathematical definition of a Gibbs process is to consider its density with respect to the distribution of a Poisson process [Jensen and Moller, 1991].
Let the reference measure P_{ref} be given by the distribution of a Poisson process,

$$P_{ref} : \mathcal{F}_{\mathcal{M}} \to [0, 1], \tag{2.3}$$

which thus defines a Poisson process. Note that for a bounded set W it holds that

$$\int_{\mathcal{M}_S^W} dP_{ref}(d\aleph) = \sum_{k=0}^{\infty} \frac{|W|^k}{k!} e^{-|W|} = 1.$$

Here, we divided the space $\mathcal{X}_S(W)$ (or $\mathcal{M}_S^W$) into point patterns containing exactly k points.

2.1.4.1 Energy Function

In general, the energy function is a sum of the energy contributions of all pairs of points [Dereudre, 2019]. The energy function is a parametric function which helps to model the behaviour of processes in the class of Gibbs processes. Modelling through an energy function can therefore help to estimate a certain type of process by fitting it's parameters. The choice of the energy function will later be the basic assumption for the model which we fit using Point Process Learning. First we need to decide on

a type of energy function, and then we can estimate the parameters of this function by fitting to a particular point pattern.

Definition 2.1.11 (Energy Function)**.**
An energy function H is a measureable function

$$H : \mathcal{X}_S \to \mathbb{R} \cup \{\infty\} \tag{2.4}$$

with the following three restrictions:

- *H is non-degenerate, which means it holds that*

$$H(\varnothing) < \infty.$$

- *H is hereditary, which means that for any $\mathbb{y} \in \mathcal{X}_S$ and $y \in \mathbb{y}$ it holds that*

$$H(\mathbb{y}) < \infty \implies H(\mathbb{y} \setminus \{y\}) < \infty.$$

- *H is stable, i.e., there is a constant c such that for any $\mathbb{x} \in \mathcal{X}_S$ it holds that*

$$H(\mathbb{x}) \geq c \cdot |\mathbb{x}|,$$

where $|\cdot|$ denotes the carnality of the pattern.

A classical and simple example of an energy function is the so called pairwise interaction function, which is given by

$$H(\mathbb{x}) = \sum_{x,y \in \mathbb{x}} g(\|x - y\|), \tag{2.5}$$

where $g : \mathbb{R}_+ \to \mathbb{R} \cup \{\infty\}$. The conditions given in Definition 2.1.11 are then fulfilled [Dereudre, 2019]. The energy function is then mainly dependent on the distance between points and not on the position of the individual points.

2.1.4.2 Exponential representation

With help of the energy function we can now define a class of finite volume Gibbs densities on a set W with positive measure.

Definition 2.1.12 (Finite Volume Gibbs Measure)**.**
Let $W \in \mathcal{S}$ be a set with positive and finite measure. Let $\beta \in \mathbb{R}_+$, $z \in \mathbb{R}_+ \setminus \{0\}$ and let H be an energy function. Then we call

$$f_W^{\beta,z}(\mathbb{x}) = \frac{1}{C_W^{\beta,z}} z^{|\mathbb{x} \cap W|} e^{-\beta H(\mathbb{x})} P_{ref}(d\mathbb{x}) \tag{2.6}$$

the density of a finite volume Gibbs measure in exponential representation, where P_{ref} is the distribution of a Poisson process, usually the unit rate Poisson process.

The factor $C_W^{\beta,z} = \int_{\mathcal{M}_S} z^{|\aleph \cap W|} e^{-\beta H(\aleph)} \, P_{ref}(d\aleph)$ is a normalizing constant, the so-called partition function, which ensures that we have a proper probability density function.

We still need to ensure that the normalizing constant is finite and greater than zero, i.e,

$$0 < C_W^{\beta,z} = \int_{\mathcal{M}_S} z^{|\aleph \cap W|} e^{-\beta H(\aleph)} \, P_{ref}(d\aleph) < \infty.$$

If this is the case, the distribution is well-defined.

Lemma 2.1.1. *For a normalizing constant $C_W^{\beta,z}$ of a parametric Gibbs density, defined as in Definition 2.1.12, it holds that*

$$0 < C_W^{\beta,z} < \infty, \tag{2.7}$$

if H is an energy function.

Proof. We can use the stability and show that

$$C_W^{\beta,z} \leq e^{-|W|} \sum_{i=0}^{\infty} \frac{\left(ze^{\beta c}|W|\right)^n}{n!} = \exp(|W|(ze^{\beta c} - 1)) < \infty.$$

The factor c is given as defined in the definition for the energy function.
By the non-degeneracy of H we know that

$$C_W^{\beta,z} \geq P_{ref}(\emptyset) e^{-\beta H(\emptyset)} = e^{-|W|} e^{-\beta H(\{\emptyset\})} > 0.$$

□

2.1.5 Intensities of Gibbs Processes and the GNZ Formula

With the previously given definitions we can now derive the first fundamental equation, the GNZ-formula, for the field of Point Process Learning. We however first need to introduce the concept of Papangelou conditional intensity, which was introduced by Papangelou [1974]. The conditional intensity expresses the likelihood of the occurring of a new point with respect to a position under the condition of an already existing point pattern.

Definition 2.1.13 (Papangelou Conditional Intensity)**.**
Let f with finite support W be a Gibbs density function with respect to the unit rate Poisson process and let $\aleph$ be a realization of that Gibbs process. Then the Papangelou conditional intensity $\lambda : S \times \mathcal{X}_s$ for $u \in W \setminus \aleph$ is given by,

$$\lambda(u; \aleph) = \frac{f(\aleph \cup \{u\})}{f(\aleph)}. \tag{2.8}$$

Note that $\aleph$ is a finite point pattern.

Note that if we consider $\lambda(u; X)$ where X is now a point process, $\lambda(u; X)$ turns into a random variable for a fixed $u \in S$, i.e. it is a random field on S. Also note that with the exponential representation of f we can rewrite the conditional intensity as

$$\begin{aligned}\lambda(u; X) &= \frac{C_W^{\beta,z}}{C_W^{\beta,z}} z^{W(X)+1-W(X)} e^{-\beta(H(X\cup\{u\})-H(X))} \\ &= z e^{-\beta(H(X\cup\{u\})-H(X))}.\end{aligned} \tag{2.9}$$

Remark (Notation).
When we express the conditional intensity for a fixed $\aleph$ as a realization from X we sometimes write $\lambda_X(u; \aleph)$ to express that $\lambda(\cdot;\cdot)$ is the conditional intensity resulting from the Gibbs density of X.

We next prove the following theorem, the Georgii-Nguyen-Zessin (GNZ) formula, which is the fundamental basis of the method of Cronie et al. [2023].

Theorem 2.1.2 (GNZ formula)**.**
Let X be a point process in $W \subset S$, whose density function has an exponential representation. Let h be a non-negative and integrable function with $h : S \times \mathcal{X}_S \to \mathbb{R}$, then the following equation holds, which states

$$\mathbb{E}\left[\sum_{x\in X} h(x, X \setminus \{x\})\right] = \int_W \mathbb{E}\left[h(u, X)\lambda_X(u; X)\right]\, du. \tag{2.10}$$

Proof. We have that

$$\mathbb{E}\left[\sum_{x\in X} h(x, X \setminus \{x\})\right] = \frac{1}{C_W^{\beta,z}} \int_{\mathcal{M}_S^W} \sum_{x\in\aleph} h(x, \aleph \setminus \{x\}) z^{|\aleph\cap W|} e^{-\beta(H(\aleph))}\, P_{ref}(d\aleph),$$

where we take P_{ref} to be the unit rate Poisson process distribution. We now factor out the event of $\aleph$ having exactly n points, which results in the above being given by

$$\begin{aligned}&\frac{e^{-|W|}}{C_W^{\beta,z}} \sum_{n=1}^{\infty} \frac{z^n}{n!} \sum_{k=1}^{n} \int_W \cdots \int_W h(x_k, \{x_1, \ldots, x_n\} \setminus x_k) e^{-\beta H(\{x_1,\ldots,x_n\})}\, dx_1 \cdots dx_n \\ = &\frac{e^{-|W|}}{C_W^{\beta,z}} \sum_{n=1}^{\infty} \frac{z^n}{n-1!} \int_W \cdots \int_W h(u, \{x_1, \ldots, x_{n-1}\} \setminus u) e^{-\beta H(\{x_1,\ldots,x_{n-1}\})} \\ &\qquad e^{\beta H(\{x_1,\ldots,x_{n-1}\})} e^{-\beta H(\{x_1,\ldots,x_{n-1}\}\cup u)}\, dx_1 \cdots dx_{n-1} du.\end{aligned}$$

We can now again rewrite the integral with help of the reference measure, which

then results in

$$\begin{aligned}
&\frac{z}{C_W^{\beta,z}} \int_W \int_{\mathcal{M}_S} h(u, \aleph) z^{|\aleph \cap W|} e^{-\beta H(\aleph)} e^{-\beta(H(\aleph \cup \{u\}) - H(\aleph))} \; P_{ref}(d\aleph) \; du \\
&= \int_W \int_{\mathcal{M}_S} h(u, X) z e^{-\beta(H(\aleph \cup \{u\}) - H(\aleph))} \; P_W^{\beta,z}(d\aleph) \; du \\
&= \int_W \mathbb{E}[h(u, X) \lambda(u; X)] \; du,
\end{aligned}$$

where the last step follows from the remark of Definition 2.1.13.

□

The GNZ formula is a very powerful tool. In fact the GNZ formula contains all the information about the Gibbs density $f_W^{\beta,z}$. To be more precise consider the following theorem.

Theorem 2.1.3. *Let $W \subset S$ be a bounded set with positive measure. Let P be a probability measure on $\mathcal{M}_S^W$ such that for any measurable and non-negative function $h : S \times \mathcal{M}_S^W \to \mathbb{R}$ it holds that,*

$$\mathbb{E}_P \left[\sum_{x \in X} h(x, X \setminus \{x\}) \right] = \mathbb{E}_P \left[\int_W h(u, X) z e^{-\beta(H(X \cup \{u\}) - H(X))} \; du \right], \tag{2.11}$$

where $\mathbb{E}_P$ denotes the expectation with respect to the probability measure P. Then it holds that P has density $f_W^{\beta,z}$, where $f_W^{\beta,z}$ denotes the Gibbs density according to Definition 2.1.12.

Proof. For the proof consider the measure $Q = z^{-N} \mathbb{1}_{H < \infty} z e^{\beta H} P$, where $N = \mu(W)$ and $\mu \in \mathcal{M}_S^W$. Then

$$\begin{aligned}
&\int_{\mathcal{M}_S^W} \sum_{x \in \aleph} h(x, \aleph \setminus \{x\}) \; Q(d\aleph) \\
&= \int_{\mathcal{M}_S^W} \sum_{x \in \aleph} h(x, \aleph \setminus \{x\}) z^{-|\aleph \cap W|} \mathbb{1}\{H(\aleph) < \infty\} z e^{\beta H(\aleph)} P(d\aleph) \\
&= \frac{1}{z} \int_{\mathcal{M}_S^W} \sum_{x \in \aleph} h(x, \aleph \setminus \{x\}) z^{-|(\aleph \setminus \{x\}) \cap W|} \mathbb{1}\{H(\aleph \setminus x) < \infty\} \mathbb{1}\{H(\aleph) - H(\aleph \setminus \{x\})) < \infty\} \\
&\quad e^{\beta H(\aleph \setminus \{x\})} e^{\beta(H(\aleph) - H(\aleph \setminus \{x\}))} P(d\aleph) \\
&= \int_{\mathcal{M}_S^W} \int_W h(u, \aleph) z^{-|\aleph \cap W|} \mathbb{1}\{H(\aleph) < \infty\} \mathbb{1}\{H(\aleph \cup \{u\}) - H(\aleph)) < \infty\} e^{-\beta(H(\aleph \cup \{u\}) - H(X))} \\
&\quad e^{\beta H(\aleph)} e^{\beta(H(\aleph \cup \{u\}) - H(\aleph))} \; du \; P(d\aleph) \\
&= \int_{\mathcal{M}_S^W} \int_W h(u, \aleph) \mathbb{1}\{H(\aleph \cup \{u\}) - H(\aleph)) < \infty\} \; Q(d\aleph).
\end{aligned}$$

We deduce that Q satisfies the Slivnyak-Mecke formula [Stoyan et al., 1995] on $\{\aleph \in \mathcal{M}_S^W, H(\aleph) < \infty\}$. That this implies that the measure Q (after normalization) is the Poisson point process restricted to $\{\aleph \in \mathcal{M}_S^W, H(\aleph) < \infty\}$. This now finishes the proof. □

With the proper selection of Q we could show that P has to be of the form given in Definition 2.1.12. This means that the GNZ fromula contains all the information about the Gibbs process. This also means, that if we estimate the conditional intensity of a process, we can reconstruct the process itself. We even go one step further and define a general Gibbs process with help of the GNZ formula.

Definition 2.1.14 (Gibbs Density)**.**
Let $W \subset S$ be a set with positive finite measure and let f be a probability density on $\mathcal{M}_S^W$ and let $\lambda(u, \cdot)$ be the Papangelou conditional intensity given as in Definition 2.1.13. We call f a Gibbs density if the GNZ formula holds for every positive and measurable function $h : S \times \mathcal{M}_S^W \to \mathbb{R}$, i.e.

$$\int_{\mathcal{M}_S^W} \sum_{x \in \mathbb{x}} h(x, \mathbb{x} \setminus \{x\}) \; f(d\mathbb{x}) = \int_{\mathcal{M}_S^W} \int_W h(u, \mathbb{x}) \lambda(u; \mathbb{x}) \; dx \; f(d\mathbb{x}).$$

The already introduced exponential form of Gibbs densities is a special representation. For a more general representation of a Gibbs density f_θ, parameterized by $\theta \in \Theta$, we split the function into its normalizing constant and the rest, i.e.

$$f_\theta(\mathbb{x}) = \frac{1}{C_\theta} g(\theta, \mathbb{x}). \tag{2.12}$$

With help of the GNZ formula we can now also define the Innovation $\mathcal{I}$, introduced by Baddeley et al. [2005], which will later be used to construct loss functions.

Definition 2.1.15.
Let X be a point process on $W \subset S$ with conditional intensity function $\lambda_X(u; \cdot)$. Let $h : S \times \mathcal{X}_S \to \mathbb{R}$ be a non-negative and integrable function. Then the associated Innovation is given by

$$\mathcal{I}^h(W, X) = \sum_{x \in X \cap W} h(x, X \setminus \{x\}) - \int_W h(u, X) \lambda_X(u; X) \; du. \tag{2.13}$$

In case of a parametric Gibbs density with the parameter $\theta \in \Theta$, where Θ is the parameter space, we denote the Innovation by

$$\mathcal{I}_\theta^h(W, X) = \sum_{x \in X \cap W} h(x, X \setminus \{x\}) - \int_W h(u, X) \lambda_\theta(u; X) \; du. \tag{2.14}$$

Note that the Innovation is zero in expectation. This follows directly from the GNZ formula.
As another consequence of the GNZ formula, by letting h be constant over its second

argument, we obtain the Campbell formula, which states

$$\mathbb{E}[\sum_{x\in X} h(x)] = \int_W h(u)\mathbb{E}[\lambda_X(u;X)]du = \int_W h(u)\rho(u)du, \tag{2.15}$$

where $\rho(\cdot)$ is the so-called intensity function of X, which reflects the (unconditional) univariate likelihood of the occurrence of a point. This, for example, allows us to calculate the expected amount of points in a set W, i.e.

$$\mathbb{E}\left[X(W)\right] = \int_W \rho_X(u)\ du = \int_W \int_{\mathcal{M}_S^W} \lambda(u;\mathbb{x})\ P(d\mathbb{x})\ du, \tag{2.16}$$

where P is the Gibbsian distribution of X.

Remark (Expectation Measure).
The term $\mathbb{E}\left[X(W)\right] = \int_W \rho_X(u)\ du$ is also known as the expectation, mean or intensity measure. It is easy to show that it fulfils the conditions for a measure $\mathcal{S} \to \mathbb{R}_+$.
If the process X is stationary the expectation measure becomes translation-invariant. We can then represent it with

$$\mathbb{E}\left[X(W)\right] = \nu|W|$$

for a suitable constant $\nu > 0 \in \mathbb{R}$, i.e. $\rho(\cdot) = \nu$.

With help of the conditional intensity, we can also define the characteristics repulsiveness and attractiveness, which describe interactions between the points of the process. If a process is attractive, points tend to cluster. If the process is repulsive, points tend to have fewer neighbours. More formally, we have the following definition.

Definition 2.1.16 (Repulsive and attractive Point Processes)**.**
We call X attractive if for $\mathbb{y} \subset \mathbb{x}$ it holds that

$$\lambda(u;\mathbb{y}) \leq \lambda(u;\mathbb{x}) \quad \forall u \in S.$$

We call X repulsive if for $\mathbb{y} \subset \mathbb{x}$ it holds that

$$\lambda(u;\mathbb{y}) \geq \lambda(u;\mathbb{x}) \quad \forall u \in S.$$

Remark (Poisson Process).
Note that a Poisson process is both attractive and repulsive, i.e., $\lambda(\cdot;\mathbb{y}) = \lambda(\cdot;\mathbb{x})$ for $\mathbb{y} \subset \mathbb{x}$. This follows from the fact that the conditional intensity is constant over the second argument, i.e.

$$\lambda(u;\cdot) = \rho(u),$$

where $\rho(\cdot)$ is the intensity function. This also implies that the term $e^{-\beta H(\cdot)}$ in the exponential form of a Gibbs density is constant, which further implies that $H(\cdot)$ is constant, meaning that there is no interaction in the process.

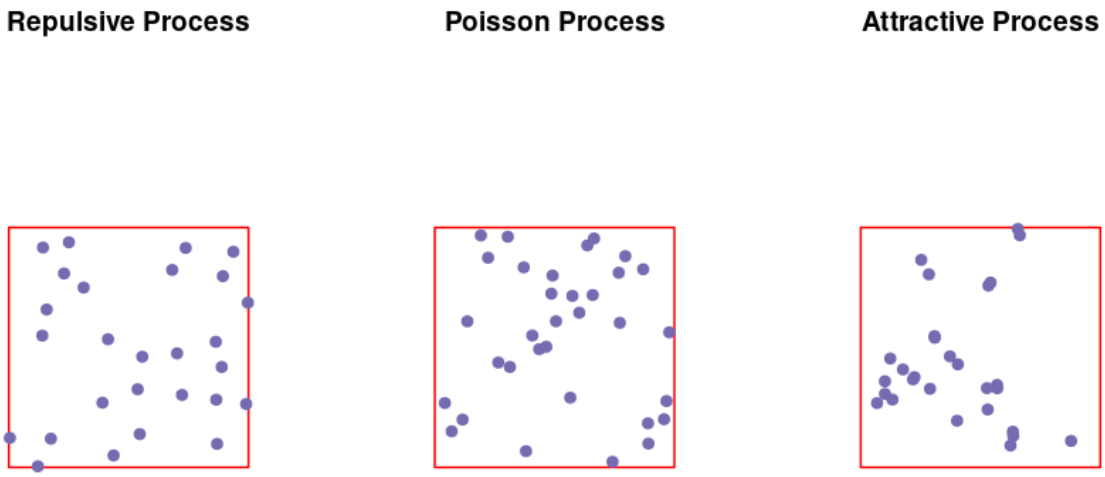


Figure 2.3: Three realizations of different point processes, where the left process is repulsive, the process generating the pattern in the middle does not have any interaction and the process in the right panel is attractive.

2.1.6 Examples of Gibbs Processes

After the theoretical introduction of Gibbs processes, we want to give some explicit examples of classes of Gibbs processes. The defined classes will be parameterized and will later be useful in the simulation study, where we will try to estimate the parameters with the help of Point Process Learning.

2.1.6.1 Strauss Process

A Strauss Process, introduced by Strauss [1975], is a pairwise interaction process where, in a ball with a certain radius R around a point, the likelihood for the appearance of another point in this ball is lower than that outside the balls around all points. It is therefore repulsive.
The Gibbs density of the Strauss process with support in a window W is given by

$$\begin{aligned} f_W^{\gamma,\beta,R}(\mathbb{x}) &= \frac{1}{C_W^{\gamma,\beta}} \beta^{|\mathbb{x}\cap W|} \exp\left[-\alpha \sum_{x,y\in\mathbb{x}} h(\|x-y\|,R)\right] \\ &= \frac{1}{C_W^{\gamma,\beta}} \beta^{|\mathbb{x}\cap W|} \gamma^{H(\mathbb{x},R)}, \end{aligned} \tag{2.17}$$

with $\gamma = e^{-\alpha}$. The inner function h of the energy function $H(\mathbb{x})$ is given by

$$h(x,R) = \mathbb{1}_{[0,R]}(x).$$

Note that the conditional intensity is now given by

$$\lambda(u;X) = \beta\gamma^{\sum_{x\in X} h(\|u-x\|)}.$$

With the parameter γ we can handle how repulsive the process will be. The parameter γ has to be in $[0,1]$ due to the following lemma.

Lemma 2.1.4. *Let $\beta, R > 0$. The Gibbs density $f_W^{\gamma,\beta}$ is well-defined if and only if $\gamma \leq 1$, i.e.*

$$\int_{\mathcal{M}_S^W} \beta^{|\mathbb{x}\cap W|}\gamma^{H(\mathbb{x},R)}P_{ref}(d\mathbb{x}) < \infty \Longleftrightarrow \gamma \leq 1.$$

Proof. "$\Rightarrow$" Assume that $\gamma > 1$ holds. With out loss of generality assume $A \subset W$ with positive measure and $\sup\{\|x-y\| | x, y \in A\} < R$ exists then the normalizing constant is given by

$$\begin{aligned}
C_W^{\beta,\gamma} &= \int_{\mathcal{M}_S^W} \beta^{|\mathbb{x}\cap W|}\gamma^{H(\mathbb{x},R)}dP_{ref}(\mathbb{x}) \\
&= \sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{H((x_1,\ldots,x_n),R)}\, dx_1 \cdots dx_n \\
&\geq \sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_A \cdots \int_A \gamma^{H((x_1,\ldots,x_n),R)}\, dx_1 \cdots dx_n \\
&\geq \sum_{n=1}^{\infty} e^{-|W|}\frac{|A|\beta^n}{n!}\beta\gamma^{\binom{n}{2}},
\end{aligned}$$

where the last step sums over all pairwise interactions. Note that we have to include every interaction, since we chose A such that every point in A interacts. When we apply the root criterion to the last term, we get

$$\limsup_{n\to\infty} \sqrt[n]{\left| e^{-|W|}\frac{|A|\beta^n}{n!}\beta\gamma^{\binom{n}{2}}\right|} = \lim_{n\to\infty} \frac{\gamma^{\frac{n-1}{2}}}{\sqrt[n]{n!}}|A|\beta = \infty,$$

when $\gamma > 1$, which shows the sum is not converging. This means that the normalizing constant does not exist.

"$\Leftarrow$" Let $\gamma \leq 1$. Now it holds that,

$$\begin{aligned}
C_W^{\beta,\gamma} &= \int_{\mathcal{M}_S^W} \beta^{|\mathbb{x}\cap W|}\gamma^{H(\mathbb{x},R)}dP_{ref}(\mathbb{x}) \\
&= \sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{H((x_1,\ldots,x_n),R)}\, dx_1 \cdots dx_n \\
&\leq \sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{0}\, dx_1 \cdots dx_n \\
&= \sum_{n=1}^{\infty} e^{-|W|}\frac{|W|\beta^n}{n!}\beta,
\end{aligned}$$

where we again can apply the root criterion to show that the sum converges, since

$$\limsup_{n\to\infty} \sqrt[n]{\left| e^{-|W|}\frac{|W|\beta^n}{n!}\beta\right|} = \lim_{n\to\infty} \frac{|W|\beta}{\sqrt[n]{n!}} = 0.$$

□

A similar proof is given by Ripley [1979]. It is however interesting to mention that Strauss intended the process to be a cluster process by setting $\gamma > 1$ [Strauss, 1975]. We also give a quick proof why the Strauss process is repulsive.

Lemma 2.1.5. *Let X be defined according to the above described density* (2.17) *with $\gamma \leq 1$, then X is repulsive.*

Proof. Let $\mathbb{x} \subset W$ be a realization of X and let $\mathbb{y}$ be a finite set of unique points in W, where it holds that $\mathbb{y} \subset \mathbb{x}$. We define $\mathbb{z}$ to be,

$$\mathbb{z} = \mathbb{x} \setminus \mathbb{y}$$

Now let $u \in W$. Then it holds for the conditional intensity that

$$\lambda(u; \mathbb{y}) = \beta\gamma^{\sum_{x \in \mathbb{x}} \rho(\|u-x\|)} \leq \beta\gamma^{\sum_{x \in \mathbb{x} \cup \mathbb{z}} \rho(\|u-x\|)} = \lambda(u; \mathbb{x})$$

since it holds that

$$\sum_{x \in \mathbb{y}} \rho(\|u - x\|) \leq \sum_{x \in \mathbb{y} \cup \mathbb{z}} \rho(\|u - x\|)$$

□

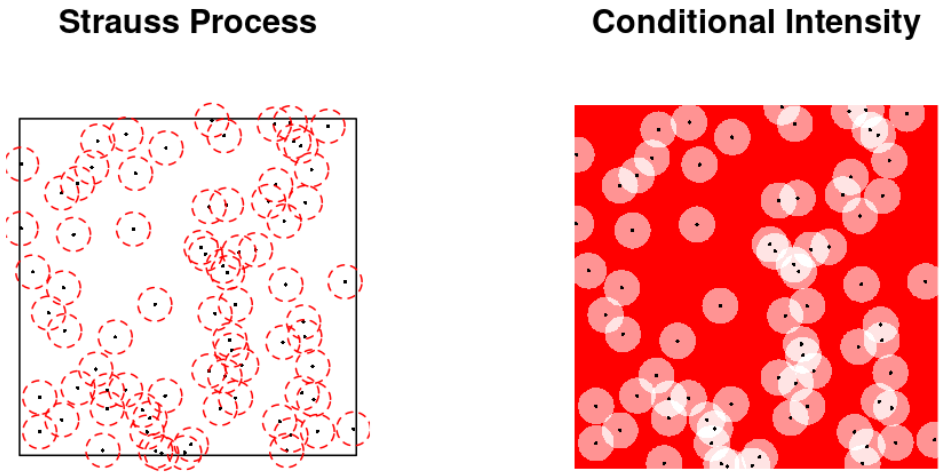


Figure 2.4: Realization of the Strauss process with parameter $R = 0.05$, $\beta = 100$ and $\gamma = 0.5$. The conditional intensity can be observed in the right panel, where a darker red indicates a higher intensity. This means points are more likely to end up in the empty space between the points. This illustrates the repulsive behaviour of the Strauss process.

2.1.6.2 Hard-Core Process

The hard-core process is a special case of the Strauss process. Here we set $\gamma = 0$, which then results in the density

$$f_W^{\beta,R}(\aleph) = \frac{1}{C_W^\beta}\beta \mathbb{1}\{\|x - y\| \geq R \; \forall x \neq y \in \aleph\}. \tag{2.18}$$

By setting γ to zero we now do not allow any points within a radius R around existing points, i.e, if X is a hard-core process then it holds that

$$\|x - y\| > R \;\; \text{a.s.}$$

for every $x, y \in X$. The conditional intensity then only maps to the values β and 0 and is given by

$$\lambda(u; X) = \begin{cases} \beta & \text{if } \; u \in W \setminus \bigcup_{x \in X} B_R(x), \\ 0 & \text{else.} \end{cases}$$

Since the Strauss process is repulsive the hard-core process is as well, which is intuitively clear, since the more points we have the smaller the area is, where we can place new points.
Hard-core processes are used in various fields, such as physics, material science, and biology, to model scenarios where objects or events can not be too close to each other [Bondesson and Fahlén, 2003].

The hard-core process is one of the processes, we will later analyze in our simulation study.

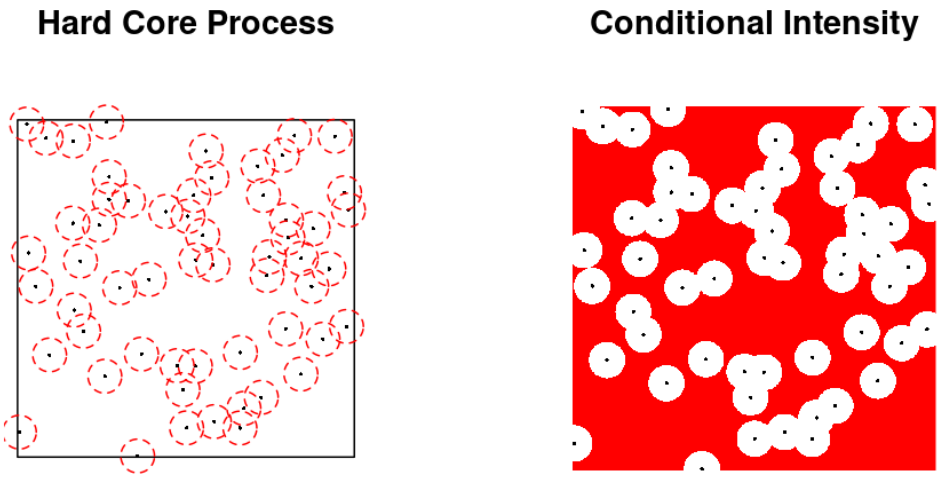


Figure 2.5: Realization of the hard-core process with parameters $R = 0.05$ and $\beta = 100$. The conditional intensity can be observed in the right picture, where a darker red indicates a higher intensity.

2.1.6.3 Area Interaction Process

The area interaction process, introduced by Baddeley and Van Lieshout [1995], can be either repulsive or attractive depending on the parameter choice [Baddeley and Van Lieshout, 1995]. Motivated by the problem with the Strauss process, which can not be extended to a cluster process [Ripley, 1979], Baddeley and Van Lieshout [1995] introduced the area interaction process. Similarly to the Strauss process, the Gibbs density is given by

$$f_W^{\beta,\gamma}(\mathbb{x}) = \frac{1}{C_W^{\beta,\gamma}} \beta^{|\mathbb{x} \cap W|} \gamma^{-A_R(\mathbb{x})}, \tag{2.19}$$

with $\gamma, \beta, R > 0$. Here $A_R(\mathbb{x})$ describes the volume of the intersection of overlapping balls with radius R around the points of X, i.e.,

$$A_R(\mathbb{x}) = \bigcup_{x \in \mathbb{x}} B_R(x).$$

For $\gamma = 1$ the process will reduce to a Poisson process, since then there is no interaction between points. The process is, similar to the Strauss process, repulsive for $\gamma < 1$, however we now can choose $\gamma > 1$ as well to obtain an attractive point process.

Lemma 2.1.6. *Let $f_W^{\beta,R}$ be defined according to Equation (2.19) and let $\beta, R > 0$ be fixed. Then the area interaction process is well-defined for all $\gamma > 0$, i.e.*

$$0 < \frac{1}{C_W^{\beta,\gamma}} < \infty \quad \forall \gamma > 0$$

holds.

Proof. For the proof we consider two cases. For the first case we assume that $\gamma > 0$, then it holds that

$$\begin{aligned}&\int_{\mathcal{M}_S^W} \beta^{|\mathbb{x}\cap W|}\gamma^{-A_R(\mathbb{x})}\ P_{ref}(d\mathbb{x})\\ =&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-A_R(x_1,\ldots,x_n)}\ dx_1\cdots dx_n.\end{aligned}$$

This gives us now

$$\begin{aligned}&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-A_R(x_1,\ldots,x_n)}\ dx_1\cdots dx_n\\ \leq&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-0}\ dx_1\cdots dx_n\\ =&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta|W|^n}{n!} < \infty.\end{aligned}$$

The normalizing constant is therefore bounded from above. It also holds that

$$\begin{aligned}&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-A_R(x_1,\ldots,x_n)}\ dx_1\cdots dx_n\\ \geq&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-n|B_R(0)|}\ dx_1\cdots dx_n\\ =&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}|W|\gamma^{-n|B_R(0)|} \geq 0.\end{aligned}$$

For the other case $\gamma \leq 1$ it holds that

$$\begin{aligned}&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-A_R(x_1,\ldots,x_n)}\ dx_1\cdots dx_n\\ \geq&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-0}\ dx_1\cdots dx_n\\ =&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta|W|^n}{n!} > 0.\end{aligned}$$

It also holds that

$$\begin{aligned}&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-A_R(x_1,\ldots,x_n)}\ dx_1\cdots dx_n\\ \leq&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}\int_W \cdots \int_W \gamma^{-n|B_R(0)|}\ dx_1\cdots dx_n\\ =&\sum_{n=0}^{\infty} e^{-|W|}\frac{\beta^n}{n!}|W|\gamma^{-n|B_R(0)|} < \infty,\end{aligned}$$

where the last inequality follows from the root criterion and using $\sqrt[n]{n!} \to \infty$. The area interaction process is therefore well-defined.

□

Note that since the Gibbs density of the area interaction process in (2.19) can easily be refactored in exponential representation, we also could show that $A_R(\cdot)$ is an energy function and then apply Lemma 2.1.1 to obtain that the normalizing constant exists.

Remark (general expression).
There is also a more general expression of the area interaction process. The term in the exponent, $-A_R(\cdot)$, can be replaced by a more general term with $-\nu(U(\cdot))$, where ν is a finite regular measure and

$$U(\mathbb{x}) = \bigcup_{x \in \mathbb{x}} Z(x), \quad Z : S \to \mathcal{S},$$

where Z assigns a point $x \in S$ to an area in S.

The Pagangelou conditional intensity of the area interaction process is given by

$$\lambda(u; X) = \beta\gamma^{-B_R(u) \setminus A_R(X)}. \tag{2.20}$$

We will later also investigate the area interaction process in our simulation study. The special thing about the area interaction process is that it can form clustered patterns but also repulsive pattern, depending on the parameterization.

Lemma 2.1.7. *Let X be the area interaction process, specified according to Equation (2.19). Then X is repulsive if $\gamma \leq 1$ and is attractive if $\gamma \geq 1$. If $\gamma = 1$ then X is a Poisson process.*

Proof. First assume $\gamma = 1$, then the conditional intensity for the area interaction process is given by

$$\lambda(u; X) = \beta.$$

We then have by definition a homogeneous Poisson process with intensity β.
Now assume $\gamma \leq 1$. Let $\mathbb{x} \subset W$ be a realization of X and let $\mathbb{y}$ be a finite point pattern in W where $\mathbb{x} \subset \mathbb{y}$. Then for a fixed $u \in W$ it holds that

$$B_R(u) \setminus \bigcup_{x \in \mathbb{x}} B_R(x) \leq B_R(u) \setminus \bigcup_{x \in \mathbb{y}} B_R(x). \tag{2.21}$$

Therefore it directly follows that for $\gamma \leq 1$,

$$\lambda(u; \mathbb{x}) \leq \lambda(u; \mathbb{y}) \quad \forall u \in W$$

and for $\gamma \geq 1$ we also can use Equation (2.21) to observe that,

$$\lambda(u; \mathbb{x}) \geq \lambda(u; \mathbb{y}) \ \ \forall u \in W.$$

□

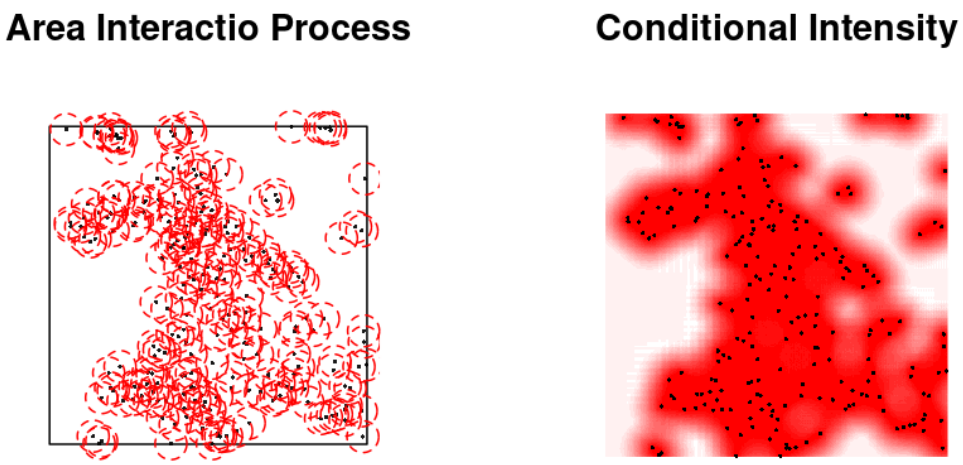


Figure 2.6: Realization of the area interaction process with parameters $R = 0.05$, $\beta = 10$ and $\gamma = 10$, which means that the process is attractive. The conditional intensity can be observed in the right picture, where a darker red indicates a higher intensity. We can clearly observe the clustering behaviour of the process.

2.2 Inference for Gibbs Processes

In this section we want to present approaches to learn the distribution of the underlying point processes from observing Gibbs patterns.
We first focus on some general approaches from the literature. Further in this section we explain the Point Process Learning method from Cronie et al. [2023], which will be the foundation for our work in the following chapters.

Note that for Gibbs densities, maximum likelihood estimation is not feasible since our estimator involves the calculation of complex normalizing constants, i.e.

$$\hat{\theta} = \arg\max_{\theta\in\Theta} f_\theta(X) = \arg\max_{\theta\in\Theta} \frac{1}{\int_{\mathcal{M}_S^W} g(\varkappa,\theta)\ P_{ref}(d\varkappa)} g(X,\theta),$$

which involves calculating the complex integral over the space $\mathcal{M}_S^W$ which is not tracable [Jensen and Moller, 1991].

For a Poisson process maximum likelihood estimation is still possible, since the normalizing constant reduces to the mean intensity of the process, since we do do not have a dependency on the point pattern. We would then need to optimize

$$\hat{\theta} = \arg\min_{\theta\in\Theta} \rho^*_\theta(x) = \arg\min_{\theta\in\Theta} \frac{\rho_\theta(x)}{\int_W \rho_\theta(u)\ du}, \quad x \in W,$$

where $\rho_\theta(\cdot)$ describes the intensity function of a Poisson process.

2.2.1 Takacs-Fiksel estimator

Because the likelihood function is not tractable, we need other methods for estimating the density. A common method for estimating Gibbs densities is the Takacs-Fiksel method, which was introduced by Takacs [1986] and Fiksel [1988]. The idea is based on the previously introduced GNZ-formula (2.10).

Definition 2.2.1 (Takacs-Fiksel estimator)**.**
Let X be a point process in $W \subset S$. Let $\theta \in \Theta$ describing a parametric Gibbs density f_θ. Then the Takacs-Fiskel estimator is given by

$$\begin{aligned}\hat{\theta}^{TF} &= \arg\min_{\theta\in\Theta} \sum_{k=1}^{K} \left[\mathcal{I}_\theta^{h_k}(W,X)\right]^2 \\ &= \arg\min_{\theta\in\Theta} \sum_{k=1}^{K} \left[\sum_{x\in X} h_k(x, X\setminus\{x\}) - \int_W h_k(u,X)\lambda_\theta(u;X)\, du\right]^2\end{aligned} \tag{2.22}$$

where $\mathcal{I}_\theta^{h_k}$ is the Innovation and $h_k : S \times \mathcal{X}_S \to R\ \forall k \in \{1,\dots,K\}$ are measurable, non-negative functions.

2.2.1.1 Maximum Pseudo-Likelihood Estimation

A commonly used version of Takacs-Fiskel estimation is the pseudo-likelihood method [jean-francois coerjolly and Dereudre, 2012]. The method was introduced by Besag [1975], in the context of random fields.

Definition 2.2.2 (Pseudo-Likelihood Estimation)**.**
Let X be a point process in $W \subset S$ and let $\Xi_\theta = \{\xi_\theta(.;X) : \theta \in \Theta\}$ be a parametric family for estimating the conditional intensity $\lambda_{\theta_0}(u;X)$ of X with $\theta_0 \in \Theta$. Then pseudo likelihood method is based on finding

$$\hat{\theta}_{PL}^{W} = \arg\min_{\theta\in\Theta} \sum_{k=1}^{K} \left[\mathcal{I}_\theta^{h_k}(W,X)\right]^2 \tag{2.23}$$

with

$$h_k = \frac{\partial}{\partial\theta_k} \log\lambda_\theta(u;X) \quad \text{for } k = \dim\theta. \tag{2.24}$$

We now observe that the Maximum Pseudo-Likelihood Estimation is a special case

of the Takacs-Fiksel estimator with explicit form

$$\sum_{k=1}^{K}\left[\mathcal{I}_{\theta}^{h_k}(W,X)\right]^2 = \sum_{k=1}^{K}\left[\sum_{x\in X} h_k(x, X\setminus\{x\}) - \int_W h_k(u,X)\lambda_\theta(u;X)\,du\right]^2$$
$$= \sum_{k=1}^{K}\left[\sum_{x\in X} \frac{\partial}{\partial\theta_k}\log\lambda_\theta(x, X\setminus\{x\}) - \int_W \left(\frac{\partial}{\partial\theta_k}\log\lambda_\theta(u;X)\right)\lambda_\theta(u;X)\,du\right]^2.$$

For more properties and convergence theorems we refer to the work of Jensen and Møller [1991]. It is important to note that the Maximum Pseudo-Likelihood Estimation is not shown to be optimal in the class of Takacs-Fiskel estimator [Coeurjolly et al., 2016].

2.3 Point Process Learning (PPL)

In this section we introduce the method from Cronie et al. [2023] and provide a general definition of PPL. This overview is important for the remaining part of the thesis, since our modification regarding data augmentation are based on this method. Tmilarly to the Takacs-Fiskel method, the presented method is inspired by the GNZ formula (2.10). We now however split the observed data into training and validation sets, i.e. we make use of cross-validation.
To split the data into training and validation sets, we need a method which separates the data points in two classes. This is done with the help of thinning, following the definition of Cronie et al. [2023].

Definition 2.3.1 (Thinning)**.**
Given a point process X in S, a thinning Z of X with retention probability $p : S \times X \to [0,1]$ may be defined as the marginal point process $Z = \{x : (x,m) \in \breve{X}, m = 1\}$ of a bi-variate marking $\breve{X} = \{(x_i, m_i)\}_{i=1,\dots,N} \subset S \times M$, $M = \{0,1\}$, of X.
Here, $m_i = m(x_i) \in M$, $i = 1,\dots,N$, for some (possibly random) marking function $m(\cdot)$, governing the retention probability. We let the reference measure on M be the counting measure. When $\breve{X}$ is independently marked, i.e. the marks are independent conditional on X, whereby the retention probability $p(u), u \in S$, does not depend on X, we say that Z is an independent thinning. If, in addition, $p(\cdot) = p \in [0,1]$, we say that Z is a p-thinning.



Figure 2.7: Realization of the Poisson process with intensity $\mu = 50$. The clear blue points indicate a thinning with retention probability $p = 0.2$.

When applying thinning on X to split the data into training and validation sets, we refer to the thinning itself as validation set, denoted by X_V, and to the remaining point as training set, denoted by $X_T = X \setminus X_V$.

Thinning in general is a powerful tool. It also allows us to easily simulate inhomogeneous Poisson processes. Here, only a homogeneous Poisson process with intensity

$$\rho^* = \sup_{u \in W} \rho(u),$$

is generated, where $\rho(\cdot)$ is the intensity function of the inhomogenous Poisson process X. Afterwards a independent thinning with retension probability

$$p : W \to [0,1]; \quad p(u) = \frac{1}{\rho^*}\rho(u)$$

can be applied to obtain a realization of the process X.

The following theorem, also presented in Cronie et al. [2023], motivates the splitting of the data into a training and validation set, which is achieved by thinning.

Theorem 2.3.1 (Theorem 1 from Cronie et al. [2023])**.**
Let X_V be an independent thinning of a point process X in S, with retention probability $p : S \to (0,1)$. Given $X_V = X \setminus X_T$, then for any non-negative or integrable function $h : S \times \mathcal{X}_S \to \mathbb{R}$ it holds that.

$$\mathbb{E}\left[\sum_{x \in X_V} h(x, X_T)\right] = \mathbb{E}\left[\sum_{x \in X_T} h(x, X_T \setminus \{x\}) \frac{p(x)}{1 - p(x)}\right] \tag{2.25}$$

Moreover, the conditional intensity of X_V is given as

$$\lambda_{X_V}(u, X_V) = p(u)\mathbb{E}[\lambda_X(u, X)|X_T] \quad a.s.$$

and when X is a Poisson process with (conditional) intensity $\rho(\cdot) = \lambda_X(\cdot, \cdot)$, then the intensity function of X_V is given as,

$$\rho_{X_V}(u) = p(u)\rho_X(u)$$

We now can define the Point Process Prediction Errors of Cronie et al. [2023].

Definition 2.3.2 (Point Process Prediction Error)**.**
Let $\Xi = \{\xi_\theta : \theta \in \Theta\}$ be an parametric estimator family and let $\mathcal{H}_\theta = \{h_\theta : \theta \in \Theta\}$ be a parametrized set of functions mapping from $W \times \mathcal{X}_S \to \mathbb{R}$. Let $\mathbb{x}, \mathbb{y} \in \mathcal{X}_S(W)$. Then we call

$$\mathcal{I}^{h_\theta}_{\xi_\theta}(W, \mathbb{x}, \mathbb{y}) = \sum_{x \in \mathbb{x}} h(x, \mathbb{y}) - \int_W h_\theta(u, \mathbb{y})\xi_\theta(u, \mathbb{y})\, du \tag{2.26}$$

Point Process Prediction Error for $\mathbb{y}$ to $\mathbb{x}$.

This already could be interpreted as a loss function when splitting the data into training and validation set. We however still need some properties about ξ_θ especially when Equation (2.26) is fitted with a training and validation set resulting from a thinning.

Theorem 2.3.2 (Theorem 2 and Corollary 1 from Cronie et al. [2023])**.**
Given a point process X with conditional intensity $\lambda_{\theta_0}(u; X)$, $u \in S$, let X_V be an arbitrary thinning of X and let $X_T = X \setminus X_V$. In addition, consider the marked point process $\breve{X} = \{(x, 0) : x \in X^T\} \cup \{(x, 1) : x \in X^V\}$ with mark space $M = \{0, 1\}$ and conditional intensity $\breve{\lambda}_{\theta_0}\{(u, m); \breve{X}\}$, $(u, m) \in S \times M$. For any parameterised model Ξ_Θ, test function $\mathcal{H}_\Theta$ and $W \subset S$, it follows that

$$\mathbb{E}\left[\mathcal{I}^{h_\theta}_{\xi_\theta}(W, X_V, X_T)\right] = \int_W \mathbb{E}\left[h_\theta(u; X_T)\left(\{\breve{\lambda}_{\theta_0}\{(u, 1); \breve{X}\} - \xi_\theta(u; X_T)\right)\right] u \tag{2.27}$$

Note, in particular, that we may let $\mathcal{H}_\Theta = \{h\}$ consist of only one function h. If further $\mathbb{E}[\breve{\lambda}_{\theta_0}\{(u, 1); \breve{X}\}^2]$ and $\mathbb{E}[h_\theta(u; X_T)^2]$, $\theta \in \Theta$, are finite for $|\cdot|$-almost every $u \in S$, then $\mathbb{E}[\mathcal{I}^{h_\theta}_{\xi_\theta}(W; X_V, X_T)] = 0$ for any $W \subset S$ if and only if

$$\begin{aligned} \xi_\theta(u; X_T) =& w_{\theta_0}(u, X)\lambda_{\theta_0}(u; X_T), \\ w_{\theta_0}(u, X) =& \mathbb{E}\left[\frac{\breve{\lambda}_{\theta_0}\{(u, 1); \breve{X}\}}{\lambda_{\theta_0}(u; X_T)}\middle| X_T\right]. \end{aligned} \tag{2.28}$$

In particular, when X_V is an independent thinning of X, based on the retention probability function $p(u) \in (0, 1)$, $u \in S$, then

$$w_{\theta_0}(u, X) = p(u)\, \mathbb{E}\left[\frac{\lambda_{\theta_0}(u; X)}{\lambda_{\theta_0}(u; X_T)}\middle| X_T\right],$$

where $w_{\theta_0}(u) = p(u)$ if X is a Poisson process, $w_{\theta_0}(u) \geq p(u)$ if X is attractive and $w_{\theta_0}(u) \leq p(u)$ if X is repulsive.

2.3.1 Method for Point Process Learning

With help of Theorem 2.3.2 we can construct a proper loss function similar to how it was done in Definition 2.1.15, by splitting the data into training and validation sets.

Definition 2.3.3 (Simple Loss Function)**.**
Let $\mathbb{x}$ be an observed point pattern and let $\mathbb{x}_T, \mathbb{x}_T \subset W$ result from an independent

thinning of $\mathbb{x}$ with retention probability $p(\cdot)$. Then we call

$$L^h_{\xi_\theta}(W, \mathbb{x}_T, \mathbb{x}_V) = \left| \sum_{x \in \mathbb{x}_V} h(x, \mathbb{x}_T) - \int_W h(u, \mathbb{x}_T) \xi_\theta(u, \mathbb{x}_T) \, du \right| \tag{2.29}$$

the simple loss function.

This loss function is now our basis for Point Process Learning.

Definition 2.3.4 (Point Process Learning)**.**
We say that any method which generates estimates $\hat{\theta}$ in Θ by exploiting Equation (2.29) *belongs to the field of Point Process Learning.*

Definition 2.3.3 already gives us a method for Point Process Learning. In Cronie et al. [2023] also the idea of cross-validation for Point Process Learning was introduced.

2.3.1.1 Cross-Validation for Point Process Learning

A known method for preventing overfitting is cross-validation [Ng et al., 1997]. Cross-validation splits data X into k parts $X^1, \ldots, X^k$. We then can repeatedly split our data into a validation-set X^i_V and a training set $X^i_T = X^i \setminus X^i_V$. Cronie et al. [2023] provided the first rigorous definition of cross-validation for point processes and looked closer at two specific cross-validation methods.

A multinomial distribution with $p_1 = \ldots = p_k = \frac{1}{k}$ can be used to independently create a mark $m(x)$ for every $x \in \mathbb{x}$. We then define our validation and test sets by

$$\mathbb{x}^i_V = \{x \in \mathbb{x} : m(x) = i\} \quad \text{and} \quad \mathbb{x}^i_T = \mathbb{x} \setminus \mathbb{x}^i_V. \tag{2.30}$$

Cronie et al. [2023] called this multinomial k-fold cross-validation. The other method which was studied was so-called Monte-Carlo cross-validation [Cronie et al., 2023]. We here independently generate k thinnings $\mathbb{x}^1_V, \ldots, \mathbb{x}^k_V$, where $\mathbb{x}^i_V$ is created by independently marking every $x \in \mathbb{x}$ with $m : \mathbb{x} \to \{0, 1\}$ where $\mathbb{P}[m(x) = 1] = p$. We then obtain

$$\mathbb{x}^i_V = \{x \in \mathbb{x} : m(x) = 1\},$$

and thus repeat this marking k times. The validation- and training-sets are here given by

$$\mathbb{x}^i_V \quad \text{and} \quad \mathbb{x}^i_T = \mathbb{x} \setminus \mathbb{x}^i_V \quad i = 1, \ldots, k$$

Motivated by the fact that a prediction error has expectation zero if and only if we make the right parameter choice, Cronie et al. [2023] introduced the following cross-validation-based loss functions.

Definition 2.3.5 (Cross-Validation Loss Function)**.**
Let $\mathbb{x}$ be an observed point pattern and let $\mathbb{x}_T^1, \mathbb{x}_V^1, \ldots, \mathbb{x}_T^k, \mathbb{x}_V^k \subset W$ be training-validation pairs resulting from independent thinning of $\mathbb{x}$ with retention probability $p(\cdot)$, depending on the choice of $j \in \{1, 2, 3\}$, we optimize

$$\begin{aligned}\mathcal{L}_j(\theta) &= \sum_{i=1}^{k} \left[\left| \sum_{x \in \mathbb{x}_V^i} h(x, \mathbb{x}_T^i) - \int_W h(u, \mathbb{x}_T^i) \xi_\theta(u, \mathbb{x}_T^i)\, du \right| \right]^j \quad j \in \{1, 2\} \\ \mathcal{L}_3(\theta) &= \left[\sum_{i=1}^{k} \sum_{x \in \mathbb{x}_V^i} h(x, \mathbb{x}_T^i) - \int_W h(u, \mathbb{x}_T^i) \xi_\theta(u, \mathbb{x}_T^i)\, du \right]^2 .\end{aligned} \tag{2.31}$$

In case of $j = 2$, for example, we try to rather have a balanced loss for every $L^{(i)}$, $i = 1, \ldots, k$ than try to reduce a single $L^{(i)}$ to zero [Ng et al., 1997]. This does indeed prevent overfitting since the variance will be reduced due to the balanced loss. The bias of our model might however increase.

2.4 Data Augmentation in Point Process Learning

As already mentioned in the beginning of the thesis, we hope to generate some regularizing effect in Point Process Learning with help of data augmentation. The goal is therefore to adapt common data augmentation procedures to the field of Point Process Learning.
There is however still the question of how we add the noise or augment data in case of point processes. We will discuss this in detail in the upcoming chapters. We however already introduce the main ideas in general and give a brief overview before diving deeper into the theoretical backgrounds.

Remark (Notation). When a point process X is is noised or disturbed we refer to the noised process as X^*. The noised realization is then indicated by $\mathbb{x}^*$.

In this thesis we present two general ways of manipulating point processes. The first method would be to leave the points in the process and just add additional noise Ψ. The noise itself can then also be generated by a point process, yielding a noised process

$$X^* = X \cup \Psi. \tag{2.32}$$

The process Ψ itself can often be chosen arbitrary and in this work we will choose Ψ to be independent of X for simplicity. Note, however, that this does not necessarily have to be the case, when adding additional data points.

Definition 2.4.1. *We call two point processes X and Y independent if the random variables $X(A)$ and $Y(A)$ are independent for every bounded set $A \in \mathcal{S}$*

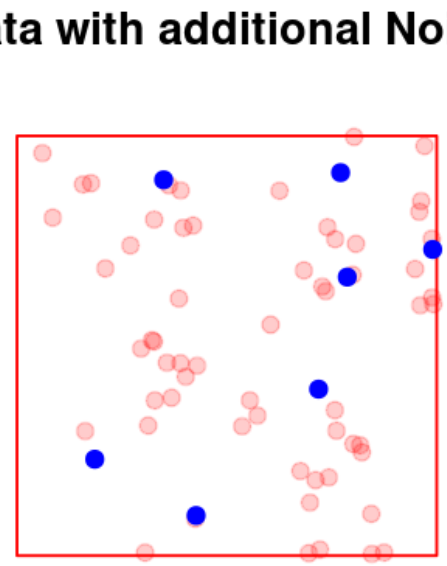


Figure 2.8: The red dots indicate the original process, where the blue points describe the the additional points we added to the original data to disturb it.

The second method will be based on shifting points. The idea is that every point in the point pattern gets shifted in space according to a distribution. Therefore we can write our disturbed point process as

$$X^* = X + \epsilon \quad \text{with } \epsilon \in \mathbb{R}^{d \times X(W)}, \quad \text{and } S = \mathbb{R}^d, \tag{2.33}$$

where ϵ, with the amount of entries as the amount of points in the process X, is an offset shifting our point process in space.

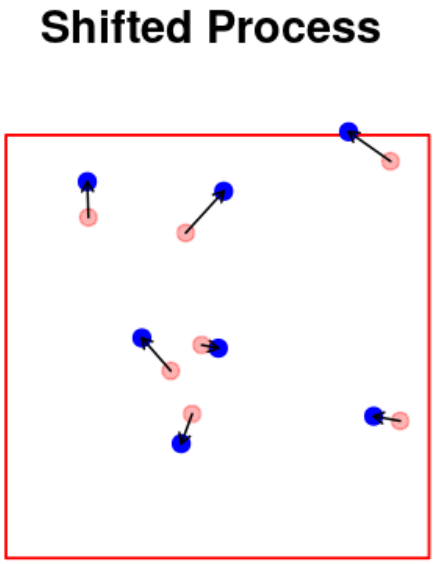


Figure 2.9: The red dots indicate the original process, where the blue points describe the new process. The arrows show how each point was shifted in space.

Both methods can of course be combined. We will, however, investigate the methods separately in this thesis. Note that also each point in the process can be shifted individually if desired.

3

Adding Noise to Point Processes

We now can present our ideas of adding noise to the data in more detail. As already indicated, this chapter is about one of two general ideas for augmenting point processes/patterns. We start with focusing on the idea of adding additional points with help of a further point process, also called noise process Ψ.
We will therefore derive properties about the superposition of different point processes, since these properties will be important for further developing Point Process Learning methods, which will involve noised data. We then investigate the behavior of the conditional intensity of a noised process. This will later play a crucial role for our method, since the estimation is based on the GNZ formula, which makes use of the conditional intensity.

The further part of the chapter will focus on ideas how to apply Point Process Learning on noised data sets. We therefore want to adapt the method from Cronie et al. [2023], which is used for estimating Gibbs densities. Several options are discussed where the noise should be included in the training process. We also discuss the behaviour of estimators with respect to the added noise.

In general, we can summarize the goals for this chapter in three main points:

- We derive properties about superposition of point processes.
- We discuss options to include noise in the training process and develop suitable methods, which are related to the already introduced method by Cronie et al. [2023]
- We further discuss behaviours of estimators with respect to the intensity of noise.

3.1 Properties of Superpositions of Point Processes

Definition 3.1.1. *A superposition $X \cup Y$ of point processes X and Y is the process resulting from the combination of the two point processes. Therefore, the random variable $X \cup Y(W)$ is describes as*

$$X \cup Y(W) = X(W) + Y(W) \quad \forall\, W \in \mathcal{S}. \tag{3.1}$$

Note that X and Y do not necessarily have to be independent.

A typical example is given by the superposition of two Poisson processes, where the superposition itself is again a Poisson process.

Lemma 3.1.1. *Let X and Y be two independent Poisson processes in $W \subset S$ with intensity functions ρ_X and ρ_Y. Then their superposition $X \cup Y$ is a Poisson process with intensity function $\rho_{X\cup Y}(\cdot) = \rho_X(\cdot) + \rho_Y(\cdot)$.*

Proof. Since X and Y are independent it holds that

$$\mathbb{P}\left[X \cup Y(A) = n\right] = \sum_{k=0}^{n} \mathbb{P}\left[N_X(A) = k\right] \cdot \mathbb{P}\left[N_Y(A) = n-k\right]. \tag{3.2}$$

Since X and Y are Poisson distributed with the integrated intensity function as means, we get

$$\begin{aligned}
\mathbb{P}\left[X \cup Y(A) = n\right] &= \sum_{k=0}^{n} \exp\left(-\int_A \rho_X(x)\ dx\right) \frac{\left(\int_A \rho_X(x)\ dx\right)^k}{k!} \cdot \\
&\qquad \exp\left(-\int_A \rho_Y(x)\ dx\right) \frac{\left(\int_A \rho_Y(x)\ dx\right)^{n-k}}{(n-k)!} \\
&= \sum_{k=0}^{n} \exp\left(-\int_A \rho_X(x) + \rho_Y(x)\ dx\right) \frac{\left(\int_A \rho_X(x)\ dx\right)^k \left(\int_A \rho_Y(x)\ dx\right)^{n-k}}{k!(n-k)!} \\
&= \frac{1}{n!} \exp\left(-\int_A \rho_X(x) + \rho_Y(x)\ dx\right) \sum_{k=0}^{n} \binom{n}{k} \left(\int_A \rho_X(x)\ dx\right)^k \left(\int_A \rho_Y(x)\ dx\right)^{n-k} \\
&= \frac{1}{n!} \exp\left(-\int_A \rho_X(x) + \rho_Y(x)\ dx\right) \left(\int_A \rho_X(x)\ dx + \int_A \rho_Y(x)\ dx\right)^n.
\end{aligned}$$

In the last step we used the binomial theorem. It now follows that

$$X \cup Y \sim Pois\left(\int_A \rho_X(x) + \rho_Y(x)\ dx\right)$$

□

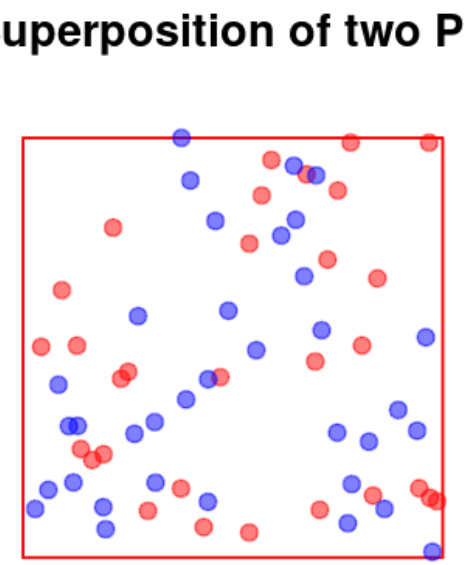


Figure 3.1: The plot indicates the superposition of two Poisson processes. The superposition is itself again a Poisson process.

Remark. Note that a thinning, which was defined in 2.3.1, also can be interpreted as a superposition of point processes. The superposition of X_V and X_T is then given by $X_V \cup X_T = X$. Note that thinnings are not independent to each other since it has to hold that $X_T(A) + X_V(A) = X(A)$ for all $A \subset W$.

3.1.1 Conditional Intensities of Superposition

We now want to investigate the behaviour of the conditional intensity of a superposition. We already saw an example for the superposition of two Poisson processes in Lemma 3.1.1 where the superposition is itself a Poisson process. We now want to study a more general case. Therefore, we consider processes where the conditional intensity is not constant over the second argument $\mathcal{X}_S$. We however make the assumption that the two processes are independent, since otherwise we would have a difficult scenario, which could heavily be influenced by the type of process or the dependence structure considered. This does not really seem to be that promising.

Theorem 3.1.2.
Let X and Y in $W \subset S$ be two arbitrary independent point processes with conditional intensities

$$\lambda_X, \lambda_Y : S \times \mathcal{X}_S \to \mathbb{R}.$$

Then, the term

$$\lambda_{X \cup Y}(u, X \cup Y) = \lambda_X(u; X) + \lambda_Y(u; Y)$$

fullfills the GNZ fromula i.e.

$$\mathbb{E}\left[\sum_{x\in X\cup Y} h(x, X\cup Y)\right] = \mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_{X\cup Y}(u, X\cup Y)\ du\right]$$

Proof. Since X and Y are independent, we can separate the sum and get

$$\mathbb{E}\left[\sum_{x\in X\cup Y} h(x, X\cup Y)\right] = \mathbb{E}\left[\sum_{x\in X} h(x, X\cup Y)\right] + \mathbb{E}\left[\sum_{x\in Y} h(x, X\cup Y)\right].$$

We can now again apply the GNZ formula to the separated sums over X and Y. We first focus on the sum over X and get

$$\mathbb{E}\left[\mathbb{E}\left[\sum_{x\in X} h(x, X\cup Y)\Big|Y\right]\right] = \mathbb{E}\left[\mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_X(u; X)\ du\Big|Y\right]\right],$$

which reduces again with the independence of X and Y and the law of total expectation to,

$$\mathbb{E}\left[\mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_X(u; X)\ du\Big|Y\right]\right] = \mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_X(u; X)\ du\right].$$

By applying the GNZ formula to the sum over Y as well, we obtain

$$\begin{aligned}\mathbb{E}\left[\sum_{x\in X\cup Y} h(x, X\cup Y)\right] &= \mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_X(u; X)\ du\right] \\ &\quad + \mathbb{E}\left[\int_W h(u, X\cup Y)\lambda_Y(u; Y)\ du\right] \\ &= \mathbb{E}\left[\int_W h(u, X\cup Y)\left(\lambda_X(u; X) + \lambda_Y(u; Y)\right)\ du\right].\end{aligned}$$

□

The paper by Decreusefond and Vasseur [2018] also stated this result, but their approach to the proof is however slightly different.

As an extension of Theorem 3.1.2 we can give an even more general statement about the superposition of point processes.

Corollary 3.1.2.1. *Let $X_1, X_2, \ldots, X_n$ be independent point processes in W with conditional intensities $\lambda_{X_1}, \lambda_{X_2}, \ldots, \lambda_{X_n}$. Then it holds that,*

$$\mathbb{E}\left[\sum_{x\in\bigcup_{i=1}^n X_i} h(x, X\cup Y)\right] = \mathbb{E}\left[\int_W h(u, \bigcup_{i=1}^n X_i)\sum_{i=1}^n \lambda_{X_i}(u; X_i)\ du\right]. \tag{3.3}$$

Remark (Superposition as Ground Process).
A superposition $\bigcup_{i=1}^{n} X_i$ can be also viewed as a ground process of a multivariate marked point process with the mark space $\{1, \ldots, n\}$ and $X_i = \breve{X}_i(\cdot \times i)$, then

$$\sum_{i=1}^{n} X_i(\cdot) = \breve{X}(\cdot \times \{1, \ldots, n\}) = X_g(\cdot) = \bigcup_{i=1}^{n} X_i.$$

In Lemma 3.1.1, about the superposition of two Poisson processes, we discovered that the superposition is itself a Poisson process. In general, this is however not the case for parametric Gibbs densities. Take for example two hard-core processes X and Y, both parametrized with the identical β and R. Then

$$\lambda_{X\cup Y}(u; X \cup Y) = \begin{cases} 2\beta & \text{if } u \in W \setminus \bigcup_{x\in X\cup Y} B_R(x) \\ 0 & \text{if } u \in \bigcup_{x\in X\cap Y} B_R(x) \\ \beta & \text{else,} \end{cases}$$

which is now a function mapping to three values. The conditional intensity of a hard-core process however only maps to two values.

It might even be the case that the conditional intensity of a superposition is not well-defined.

Corollary 3.1.2.2. *Let X and Y be point processes in W, with existing conditional intensities, λ_X and λ_Y. Then the conditional intensity $\lambda_{X\cup Y}$ is not necessarily a well-defined function from $S \times \mathcal{X}_S \to \mathbb{R}$.*

Proof. For the proof, we quickly give a counter example. Let $\aleph \in \mathcal{X}_S$ with $\lambda_X(u; \aleph) \neq \lambda_Y(u; \aleph)$ for all $u \in A$, where $A \subset W$ is a set with positive measure. Furthermore, let $\lambda_X(u; \aleph) \neq \lambda_X(u; \emptyset)$ for $u \in A$, then it holds that

$$\lambda_{X\cup Y}(u; \aleph) = \lambda_X(u; \aleph) + \lambda_Y(u; \emptyset) \neq \lambda_X(u; \emptyset) + \lambda_X(u; \aleph) = \lambda_{X\cup Y}(u; \aleph),$$

which indicates that $\lambda_{X\cup Y}$ is not a well-defined function.

□

Corollary 3.1.2.3 (GNZ with noise)**.**
Let $X^* = X \cup \Psi$ be a noised Gibbs process in $W \subset S$, where Ψ and X are processes in W. Let h be a non-negative and integrable function with $h : S \times \mathcal{X}_S \to \mathbb{R}$, then

$$\mathbb{E}\left[\sum_{x\in X^*} h(x, X^* \setminus \{x\})\right] = \int_W \mathbb{E}\left[h(u, X^*)\lambda_{X^*}(u; X^*)\right]\, du. \tag{3.4}$$

When X and Ψ are independent it is equivalent to

$$\begin{aligned} &\mathbb{E}\left[\sum_{x\in X} h(x, X^* \setminus \{x\})\right] + \mathbb{E}\left[\sum_{x\in \Psi} h(x, X^* \setminus \{x\})\right] \\ &= \int_W \mathbb{E}\left[h(u, X^*)\lambda_X(u; X)\right]\, du + \int_W \mathbb{E}\left[h(u, X^*)\lambda_\Psi(u; \Psi)\right]\, du \end{aligned}$$

The last equation directly follows from Theorem 3.1.2, which allows us to separate the conditional intensities.

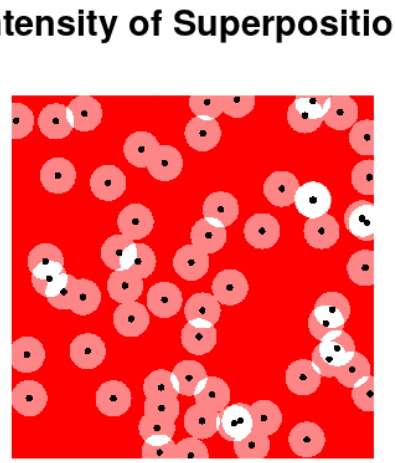


Figure 3.2: The plot shows the conditional intensity of the superposition of two hard-core processes. Note that the superposition itself is not a hard-core process anymore.

3.1.2 Repulsive and Attractive Superpositions

With the previously derived Corollary 3.1.2.1 we now can make assumptions about the repulsiveness and the attractiveness of the superposition.

Theorem 3.1.3. *Let $\bigcup_{i=1}^{n} X_i$ be the superposition of n independent attractive point processes, then $\bigcup_{i=1}^{n} X_i$ is weakly attractive, which means that*

$$\lambda_{\bigcup_{i=1}^{n} X_i}(x; \bigcup_{i=1}^{n} \aleph_i \setminus \{x\}) \geq \lambda_{\bigcup_{i=1}^{n} X_i}(x; \emptyset).$$

If $X_1, \ldots, X_n$ are repulsive then the superposition $\bigcup_{i=1}^{n} X_i$ is weakly repulsive i.e.,

$$\lambda_{\bigcup_{i=1}^{n} X_i}(x; \bigcup_{i=1}^{n} \aleph_i \setminus \{x\}) \leq \lambda_{\bigcup_{i=1}^{n} X_i}(x; \emptyset).$$

Proof. The conditional intensity of $\bigcup_{i=1}^{n} X_i$ is given by

$$\lambda_{\bigcup_{i=1}^{n} X_i}(x; \bigcup_{i=1}^{n} \aleph_i \setminus \{x\}) = \sum_{i=1}^{n} \lambda_{X_i}(x; \aleph_i \setminus \{x\}).$$

Then it holds that

$$\sum_{i=1}^{n} \lambda_{X_i}(x; \aleph_i) \geq \sum_{i=1}^{n} \lambda_{X_i}(x; \emptyset) \quad \forall i \in \{1, \ldots, n\}$$

if X_i is attractive and

$$\sum_{i=1}^{n} \lambda_{X_i}(x; \aleph_i) \leq \sum_{i=1}^{n} \lambda_{X_i}(x; \emptyset) \quad \forall i \in \{1, \ldots, n\}$$

if X_i is repulsive.

□

Note that strict repulsiveness or attractiveness does not necessarily hold for the superposition. Take for example the superposition of a hard-core process X and a Poisson process Y with intensity μ, then the realization of the superposition is given by

$$\mathbb{x} \cup \mathbb{y} = \{x_1, \dots, x_n, y_1, \dots, y_m\}.$$

The conditional intensity of the superposition is given by

$$\lambda_{X\cup Y}(u, \mathbb{x} \cup \mathbb{y}) = \lambda_X(u, \mathbb{x}) + \lambda_Y(u, \mathbb{y}) = \beta \mathbb{1}\{W \setminus \bigcup_{x\in\mathbb{x}} B_R(x)\} + \mu.$$

Now consider $\mathbb{z} = \{x_1, \dots, x_n, y_1, \dots, y_{m-1}\} \subset \mathbb{x} \cup \mathbb{y}$. It then holds that

$$\lambda_{X\cup Y}(u, \mathbb{x} \cup \mathbb{y}) = \lambda_X(u, \mathbb{x}) + \lambda_Y(u, \mathbb{y}) > \lambda_X(u, \mathbb{z}) + \lambda_Y(u, \emptyset) = \lambda_{X\cup Y}(u, \mathbb{z}),$$

for $u \in \bigcup_{y\in\{y_1,\dots,y_{m-1}\}} B_R(y)$, which indicates that the process is not repulsive even though X and Y are repulsive.

Remark. Consider a parametric Gibbs density f_θ with conditional intensity λ_θ, parameterized by $\theta \in \Theta$. Then it holds that

$$\lambda_\theta(u, \mathbb{x}) \leq \lambda_\theta(u; \mathbb{x}^*) = \lambda_\theta(u; \mathbb{x} \cup \psi) \quad \forall u \in W \tag{3.5}$$

if X is attractive and if X is repulsive it holds that,

$$\lambda_\theta(u; \mathbb{x}) \geq \lambda_\theta(u; \mathbb{x}^*) = \lambda_\theta(u; \mathbb{x} \cup \psi) \quad \forall u \in W, \tag{3.6}$$

where ψ is a realization of an arbitrary noise process Ψ.

We now want to discuss different options for how to insert noise into the estimation procedure. In the next section we define different methods based on our findings. We will present methods where only the validation set or the training set is noised. We also discuss a method where both the training and the validation sets are noised. It is important to note that every presented version's performance is heavily depending on the underlying process and its parametric family of estimators. It is therefore hard to tell which option is the best. We thus simply present and analyse different options.

3.2 Adding Noise to Validation and Training Data

We first want to focus on Point Process Learning with additional noise. The method from Cronie et al. [2023] is based on splitting the selected data into a training set X_T and a validation set X_V, which are thinned point processes. We now want to investigate the thinning of noised processes and apply our findings to the method from Cronie et al. [2023].

Theorem 3.2.1. *Let X^* be our noised process in W and let X_V^* be an independent thinning with retention probability $p(\cdot)$, splitting X^* into training and validation sets X_V^* and $X_T^* = X^* \setminus X_V^*$. Furthermore, if $\theta_0 \in \Theta$ is unique and it holds that $\mathbb{E}[L^h_{\xi_{\theta_0}}(W, X_T^*, X_V^*)] = 0$ with*

$$L^h_{\xi_\theta}(W, X_T^*, X_V^*) = \sum_{x \in X_V^*} h(x, X_T^*) - \int_W h(u, X_T^*)\xi_\theta(u, X_T^*)\, du, \tag{3.7}$$

then it follows that

$$\xi_{\theta_0}(u, X_T^*) = w(u, X_T^*, X_V^*)\lambda_{X^*}(u; X_T^*) \quad a.e.,$$

where w is given by

$$w(u, X_T^*, X_V^*) = \frac{p(u)}{\lambda_{X^*}(u; X_T^*)}\mathbb{E}[\lambda_{X^*}(u; X^*)|X_T^*].$$

Proof. The proof is based on the theorems of Cronie et al. [2023] and we quickly repeat the steps. First we use Theorem 2.3.1 to obtain

$$\mathbb{E}\left[\sum_{x \in X_V^*} h(x, X_T^*)\right] = \mathbb{E}\left[\sum_{x \in X_T^*} h(x, X_T^*)\frac{p(x)}{1 - p(x)}\right].$$

The rest directly follows from Theorem 2.3.2, which gives us the following equation:

$$\begin{aligned}
\mathbb{E}\left[\sum_{x \in X_V^*} h(x, X_T^*)\right] =& \mathbb{E}\left[\int_W h(u, X_T^*)\frac{p(u)}{1 - p(u)}\lambda_{X_T^*}(u; X_T^*)\, du\right] \\
=& \mathbb{E}\left[\int_W h(u, X_T^*)p(u)\mathbb{E}[\lambda_{X^*}(u; X^*)|X_T^*]\, du\right] \\
=& \mathbb{E}\left[\int_W h(u, X_T^*)\xi_{\theta_0}(u, X_T^*)\, du\right].
\end{aligned}$$

□

Remark. The derivation follows exactly the derivation of the estimator in our reference paper [Cronie et al., 2023]. Note that in this case we could allow Ψ to be dependent on X in (3.4). We just need to have the independent thinning splitting the noise into validation and training sets.

3.2.1 Definition of the Method

We now give a description of the method using noise in the validation and training sets. Let X be our process with realization $\mathbb{x}$ and let Ψ be the noise process with realization ψ.

- We first consider a parametric family Ξ_Θ which contains the conditional intensity of the process X without the noise.
- We take a realization ψ of the noise process Ψ and noise our data set $\mathbb{x}$.
- We split our noised data $\mathbb{x}^* = \mathbb{x} \cup \psi$ into training and validation sets k times, which results in $\mathbb{x}_T^{1*}, \mathbb{x}_V^{1*}, \ldots, \mathbb{x}_T^{k*}, \mathbb{x}_V^{k*}$.
- We optimize the loss function and get an estimate

$$\hat{\theta} = \arg\min_{\theta\in\Theta} \mathcal{L}_j(\theta) = \arg\min_{\theta\in\Theta} \sum_{i=1}^{K} \left[\left| \sum_{x\in\mathbb{x}_V^{i*}} h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*} \xi_\theta(u, \mathbb{x}_T^{i*})\ du) \right| \right]^j,$$

where $j \in \{1, 2\}$ depending on the choice of loss function. In the case of the L_3-loss we minimize

$$\hat{\theta} = \arg\min_{\theta\in\Theta} \mathcal{L}_3(\theta) = \arg\min_{\theta\in\Theta} \left[\sum_{i=1}^{K} \sum_{x\in\mathbb{x}_v^{i*}} h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*}) \xi_\theta(u, \mathbb{x}_T^{i*})\ du \right]^2.$$

Note that the data sets $\mathbb{x}^{i*}$ are slightly different copies from each other, which is a typical scenario for data augmentation [van Dyk and Meng, 2001].

3.2.2 Remarks on the Method

Before we perform a simulation study with the introduced method, we need to discuss some behaviours of the method, when estimating the process.

3.2.2.1 Estimator Family for the noised Process

The estimator derived in the section is a nice result. There is, however, one fact that can be a drawback in its current form. In our setup, we normally have a parameterized estimators family given as

$$\Xi_\theta = \{\xi_\theta : \theta \in \theta\}.$$

Elements of this class, however, usually describe the conditional intensities of X without the added noise. It can therefore often be assumed that it holds,

$$w(u, X_T, X_V)\lambda_X(\cdot; X_T) \in \Xi_\theta \tag{3.8}$$

However, this is not necessarily the case for the intensity function of the process $\lambda_{X^*}(\cdot; X^*)$, i.e.

$$w(u, X_T^*, X_V^*)\lambda_{X^*}(\cdot; X_T^*) = p(u)E[\lambda_{X^*}(u; X^*)|X_T^*] \notin \Xi_\theta$$

For illustration consider the following simple example. Let X be a hard-core process with unknown parameters R and β. The estimator family is therefore given based on the already introduced density of the hard-core process with

$$\Xi_{\beta,R} = \{w\lambda_{\beta,R} : \beta, R \in (0,\infty)\},$$

where $\lambda_{\beta,R}$ is given in Equation (3.8) and the parametrized conditional intensity of the training set is, as already described in 2.1.6.2, given by

$$\lambda_{\beta,R}(u; X_T) = \begin{cases} \beta & \text{if } \; u \in W \setminus \bigcup_{x \in X_T} B_R(x) \\ 0 & \text{else.} \end{cases} .$$

Now assume that our noise process Ψ is a homogeneous Poisson process with intensity μ. Then the conditional intensity of the superposition X^* is given by

$$\lambda_{X^*}(u, X^*) = \begin{cases} \beta + \mu & \text{if } \; u \in W \setminus \bigcup_{x \in X} B_R(x) \\ \mu & \text{else.} \end{cases}$$

Therefore the intensity function of the noised process is clearly not a member of our previously defined parametric family Ξ_θ, since it holds for fixed $\beta, R > 0$ that,

$$w(u, X_T^*, X_V^*)\lambda_{\beta,R}(u, X_T^*) = \mathbb{E}[\lambda_{\beta,R}(u, X^*)|X_T^*] \neq \mathbb{E}[\lambda_{X^*}(u, X^*)|X_T^*].$$

An illustration can be seen in Figure 3.3.

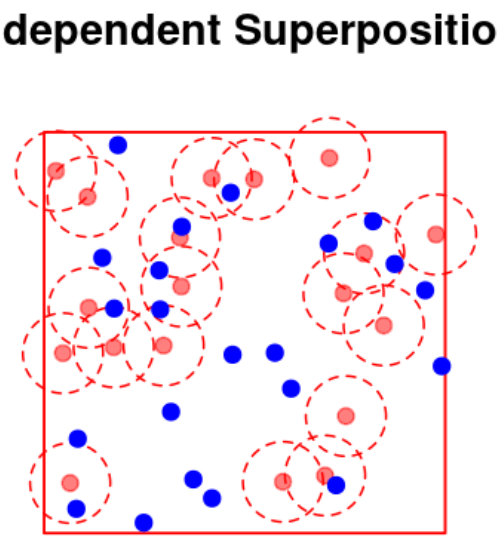


Figure 3.3: The plot shows the the superposition of two independent processes. The red colored points indicate the hard-core process. The blue points are realized from a Poisson process. The circle around the red points indicates the parameter R of the hard-core process which was chosen to be $R = 0.1$. Note that there are blue points in a circle. This means the intensity is not zero there.

As a solution we could extend the estimator family by for example adding an offset μ as a parameter. In general however this extension would heavily depend on the noise process Ψ and would also lead to a huge increase of complexity. We also only have interest in estimating the distribution of X. We therefore consider this option to be not really promising and we will not investigate it any deeper.
In general it is however not a problem to not have the best possible estimator ξ_{θ_0} in our estimator family Ξ, it might even be a desired effect. We would then trade a slightly higher bias against a lower variance in the optimal case. Indeed restricting the amount of parameters in a distribution can simplify a model. Therefore it is worth considering to simply not adapt the parametric family and estimate the parameters β and R based on the noised data.

3.2.2.2 Analysis of the Estimator

We now further want to analyze the behaviour of our estimator if noise is added. The behaviour is again depending on the function h. A general statement is hard to give. Therefore we restrict the learning process and derive some properties about the estimators.

We first fix the test function h with

$$h(x, X_T \setminus \{x\}) = \frac{1}{\xi_\theta(x, X_T)}, \tag{3.9}$$

where $\theta \in \Theta$. The function h in Equation (3.9) is chosen as in our reference paper from Cronie et al. [2023]. This idea for the test function was first introduced by Stoyan and Grabarnik [1991].
We also demand that the weight function is constant since it also depends on the specific underlying process and is therefore hard to deal with. As a constant for w we take the proposed approximation from Cronie et al. [2023] which they also used for the simulation study. For a fixed p the approximation is given by,

$$\begin{aligned} w(u, X_T, X_V) &= p\frac{\mathbb{E}[\lambda_\theta(u; X)|X_T]}{\lambda_\theta(u; X_T)} = p\frac{\lambda_\theta(u; X_T) + \mathbb{E}[\lambda_\theta(u; X_V)|X_T]}{\lambda_\theta(u; X_T)} \\ &\approx p + \frac{p^2}{1-p} \end{aligned} \tag{3.10}$$

Theorem 3.2.2. *Let X be a repulsive process in W and let λ_θ be the conditional intensity resulting from parametric Gibbs density for X. Assume the fixed test function, as in Equation* (3.9)*, and an approximated weight function, as in Equation* (3.10)*, for the training process. Then for a fixed $\theta \in \Theta$ it holds that*

$$L^h_{\xi_\theta}(W, X_T, X_V) \leq L^h_{\xi_\theta}(W, X^*_T, X^*_V)$$

Proof. For the proof we use Remark 3.1.2. We start with

$$L^h_{\xi_\theta}(W, X_T, X_V) = \sum_{x \in X^*_V} h(x, X^*_T \setminus \{x\}) - \int_W h(u, X^*_T)\xi_\theta(u, X^*_T)\, du, \tag{3.11}$$

which by the choice of h and the fixed weight $w = p + p^2(1-p)^{-1}$ results in

$$L^h_{\xi_\theta}(W, X_T, X_V) = \sum_{x \in X_V} \frac{1}{w\lambda_\theta(x, X_T \setminus \{x\})} - 1.$$

Since X is repulsive we know that $\lambda_\theta(x, X_T) > \lambda_\theta(x, X^*_T)$, which then results in

$$\begin{aligned} L^h_{\xi_\theta}(W, X_T, X_V) \leq& \sum_{x \in X_V} \frac{1}{w\lambda_\theta(x, X^*_T \setminus \{x\})} - 1 \\ \leq& \sum_{x \in X^*_V} \frac{1}{w\lambda_\theta(x, X^*_T \setminus \{x\})} - 1 \\ =& L^h_{\xi_\theta}(W, X^*_T, X^*_V) \end{aligned}$$

□

In this case $\hat{\theta}$ needs to be selected in a way that artificially increases the value of $\xi_\theta(x, X^*_T \setminus \{x\})$, to compensate the described effect from Theorem 3.2.2. This will lead to a biased estimator in the case of a unique $\theta_0 \in \Theta$.
In the case of X being attractive it holds that

$$\xi_\theta(x, X_T \setminus \{x\}) \leq \xi_\theta(x, X^*_T \setminus \{x\}).$$

We can't however give a proper statement about θ without more knowledge about $\xi_\theta(x, X_T^* \setminus \{x\})$ since the sum over the noised validation set again increases the term

$$\sum_{x \in X_V^*} \frac{1}{\xi_\theta(x, X_T^* \setminus \{x\})} - 1.$$

Intuitively noising only the training data would be the first choice. We will investigate this option in the next section.
Note however that the method from Cronie et al. [2023] belongs to the field of unsupervised learning [Alloghani et al., 2020], even though we have a validation set $\mathbb{X}_V$. The validation set however does not describe our true parameters, which we want to learn, directly. Also both the training set $\mathbb{X}_T$ as also the validation set $\mathbb{X}_V$ have impact on the choice of the parameters in the described method from Cronie et al. [2023] when optimizing Equation (2.29). This is usually not the case in supervised learning, where often only the training set is used to optimize the parameters of the estimator [Cerulli, 2023]. Therefore the choice of noising training and validation set is valid.

3.3 Noising the Test Function

We now want to derive the foundation for other methods to insert noise into the training process. As already indicated we want to investigate the impact of only noising the training data. Therefore we will not add any noise to the validation set. The term of interest is then given by

$$\sum_{x \in X_V} h(x, X_T^* \setminus \{x\}). \tag{3.12}$$

We now assume that the noise Ψ is independent of the training set X_T^*. To obtain some insights about (3.12) it would be great if we could apply Theorem 1 from Cronie et al. [2023], which was stated in 2.3.1. We however have the issue that X_T^* and X_V do not describe an independent thinning with retention probability p. We therefore need to find a work around.

Theorem 3.3.1. *Let Ψ be a point process independent of X. Let $h : S \times \mathcal{X}_S \to \mathbb{R}$ be a non-negative or integrable function. Then the following function*

$$L^{(h)}(X, \Psi) = \sum_{x \in X_V} h(x, X^*) - \int_W h(u, X^*) \lambda_X(u; X)\, du \tag{3.13}$$

has mean zero for a fixed noise pattern, i.e. $\mathbb{E}\left[L^{(h)}(X, \Psi) \middle| \Psi = \psi\right] = 0$ with $\psi \in \mathcal{X}_S$. Where $X^* = X \cup \Psi$ can be interpreted as our training data with additional noise.

Proof. Note that we demanded that the points in Ψ and X are independent. Which means that the points of X are not effected by the pattern Ψ. For a fixed ψ we can define $h^* : S \times \mathcal{X}_S \to \mathbb{R}$ also integrable and non-negative with,

$$h^*(u, X) = \mathbb{E}[h(u, X \cup \psi)]$$

We now apply the GNZ formula with the function h^*. We then obtain

$$\mathbb{E}\left[\sum_{x \in X} h^*(x, X \setminus \{x\})\right] = \mathbb{E}\left[\int_W h(u, X \cup \psi) \lambda_X(u; X)\, du\right]$$

□

We now again can construct a loss function similar to the loss function from Croine et al. in Equation (2.29).

Corollary 3.3.1.1. *Let X be the process in W and let X_V be an independent thinning with retention probability $p(\cdot)$ splitting X into training and validation sets X_V and $X_T = X \setminus X_V$. Let Ψ be an independent noise process and $X_T^* = X_T \cup \Psi$.*

Furthermore if $\theta_0 \in \Theta$ is unique and it holds that $\mathbb{E}[L^{h^*}_{\xi_{\theta_0}}(W, X_T, X_V)|\Psi = \psi] = 0$ for a fixed $\psi \in \mathcal{X}_S$ with

$$L^{h^*}_{\xi_\theta}(W, X_T, X_V) = \sum_{x \in X_V} h(x, X_T^*) - \int_W h(u, X_T^*)\xi_\theta(u, X_T)\ du, \tag{3.14}$$

then it follows that,

$$\xi_{\theta_0}(u, X_T) = w(u, X_T, X_V)\lambda_X(u; X_T) \quad a.e.$$

where w is given by

$$w(u, X_T, X_V) = \frac{p(u)}{\lambda_X(u; X_T)} E[\lambda_X(u; X)|X_T].$$

Proof. The statement directly follows from Theorem 2.3.2 and Theorem 3.3.1. Note that with Theorem 3.3.1 it holds for a fixed noise process ψ with $X^* = X \cup \psi$ that

$$\begin{aligned}
0 =& \mathbb{E}\left[\sum_{x \in X_V} h(x, X_T^*)\right] - \mathbb{E}\left[\int_W h(u, X_T^*)\xi_{\theta_0}(u, X_T)\ du\right] \\
=& \mathbb{E}\left[\int_W h(u, X_T^*)p(u)\mathbb{E}\left[\lambda_X(u, X)|X_T\right]\ du\right] - \mathbb{E}\left[\int_W h(u, X_T^*)\xi_{\theta_0}(u, X_T)\ du\right] \\
=& \mathbb{E}\left[\int_W h(u, X_T^*)\left(w(u, X_T, X_V)\lambda_X(u; X_T) - \xi_{\theta_0}(u, X_T)\right)\ du\right],
\end{aligned}$$

which completes the proof, since h is arbitrary. □

3.3.1 Interpretation

The presented loss function in Corollary 3.3.1.1 is almost identical to the loss function of the method of Cronie et al. [2023], given in Equation (2.3.3), except for the change in the test function. However, it becomes interesting when cross validation is applied. For each training set x_T^i a new realization ψ_i of the noise process can be generated. Accordingly, variation can be introduced in the test function h. This is then just a special case of Takacs-Fiksel estimation. For example take $\psi_1, \ldots, \psi_n$ realized from a noise processes independent of X. We then again define

$$h_i^*(x, X \setminus \{x\}) = h(x, X \cup \psi_i \setminus \{x\}) \quad i \in \{1, \ldots, K\}.$$

We then can then optimize

$$\sum_{i=1}^{K} \left[\mathcal{I}_\theta^{h_i^*}(W, X)\right]^2.$$

Remark (Function Class).
For applying this method to Point Process Learning we would need to adapt the

current implementation of Cronie et al. [2023], since the the current model is restricted to only a small amount of possible test functions, which have the condition that

$$h(u, X) = f(\xi_\theta(u, X)). \tag{3.15}$$

We therefore would need to adapt $\mathcal{H}$ in a way such that we also can insert the noised test set into h.

3.3.2 Definition of the Method

We now give a description of the method using noise only in the training set and only inserting the noised data into the test function h.

Let X be our process with realization $\mathbb{x}$ and let Ψ be the noise process with realization ψ.

- We first define a parametric family Ξ_Θ which describes the conditional intensity of the process X without the noise.
- We split our data $\mathbb{x}$ into training $\mathbb{x}_T^1, \dots, \mathbb{x}_T^k$ and validation sets $\mathbb{x}_V^1, \dots, \mathbb{x}_V^k$.
- We then only noise the training sets $\mathbb{x}_T^1, \dots, \mathbb{x}_T^k$ with the realization $\psi_1, \dots, \psi_k$ of the noised process Ψ.
- We optimize the loss function and get an estimator by

$$\hat{\theta} = \arg\min_{\theta\in\Theta} \mathcal{L}_j = \arg\min_{\theta\in\Theta} \sum_{i=1}^{K} \left[\left| \sum_{x\in\mathbb{x}_V^i} h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*}) \xi_\theta(u, \mathbb{x}_T^i)\, du \right| \right]^j$$

with $j \in \{1, 2\}$ or

$$\hat{\theta} = arg\min_{\theta\in\Theta} \mathcal{L}_3 = \arg\min_{\theta\in\Theta} \left[\sum_{i=1}^{K} \sum_{x\in\mathbb{x}_v^i} h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*}) \xi_\theta(u, \mathbb{x}_T^i)\, du \right]^2,$$

if $j = 3$.

3.4 Other Methods for Inserting Noise

There are still some remaining options to insert noise into the training procedure. So far we discussed the option of training on a noised validation and training set. We also discussed the option to slightly change the test function h by inserting a noised training set only into the test function.

3.4.1 Noise in the Training Set

We could now also include the noised training set into the estimator ξ_θ. In this case we would minimize the following term to obtain $\hat{\theta}$.

$$\begin{aligned}\hat{\theta} &= \arg\min_{\theta\in\Theta} L^h_{\xi_\theta}(W, X_T^*, X_V) \\ &= \arg\min_{\theta\in\Theta} \left| \sum_{x\in X_V} h(x, X_T^*) - \int_W h(u, X_T^*)\xi_\theta(u, X_T^*)\, du \right|.\end{aligned} \tag{3.16}$$

The impact of the noise in the estimator is again hard to quantify without further knowledge about h and the parametric family Ψ. We therefore again consider h to be $\xi_\theta(u, X_T^*)^{-1}$ and w to be constant, approximated with $w = p + p^2(1-p)^{-1}$.

Lemma 3.4.1. *Let X be a process in W and let λ_θ, with $\theta \in \Theta$, model the parametric Gibbs density for X. Assume the fixed test function, as in Equation* (3.9) *and an approximated weight function, as in Equation* (3.10)*, for the training process. Then for a fixed $\theta \in \Theta$ it holds that*

$$L^h_{\xi_\theta}(W, X_T, X_V) \leq L^h_{\xi_\theta}(W, X_T^*, X_V),$$

if X is repulsive and

$$L^h_{\xi_\theta}(W, X_T, X_V) \geq L^h_{\xi_\theta}(W, X_T^*, X_V),$$

if X is attractive.

Proof. By the choice of h and the fixed weight $w = p + p^2(1-p)^{-1}$ the inovation results into

$$L^h_{\xi_\theta}(W, X_T, X_V) = \sum_{x\in X_V} \frac{1}{w\lambda_\theta(x, X_T \setminus \{x\})} - 1.$$

For a fixed $\theta \in \Theta$ it holds that

$$\sum_{x\in X_V} \frac{1}{\xi_\theta(x, X_T)} \geq \sum_{x\in X_V} \frac{1}{\xi_\theta(x, X_T^*)},$$

if X is repulsive, and if X is attractive it holds that

$$\sum_{x\in X_V} \frac{1}{\xi_\theta(x, X_T)} \leq \sum_{x\in X_V} \frac{1}{\xi_\theta(x, X_T^*)}.$$

This directly implies that

$$L^h_{\xi_\theta}(W, X_T, X_V) \leq L^h_{\xi_\theta}(W, X_T^*, X_V),$$

if X is repulsive and

$$L^h_{\xi_\theta}(W, X_T, X_V) \geq L^h_{\xi_\theta}(W, X_T^*, X_V),$$

if X is attractive. □

In case of minimizing the loss function L^h_ξ the parameter θ would then need to be adapted accordingly. This also implies that the estimator would be biased if θ_0 is unique.

3.4.2 Noise in the Validation Set

The other option would be to insert the noise only in the validation set. We would then need to minimize the following term.

$$\begin{aligned}\hat{\theta} &= \arg\min_{\theta\in\Theta} L^h_{\xi_\theta}(W, X_T, X_V^*) \\ &= \arg\min_{\theta\in\Theta} \left| \sum_{x\in X_V^*} h(x, X_T) - \int_W h(u, X_T)\xi_\theta(u, X_T)\ du \right|.\end{aligned} \tag{3.17}$$

Since h is a non-negative function it holds for a fixed weight or a test function, which only depends on the training set X_T that,

$$\sum_{x\in X_V^*} h(x, X_T) = \sum_{x\in X_V} h(x, X_T) + \sum_{x\in\Psi} h(x, X_T) \geq \sum_{x\in X_V} h(x, X_T).$$

We can summarize the result in the following Lemma.

Lemma 3.4.2. *Let X be a process in W and let w be constant in our estimator function $\xi_\theta(u, X_T) = w\lambda_\theta(u; X_T)$. Then it holds that*

$$L^h_{\xi_\theta}(W, X_T, X_V) \leq L^h_{\xi_\theta}(W, X_T^*, X_V).$$

We then again have to compensate the change by adapting $\theta \in \Theta$ to minimize equation 3.17.

3.5 Summary

In this chapter we introduced various options to insert noise into the training process. We investigated the options of noising the training data or the validation data. We presented two main methods, where the first method was based on using the method of Cronie et al. [2023] with noised data and the second method was based on only noising the input of the test function h.
Noising the data can generate similar copies of the process which then can have a regularizing effect on the estimation. Combined with the cross validation we could obtain a regularizing effect by slightly changing the data set every time.
However, this is only a vague assumption, as we have no further information in the present form of the method. To make more precise statements about the effect of the noise, we need a more explicit form of the test function h and of ξ. We therefore assume that the success is strongly dependent on both the point process and the choice of test function. With a more precise and less general representation of the loss function, statements about variance and bias could possibly be made. We therefore refrain from further analysis for the time being and examine the behavior in a simulation study. Subsequently, the results can still be interpreted and explained.
Irrespective of the results of the simulation study, we nevertheless presented interesting results and methods. On the one hand, properties of superpositions of point processes were derived and were for example used for an interesting form of Takacs-Fiskel estimation.

So far we talked only about the training with additional noise. The question that remains is how to add the noise. One of the simplest ways of doing that is to just take a homogeneous Poisson process with intensity μ as the noise process. This would be a simple way to noise the already existing data.
This might however not be the optimal choice depending on the original process. Finding a computationally efficient and solid solution for generating efficient data can be a hard challenge [van Dyk and Meng, 2001]. Often heuristics are applied to generate new data [van Dyk and Meng, 2001]. Therefore finding a suitable point process Ψ to noise our data is not necessarily easy.
Especially if the characteristics of the original process, like for example repulsiveness or attractiveness, shall be included. In Section 7.1 we quickly talk about such a model and how it could be realized. However, we will postpone a detailed analysis and definition for future work, as this would go beyond the scope of this thesis. The idea is on the one hand to use thinnings to control the intensity of the noise process depending on the position $u \in W$, on the other hand we could use the area interaction process to model noise processes that are repulsive and processes that are attractive. The sampling would then have to happen conditionally on the already existing data points.

We should also add that there are more options for manipulating the dataset with additional data points. A data set can also be varied by removing points or replacing some data points with new data points.

4

Simulation Study for the Additional Noise Method

We discussed various methods how to insert the noise into the data set. We will however only focus on experiments based on the method introduced in 3.2.1, where we noised training and validation sets. This in general seems like a interesting choice and more promising than only to noise the validation set or test set. The idea where we noise the input of the test function h described in 3.3 also seems really promising. There will be however follow up work to the paper of Cronie et al. [2023] where a more general set $\mathcal{H}$ for test functions will be proposed. Investigating the impact of noise in the test function will then be more interesting and simpler. We will explain this in more detail in the future work Section 7.1 in the end of the thesis. As a simple and neutral way of adding noise to the data we will chose our noise process to be the Poisson process where we then can investigate the performance with respect to the noise intensity. More about the choice of the noise process can be also found in the future work Section 7.1.

We will perform experiments on two different already introduced point processes. The first process will be the hard-core process introduced in Section 2.1.6.2. The second process will be the area interaction process, introduced in Section 2.1.6.3. We will parameterize the area interaction process in a way that it will be attractive. Since the hard-core process is repulsive we will have a repulsive and attractive process where we apply Point Process Learning.

For the simulation study we assume our space S to be $[0,1]^2$. For the evaluation we use three different metrics. First we are interested in the $MSE(\hat{\theta})$, which is given by

$$MSE(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta_0)^2], \tag{4.1}$$

where $\hat{\theta}$ is given by the mean of the results of n simulations,i.e.

$$MSE(\hat{\theta}) \approx (\frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i - \theta_0)^2 \tag{4.2}$$

where $\hat{\theta}_i$ is the optimal θ for the i-th experiment.
The MSE can be decomposed into bias and variance as shown in Section 1.2. The bias is given by

$$\text{Bias}(\hat{\theta}) = \mathbb{E}[(\hat{\theta}] - \theta_0) \approx (\frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i - \theta_0) \tag{4.3}$$

and the variance is given by

$$\mathrm{Var}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^2] \approx \frac{1}{n}\sum_{i=1}^{n}(\hat{\theta}_i - \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i)^2. \tag{4.4}$$

4.1 Experimental Setup

For investigating additional noise we carry out the following steps in the experiment.

- We fix the parameters k and I where k describes the amount of thinnings we consider for the cross validation. I describes the number of simulations for estimating the already described metrics.
- We fix $\theta_0 \in \Theta$ which will describe the distribution of the generated point process. We refer to the true conditional intensity as $\lambda_{\theta_0}(u; X)$.
- We then fix a vector n which stores different noise intensities which we want to investigate. The noise intensity then describes the intensity of a homogenous Poisson process which we will use for adding additional points.
- We then iterate over I and n and for every I we generate a new process according to θ_0 and k noise processes according to the entry of n, where the superpositions is then used for training. the vector n stores ascending intensities for the noise process.

For a better overview we quickly describe the bullet points in an algorithm.

Require:
$I > 0$: the number of iterations,
$k > 0$: the amount of thinnings for the cross validation,
N: a vector of noise intensities,
$\theta_0 \in \Theta$: parameter describing the true distribution.

for $i \in I$ **do**
  Generate process X according to θ_0.
  for $n \in N$ **do**
    Generate noise processes as Poisson processes $\Psi_1, \ldots, \Psi_k$ with intensity n.
    Generate superpositions $X_1^*, \ldots, X_k^*$.
    Train the model with thinnings of $X_1^*, \ldots X_k^*$ with retention probability p.
    Save estimated $\hat{\theta}$.
  end for
  calculate desired statistics with θ_0 and $\hat{\theta}$.
end for

The training it self is described in Chapter 3 for the method 3.2.1.
The L_2-loss function is used.

4.1.1 Hard-Core Process

For the hard-core process we have the two parameters β and R which we can fix. For generating a realization of the hard core process we choose the parameters to be $\beta = 100 \quad R = 0.05$. As an observation window we choose the unit square. We choose I to be 100, which means that we execute the experiment 100 times for each level of noise. The remaining hyper-parameters are given as

$$\begin{aligned} \text{noise intensity} &= \{0, 0.5, 1, 1.5, 2, 3, 4, 5, 6, 7, 8, 10, 15\}, \\ p &= \{0.2, 0.5, 0.8\}, \\ k &= \{1, 3, 10\}, \\ \text{test function } h &= \lambda_\theta(u; \aleph)^{-1}, \end{aligned}$$

where the noise intensity describes the amount of added Poisson noise.

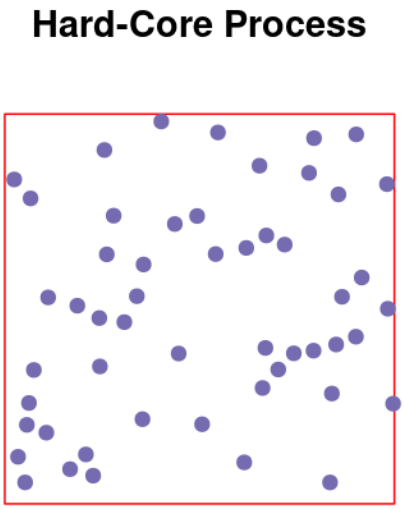


Figure 4.1: Realization of the hard-core process with the proposed parameters $\beta = 100$ and $R = 0.05$.

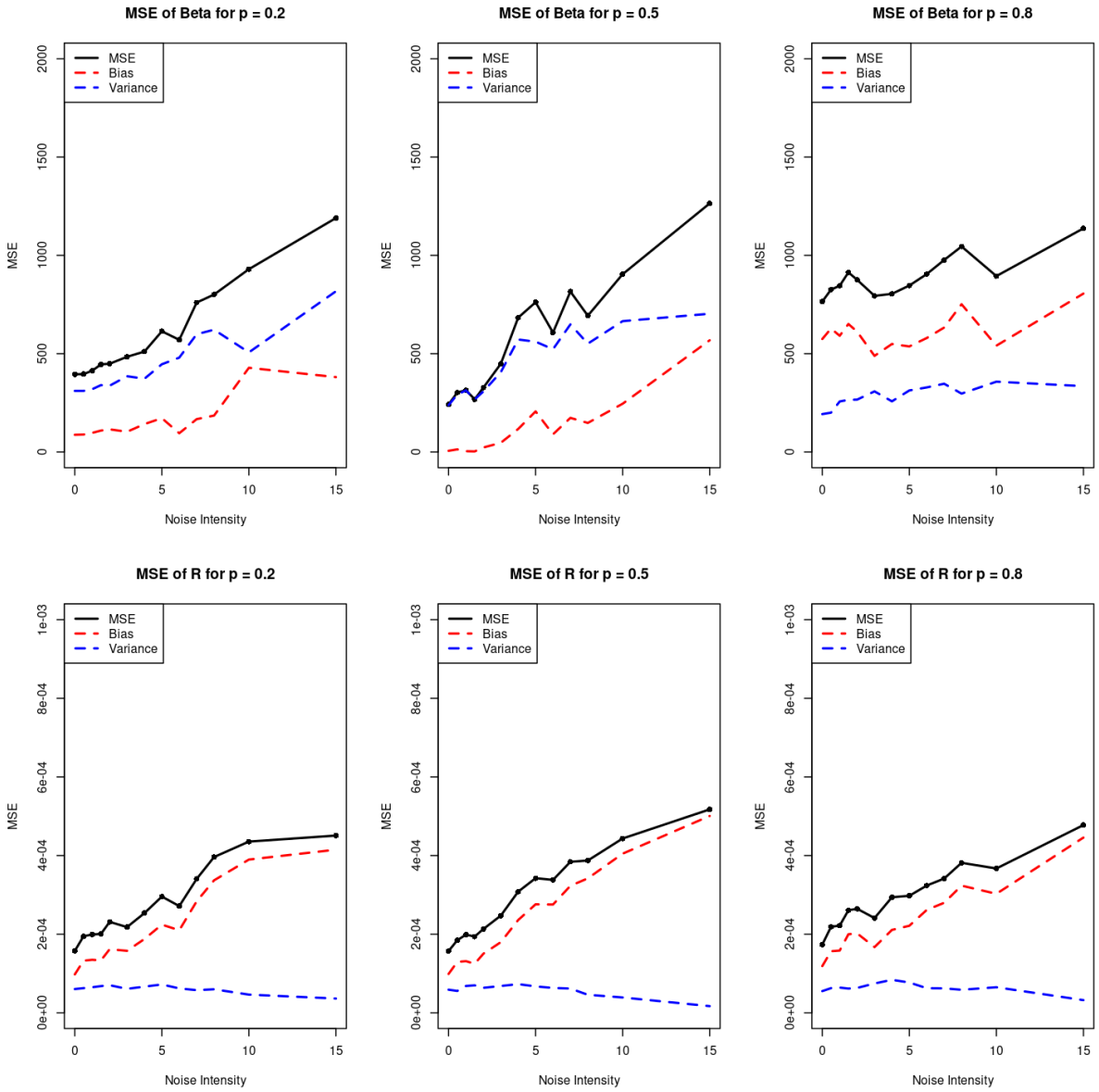


Figure 4.2: MSE, variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 1$ was chosen.

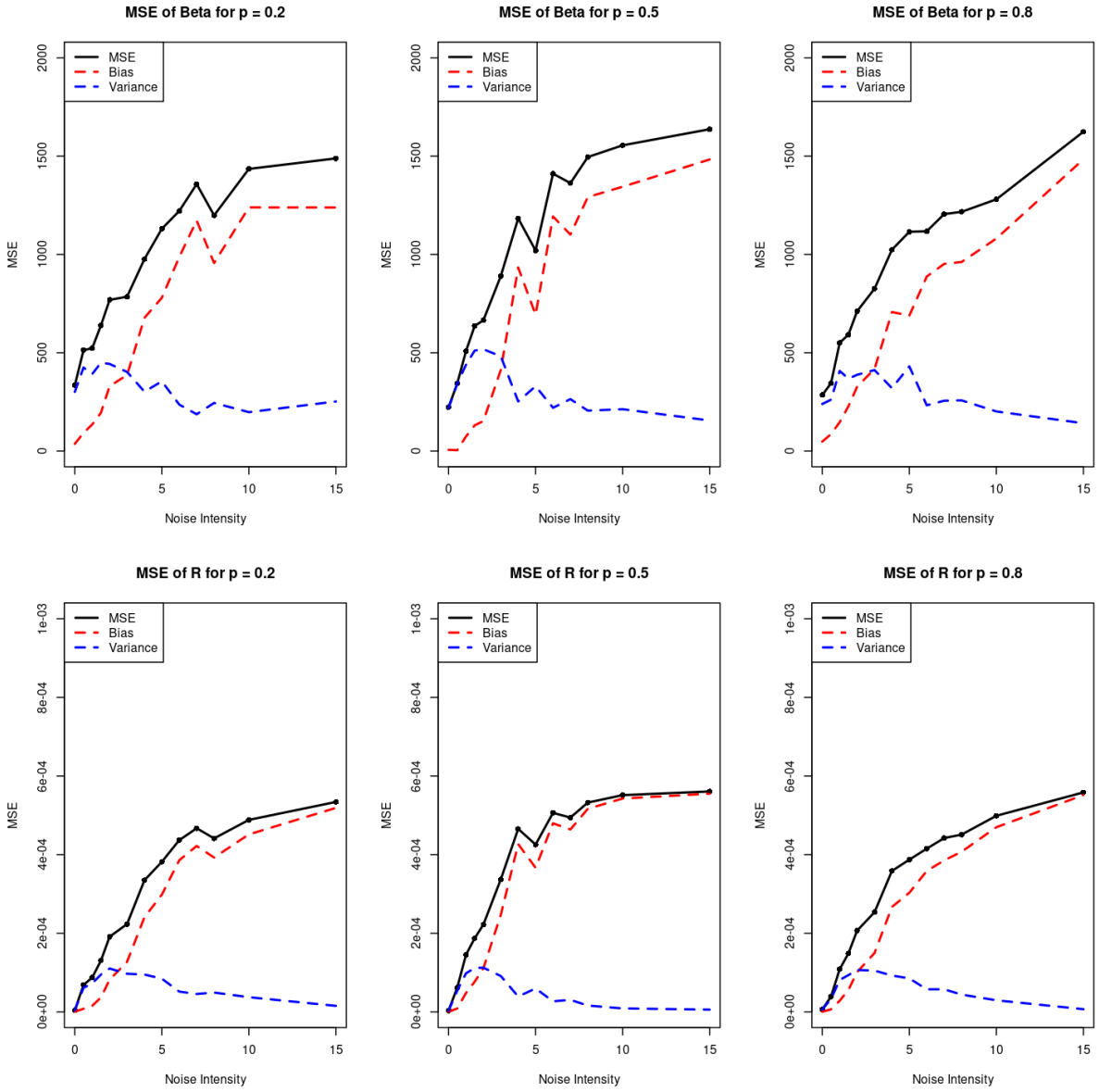


Figure 4.3: MSE, variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 3$ was chosen.

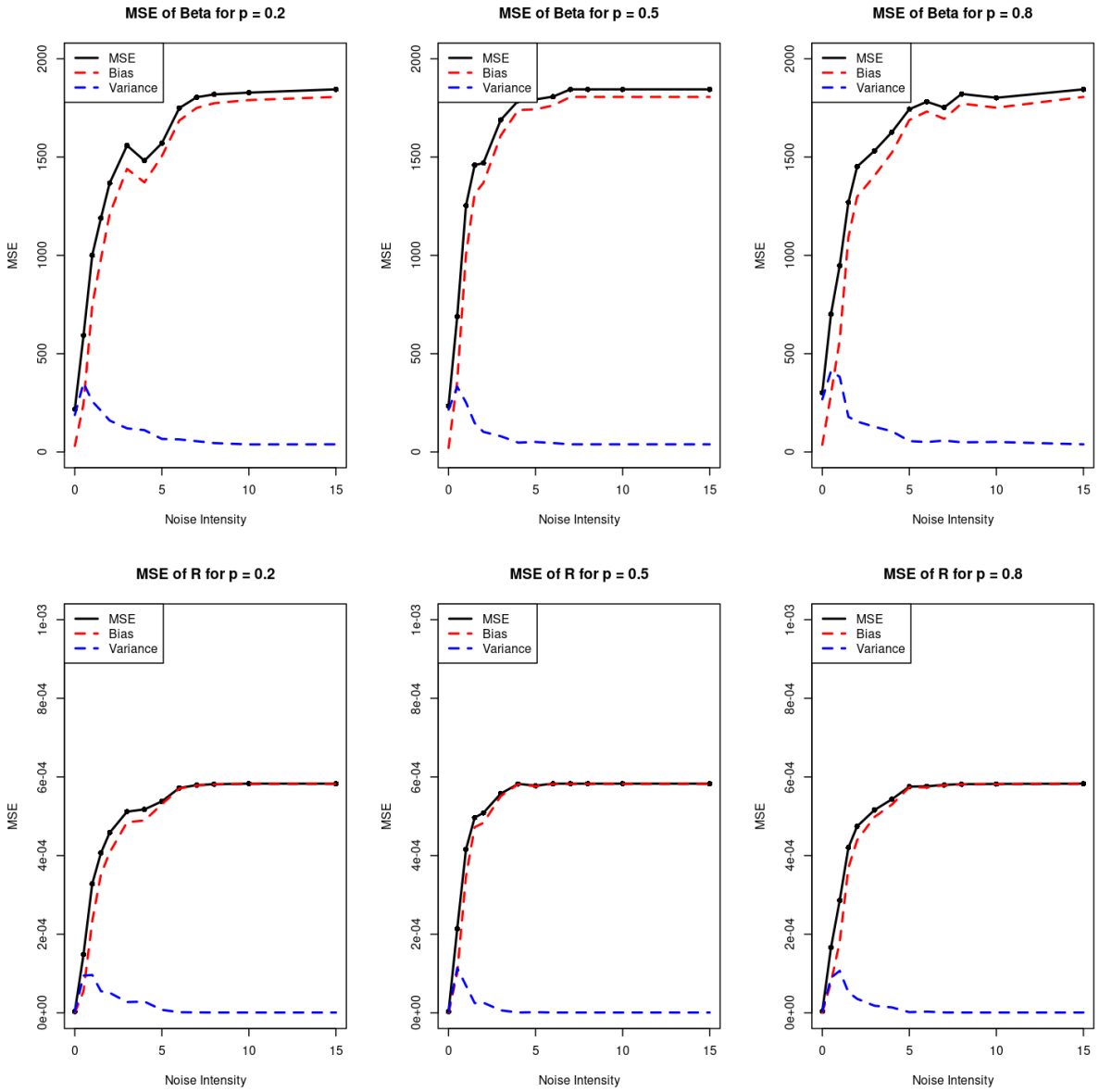


Figure 4.4: MSE, variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 10$ was chosen.

4.1.2 Area Interaction Process

For the area interaction process we have to do some modification since the spatstat implementation by Baddeley and Turner [2005] differs from the one in the literature. Recall the already introduced density (2.1.6.3), i.e.

$$f_W^{\beta,\gamma}(\mathbb{x}) = \frac{1}{C_W^{\beta,\gamma}} \beta^{|\mathbb{x} \cap W|} \gamma^{-A_R(\mathbb{x})}.$$

The spatsat version however is given by

$$f_W^{\beta,\gamma}(X) = \frac{1}{C_W^{\beta,\gamma}} \beta^{|\mathbb{x} \cap W|} \eta^{\frac{-A_R(\mathbb{x})}{\pi R^2} - |\mathbb{x} \cap W|},$$

which means that the area A_R is always scaled with respect to the unit circle. The corresponding conditional intensity is therefore given by

$$\lambda_\theta(u; X) = \beta \eta^{C_R(X,u)},$$

where the term in the exponent is given by

$$C_R(X, u) = -B_R(u) \setminus \bigcup_{x \in X} B_R(x) + 1.$$

Spatstat also only allows the parameter η to be greater than one. This is however not a problem since it was the intention in the first place.
Also note that we now have three parameters, with

$$\beta \in (0, \infty), \quad \eta \in [1, \infty), \quad R \in (0, \infty).$$

For the simulation study we will however fix β and find the optimal solution for η and R. This is due to the exponentially increasing computing time when adding a new parameter, since the algorithm iterates over an n-dimensional grid, where n is the number of parameters. Also computing the conditional intensity for the area interaction process does take way longer, since it involves calculating areas of non trivial shapes. We choose the parameter β to be fixed, since the interaction between η and R seems to be interesting.

For generating a realization of the area interaction process we choose the parameters to be

$$\eta = 2 \quad R = 0.05 \quad \beta = 50.$$

As an observation window we choose the unit square. We choose N to be 50 because of drastically increased computing time. The remaining hyper-parameters are given by

$$\begin{aligned} \text{noise intensity} &= \{0, 0.5, 1, 1.5, 2, 3, 4, 5, 6, 7, 8, 10, 15\}, \\ p &= \{0.2, 0.5, 0.8\}, \\ k &= \{1, 3, 10\}, \\ \text{test function } h &= \lambda_\theta(u; \Join)^{-1}, \end{aligned}$$

where the noise intensity describes the amount of added Poisson noise.

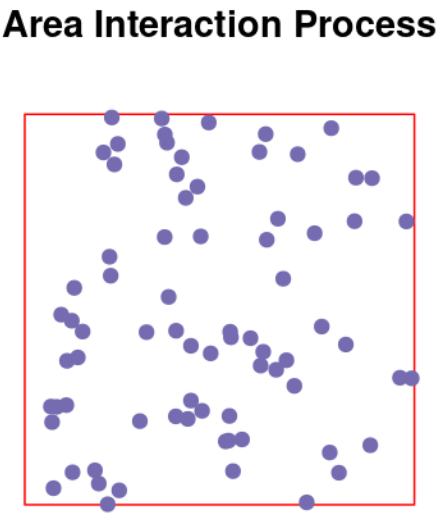


Figure 4.5: Realization of the area interaction process with the proposed parameters $\beta = 50$, $\eta = 2$ and $R = 0.05$.

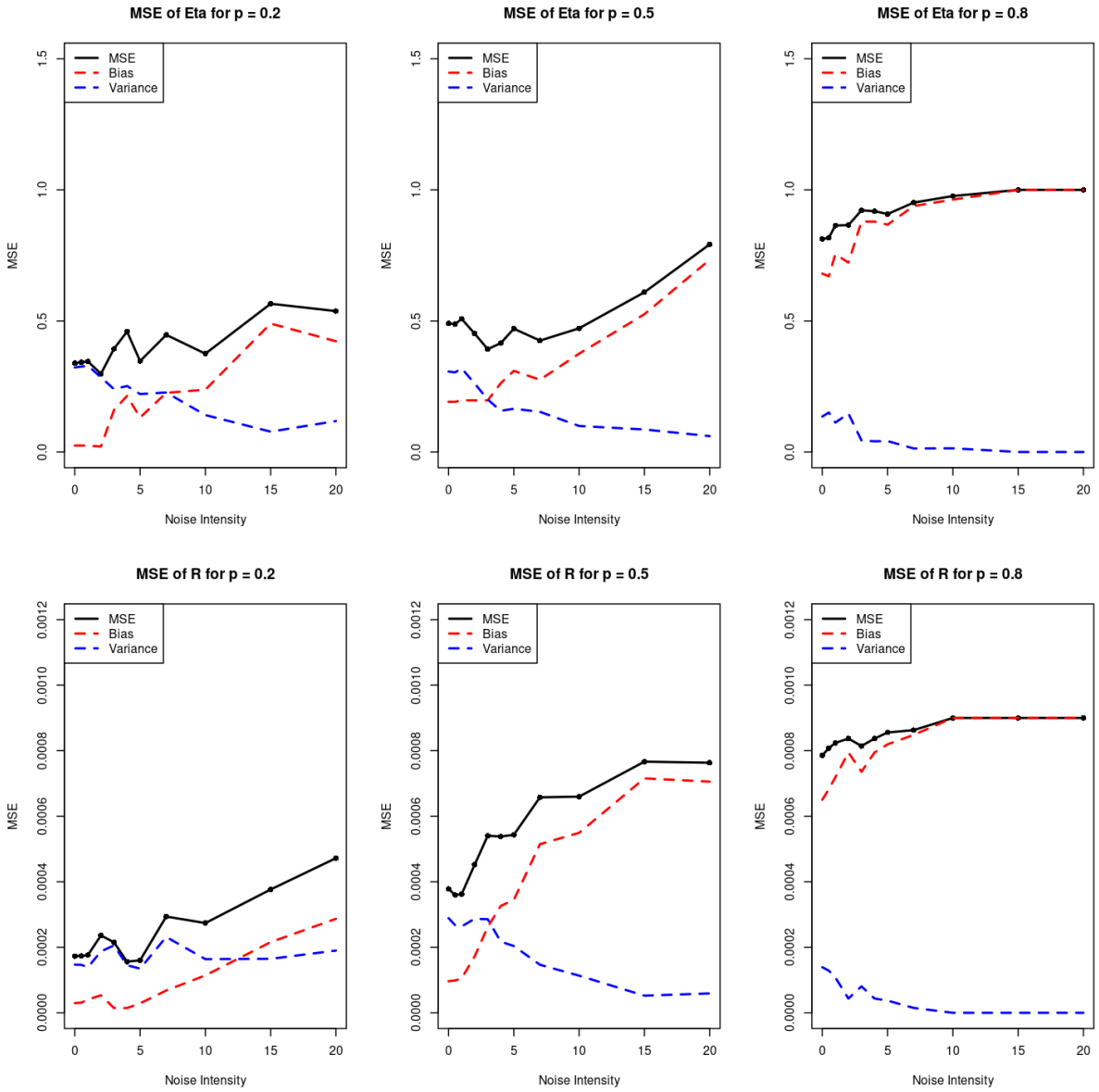


Figure 4.6: MSE, variance and squared bias of the parameters η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 1$ was chosen.

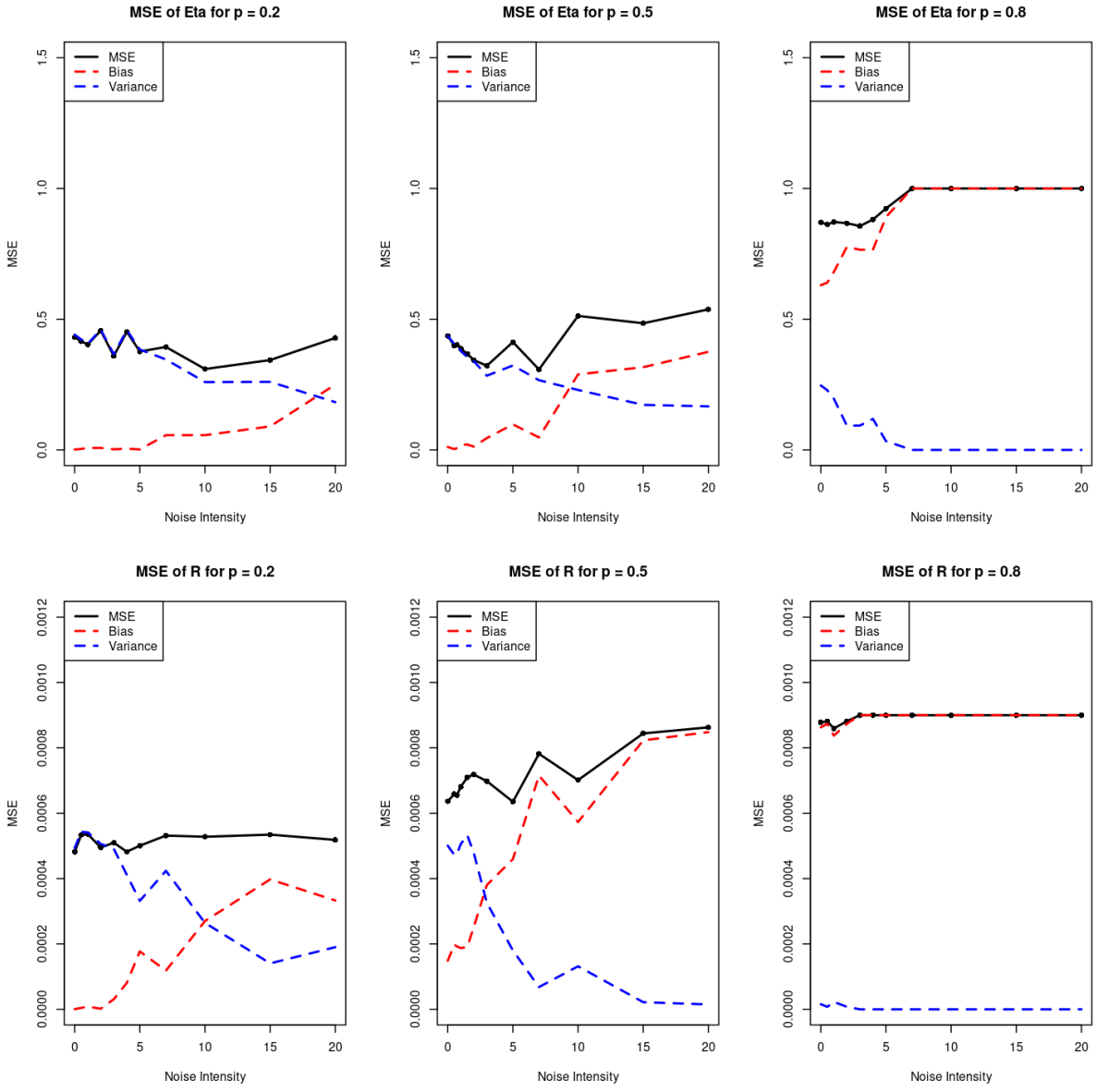


Figure 4.7: MSE, variance and squared bias of the parameter η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 3$ was chosen.

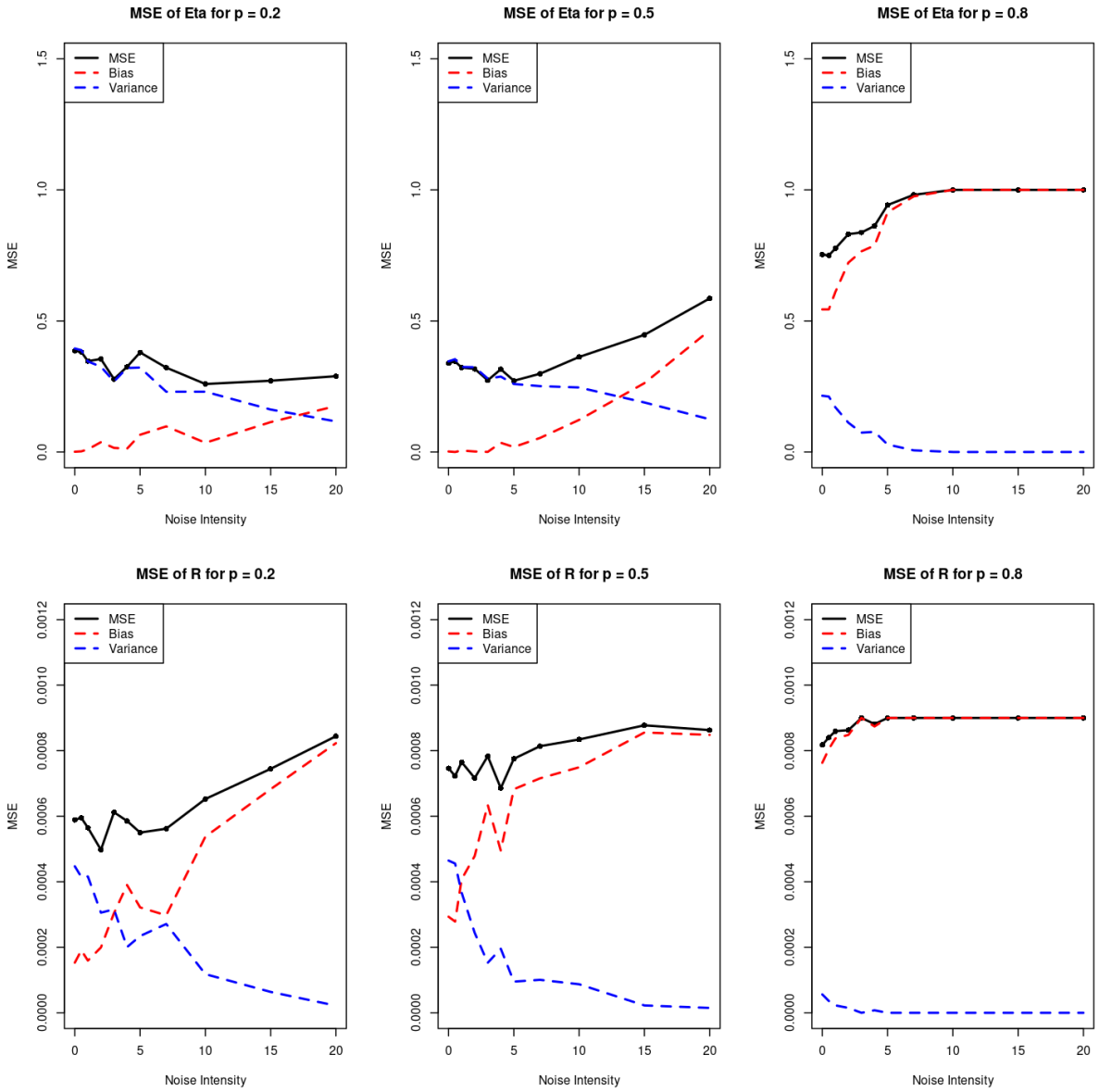


Figure 4.8: MSE, variance and squared bias of the parameter η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the noise intensity. $k = 10$ was chosen.

4.2 Results

We split this section into two parts. In the first part, we discuss and analyze the results of the hard-core process. In the second part we discuss the results of the area interaction process.

4.2.1 Analysis of the Hard Core Process

In general, the application of data augmentation to the hard-core process has not really worked well. The MSE did not improve remarkably. In general, it even increased proportionally to the amount of added noise. The higher k is, the stronger and faster the MSE increases for both R and β. Overall the results for the hard-core process are not good.

The hard-core process seems really sensitive to noise, which can affect the estimation a lot. The following Lemma gives some insights into this sensitivity.

Lemma 4.2.1. *Let X be the hard-core process, parameterized by β, R, and let X^* be noised by a homogenous Poisson process conditioned on having n points, i.e. a binomial point process. Then it holds that*

$$\min_{x,y\in X}\|x-y\| \geq \min_{x,y\in X^*}\|x-y\|. \tag{4.5}$$

Furthermore the probability that the strict inequality in Equation (4.5) holds is given by

$$\begin{aligned}\mathbb{P}[\min_{x,y\in X}\|x-y\| > \min_{x,y\in X^*}\|x-y\|] &= 1-\left(\frac{|W\setminus\bigcup_{x\in X}B_{\min_{x,y\in X}}(x)|}{|W|}\right)^n \\ &\geq 1-\left(\frac{|W\setminus\bigcup_{x\in X}B_R(x)(x)|}{|W|}\right)^n,\end{aligned}$$

where strict inequality holds if $W\subset\bigcup_{x\in X}B_R(x)$.

Proof. The first statement is clear, since we add points. The second statement follows from the fact that each noise point appears uniformly distributed on W. The minimal distance between points is bounded by R.

□

This simple statement has some consequences on the estimation. It immediately implies that

$$\mathbb{P}[\exists x\in X_V^*|\lambda_{\beta,R}(x;X_T^*)=0] = 1-\left(\frac{|W\setminus\bigcup_{x\in X}B_R(x)(x)|}{|W|}\right)^{|\Psi|},$$

which then implies that

$$\mathbb{P}[L^h_{\beta,R}(W,X_T^*X_V^*)=\infty] = 1-\left(\frac{|W\setminus\bigcup_{x\in X}B_R(x)(x)|}{|W|}\right)^{|\Psi|},$$

with $h(u, X_T^*) = (\lambda_{\beta,R}(u; X_T^*))^{-1},\ u \in W$. This in general leads to an underestimation of the parameter R. The effect is even stronger when considering cross-validation. Our loss is immediately infinity if one of the splittings results in a loss of infinity, i.e.

$$L^h_{\beta,R}(W, X_T^* X_V^*) = \infty \Longrightarrow \mathcal{L}_j(W, X^*, \beta, R) = \infty \quad j \in \{1, 2, 3\}$$

The following Lemma illustrates the limit behavior for the noise intensity and gives us an intuition why the estimation gets worse for higher noise.

Lemma 4.2.2. *Let $X \neq \emptyset$ be a process noised by a homogeneous Poisson process Ψ with noise intensity n. Then for any $\hat{R}, \hat{\beta} > 0$ it follows that*

$$\lim_{n\to\infty} \lambda_{\hat{R},\hat{\beta}}(\cdot, X^*) = 0.$$

Proof. Consider $B_{\hat{R}(x)} \setminus W$ for $x \in X$ Then the probability of $\lambda_{\beta,R}(\cdot, X) = 0$ is bounded by,

$$\mathbb{P}[\lambda_{\beta,R}(\cdot, X) = 0] = \mathbb{P}[B_{\hat{R}(x)} \cap \Psi \neq \emptyset] \geq 1 - \left(\frac{|W \setminus B_{\hat{R}}(x)|}{|W|}\right)^{|\Psi|} \xrightarrow{n\to\infty} 1.$$

□

Note that this sensitivity, especially for the cross validation, is hardly related to the choice of the test function h. Another choice could yield better results.

It should however be said that the use of data augmentation in the hard-core process is by no means useless. It can definitely be useful in special cases.
For the experiment we always estimated the weights w which is given as

$$w_\theta(u, X_T, X_V) = p(u)\frac{\mathbb{E}[\lambda_\theta(u; X)|X_T]}{\lambda_\theta(u; X_T)} = \begin{cases} p(u) & \text{if} \quad u \notin \bigcup_{x\in X} B_R(x) \\ 0 & \text{else.} \end{cases}$$

This is important to model the repulsive behavior of the hard-core process. The estimation of the weight however is really time consuming, since we have to calculate the conditional intensity of each point in the window. Therefore it is also an option to approximate the weight with

$$w_\theta(u, X_T, X_V) \approx p + \frac{p^2}{1-p}.$$

Doing this makes our estimator biased in case of the hard-core process since with the repulsiveness of the process and Equation (4.2.1) we get

$$w_\theta(u, X_T, X_V)\lambda_\theta(u; X_T) \leq \left(p + \frac{p^2}{1-p}\right)\lambda_\theta(u, X_T).$$

This statement is also implied by Lemma 3.4.2. To compensate for this, we can add noise to the validation set and observe that for choosing the test function to be $\lambda_\theta(\cdot;\cdot)^{-1}$ for a fixed θ we get

$$\sum_{x\in X_V} \frac{1}{\left(p + \frac{p^2}{1-p}\right)\lambda_\theta(x; X_T)} - 1 \leq \sum_{x\in X_V^*} \frac{1}{\left(p + \frac{p^2}{1-p}\right)\lambda_\theta(x; X_T)} - 1,$$

compensating for the fact that

$$\sum_{x\in X_V} \frac{1}{\left(p+\frac{p^2}{1-p}\right)\lambda_\theta(x;X_T)} - 1 < \sum_{x\in X_V} \frac{1}{w_\theta(x,X_T,X_V)\lambda_\theta(x;X_T)} - 1 = \sum_{x\in X_V} \frac{1}{p\lambda_\theta(x;X_T)} - 1.$$

The second equality is due to the fact that all points in X_V will not be in $\bigcup_{x\in X_T} B_R(x)$.

The results for additional noise in the validation set without weight estimation are surprisingly good. It even beats the pseudo likelihood method with setting $p = 0.1$, $k = 1$ and noise intensity of $n = 3$. Most likely also for the fact that the sums in Equation (4.2.1) are very similar for $p = 0.1$. An illustration can be seen in Figure 4.9.

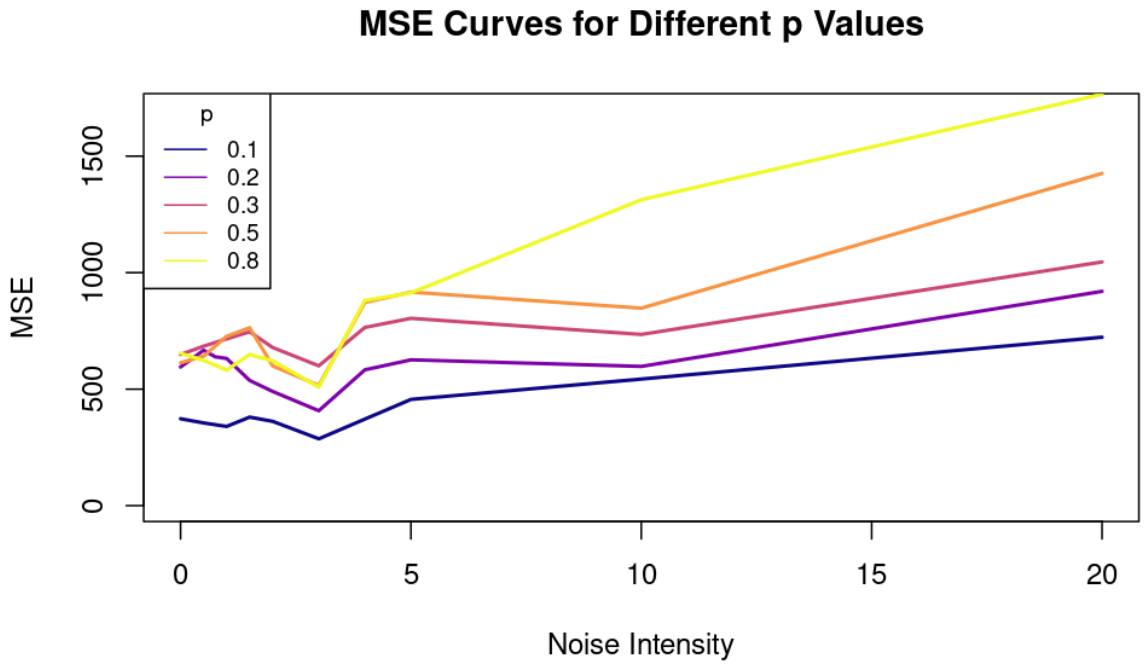


Figure 4.9: MSE without weight-estimation with respect to the noise intensity in the validation set for $p \in \{0.1, 0.2, 0.3, 0, 5, 0.8\}$. We observe an improved performance for a noise intensity of $n = 3$. For $p = 0.1$ this even beats the pseudo likelihood estimation. The parameters $k = 1$, $\beta = 100$ and $R = 0.05$ were chosen.

4.2.2 Analysis of the Area Interaction Process

In general, the results of the area interaction process are significantly better than those of the hard-core process. A lower MSE for η can be seen for $p = 0.2$ and $p = 0.5$ if the noise intensity is between $n = 3$ and $n = 10$. The subsequent increase in the noise intensity then leads to an increase in the MSE again. The effect occurs somewhat later for $p = 0.2$ than for $p = 0.5$. For $k = 3$ and $k = 10$ it is even more

pronounced. There is also a clear trade off between bias and variance. Although we obtain an increased bias with additional noise, the variance decreases.
For the parameter R, the desired effect does not occur, but there is also no real deterioration at low noise intensity. The trade off between bias and variance is however still observable.
For $p = 0.8$ the results are generally not good. There is no improvement visible when noise is added. The additional noise makes the estimation even worse. For $p = 0.8$ and additional noise, the method constantly overestimates the parameters η and R, which is most likely for the reason that the method has to compensate the fact that for high values of p,

$$L_\theta^h(W, X_T^*, X_V^*) = \sum_{x \in X_V^*} \frac{1}{\xi_{\eta,R}(x, X_T^*)} - 1 \leq \sum_{x \in X_V} \frac{1}{\xi_{\eta,R}(x, X_T)} - 1 = L_\theta^h(W, X_T, X_V)$$

holds for fixed η and R. This intuitively makes sense, since we sum also over the noised points in the validation set. The attractiveness of the area interaction process however can compensate this effect in some way, especially for lower p, since

$$\xi_{\eta,R}(x, X_T) = \mathbb{E}[\lambda_{\eta,R}(u; X)|X_T] \leq \mathbb{E}[\lambda_{\eta,R}(u; X^*)|X_T^*] = \xi_{\eta,R}(x, X_T^*) \tag{4.6}$$

for $u \in W$. This effect is also not too strong since the results for $p = 0.2$ are still very good. Note however that the described effect in Equation (4.6) only affects the estimation if the noise points in $\Psi_T \subset X_T^*$ interact with the validation points, i.e. $\|\phi - x\| < R$ for $\phi \in \Psi_T$ and $x \in X_V^*$. For small p the effect is relatively minor due to the low cardinality of X_V^* and the lack of clustering among the noise points compared to the original data points.

The method of Cronie et al. [2023] does not seem to work particularly well for the parameter R. An increase in k also leads to an increase in the MSE for the parameter R. The spatstat authors Baddeley and Turner [2005] give a warning for estimating parameters when R is to low, due to computational issues. However the effect is also visible for higher values of R and also for $k = 1$ the estimation of R is solid. This effect is however not related to the additional noise, since the effect also occurs for a noise intensity of $n = 0$.

In general the observations clearly indicate a regularizing effect of the noise. This is particularly confirmed by the observed bias variance trade off. In contrast to the hard-core process, the area interaction process is not as sensitive to noise. A slightly noisy point pattern could have also been generated by the process itself. The diversity in the dataset is particularly pronounced when we increase the parameter k, as we now have multiple similar copies, highlighting the effectiveness of data augmentation.

5

Noising Data with Shifting

For this chapter we assume our space $(S, \mathcal{S})$ to be the euclidean space $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$ with the Borel sigma algebra.
In the previous chapter we discussed options for adding additional noise to our data set to perturb the data. We therefore always increased the amount of points in a data set with help of a noise process. The original data was always a subset of the noised data, i.e.

$$X \subset X^*.$$

There is however another method for how we can disturb the given data. In this method we will not increase the amount of data points. This approach will be based on shifting each individual point of the process by an offset. This is also known as jittering

5.1 Motivation

A common way to interpret noise in a data set or measurement is by assuming that our data X^* is given by true values X and a noise part ϵ [Tuzlukov, 2018], i.e.

$$x_i^* = x_i + \epsilon_i \quad x_i \in X \quad \forall i \in \{1, \dots n\}, \quad \epsilon_i \in \mathbb{R}^d$$

where the noise term ϵ_i is unknown. However it's assumed to be a random variable with mean zero, i.e. $\mathbb{E}[\epsilon_i] = 0$. A common way is to model the noise part with a Gaussian distribution [Tuzlukov, 2018], i.e.

$$\epsilon_i \sim \mathcal{N}(0, \sigma \cdot \mathcal{I}_d),$$

where $\mathcal{I}_d$ is the identity matrix, which also ensures that the entries of the noise vector ϵ are uncorrelated.
By sampling such a vector ϵ, clean data can be noised on purpose. In the first Chapter in Section 1.2 we already showed that such an approach can have a regularizing effect by adding noise to a linear regression model in. We therefore will try a similar method which can be applied to a point process. Since we always consider the point process in an observation window W we have to deal with edge correlation problems, which will lead to some major restrictions about the model. Edge correlation can occur when data points are shifted out or in the observation window W.

5.1.1 The Distribution of the Noise

Since the Gaussian distribution has infinite support we would always have edge effect problems by noising our data with help of an offset ϵ sampled according to a Gaussian distribution no matter how we choose the window W. For a bounded window it would hold that

$$\mathbb{P}[x + \epsilon \notin W] > 0 \quad \forall x \in X \cap W.$$

To compensate points leaving the window W after being noised, we would also need to have points which can appear in W after noising. The original process should therefore have a density with support

$$W' \quad \text{with} \quad W \subset W'.$$

We would also consider every point $y \in X \setminus W$ outside the window, since also the probability for finding the noised point y^* in the window W is non-zero, i.e.

$$P[y + \epsilon \in W] > 0 \quad \forall y \in X \setminus W.$$

no matter how big the support W' of X would be.
There is however a simple solution to this problem. We simply restrict the support of the PDF of ϵ.
As a helping tool, we first define Minkiwski addition, introduced by Gilbert et al. [1988].

Definition 5.1.1 (Minkowski Addition)**.**
Let A and B be two sets in S. Then the Minkowski addition of A and B is defined by

$$A \oplus B = \{a + b |\ a \in A,\ b \in B\}.$$

The Minkowski subtraction of A and B is given by

$$A \ominus B = \{a - b |\ a \in A,\ b \in B\}.$$

With help of the Minkowski addition we now can define the critical areas where point from outside the window can interact.

Definition 5.1.2. *We define the outer ring of W with*

$$\delta^+(W) = W \oplus B_\delta(0),$$

and the inner ring of W with

$$\delta^-(W) = W \cap \left(\partial\overline{W} \oplus B_\delta(0)\right).$$

Lemma 5.1.1 below gives us a proper statement about areas which interact with the window W when points are shifted.

Lemma 5.1.1. *Let ϵ be a random variable with support $B_\delta(0)$, $\delta > 0$. Then the following statements hold:*

- $\mathbb{P}[x + \epsilon \in W] > 0 \quad \Rightarrow \quad x \in \delta^+(W)$
- $\mathbb{P}[x + \epsilon \notin W] > 0$ *and* $x \in W \quad \Rightarrow \quad x \in \delta^-(W)$

The proof of Lemma 5.1.1 is trivial. To summarize, we can say that if we move the points of our process in the window W we have to use the space $\delta^+(W)$ for the edge effects, where δ is the radius of the support of the noise vector ϵ. This must be taken into account when using the method of Cronie et al. [2023].

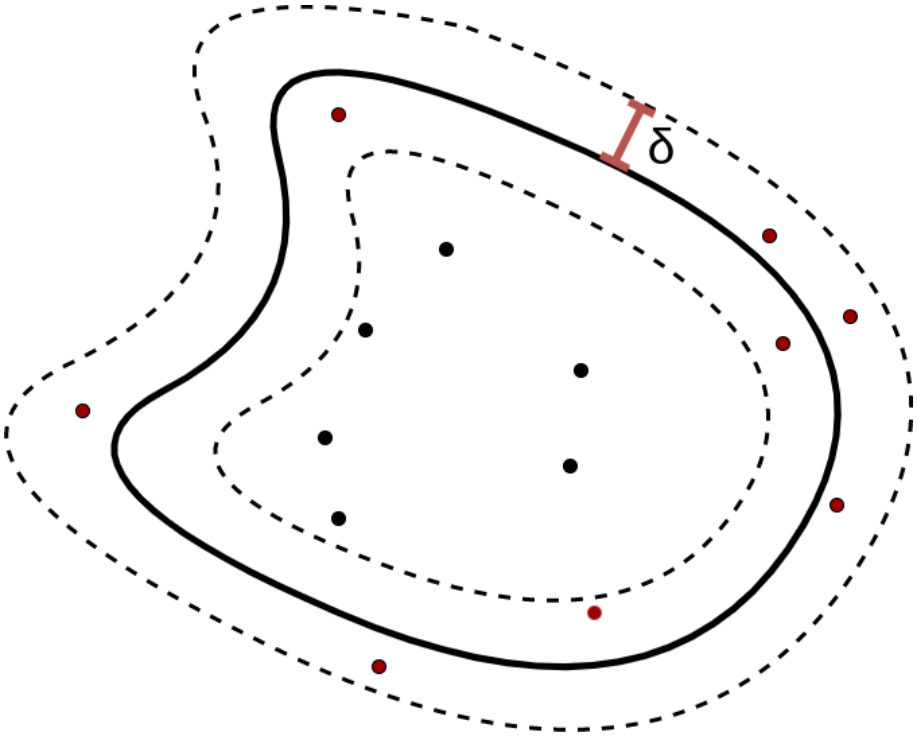


Figure 5.1: Sketch of the described resulting windows. The solid line describes the window W. The outer dashed line describes the support $\delta^+(W)$ of the process. Red dots indicate points that may be shifted out of or into the window W after a shift. The inner dotted line indicates $\delta^-(W)$.

5.2 Method Description

We now define the method where we insert the noise via jittering. As input we have our point pattern $\mathbb{x}'$ and an observation window W', which later needs to be adapted.

- We first fix a PDF f of a random variable ϵ with

$$\text{supp}(f) = B_\delta(0).$$

  We also demand that

$$\mathbb{E}[\epsilon] = 0.$$

- We define our window where we later want to use Point Process Learning

$$W = \delta^-(W').$$

- We then sample $\epsilon^1, \ldots, \epsilon^K$ with,

$$\epsilon^j = (\epsilon_1^j, \ldots, \epsilon_n^j) \quad j \in \{1, \ldots, K\}, \tag{5.1}$$

  where $n = |\mathbb{x}'|$ describes the amount of data points in the data set.
- We then can generate noised data sets $\mathbb{x}^{1*}, \ldots, \mathbb{x}^{n*}$ with

$$\mathbb{x}^{j*} = \{z \in W |\ z = \mathbb{x}^j + \epsilon^j\}.$$

- We apply a thinning to split our noised data $\mathbb{x}^{j*}$ into training $\mathbb{x}_T^{j*}$ and validation set $\mathbb{x}_V^{j*}$, with $j \in \{1, \ldots, K\}$.
- We then minimize the following loss function to obtain an estimator $\hat{\theta}$. Our estimator is then given by

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \mathcal{L}_j(\theta) = \arg\min_{\theta \in \Theta} \sum_{i=1}^{K} \left[\sum_{x \in \mathbb{x}_V^{i*}} \left| h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*}) \xi_\theta(u, \mathbb{x}_T^{i*})\ du \right| \right]^j,$$

  with $j \in \{1, 2\}$. And for the L_3 loss we optimize

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \mathcal{L}_3(\theta) = \arg\min_{\theta \in \Theta} \left[\sum_{i=1}^{K} \sum_{x \in \mathbb{x}_V^{i*}} h(x, \mathbb{x}_T^{i*}) - \int_W h(u, \mathbb{x}_T^{i*}) \xi_\theta(u, \mathbb{x}_T^{i*})\ du \right]^2.$$

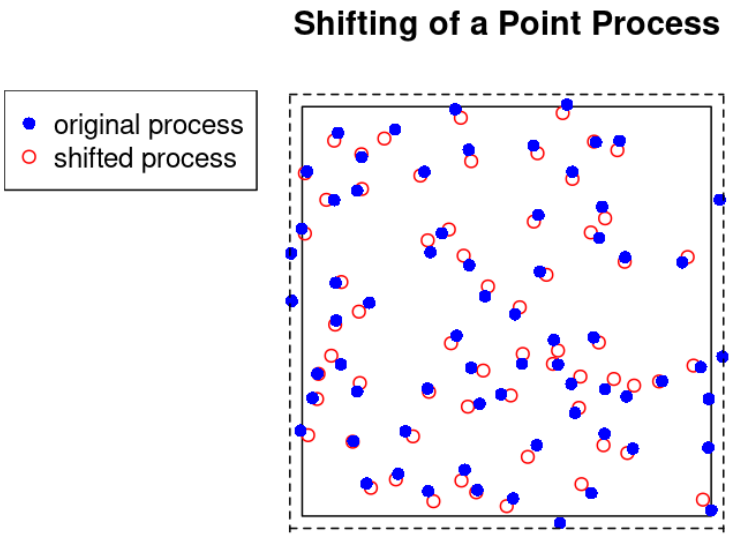


Figure 5.2: In blue we see the original process observed on an extended window W^* with $\delta^+(W) \subset W^*$, indicated by the dotted line. The red point pattern shows the shifted process restricted to the original window W. Note that some points are shifted out or shifted into W.

5.3 Statistical Properties

For inferring over a window $W = \delta-(W')$ we always need our data to be in a window W' which includes all balls around every point in W with radius δ. The choice of delta is therefore also limited. By choosing delta too high we could end up with the empty set for W. To be more specific consider the following Theorem.

Theorem 5.3.1. *Let W' be the given window for the process X and let r be given by*

$$r = \sup\{r \in \mathbb{R} |\ B_r(x) \subset W' \ \ for \ \ x \in W\}.$$

If $\delta > r$, then the resulting set $W = \delta^-(W')$ will be the empty set.

Proof. Assume $\delta > r$ and that W is not the empty set. Then there exists a point $x \in W$ where for all $y \in \partial\delta^+(W)$ it holds that $\|x - y\| > \delta$. Since else there would be a $y \in \partial W'$ with

$$x \in B_\delta(y).$$

However if this is the case there would also fit a ball around x with radius δ inside of $\delta^+(W)$. Therefore r was not chosen as Defined in 5.3.1, which is a contradiction.

□

As a result we see that the shape of $\delta^+(W)$ has to be considered regarding the choice of δ.
The GNZ formula will still hold and therefore the method of Cronie et al. [2023] still works. To be more precise, consider the following Theorem.

Theorem 5.3.2. *Let $W \subset W'$ be a compact set with positive measure and let X be a process on W' then the GNZ formula still holds for the process $X \cap W$, i.e.*

$$\mathbb{E}\left[\sum_{x \in X \cap W} h(x, X \cap W \setminus \{x\})\right] = \mathbb{E}\left[\int_W h(u, X \cap W)\lambda_X(u; X \cap W)\, du\right],$$

with $h : S \times \mathcal{X}_S \to \mathbb{R}$ being integrable and non-negative.

Proof. Note that we can rewrite the intersection as an indicator function, i.e.

$$\mathbb{E}\left[\sum_{x \in X \cap W} h(x, X \cap W \setminus \{x\})\right] = \mathbb{E}\left[\sum_{x \in X} h(x, X \cap W \setminus \{x\})\mathbb{1}_{x \in W}\right].$$

The function $h(x, X \cap W\{x\})\mathbb{1}_{x \in W}$ still fulfills the condition for the GNZ formula, which implies that

$$\begin{aligned}\mathbb{E}\left[\sum_{x \in X} h(x, X \cap W \setminus \{x\})\mathbb{1}_{x \in W}\right] &= \mathbb{E}\left[\int_{W'} h(u, X \cap W)\lambda_X(u; X \cap W)\mathbb{1}_{u \in W}\, du\right] \\ &= \mathbb{E}\left[\int_W h(u, X \cap W)\lambda_X(u; X \cap W)\, du\right].\end{aligned}$$

□

Shifting the data points is in general changing the overall distribution. If we however have a stationary process the amount of expected points within the window will not change after jittering. This is very powerful since it gives us an assumption we can make before we try to infer the noised data.

Theorem 5.3.3. *Let X be a stationary (Gibbs) point process observed in a window $W \subset S = \mathbb{R}^d$ with positive measure and let ϵ be a vector of random variables with mean zero and support $B_\delta(0)$. Then it holds that*

$$\mathbb{E}[X(W)] = \mathbb{E}[X^*(W)], \tag{5.2}$$

where $X^* = X + \epsilon$ is the noised process. Furthermore the process X^* is also stationary.

Proof. Let μ be the probability measure describing the distribution of ϵ with support $B_\delta(0)$. First consider the following term which describes the term $\mathbb{E}[X^*(W)]$.

$$\mathbb{E}[X^*(W)] = \mathbb{E}[|X + \epsilon \cap W|] = \mathbb{E}\left[\sum_{x\in X} \int_{B_\delta(x)\cap W} u - x \ \mu(du)\right]$$

We define the right term as $g(W)$. We first show that $g(\cdot)$ is a translation-invariant measure.

It clearly holds that $g(\emptyset) = 0$ and that $g(W) \geq 0$ for all $W \subset S$. And for disjoint sets W_1 and W_2 in S it holds that

$$g(W_1) + g(W_2) = \mathbb{E}\left[\sum_{x\in X} \int_{B_\delta(x)\cap W_1} u - x \ \mu(du)\right] + \mathbb{E}\left[\sum_{x\in X} \int_{B_\delta(x)\cap W_2} u - x \ \mu(du)\right],$$

where we now use the linearity of the expectation and the integral to obtain

$$g(W_1) + g(W_2) = \mathbb{E}\left[\sum_{x\in X} \int_{B_\delta(x)\cap (W_1\cup W_2)} u - x \ \mu(du)\right] = g(W_1 \cup W_2).$$

We showed that $g(\cdot)$ is a measure. We however still need to show that it is translation-invariant. We prove this with contradiction. Assume $g(\cdot)$ is not invariant. Then there exists a set $A \subset S$ and $A^* \subset S$ with

$$A^* = A + t, \quad t \in S \text{ and } g(A) \neq g(A^*).$$

Without loss of generality we assume that $g(A) < g(A^*)$. We now want to apply the GNZ formula. Now it holds that

$$g(A) = \mathbb{E}\left[\sum_{x\in X} \int_{B_\delta(x)\cap A} u - x \ \mu(du)\right] = \mathbb{E}\left[\sum_{x\in X\cap \delta^+(A)} \int_{B_\delta(x)\cap A} u - x \ \mu(du)\right].$$

Using the GNZ formula and and applying the same trick to $g(A^*)$ we get $g(A) < g(A^*)$ is equivalent to

$$\begin{aligned} &\mathbb{E}\left[\int_{\delta^+(A)} \left(\int_{B_\delta(y)\cap A} u - y \ \mu(du)\right) \lambda_X(y; X) \ dy\right] \\ <\ &\mathbb{E}\left[\int_{\delta^+(A^*)} \left(\int_{B_\delta(y)\cap A^*} u - y \ \mu(du)\right) \lambda_X(y; X) \ dy\right]. \end{aligned}$$

Since we defined $A^* = A + t$ we can rewrite the equation to

$$\begin{aligned} &\mathbb{E}\left[\int_{\delta^+(A)} \left(\int_{B_\delta(y)\cap A} u - y \ \mu(du)\right) \lambda_X(y; X) \ dy\right] \\ <\ &\mathbb{E}\left[\int_{\delta^+(A)} \left(\int_{B_\delta(y+t)\cap A^*} u - y \ \mu(du)\right) \lambda_X(y+t; X) \ dy\right]. \end{aligned}$$

We also know that for a fixed $x \in S$ it holds that

$$\int_{B_\delta(x)\cap A} u - x \ \mu(du) = \int_{B_\delta(u+t)\cap A+t} u - x \ \mu(du) \leq 1 < \infty.$$

Therefore it also has to hold that

$$\mathbb{E}\left[\int_{\delta(A)} \lambda_X(y, X)\ dy\right] < \mathbb{E}\left[\int_{\delta(A)} \lambda_X(y+t, X)\ dy\right]$$

which would result with the Campbell formula to

$$\mathbb{E}[X(A)] < \mathbb{E}[X(A^*)].$$

This is however a contradiction to the stationarity of X. Therefore $g(\cdot)$ is a translation-invariant measure on S.

As a last step we now have to show that $\mathbb{E}[X(W)] = g(W)$ for all $W \subset S$. First note that the following term is bounded with

$$X(\delta^-(W)) \leq \sum_{x \in X} \int_{B_\delta \cap W} u - x\ \mu(du) \leq X(\delta^+(W)),$$

therefore it holds that

$$\mathbb{E}[X(\delta^-(W))] \leq g(W) \leq \mathbb{E}[X(\delta^+(W))].$$

Since both measures are translation invariant we can rewrite the equation as

$$\mathbb{E}[X(\delta^-(W))] = \nu_1|\delta^-(W)| \leq g(W) = \nu_2|W| \leq \nu_1|\delta^+(W)| = \mathbb{E}[X(\delta^+(W))],$$

where $\nu_1, \nu_2 \in \mathbb{R}^+$ and $|\cdot|$ indicates the Lebesgue measure. Now take W to be a cube scaled with radius r i.e. $W = [-r, r]^d$, with $r > \delta$. Then we get

$$\begin{aligned} & \nu_1|\delta^-(W)| \leq \nu_2|W| \leq \nu_1|\delta^+(W)| \\ \implies\ & \nu_1(2(r-\delta))^d \leq \nu_2(2r)^d \leq \nu_1(2(r+\delta))^2 \\ \implies\ & \nu_1\frac{(2(r-\delta))^d}{(2r)^d} \leq \nu_2 \leq \nu_1\frac{(2(r+\delta))^2}{(2r)^d}. \end{aligned}$$

We observe that for $r \to \infty$ it holds that

$$\nu_1 \leq \nu_2 \leq \nu_1.$$

Since W was arbitrary we now observe that $\nu_1 = \nu_2$ which directly implies that

$$\mathbb{E}[X(W)] = \mathbb{E}[X^*(W)].$$

□

Theorem 5.3.3 is really strong. We now know that if we have a stationary point process and we shift data points according to a distribution with finite support, we can expect the same amount of points in our observation window W as before the shift.

Theorem 5.3.4. *Let X be a point process in $S = \mathbb{R}^2$ and let X^* be the process resulting from jittering the points with a uniform distribution of support $B_\delta(0)$. Let*

$$\lim_{r\to\infty} \frac{|B_r(a) \cap X|}{|B_r(a)|} = \lambda \quad \forall a \in S.$$

Let X be bounded by a constant ν which means that for all $k > \nu|A|$, with $A \in \mathcal{S}$ bounded, it holds that

$$P(X(A) = k) = 0 \quad a.s..$$

Then for $\delta \to \infty$ the process X^* will be a homogeneous Poisson process with intensity λ.

Proof. For the proof we first show that $X^*(A)$ is Poisson distributed in the limit. Therefore we first analyze the random variable $X^*(A)$ for a bounded set A.
Let $r(A)$ and $c(A)$ be defined as

$$r(A) = \arg\min_{r\in\mathbb{R}} \{r, a | A \subset B_r(c(A))\}.$$

and

$$c(A) = \frac{1}{|A|} \int_A x \; dx$$

We now can decompose the random variable $X^*(A)$ into a sum of two random variables, i.e.

$$X^*(A) = Z^1_\delta(A) + Z^2_\delta(A),$$

where we define $Z^1_\delta(A)$ as

$$Z^1_\delta(A) = |\ \{x + \epsilon \in A | x \in B_{\delta+r(A)}(c(A)) \setminus B_{\delta-r(A)}(c(A)) \cap X\}|$$

and $Z^2_\delta(A)$ as

$$Z^2_\delta(A) = |\ \{x + \epsilon \in A | x \in B_{\delta-r(A)}(c(A)) \cap X\}|.$$

Note that we in general only need to consider the points of X in $B_{\delta+r(A)}(c(A))$ for the random variable $X^*(A)$. We can now state the discrete PDF by

$$\mathbb{P}(X(A) = k) = \sum_{i=0}^{n} \mathbb{P}(Z^1_\delta(A) = i)\mathbb{P}(Z^2_\delta(A) = k - i),$$

where n is given by $|B_{\delta+r(A)}(c(A)) \cap X|$. We now split up the sum and show that the right term goes to zero for δ to infinity.

$$\mathbb{P}(X(A) = k) = \mathbb{P}(Z^1_\delta(A) = 0)\mathbb{P}(Z^2_\delta(A) = k) + \sum_{i=1}^{n} \mathbb{P}(Z^1_\delta(A) = i)\mathbb{P}(Z^2_\delta(A) = k - i).$$

Now we focus on the right sum and observe that

$$\sum_{i=1}^{n} \mathbb{P}(Z^1_\delta(A) = i)\mathbb{P}(Z^2_\delta(A) = k - i) \leq \sum_{i=1}^{n} \mathbb{P}(Z^1_\delta(A) = i).$$

For a point in $x \in B_{\delta+r(A)}(c(A)) \setminus B_{\delta-r(A)}(c(A))$,

$$\mathbb{P}(x+\epsilon \in A) = \frac{|A \cup B_{\delta+r(A)}(c(A))|}{|B_{\delta+r(A)}(0)|} \leq \frac{|B_{r(A)}(0)|}{|B_{\delta+r(A)}(0)|} = \frac{r(A)^2}{(\delta+r(A))^2}.$$

We then can bound our previous sum by

$$\sum_{i=1}^{n} \mathbb{P}(Z_\delta^1(A)=i) = \sum_{i=1}^{m} \mathbb{P}(Z_\delta^1(A)=i) \leq \sum_{i=1}^{m} \binom{m}{i} \left(\frac{r(A)^2}{(\delta+r(A))^2}\right)^i,$$

where m is the amount of points given in the outer ring, with

$$m = |X \cap B_{\delta+r(A)}(c(A)) \setminus B_{\delta-r(A)}(c(A))| \leq \nu |B_{\delta+r(A)}(c(A)) \setminus B_{\delta-r(A)}(c(A))| = \nu 4\delta r(A).$$

With the binomial theorem we now get that

$$\sum_{i=1}^{m} \binom{m}{i} \left(\frac{r(A)^2}{(\delta+r(A))^2}\right)^i = \sum_{i=0}^{m} \binom{m}{i} \left(\frac{r(A)^2}{(\delta+r(A))^2}\right)^i - 1 = \left(1+\frac{r(A)^2}{(\delta+r(A))^2}\right)^m - 1.$$

Note that it hods that

$$\lim_{x\to\infty} \left(1+\frac{1}{x^2}\right)^x = 0.$$

Therefore if δ goes to infinity we get

$$\left(1+\frac{r(A)^2}{(\delta+r(A))^2}\right)^m - 1 \leq \left(1+\frac{r(A)^2}{(\delta+r(A))^2}\right)^{\nu 4\delta r(A)} - 1 \xrightarrow{\delta\to\infty} = 1-1 = 0.$$

This now means that

$$X^*(A) \xrightarrow{\delta\to\infty} Z_\delta^2(A)$$

in distribution, and therefore the distribution of $X^*(A)$ is given by

$$\begin{aligned}
\lim_{\delta\to\infty} P(X^*(A)=k) &= \lim_{\delta\to\infty} (Z_\delta^1(A)=0) P(Z_\delta^2(A)=k) \\
&= \lim_{\delta\to\infty} (Z_\delta^1(A)=0) \binom{(n-m)}{k} \left(\frac{|A|}{|B_{\delta-r(A)}|}\right)^k \left(\frac{|A|}{|B_{\delta-r(A)}|}\right)^{n-m-k} \\
&= \lim_{\delta\to\infty} \binom{(n-m)}{k} \left(\frac{|A|}{|B_{\delta-r(A)}|}\right)^k \left(\frac{|A|}{|B_{\delta-r(A)}|}\right)^{n-m-k}.
\end{aligned}$$

And by the assumption that $\frac{n}{B_\delta(c(A))} \to \lambda$, we can apply the Poisson limit theorem [Barbour and Eagleson, 1987] and get that,

$$\lim_{\delta\to\infty} P(X^*(A)=k) = e^{-|A|\lambda} \frac{(|A|\lambda)^k}{k!}$$

We now still have to show that the random variables $X^*(A_1)$ and $X^*(A_2)$, with disjoint sets $A_1, A_2 \subset S$ are independent for $\delta \to \infty$. Note that we can apply the same trick as before. We assume $A_1 \cup A_2$ to be one set. We then again can generate $r(A_1 \cup A_2)$, $c(A_1 \cup A_2)$ and show that for $\delta \to \infty$ it holds that

$$\lim_{\delta\to\infty} P(X^*(A_1 \cup A_2)=k) = \lim_{\delta\to\infty} (Z_\delta^1(A_1 \cup A_2)=0) P(Z_\delta^2(A_1 \cup A_2)=k).$$

By now defining n as

$$n = |B_{\delta - r(A_1 \cup A_2)}(c(A_1 \cup A_2)) \cap X|,$$

we can calculate the probability for $P(X^*(A_1) = k, X^*(A_2) = l)$ for $\delta \to \infty$ with

$$\begin{aligned}\lim_{\delta\to\infty} P(X^*(A_1) = k, X^*(A_2) = l) = &\lim_{\delta\to\infty} \binom{n}{k} \left(\frac{|A_1|}{|B_{\delta - r(A_1\cup A_2)}(c(A_1 \cup A_2))|}\right)^k \\ &\times \lim_{\delta\to\infty} \binom{n-k}{l} \left(\frac{|A_2|}{|B_{\delta - r(A_1\cup A_2)}(c(A_1 \cup A_2))|}\right)^l \\ &\times \lim_{\delta\to\infty} \left(\frac{|A_1 \cup A_2|}{|B_{\delta - r(A_1\cup A_2)}(c(A_1 \cup A_2))|}\right)^{n-k-l}.\end{aligned}$$

The limit is again explicitly given with the Poisson limit theorem, which now lets us finish the proof since

$$\begin{aligned}\lim_{\delta\to\infty} P(X^*(A_1) = k, X^*(A_2) = l) =& e^{-|A_1|\lambda}\frac{(|A_1|\lambda)^k}{k!} e^{-|A_2|\lambda}\frac{(|A_2|\lambda)^l}{l!} \\ =& \lim_{\delta\to\infty} P(X^*(A_1) = k) P(X^*(A_2) = l),\end{aligned} \tag{5.3}$$

which shows the independence.

□

Remark (Stationarity).
The condition in Theorem 5.3.4 that,

$$\lim_{r\to\infty} \frac{|B_r(a) \cap X|}{|B_r(a)|} = \lambda \quad \forall a \in S$$

follows from the assumption that X is stationary. This is shown in Daley et al. [2003].

5.3.1 Interpretation

The choice of the distribution of ϵ is arbitrary under the restrictions given in the method description 5.2. However a natural choice would be a uniform distribution with the PDF

$$f(x) = \frac{1}{\pi\delta^2} \mathbb{1}_{B_\delta(0)}(x). \tag{5.4}$$

Another suitable and intuitive distribution would be the truncated normal distribution, which can be viewed by a normal distribution with a limited support [Horrace, 2005].

Assume given our selected data $\varkappa'$ in $\delta^+(W)$ depending on the choice of δ we might lose some information since we will later only train on a subset $W \subset \delta^+(W)$. The smaller the subset gets, the more information we might lose. Therefore the shape of W should be appropriate. With Theorem 5.3.1 we know that if we for example only have a long narrow shape for $\delta^+(W)$ the choice of delta might be even more

limited.

Theorem 5.3.3 gives us inside into the amount of expected points in the observation window. Note that with our previous method, where we added artificial data, we always expected more points in the observation window after noising the data. Learning with the same amount of points in expectation can be beneficial for estimating parameters related to the amount of points in a window.

Theorem 5.3.4 shows that for increasing delta we remove the dependencies in the point process. When carefully used, this can prevent overfitting. Note for example that shifting the data points will result in a "smoother" distribution of the points. For illustration consider the offset ϵ be distributed according to the uniform distribution in Equation (5.4) and let ρ_ω be the kernel density estimator with support W and a suitable bandwidth ω over the realization of the point process X. The kernel density estimation ρ^*_ω of the shifted process X^* is then given by the convolution

$$\rho^*_\omega(s) = \int_{W'} \rho_\omega(x) \frac{\mathbb{1}_{B_\delta(0)}(s-x)}{|B_\delta(0)|}\, dx = \frac{1}{|B_\delta(s)|} \int_{B_\delta(s)} \rho_\omega(x)\, dx.$$

Note that $\rho^*_\omega(s)$ describes the average density in $B_\delta(s)$. This will lead to a smoothing effect. An illustration can be seen in 5.3. A similar idea is for example used in the area of computer vision, when an image is blurred, by choosing a kernel with a uniform distribution [Pulfer, 2019].

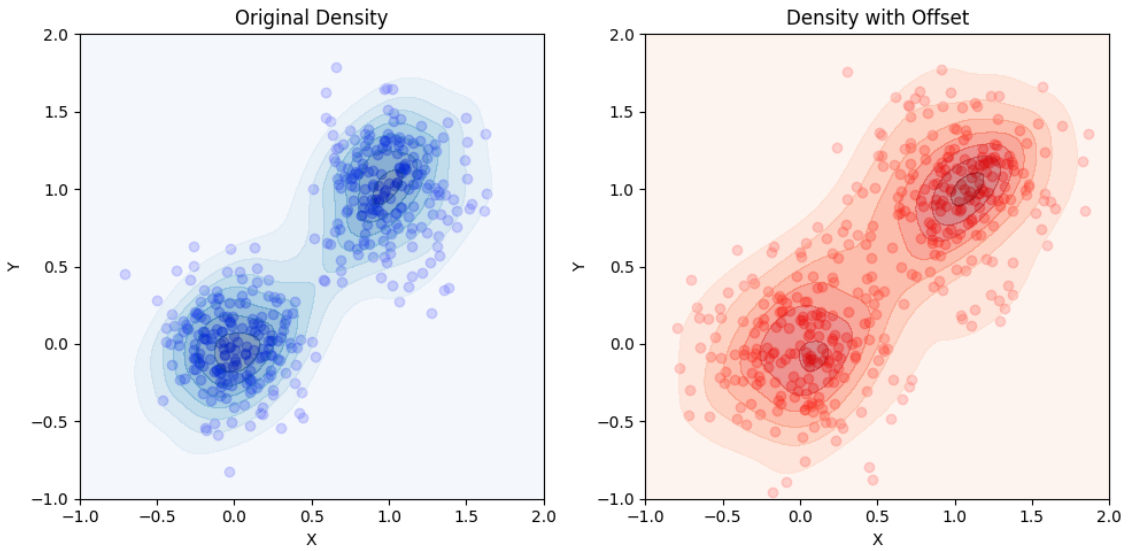


Figure 5.3: In blue we see a kernel density estimation without jittering of points. The right image in red shows the kernel density estimation after jittering the points according to a uniform distribution with support $B_\delta(0)$ and $\delta = 0.3$. We observe a smoothing effect for the kernel density of shifted data set. The dataset contains $n = 200$ points generated by two normally distributed random variables with $\sigma = 0.07$. The bandwidth was chosen according to the Scott's Rule by Scott [2010].

Remark. Note that we again could decide if we only noise the training or the validation set. We however decide to not investigate that any further, since it does not yield any interesting results.

6

Simulation Study for the Shifting Method

We are also conducting a simulation study regarding the shifting method, which we introduced in Chapter 5. As in the previous simulation study, which investigated the addition of additional data points, two processes will be examined. As before, we also investigate the influence on the hard-core process 2.1.6.2 and the influence on the area interaction process 2.1.6.3.
For the evaluation we again use three different metrics. First we are interested in the $MSE(\hat{\theta})$, which is given by

$$MSE(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta_0)^2] \approx (\frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i - \theta_0)^2, \tag{6.1}$$

where $\hat{\theta}_i$ is the optimal θ for the i-th experiment.
The MSE can be decomposed into bias and variance as shown in Section 1.2. The bias, which is given by,

$$\text{Bias}(\hat{\theta}) = \mathbb{E}[\hat{\theta} - \theta_0] \approx (\frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i - \theta_0) \tag{6.2}$$

and variance, which is given by

$$\text{Var}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^2] \approx \frac{1}{n}\sum_{i=1}^{n}(\hat{\theta}_i - \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_i)^2. \tag{6.3}$$

6.1 Experiments

The experiment for jittering data points has a similar setup to the previous experiment. The steps for it are given here.

- We again fix the parameters k and I where k describes the amount of thinnings we consider for the cross validation. I describes the amount of iterations for estimating the already described metrics.
- We fix $\theta_0 \in \Theta$ which will describe the distribution of the generating point process. We refer to the true conditional intensity as $\lambda_{\theta_0}(u; X)$.
- We then fix a vector Δ which stores different supports of the distribution used to shift the data points.

- We then iterate over I and Δ and for every I we generate a new data process according to θ_0 on a window size depending on $\delta \in \Delta$. The window size is given by
$$W' = [-\delta, 1+\delta]^2 \tag{6.4}$$
- For every δ in Δ we generate k offsets $\epsilon_1, \ldots, \epsilon_n$ with help of the truncated normal distribution of support δ; the standard divination s is chosen to be $s = \delta$.
- We generate shifted data $X_1^*, \ldots, X_k^*$. The shifted data sets are then used for training according to the method in 5.2.
- We then train the shifted data on the reduced window W where W is now the unit square given by,
$$W = [0,1]^2.$$

The algorithm which we implemented is therefore similar to the algorithm used for training with additional noise.

Require:
$I > 0$: the amount of iterations,
$k > 0$: the amount of thinnings for the cross validation,
Δ: a vector storing the support of the shift,
$\theta_0 \in \Theta$: parameter describing the true distribution.

for $i \in I$ **do**
  Generate the process X according to θ_0 in W'.
  for $n \in N$ **do**
    Generate shift vectors $\epsilon_1, \ldots, \epsilon_k$.
    Generate shifted processes $X_1^*, \ldots, X_k^*$.
    Restrict the points to W.
    Train the model with thinnings $X_V^{1*}, \ldots, X_V^{k*}$ with retention probability p.
    Save the estimate $\hat{\theta}$.
  end for
  Calculate desired statistics with θ_0 and $\hat{\theta}$.
end for

6.1.1 Hard-Core Process

For the hard-core process we have the two parameters β and R which we can fix. For generating a realization of the hard-core process we choose the parameters to be

$$\beta = 100 \quad R = 0.05.$$

As an observation window we choose the unit square. We choose I to be 100, which means that we execute the experiment 100 times for each level of noise. The remaining hyper-parameters are given as

$$\begin{aligned} \text{deltas} &= \{0, 0.001, 0.002, 0.003, 0.005, 0.006, 0.007, 0.01, 0.015, 0.02, 0.03, 0.05, 0.1\}, \\ p &= \{0.2, 0.5, 0.8\}, \\ k &= \{1, 3, 10\}, \\ \text{test function } h &= \lambda_\theta(u; \aleph)^{-1}, \end{aligned}$$

where the deltas describe the support of the offset distribution.

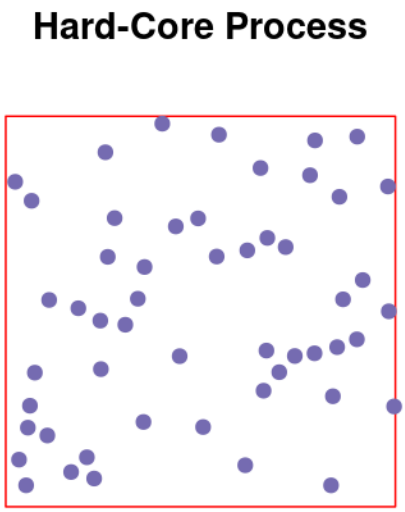


Figure 6.1: Realization of the hard-core process with the proposed parameters $\beta = 100$ and $R = 0.05$.

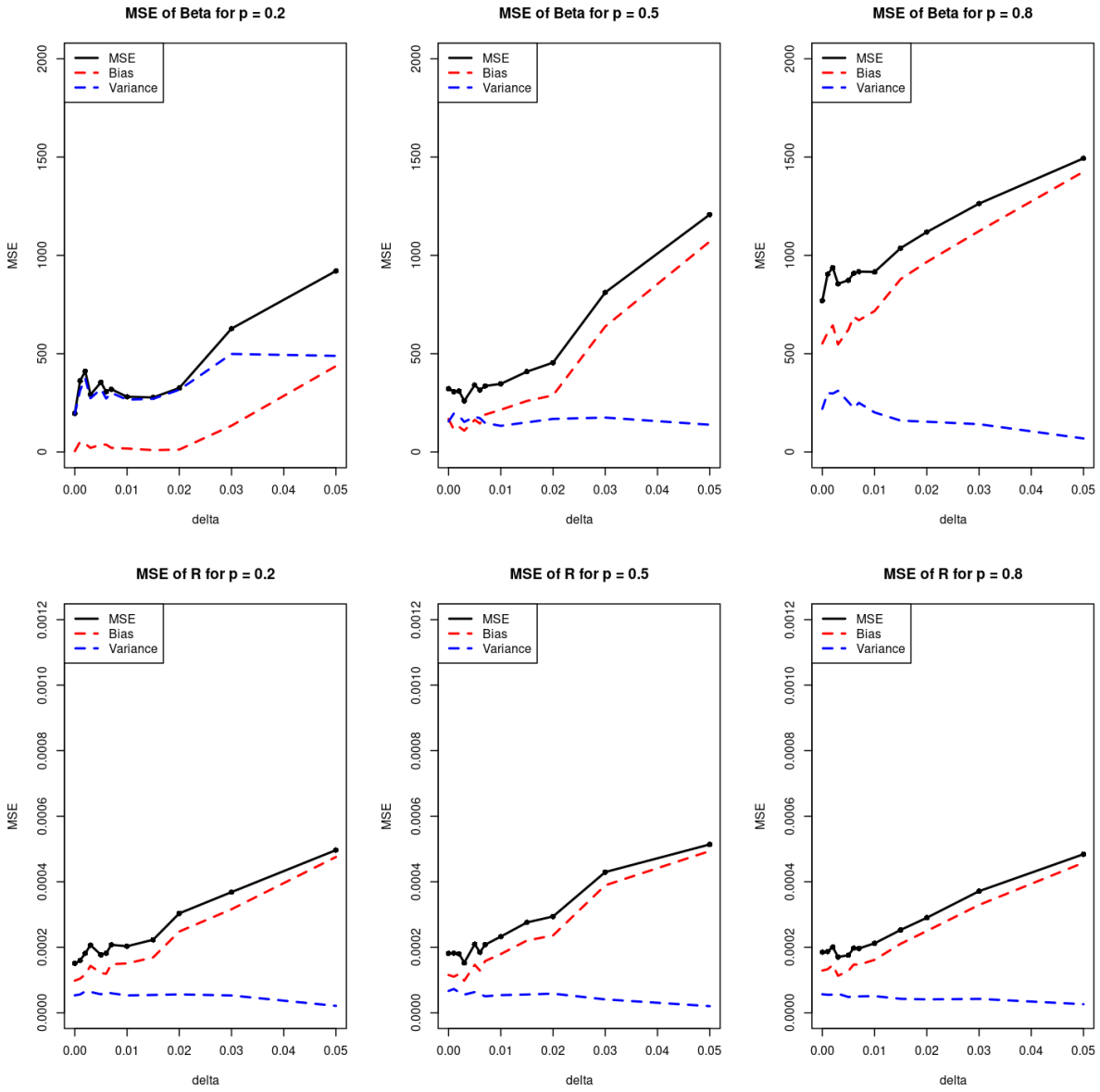


Figure 6.2: MSE, Variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support of the distribution for jittering the data points. $k = 1$ was chosen.

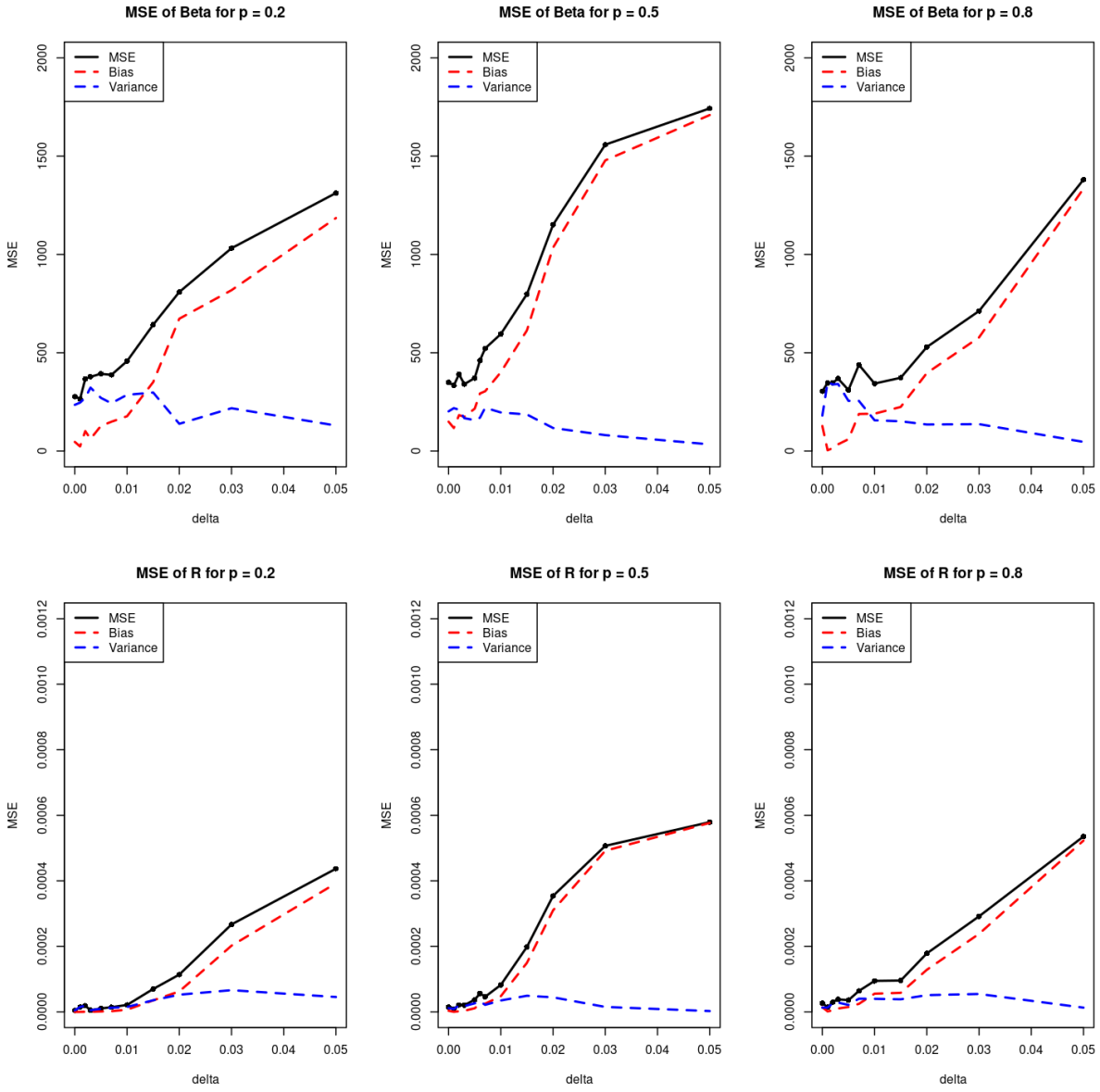


Figure 6.3: MSE, variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support of the distribution for jittering the data points. $k = 3$ was chosen.

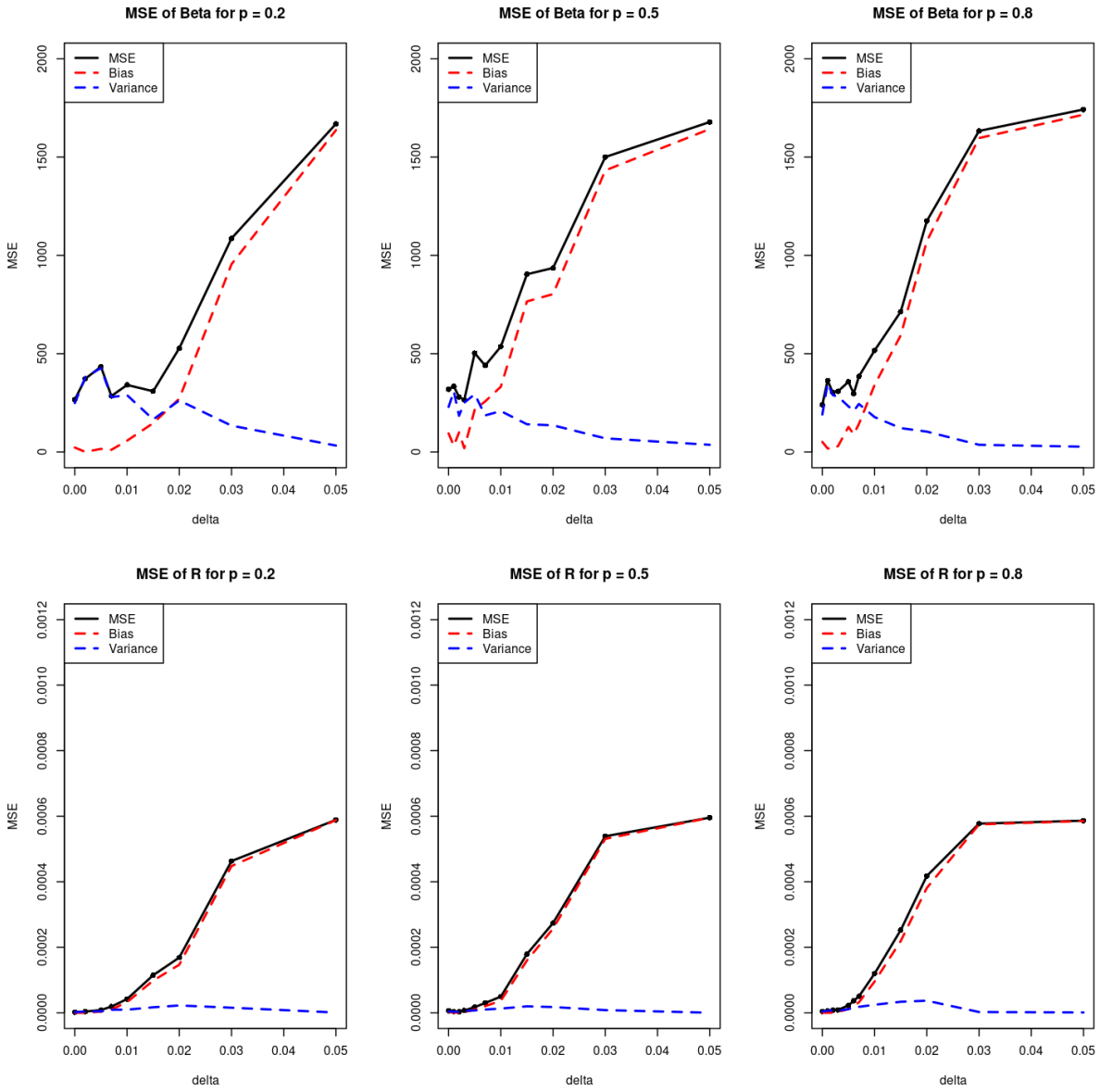


Figure 6.4: MSE, variance and squared bias of the parameters β and R for the Hard core process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support of the distribution for jittering the data points. $k = 10$ was chosen.

6.1.2 Area Interaction Process

We again use the spatstat version of the area interaction process, which was already discussed in the last simulation study in Section 4.1.2. We again fix the parameter β and only investigate the parameters R and η. This is again motivated by lower computing time and complexity.

For generating a realization of the area interaction process we choose the parameters to be

$$\eta = 2 \quad R = 0.05 \quad \beta = 50.$$

As observation window we choose the unit square. We choose N to be 50 because of drastically increased computing time. The remaining hyper-parameters are given by

$$\begin{aligned} \text{noise intensity} &= \{0, 0.5, 1, 1.5, 2, 3, 4, 5, 6, 7, 8, 10, 15\}, \\ \text{deltas} &= \{0.002, 0.005, 0.007, 0.01, 0.015, 0.02, 0.03, 0.05, 0.1\}, \\ p &= \{0.2, 0.5, 0.8\}, \\ k &= \{1, 3, 10\}, \\ \text{test function } h &= \lambda_\theta(u; \mathbb{x})^{-1}, \end{aligned}$$

where the deltas describe the support of the offset distribution.

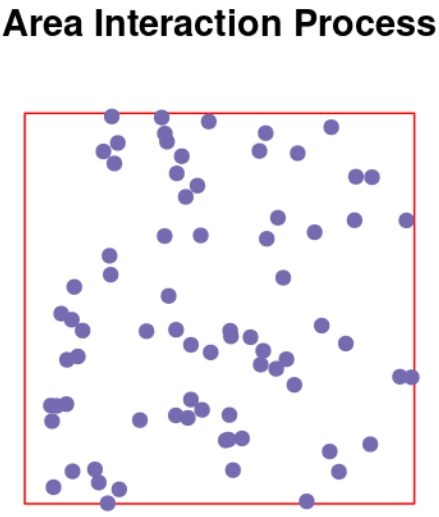


Figure 6.5: Realization of the area interaction process with the proposed parameters $\beta = 50$, $\eta = 2$ and $R = 0.05$.

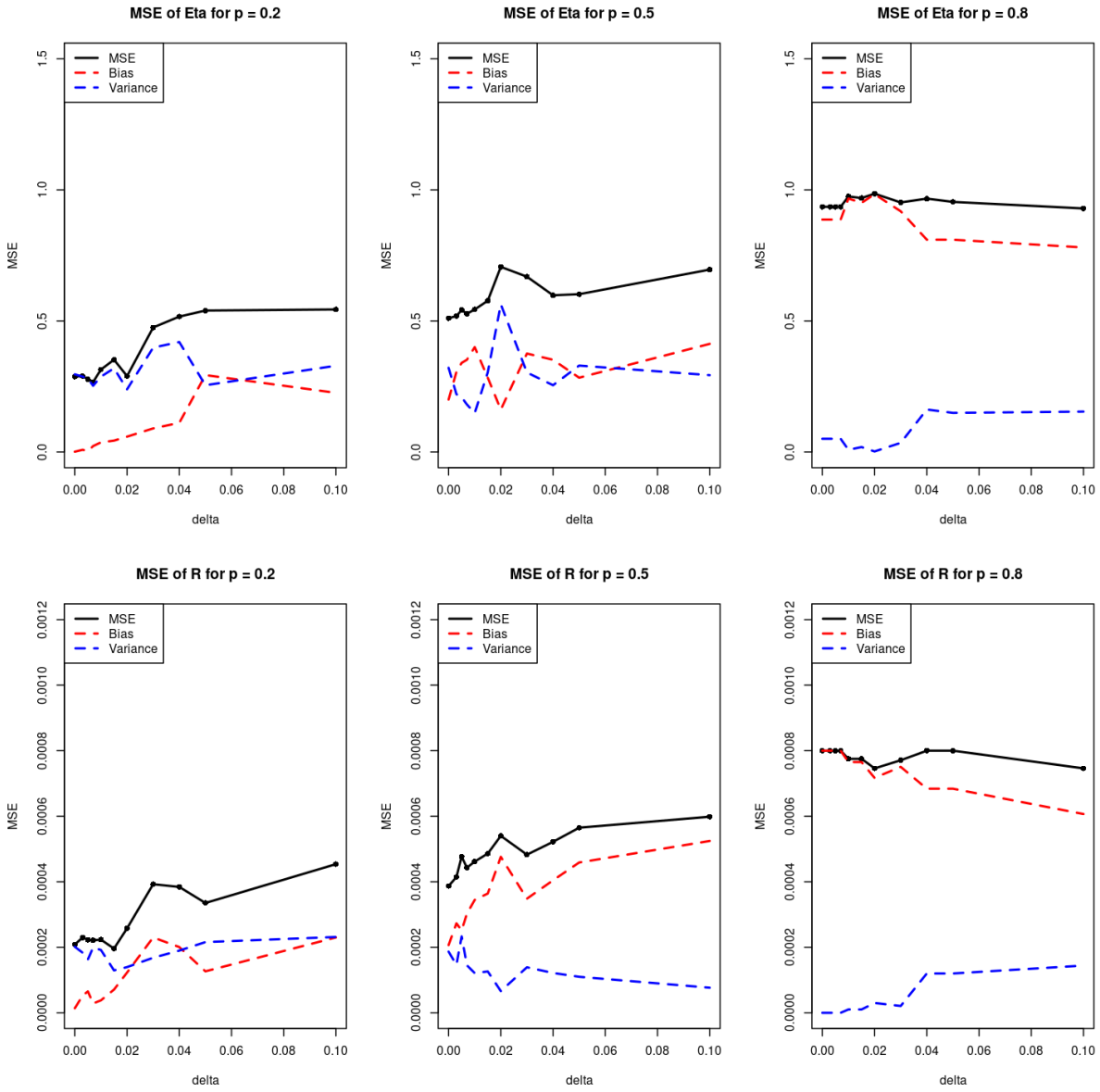


Figure 6.6: MSE, variance and squared bias of the parameter η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support for the jittering variable ϵ. $k = 1$ was chosen.

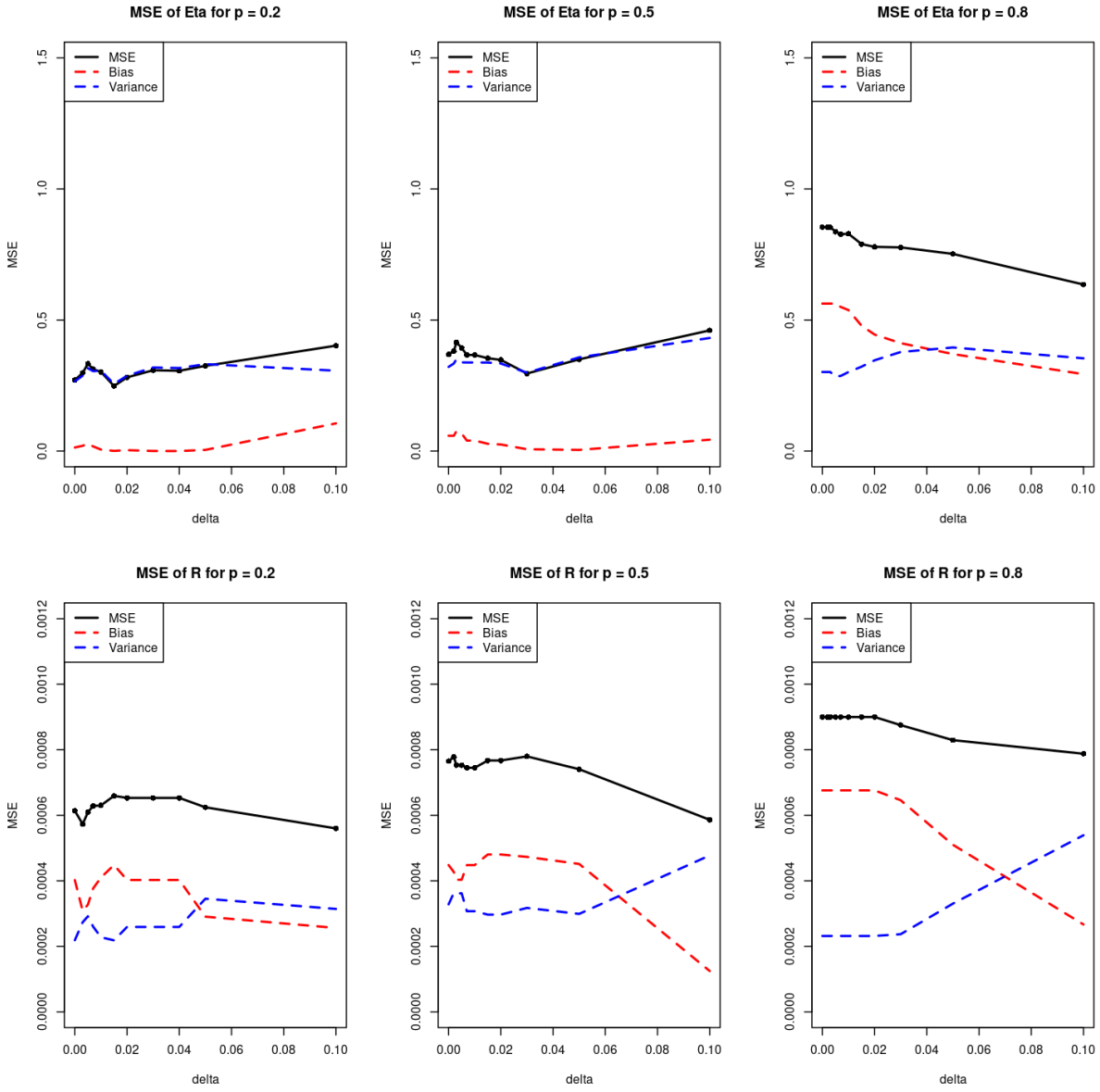


Figure 6.7: MSE, variance and squared bias of the parameter η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support for the jittering variable ϵ. $k = 3$ was chosen.

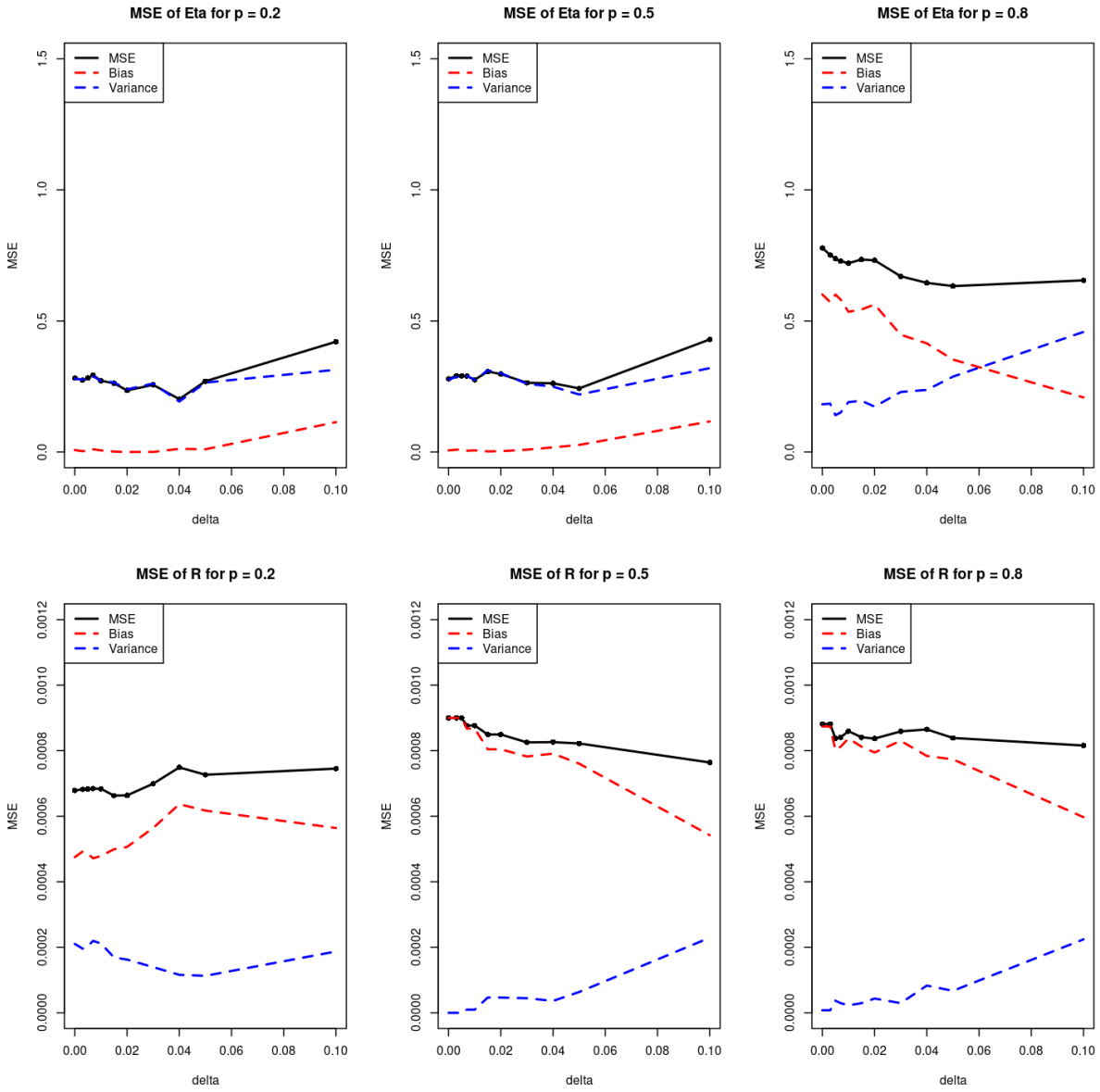


Figure 6.8: MSE, variance and squared bias of the parameter η and R for the area interaction process with $p \in \{0.2, 0.5, 0.8\}$. The x-axis indicates the support for the jittering variable ϵ. $k = 10$ was chosen.

6.2 Results

We again separate the analysis of the two point processes. We first discuss the results from the hard-core process and then the results from the area interaction process.

6.2.1 Analysis of the Hard-Core Process

In the case of the hard-core process, no significant improvement in performance can be observed. For $k = 1$, $k = 3$, $k = 10$ the MSE for R and for β increases in relation to the parameter δ. The higher k is, the faster the MSE increases. For small δ, however, the performance is almost identical. Therefore similar to the previous method for noising the data, the hard-core process seems to be to sensitive to noise.

The increase of the MSE is most likely based on a similar reason as in the last simulation study. When the points on the grid get shifted, the minimum distance between points of a pattern decreases. For illustration we made a Monte-Carlo experiment, which can be seen in Figure 6.9. This implies the underestimation of R and therefore also disturbs the estimation of β. We already gave a more detailed analysis for that in the last simulation study in Chapter 4.

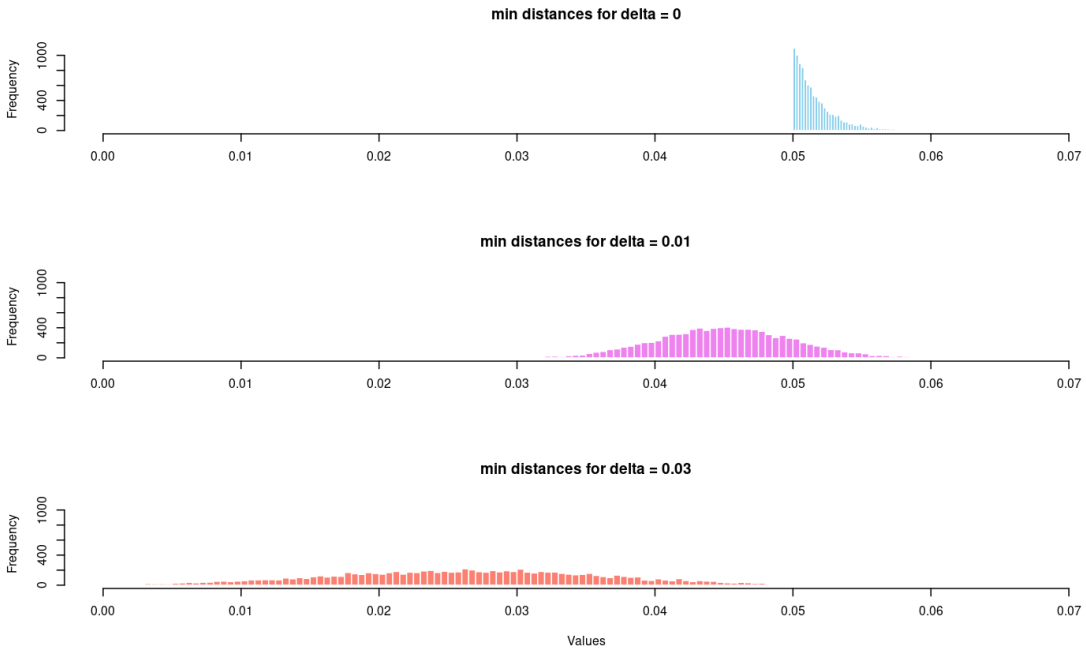


Figure 6.9: Distribution of the minimum distance between points of the hard-core process after jittering, based on 10000 generated samples. We clearly observe that when the hard-core process is shifted, R will be underestimated.

However this effect can also be used as an advantage, since for smaller values of R and β, the method from Cronie et al. [2023] overestimates the parameter R. An illustration can be seen in Figure 6.10, where we $\beta = 50$ was chosen, leading to fewer

points in the grid.

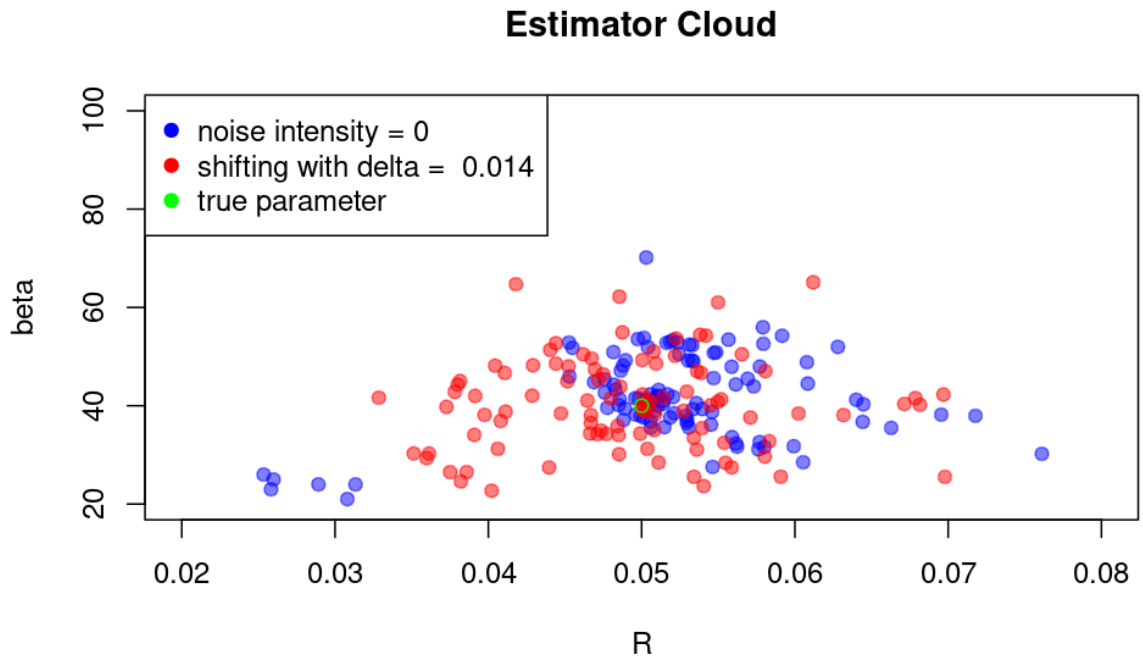


Figure 6.10: Estimator Cloud over 100 experiments. The blue points indicate the estimator $\hat{\theta}$ without jittering the data points.The red points describe the estimator for the data set with shifted points. We chose $k = 5$, $R = 0.05$ and $\beta = 40$. The bias of both estimators is different.

6.2.2 Analysis of the Area Interaction Process

The shifting method also shows promising regulatory effects for the area interaction process. This can be clearly observed for $k = 3$ and especially for $k = 10$ for the parameter η. For $k = 1$ no real improvement can be seen. For higher k the estimation of the parameter R is again poor. However, shifting the data points also has a regulating effect here.

We assume that the cluster structure of the area interaction process is maintained by slightly shifting the data points. Points that are close to each other often remain within range for smaller δ after jittering, where the points still interact with each other. This allows realistic copies of the original process to be created. These can then be successfully used for data augmentation. If δ is too large, the cluster structure is canceled out. We already gave an intuition, which can be seen in Figure 5.3. The points no longer interact and the data set is too noisy. The regulating effect could be seen particularly well for $k = 10$. It can be assumed that if we have more copies, the regulating effect is also stronger.

7
Conclusion

Motivated by the regularizing effect of data augmentation, we have tried to use this concept for the interference of point processes. To this end, we started to use the method developed by Cronie et al. [2023] and motivate it with background from the theory of point processes. With this method, we also introduced the term Point Process Learning, which, with its introduction, should be considered the basis for our work. In general, there are several methods that can be associated with Point Process Learning. For example, Cronie et al. [2023] linked cross validation with Point Process Learning [Cronie et al., 2023]. Our work had the task of combining the concept of data augmentation with Point Process Learning. As motivated at the beginning of the thesis, the idea of data augmentation is to create slightly modified copies of existing data and then incorporate these into the training process. Two explicit examples of this were presented at the beginning of the thesis, based on linear regression and deep learning. The regulating effect was also demonstrated.

However, the explicit question now arises as to how data augmentation can be applied to point processes. We have developed two general concepts for this. The first concept was to add artificial new points to an already existing point pattern and thereby slightly modify it.
The additional points are generated by a so-called noise process, and then our original point process is generated from the superposition of the original process and the noise process. The noise process is generally freely electable. It can be dependent or independent of the original process. However, in this chapter, we did not address the explicit choice of the noise process. This remains open for future work. Instead, we focused on the properties of the superposition. For example, we showed how the Papangelou conditional intensity behaves for the superposition of two independent processes. These insights were important for incorporating the described method into Point Process Learning. The idea now is to create several similar copies of the dataset and then use the method of Cronie et al. [2023] to find an estimator that best describes all the copies. The estimator is constrained using a parameterized family that describes the original process. The regularizing effect of this idea was demonstrated with the subsequent simulation study. For a parameterized family describing the area interaction process, an improvement was observed with the correct intensity of the noise process. Here, the noise process was chosen as a Poisson process. We also tested the method with the introduced hard-core process. Here, no improvement was observed. It would be interesting to know whether more thoughtful noise processes might potentially yield better results in this case.

The second introduced method for noising the point processes is based on a very intuitive idea. Each point in the point pattern should be shifted within a certain area around it. As a result, the resulting pattern is similar to the original pattern when small shifts are applied. However, during the jittering process, it's possible for points to be shifted out of our observation window. To compensate for this problem, we developed a method that first observes the process in a larger area and then considers also points that are shifted into our observation window. We showed that for stationary processes, the expected number of points in a window before and after jittering is equal. This also motivates the use of this method, as this was not the case in the previous method. As the shifting becomes more significant, the spatial dependence between points diminishes. We also showed that under certain conditions, this dependence disappears entirely as the jittering becomes stronger. One speculation was that this approach can help avoid overfitting.
Our simulation study partially confirms our speculations. Promising results were also obtained for the area interaction process using this method. It was observed that for a well-chosen shifting range, a regularazing effect can be achieved.

In summary, we have developed and presented the basis and mathematical foundations of a promising method. These developed methods do not necessarily have to be used for determining the correct parameters. They could also be useful in the future for cleaning noisy data or generating potentially lost data points. The theoretical foundations have been laid out for this purpose. However, the thesis also raises many other interesting questions and approaches, which we will briefly discuss in the next paragraph. Overall, it can be said that data augmentation can be successfully applied in the field of Point Process Learning.

7.1 Future Work

As the thesis has already shown, this topic can have many facets and it is difficult to include all aspects and possibilities in one work. It will therefore be discussed at the end what possibilities still exist and where future work can continue.

First of all, it should be noted that in our simulation study we used a Poisson process as the noise process. In Chapter 3 we have already discussed other noise processes. The effect on the training process would be interesting to observe, as a more carefully chosen noise process could, for example, better preserve the properties of the hard-core process and the training would therefore not react so sensitively to noise.

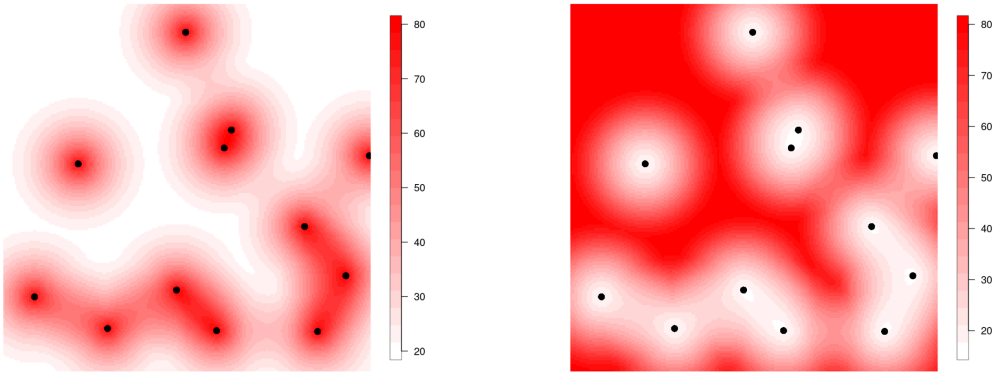


Figure 7.1: Conditional Intensity of the area interaction process applied on a process X with, left, $\eta = 2$ and, right, $\eta = 0.5$. Depending on the parameter η new points could be sampled according to an attractive or repulsive process. The process X is a homogenous Poisson process.

The area interaction process can model attractive and repulsive behavior depending on the parameterization, as we already showed in Chapter 2. It would be interesting to make use of that and design a noise process based on these characteristics. This noise process would then depend on the already existing data and can therefore model an assumption of the dataset. An illustration can be seen in Figure 7.1, where the conditional intensity depending on the process X is given with two different parameterizations of the area interaction process. The idea for the method seems really promising and offers nice mathematical insights. However, it needs further investigation and would be therefore interesting for follow-up work.
It would be also interesting to investigate the behaviour not only on other process types but also on real data sets. Because in the end this would be a useful application of Point Process Learning.

In Chapter 3 We also discussed the option of only noising the test function h. We however decided against testing this in the simulation study because the choice of the test function was very limited. There are however also different approaches to choose h. When testing out different classes of functions for the test function, the noise in the test function can be an interesting aspect.

Bibliography

Ottmar Cronie, Mehdi Moradi, and Christophe A N Biscio. A cross-validation-based statistical theory for point processes. *Biometrika*, page asad041, 06 2023. ISSN 1464-3510. doi: 10.1093/biomet/asad041. URL https://doi.org/10.1093/biomet/asad041.

Pau Aragó, C. Díaz-Avalos, Pablo Juan, and Patrica Salvador. Spatial point process modeling applied to the assessment of risk factors associated with forest wildfires incidence in castellón, spain. *European Journal of Forest Research*, 135, 06 2016. doi: 10.1007/s10342-016-0945-z.

Yosihiko Ogata. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, 1998. URL https://EconPapers.repec.org/RePEc:spr:aistmt:v:50:y:1998:i:2:p:379-402.

Nicolas Picard, Avner Bar-Hen, Frédéric Mortier, and Joel Chadoeuf. The multi-scale marked area-interaction point process: a model for the spatial pattern of trees. *Scandinavian journal of statistics*, 36(1):23–41, 2009.

J Daniel Kelly, Junhyung Park, Ryan J Harrigan, Nicole A Hoff, Sarita D Lee, Rae Wannier, Bernice Selo, Mathias Mossoko, Bathe Njoloko, Emile Okitolonda-Wemakoy, et al. Real-time predictions of the 2018–2019 ebola virus disease outbreak in the democratic republic of the congo using hawkes point process models. *Epidemics*, 28:100354, 2019.

Van Lieshout. *Theory of spatial statistics: A concise introduction*. CRC Press, 2019.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Renaud Alie, David A. Stephens, and Alexandra M. Schmidt. On data augmentation in point process models based on thinning procedures, 2023.

Ryan Prescott Adams, Iain Murray, and David J. C. MacKay. Tractable nonparametric bayesian inference in poisson processes with gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 9–16, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553376. URL https://doi.org/10.1145/1553374.1553376.

Alex Hernández-García and Peter König. Data augmentation instead of explicit regularization, 2020.

David A van Dyk and Xiao-Li Meng. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10(1):1–50, 2001. doi: 10.1198/10618600152418584. URL https://doi.org/10.1198/10618600152418584.

Samprit Chatterjee and Ali S Hadi. Influential observations, high leverage points, and outliers in linear regression. *Statistical science*, pages 379–393, 1986.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

David Dereudre. *Introduction to the Theory of Gibbs Point Processes*, pages 181–229. Springer International Publishing, Cham, 2019. ISBN 978-3-030-13547-8. doi: 10.1007/978-3-030-13547-8_5. URL https://doi.org/10.1007/978-3-030-13547-8_5.

Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*. Springer, 2008.

Daryl J Daley, David Vere-Jones, et al. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

Jan Grandell. Point processes and random measures. *Advances in Applied Probability*, 9(3):502–526, 1977. ISSN 00018678. URL http://www.jstor.org/stable/1426111.

Jens Ledet Jensen and Jesper Moller. Pseudolikelihood for Exponential Family Models of Spatial Point Processes. *The Annals of Applied Probability*, 1(3):445 – 461, 1991. doi: 10.1214/aoap/1177005877. URL https://doi.org/10.1214/aoap/1177005877.

F. Papangelou. The conditional intensity of general point processes and an application to line processes. *Zeitschrift fuer Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 28(3):207–226, 1974. ISSN 1432-2064. doi: 10.1007/bf00533242. URL http://dx.doi.org/10.1007/BF00533242.

D. Stoyan, W.S. Kendall, and J. Mecke. *Stochastic Geometry and Its Applications*. Inorganic Chemistry. Wiley, 1995. ISBN 9780471950998. URL https://books.google.se/books?id=oFDvAAAAMAAJ.

A. Baddeley, R. Turner, J. Møller, and M. Hazelton. Residual analysis for spatial point processes. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(5):617–666, 2005. ISSN 13697412, 14679868. URL http://www.jstor.org/stable/3647612.

David J. Strauss. A model for clustering. *Biometrika*, 62(2):467–475, 08 1975. ISSN 0006-3444. doi: 10.1093/biomet/62.2.467. URL https://doi.org/10.1093/biomet/62.2.467.

B. D. Ripley. Algorithm as 137: Simulating spatial patterns: Dependent samples from a multivariate density. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):109–112, 1979. ISSN 00359254, 14679876. URL http://www.jstor.org/stable/2346831.

Lennart Bondesson and Jessica Fahlén. Mean and variance of vacancy for hard-core disc processes and applications. *Scandinavian Journal of Statistics*, 30(4): 797–816, 2003.

Adrian J Baddeley and MNM Van Lieshout. Area-interaction point processes. *Annals of the Institute of Statistical Mathematics*, 47:601–619, 1995.

R Takacs. Estimator for the pair–potential of a gibbsian point process. *Statistics: A Journal of Theoretical and Applied Statistics*, 17(3):429–433, 1986.

Thomas Fiksel. Estimation of interaction potentials of gibbsian point processes. *Statistics*, 19:77–86, 01 1988. doi: 10.1080/02331888808802074.

jean-francois coerjolly and David Dereudre. Takacs-fiksel method for stationary marked gibbs point processes. *Scandinavian Journal of Statistics*, 39(3):416–443, 2012. ISSN 03036898, 14679469. URL http://www.jstor.org/stable/41679804.

Julian Besag. Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 24(3):179–195, 1975. ISSN 00390526, 14679884. URL http://www.jstor.org/stable/2987782.

Jens Ledet Jensen and Jesper Møller. Pseudolikelihood for exponential family models of spatial point processes. *The Annals of Applied Probability*, 1(3):445–461, 1991. ISSN 10505164. URL http://www.jstor.org/stable/2959747.

Jean-François Coeurjolly, Yongtao Guan, Mahdieh Khanmohammadi, and Rasmus Waagepetersen. Towards optimal takacs–fiksel estimation, 2016.

Andrew Y Ng et al. Preventing" overfitting" of cross-validation data. In *ICML*, volume 97, pages 245–253. Citeseer, 1997.

Laurent Decreusefond and Aurélien Vasseur. Stein's method and papangelou intensity for poisson or cox process approximation, 2018.

Dietrich Stoyan and Pavel Grabarnik. Second-order characteristics for stochastic structures connected with gibbs point processes†. *Mathematische Nachrichten*, 151:95–100, 1991. URL https://api.semanticscholar.org/CorpusID:122022238.

Mohamed Alloghani, Dhiya Al-Jumeily, Jamila Mustafina, Abir Hussain, and Ahmed J. Aljaaf. *A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science*, pages 3–21. Springer International Publishing, Cham, 2020. ISBN 978-3-030-22475-2. doi: 10.1007/978-3-030-22475-2_1. URL https://doi.org/10.1007/978-3-030-22475-2_1.

Giovanni Cerulli. *The Basics of Machine Learning*, pages 1–17. Springer International Publishing, Cham, 2023. ISBN 978-3-031-41337-7. doi: 10.1007/978-3-031-41337-7_1. URL https://doi.org/10.1007/978-3-031-41337-7_1.

Adrian Baddeley and Rolf Turner. spatstat: An R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42, 2005. doi: 10.18637/jss.v012.i06.

Vyacheslav Tuzlukov. *Signal processing noise*. CRC Press, 2018.

E.G. Gilbert, D.W. Johnson, and S.S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal on Robotics and Automation*, 4(2):193–203, 1988. doi: 10.1109/56.2083.

A. D. Barbour and G. K. Eagleson. An improved poisson limit theorem for sums of dissociated random variables. *Journal of Applied Probability*, 24(3):586–599, 1987. doi: 10.2307/3214091.

William C. Horrace. Some results on the multivariate truncated normal distribution. *Journal of Multivariate Analysis*, 94(1):209–221, 2005. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2004.10.007. URL https://www.sciencedirect.com/science/article/pii/S0047259X04002167.

Erich-Matthew Pulfer. Different approaches to blurring digital images and their effect on facial detection. 2019.

David Scott. Scott's rule. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2, 07 2010. doi: 10.1002/wics.103.